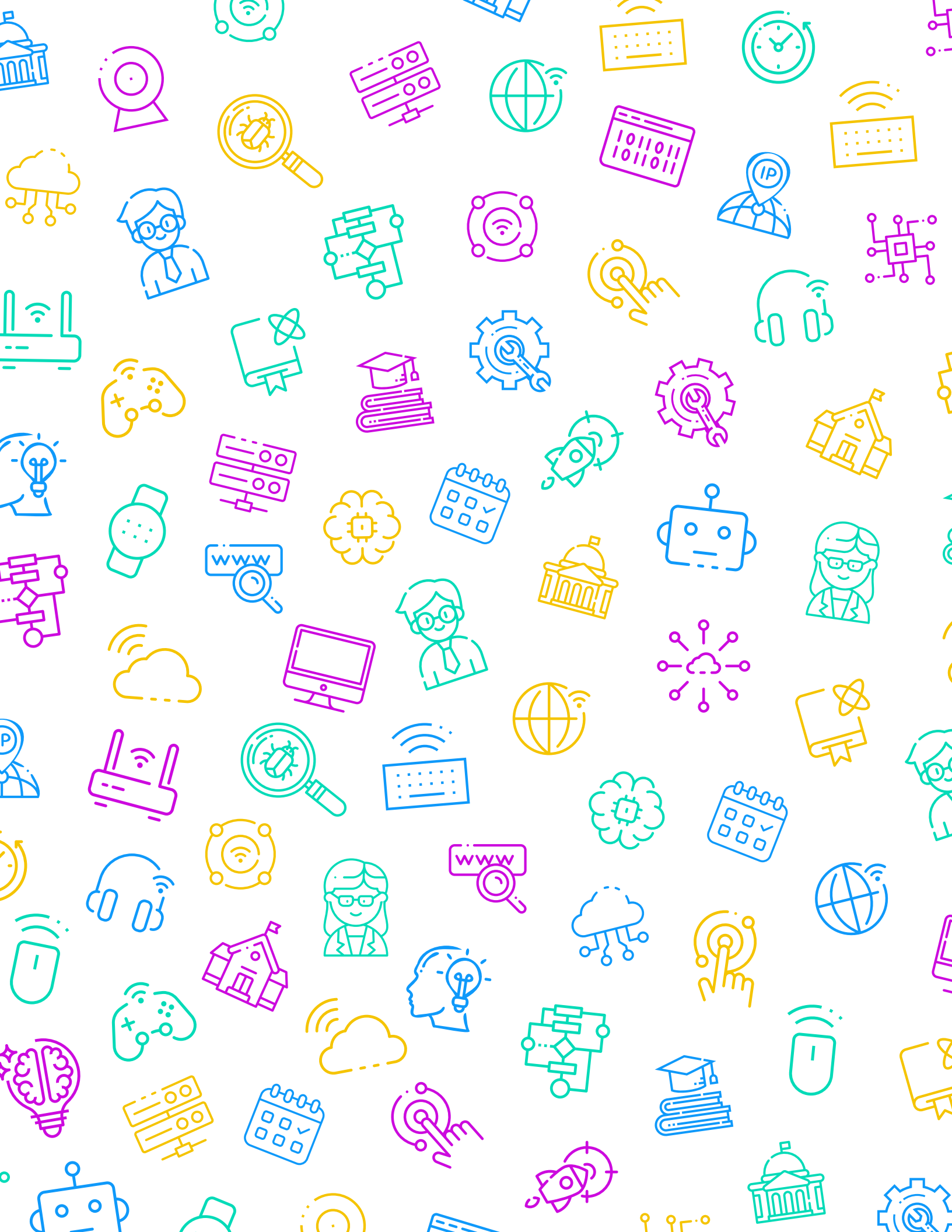


APRENDO A PROGRAMAR

Programación para III y IV medio

Alineado curricularmente con Pensamiento
Computacional y Programación

IdeoDigital



Aprendo a Programar

Editores

María Cristrina Cid Cartes

Loraine Schlotterbeck Byrne

Curadores

Sebastián Godoy Reyes

Marisel Mateluna Jorquera

Diseñadora Gráfica

Tamara Oyanedel

IdeoDigital

Proyecto país originado gracias a una alianza estratégica entre Fundación Kodea y BHP Foundation, que busca crear las condiciones necesarias para implementar las Ciencias de la Computación en el sistema escolar público chileno, para que miles de niños, niñas y adolescentes, se conviertan en protagonistas de la sociedad digital del siglo XXI.

Líder de Proyecto

Claudia Jaña

Gerente de Educación- Fundación Kodea



IdeoDigital

Ciencias de la Computación en el aula

Desarrolla

fundación **kodea**

Apoya y respalda

BHP Foundation

APRENDO A PROGRAMAR

Programación para III y IV medio

Alineado curricularmente con Pensamiento
Computacional y Programación

Índice

01	Introducción	7
02	Unidad 01	15
	¿Qué es el pensamiento computacional y la programación?	
03	Unidad 02	167
	La resolución de problemas y las máquinas	
04	Unidad 03	309
	¿Cómo nos ayuda el computador para resolver problemas matemáticos?	

05 **Unidad 04** 403

¿Cómo se hace una app móvil?

06 **Referencias** 593

07 **Glosario** 595

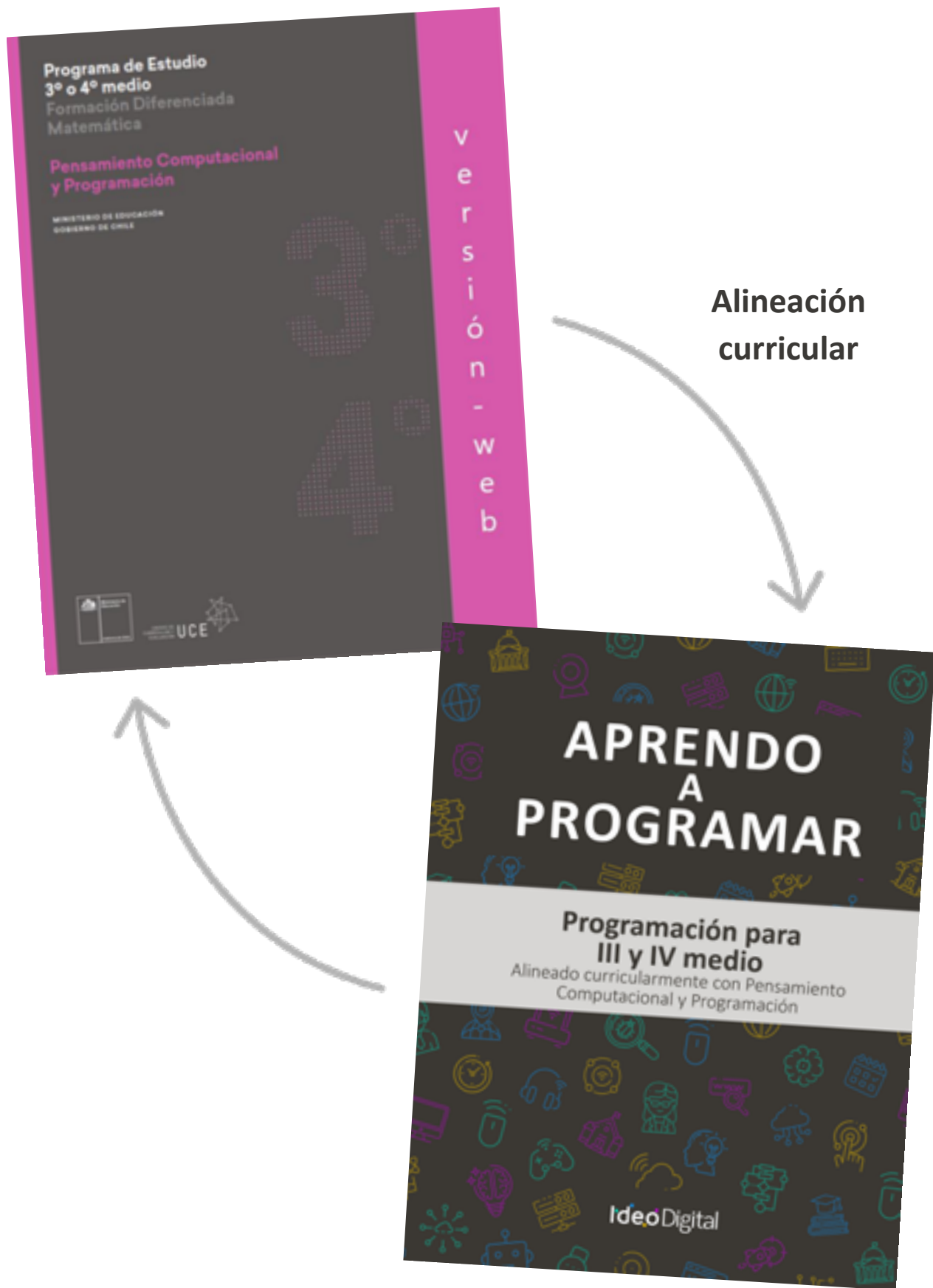
Introducción

Presentación del programa

Aprendo a programar: programación integrada con Pensamiento computacional y Programación pone el foco en relevar los Objetivos de Aprendizaje de las Bases Curriculares desde la lógica de la programación para los niveles de III y IV medio.

Para efectos pedagógicos significativos y coherentes con los intereses de los estudiantes, la integración de aprendizajes está enfocada en lecciones con y sin conexión, con la intención de incrementar instancias de aprendizaje que necesitan ser resueltas por medio de la programación, la interacción con equipos computacionales y el uso de materiales y recursos educativos que permiten a los estudiantes avanzar desde ideas concretas u otras más abstractas. Asimismo, se busca una integración coherente con los conocimientos y habilidades propias del Pensamiento Computacional y Programación para estos niveles, como son los fundamentos de ciencias de la computación, el pensamiento crítico, la creatividad, la comunicación efectiva y la colaboración.

En cada una de las lecciones se favorece la adquisición de un lenguaje de programación que facilita la resolución de problemas en diferentes ambientes de enseñanza aprendizaje.



Propósito de Aprendo a Programar

El propósito de las lecciones es relevar estrategias didácticas asociadas al aprendizaje de la programación y al desarrollo del pensamiento técnico y tecnológico, que permitan abordar de manera simultánea los objetivos de aprendizaje prescritos en las Bases Curriculares vigentes para la asignatura de Pensamiento Computacional y Programación.

Esta propuesta de aprendizaje busca dar continuidad al Programa Fundamentos de la Computación de 1º básico a 2º año medio disponible en currículum nacional, fortaleciendo conceptos y habilidades del pensamiento computacional, la resolución de problemas tecnológicos, el diseño sistemas y la comprensión del mundo a través de la tecnología, el ambiente y la sociedad.

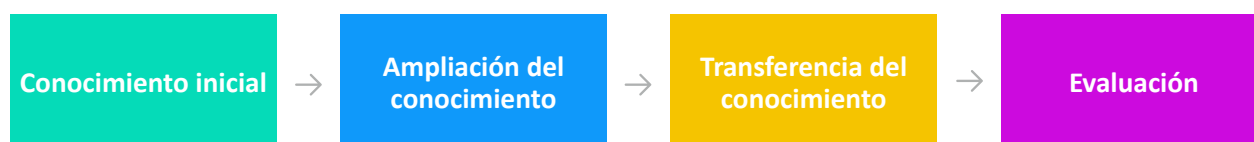
Para el uso e implementación de las lecciones se sugieren propósitos, secuencias de aprendizaje, momentos de preparación, vocabulario y los objetivos de cada una, las cuales potencian e integran activamente los aprendizajes de los estudiantes a través de la resolución de problemas, el desarrollo de habilidades y la aplicación creativa de la programación, teniendo en cuenta el tiempo disponible y las particularidades de cada contexto escolar.

Las actividades consolidadas en cada lección se suman a una serie de experiencias de aprendizaje, con el objetivo de profundizar y afianzar el conocimiento de los contenidos vistos, así como también fortalecer las habilidades abordadas en cada unidad. Además, propone desafíos a los estudiantes, que los docentes podrán utilizar como ticket de salida y parte de la evaluación de proceso, según la pertinencia y atingencia de los avances de sus estudiantes, ya que la idea es facilitar el proceso de enseñanza aprendizaje de los y las estudiantes.¹

Cada lección será valorada, como una evaluación de proceso, con el fin de enriquecer los conocimientos adquiridos, a través de desafíos que serán considerados como una estrategia de evaluación de salida, retroalimentando de manera constante, así como también desarrollando la metacognición y metaevaluación de los estudiantes.

Aprendo a programar entrega una serie de sugerencias al docente, recomendaciones de recursos didácticos complementarios, como videos, tutoriales y bibliografía dispuesta tanto para profesores como para los y las estudiantes.²

Para poder abordar las lecciones, en cada una de ellas se sugiere la mencionada secuencia de aprendizaje, la cual está asociada a tiempos para cada instancia, siendo flexible de acuerdo con la asignación de horas de libre disposición que considere cada establecimiento:



^{1 y 2} <https://www.curriculumnacional.cl/portal/Diferenciado-Humanista-Cientifico/Matematica/Pensamiento-computacional-y-programacion/>

Objetivos de Aprendizaje de Pensamiento computacional y programación³ que abordan las lecciones de Aprendo a Programar

Aprendo a Programar contempla cuatro unidades, diseñadas de manera progresiva, considerando los Objetivos de Aprendizaje de Pensamiento computacional y programación.

Objetivos de Aprendizaje Pensamiento Computacional y programación 3° o 4° medio		Lecciones que cubren OA
OA 1.	Aplicar conceptos de Ciencias de la Computación – abstracción, organización lógica de datos, análisis de soluciones alternativas y generalización– al crear el código de una solución computacional.	Desde la lección 1 a la 25.
OA a.	Construir y evaluar estrategias de manera colaborativa al resolver problemas no rutinarios.	Desde la lección 5 a la 25.
OA d.	Argumentar, utilizando lenguaje simbólico y diferentes representaciones para justificar la veracidad o falsedad de una conjetura, y evaluar el alcance y los límites de los argumentos utilizados.	Desde la lección 5 a la 25.
OA g.	Elaborar representaciones, tanto en forma manual como digital, y justificar cómo una misma información puede ser utilizada según el tipo de representación.	Desde la lección 12 a la 25.
OA k.	Analizar y evaluar el impacto de las tecnologías digitales en contextos sociales, económicos y culturales.	Desde la lección 5 a la 11.

³ <https://www.curriculumnacional.cl/portal/Diferenciado-Humanista-Cientifico/Matematica/Pensamiento-computacional-y-programacion/>

Objetivos de Aprendizaje Pensamiento Computacional y programación 3° o 4° medio		Lecciones que cubren OA
Actitudes	Pensar con perseverancia y proactividad para encontrar soluciones innovadoras a los problemas.	Desde la lección 1 a la 25.
Actitudes	Interesarse por las posibilidades que ofrece la tecnología para el desarrollo intelectual, personal y social del individuo.	Desde la lección 5 a la 11.
Actitudes	Aprovechar las herramientas disponibles para aprender y resolver problemas.	Desde la lección 5 a la 25.
OA 2.	Representar diferentes tipos de datos en una variedad de formas que incluya textos, sonidos, imágenes y números.	Desde la lección 26 a la 33.
OA 3.	Desarrollar y programar algoritmos para ejecutar procedimientos matemáticos, realizar cálculos y obtener términos definidos por una regla o patrón.	Desde la lección 34 a la 48.
OA g.	Elaborar representaciones, tanto en forma manual como digital, y justificar cómo una misma información puede ser utilizada según el tipo de representación.	Desde la lección 26 a la 33.
OA i.	Buscar, seleccionar, manejar y producir información matemática/cuantitativa confiable a través de la web.	Desde la lección 34 a la 48.
Actitudes	Valorar las TIC como una oportunidad para informarse, investigar, socializar, comunicarse y participar como ciudadano.	Desde la lección 34 a la 48.
Actitudes	Pensar con perseverancia y proactividad para encontrar soluciones innovadoras a los problemas.	Desde la lección 34 a la 48.

Objetivos de Aprendizaje Pensamiento Computacional y programación 3° o 4° medio		Lecciones que cubren OA
Actitudes	Pensar con perseverancia y proactividad para encontrar soluciones innovadoras a los problemas.	Desde la lección 34 a la 48.
Actitudes	Aprovechar las herramientas disponibles para aprender y resolver problemas.	Desde la lección 34 a la 48.
OA 4.	Crear aplicaciones y realizar análisis mediante procesadores simbólicos, de geometría dinámica y de análisis estadístico.	Desde la lección 49 a la 73.
OA d.	Argumentar, utilizando lenguaje simbólico y diferentes representaciones, para justificar la veracidad o falsedad de una conjetura, y evaluar el alcance y los límites de los argumentos utilizados.	Desde la lección 59 a la 63.
OA g.	Elaborar representaciones, tanto en forma manual como digital, y justificar cómo una misma información puede ser utilizada según el tipo de representación.	Desde la lección 64 a la 73.
OA i.	Buscar, seleccionar, manejar y producir información matemática/cuantitativa confiable a través de la web.	Desde la lección 59 a la 63.
Actitudes	Pensar con conciencia, reconociendo que los errores ofrecen oportunidades para el aprendizaje.	Desde la lección 49 a la 73.
Actitudes	Interesarse por las posibilidades que ofrece la tecnología para el desarrollo intelectual, personal y social del individuo.	Desde la lección 49 a la 73.
Actitudes	Valorar las TIC como una oportunidad para informarse, investigar, socializar, comunicarse y participar como ciudadano.	Desde la lección 59 a la 63.

Objetivos de Aprendizaje Pensamiento Computacional y programación 3° o 4° medio		Lecciones que cubren OA
OA 5.	Desarrollar aplicaciones para dispositivos móviles y para dispositivos provistos de sensores y mecanismos de control.	Desde la lección 82 a la 98.
OA 6.	Utilizar la tecnología digital, y la información personal y privada que esta contiene, de una forma creativa, respetuosa y responsable.	Desde la lección 74 a la 81.
OA a.	Construir y evaluar estrategias de manera colaborativa al resolver problemas no rutinarios.	Desde la lección 82 a la 98.
OA d.	Argumentar, utilizando lenguaje simbólico y diferentes representaciones, para justificar la veracidad o falsedad de una conjetura, y evaluar el alcance y los límites de los argumentos utilizados.	Desde la lección 74 a la 81.
OA j.	Desarrollar un trabajo colaborativo en línea para discusión y resolución de tareas matemáticas, usando herramientas electrónicas de productividad, entornos virtuales y redes sociales.	Desde la lección 82 a la 98.
Actitudes	Trabajar colaborativamente en la generación, desarrollo y gestión de proyectos y la resolución de problemas, integrando las diferentes ideas y puntos de vista.	Desde la lección 82 a la 98.
Actitudes	Pensar con conciencia, reconociendo que los errores ofrecen oportunidades para el aprendizaje.	Desde la lección 74 a la 98.
Actitudes	Interesarse por las posibilidades que ofrece la tecnología para el desarrollo intelectual, personal y social del individuo.	Desde la lección 74 a la 98.

Unidad 01

¿Qué es el pensamiento computacional y la programación?

Contenido 1:

Introducción al pensamiento computacional

Contenido 2:

Creación de animaciones

Contenido 3:

Creación de videojuegos

Contenido 1

Introducción al Pensamiento Computacional

Resumen

- **Lección 1:** Programación con papel cuadriculado
- **Lección 2:** Introducción a los desafíos en línea
- **Lección 3:** Programación de relevos
- **Lección 4:** Depuración del código

Objetivos

- **OA 1.** Aplicar conceptos de Ciencias de la Computación –abstracción, organización lógica de datos, análisis de soluciones alternativas y generalización– al crear el código de una solución computacional.
- **OA a.** Construir y evaluar estrategias de manera colaborativa al resolver problemas no rutinarios.
- **OA g.** Elaborar representaciones, tanto en forma manual como digital, y justificar cómo una misma información puede ser utilizada según el tipo de representación.

Referencias

- **Code Studio**- Code.org
- **Cuantrix**- Fundación Televisa
- **CSTA**- Computer Science Teachers Association

Lección 1

Programación con papel cuadriculado

Propósito

Al “programarse” unos a otros para hacer dibujos, los estudiantes tendrán la oportunidad de experimentar los conceptos clave de la programación de una forma divertida y accesible.

Al principio de la clase, los estudiantes usarán símbolos para instruirse entre ellos a pintar cuadrados en un papel cuadriculado, a fin de reproducir una imagen existente. Si dispone de tiempo, la Lección puede terminar con imágenes creadas por los mismos estudiantes.

Los objetivos de esta Lección son desarrollar habilidades de pensamiento crítico, sembrar interés por este curso e introducir algunos conceptos fundamentales de la programación que serán usados a lo largo del curso. Al introducir conceptos básicos, como secuenciación o algoritmos, a través de una Lección sin conexión, incluso los estudiantes que no se sienten familiarizados con un computador podrán sentar las bases para comprender estos temas. En esta Lección, los estudiantes aprenderán cómo desarrollar un algoritmo y codificarlo en un programa.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (30 min)

Transferencia del conocimiento (15 min)

Evaluación (10 min)

Objetivos

Los estudiantes serán capaces de:

- Restructurar una secuencia de pasos en un programa codificado.

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los Profesores:

- Video – [Lección en acción: programación con papel cuadriculado](#) (no olvide activar los [subtítulos al español](#)).
- [Guía del profesor – Programación con papel cuadriculado](#).
- [Guía del profesor – Evaluación – Programación con papel cuadriculado](#).

Para los estudiantes:

- [Guía de trabajo – Programación con papel cuadriculado](#).
- Video – [Programación con papel cuadriculado](#).
- Evaluación – [Programación con papel cuadriculado](#).

Vocabulario

- **Algoritmo:** lista de pasos para realizar una tarea.

- Explicar las limitaciones de traducir problemas desde el lenguaje humano al lenguaje de las máquinas para facilitar el proceso de desarrollo del programa.

Preparación

- Vea el video [Lección en acción](#) en sección recursos para profesores (opcional).
- Imprima una guía de trabajo y una evaluación para cada estudiante.
- Asegúrese de que cada estudiante tenga su [Bitácora](#).

- **Programa:** algoritmo que ha sido codificado de forma que pueda ser ejecutado por una máquina.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Introducción a la programación con papel cuadriculado

En esta Lección, los estudiantes codificarán instrucciones para guiarse unos a otros a hacer dibujos, sin que el resto del grupo vea la imagen original. Esta sección contextualizará el ejercicio para la clase.

Mostrar: vean uno de los videos a continuación para contextualizar a los estudiantes sobre la clase de cosas que puede hacer un robot:

- [Robot Honda Asimo](#) (1:51).
- [Robot diseñador de huevos](#) (3:15).
- [Robot bailarín de Lego](#) (1:35).
- Análisis: ¿cómo creen que el robot sabe cómo hacer las cosas que hace? ¿los robots tienen un cerebro similar al nuestro?

Este breve análisis tiene por objetivo hacer énfasis en que los robots, a pesar de que pareciesen comportarse como humanos, realmente sólo responden a su programa. Es probable que los estudiantes hagan referencia a algunos robots de las películas y de la televisión con comportamientos más humanos. Guíelos a considerar robots que hayan visto u oído en la vida real, como el Roombas, o incluso asistentes digitales como Alexa o el asistente de Google.

Guíe el análisis hacia una conversación sobre cómo las personas programan a los robots para hacer cosas específicas, a través de comandos específicos.

Ampliación del conocimiento (30 min)

Práctica en conjunto

En esta Lección, los estudiantes tomarán los roles de programador y robot. En una hoja de papel cuadriculado, pintarán cuadrados de acuerdo con los programas que se hayan escrito los unos a los otros.

Distribuir: los estudiantes usarán una cuadrícula de 4x4 (u hojas de papel cuadriculado seccionado en cuadrículas de 4x4). También necesitarán una imagen de modelo.

Mostrar: proyecte estos comandos o escríbalos en la pizarra. No estarán mucho tiempo ahí, sólo lo suficiente para ayudar a los estudiantes a hacer la transición de algoritmos a programas.

[Moverse un cuadrado a la derecha](#)
[Moverse un cuadrado a la izquierda](#)
[Moverse un cuadrado arriba](#)

[Moverse un cuadrado abajo](#)
[Rellenar el cuadrado](#)

Generar una instancia de aprendizaje donde los programadores serán los estudiantes y el docente sigue las instrucciones al pie de la letra (como si fuese un robot). Luego, dividimos en grupos para que todos tengan su turno.

Modelar: muestre la imagen que usará de ejemplo y la cuadrícula en blanco que rellenará con su Sistema de Ejecución Automática (SEA). Asegúrese de que las instrucciones, la cuadrícula y la imagen permanezcan visibles al mismo tiempo.



Comentarios: Presentar un robot imaginario que funciona con un Sistema de Ejecución Automática (SEA). Esto significa que reacciona de forma automática a sus instrucciones, pero sólo a las que pueda entender.

Empiecen en la esquina superior izquierda. Guíen mi SEA diciéndome las instrucciones en voz alta.

Modelar: a continuación, la clase podría darle instrucciones como éstas. Cuando escuche una instrucción que pretenda seguir, asegúrese de repetir dicha instrucción en voz alta, de manera que los estudiantes puedan llevar un registro de sus movimientos.

[Moverse un cuadrado a la derecha](#)
[Rellenar el cuadrado](#)
[Moverse un cuadrado a la derecha](#)
[Moverse un cuadrado abajo](#)
[Rellenar el cuadrado](#)

Continúe con la Lección hasta completar la cuadrícula de ejemplo.

Captar: escriba cada uno de los comandos, de manera que los estudiantes puedan ver todos los pasos realizados para dibujar la imagen.

[Moverse un cuadrado a la derecha](#)
[Rellenar el cuadrado](#)
[Moverse un cuadrado a la derecha](#)
[Moverse un cuadrado abajo](#)
[Rellenar el cuadrado](#)

Recordar la definición de algoritmos. Como programadores podemos entenderlos fácilmente. PERO ¿qué pasa si queremos escribir el algoritmo para un dibujo como este?

Mostrar: muestre una imagen más complicada, como ésta.



A continuación, comience a escribir algunas de las instrucciones para replicar la imagen. Con suerte, los estudiantes verán que escribir todo a mano podría fácilmente volverse una pesadilla.

Mostrar: muestre esta lista de símbolos.

- Moverse un cuadrado a la derecha
- Rellenar el cuadrado
- Moverse un cuadrado a la derecha
- Moverse un cuadrado a la derecha
- Rellenar el cuadrado
- Moverse un cuadrado abajo
- Moverse un cuadrado a la izquierda
- Rellenar el cuadrado
- Moverse un cuadrado a la izquierda
- Moverse un cuadrado a la izquierda
- Rellenar el cuadrado
- ¡¡Y 12 instrucciones más!!

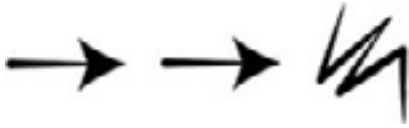
Análisis: ¿cómo podemos usar estos símbolos para facilitar nuestras instrucciones?



Continúe explorando ideas que apoyen la transición de instrucciones verbales a símbolos. Una vez que los estudiantes comprendan la idea, indique que el texto:

El objetivo de este análisis es llegar a la idea de que los estudiantes pueden usar símbolos para representar frases completas. Una vez que lo comprendan, coménteles que pasar de enlistar pasos detallados a codificarlos, se llama “programación”.

“Moverse un cuadrado a la derecha, moverse un cuadrado a la derecha, rellenar el cuadrado” Ahora corresponde al programa:

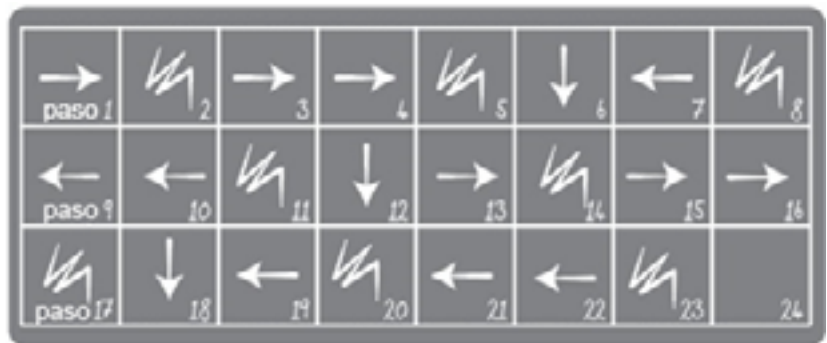
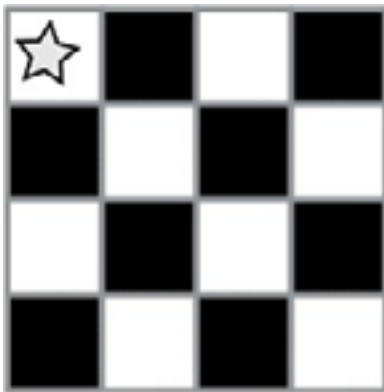


Modelar: ahora, que la clase le ayude a dibujar la imagen grande usando sólo símbolos. Por ahora, no se preocupe si se incluyen pasos innecesarios; si el programa funciona para recrear la imagen, considérela correcta.

En este punto, los estudiantes podrían estar emocionados y animados dando sugerencias. Si entienden lo esencial del ejercicio, este es un buen momento para analizar otras alternativas para rellenar la misma cuadrícula. Si aún no están listos, guarde esta idea para otro día y realice otro ejemplo.

Vea una solución de ejemplo a continuación:

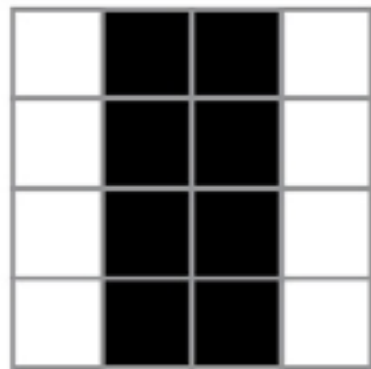
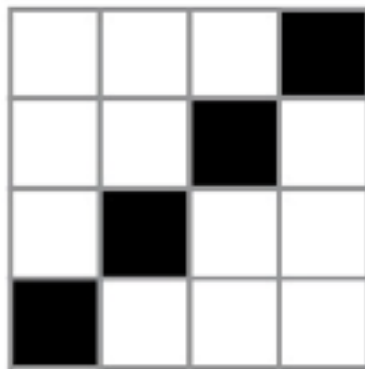
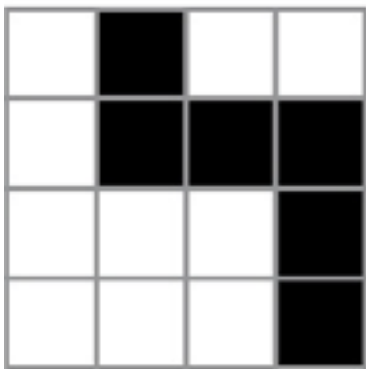
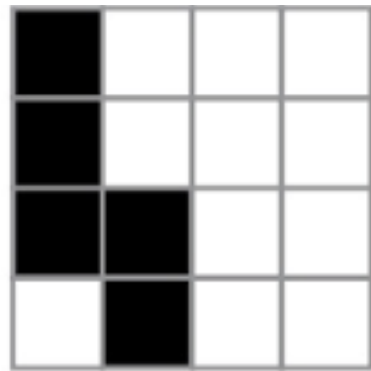
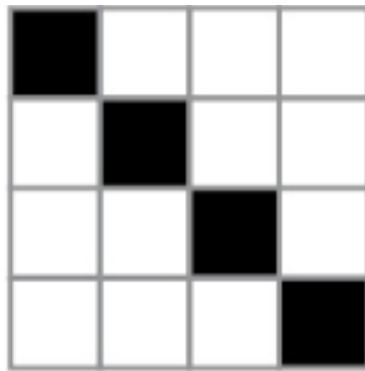
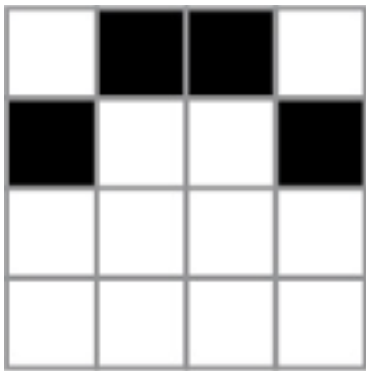
Note como hemos escrito el programa de izquierda a derecha, como se leería un libro en español. Algunos estudiantes prefieren este método, mientras que otros prefieren empezar cada línea de la cuadrícula en una nueva línea de la hoja. La forma en que escriban el programa no importa mucho, mientras los demás estudiantes puedan leer y seguir el programa.



Turno del estudiante

Grupo: divida a los estudiantes en parejas o pequeños grupos.

- Que cada grupo/pareja elija una imagen de la guía de trabajo.
- Discutan con sus compañeros el algoritmo necesario para dibujar la imagen elegida.
- Conviertan el algoritmo en un programa, usando los símbolos.
- Intercambien los programas con otros grupos/parejas, para que dibujen las imágenes de otros.
- Elijan otra imagen y ¡volvamos a empezar!



Transferencia del conocimiento (15 min)

Escribir en la bitácora y charla rápida

El acto de escribir en su bitácora sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para la bitácora:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?
- Dibuja otra imagen que podrías codificar. ¿Puedes escribir el programa que corresponde con la imagen?
- ¿Qué otro tipo de robots podríamos programar si cambiáramos lo que significan las flechas?

Evaluación (10 min)

- Entregue la Evaluación – Programación con papel cuadrículado. Luego de haber explicado claramente las instrucciones, permita que los estudiantes realicen la evaluación de forma individual.

- Gracias a la Lección previa, esto no debería significar ningún problema para ellos.

Experiencias de aprendizaje de profundización

Use estos Contenidos para ampliar el aprendizaje de los estudiantes. Se pueden usar como Contenidos extras fuera del aula.

Cada vez mejor

- Que su clase intente hacer sus propias imágenes.
- ¿Pueden descifrar cómo codificar las imágenes que acaban de crear?

Desafío de curso

- Dibuje una imagen en una cuadrícula de 5x5 en lugar de una 4x4.

Lección 2

Introducción a los desafíos en línea

Propósito

En este conjunto de desafíos, los estudiantes comenzarán con una introducción (o repaso, dependiendo de la experiencia de su clase) del espacio de trabajo en línea de Code.org. Habrá videos indicando las funciones básicas del espacio de trabajo, como los botones ejecutar (run), reiniciar (reset) y paso (step). En estos videos también se aborda cómo arrastrar, borrar y conectar bloques Blockly. Luego, sus estudiantes pondrán en práctica sus habilidades de secuenciación y depuración en un laberinto. A partir de ahí, los estudiantes verán nuevos tipos de desafíos, como la recolectora, el artista o la cosechadora, mientras aprenden lo básico de los bucles.

Comprendemos que en cada sala de clases hay un espectro de comprensión diferente para cada tema. Algunos estudiantes pueden ser muy hábiles con los computadores mientras que otros pueden tener muy poca experiencia con ellos. Con el objetivo de nivelar el área de juego (y de aprendizaje), hemos desarrollado esta “etapa de reforzamiento” para el curso D. Esto puede ser tanto una introducción como un repaso de cómo usar Code.org y conceptos básicos de las Ciencias de la Computación.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Lección puente: programación (10 min)

Ampliación del conocimiento (30 min)

Transferencia del conocimiento (10 min)

Objetivos

Los estudiantes serán capaces de:

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- Video – [Pair programming.](#)
- Puntos clave – [Programación en parejas.](#)

Vocabulario

- **Bug (error):** parte de un programa que no funciona correctamente.
- **Depurar (debugging):** encontrar y solucionar los problemas en un algoritmo o programa.
- **Bucle (loop):** la acción de hacer algo una y otra vez.
- **Programa:** algoritmo que ha sido codificado de forma que pueda ser ejecutado por una máquina.
- **Programación:** el arte de crear un programa.

- Ordenar comandos de movimientos como pasos secuenciales de un programa.
- Modificar un programa existente para reparar errores.
- Descomponer una secuencia larga de instrucciones en una secuencia de repetición más corta.

Preparación

- Realice los desafíos para encontrar cualquier área potencialmente problemática para su clase.
- Asegúrese de que cada estudiante tenga su [Bitácora](#).

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Introducción

Los estudiantes aprenderán muchos conceptos nuevos o repasarán muchos conceptos básicos. En base a la experiencia de su clase, puede revisar las palabras programa, programación, bug, depurar, ciclo o bucle en el glosario del curso o pasar directamente a la Lección puente. Si las definiciones no están cubiertas explícitamente, recomendamos usar las palabras en oraciones.

Lección puente: programación (10 min.)

Esta Lección ayudará a sus estudiantes a traer los conceptos sin conexión de “programación con papel cuadriculado” al mundo digital al cual se están sumergiendo.

Revisión de desafíos en línea en conjunto

Elija un desafío de la Lección. Para esta Lección recomendamos progresar con [estos desafíos](#). Divida a los estudiantes en grupos de tres o cuatro. Pídales que programen a “Red”, de Angry Birds, para que llegue al cerdo, usando las flechas de “programación con papel cuadriculado”.

La clase no necesitará usar el último símbolo.



Una vez que toda la clase tenga una respuesta, discutan el camino a seguir en conjunto como clase.

Ampliación del conocimiento (30 min.)

Desafíos en línea

Los profesores son de vital importancia en la educación de las Ciencias de la Computación y juegan un rol fundamental para producir un ambiente vibrante y colaborativo en la sala de clases. Durante las actividades en línea, el rol del profesor es alentar y apoyar. Los desafíos en línea están estructurados para estar centrados en el estudiante, así que los profesores deben evitar involucrarse cuando los alumnos tengan problemas para resolverlos. Algunas ideas de cómo hacerlo son:

- Use la programación en parejas cada vez que sea posible durante la Lección.
- A través de desafíos o preguntas, anime a los estudiantes para que busquen respuestas con sus respectivas parejas.
- Las preguntas sin responder pueden ser delegadas a un grupo cercano, que podría ya tener la

respuesta.

- Recuérdeles usar el proceso de depuración antes de que usted se acerque a ayudar.
- Pida a los estudiantes que describan el problema que estén viendo. ¿Qué se supone que debe hacer?, ¿qué hace?, ¿qué te dice eso?
- Recuérdeles que la frustración es un paso en el camino del aprendizaje y que la perseverancia dará sus frutos.
- Si un estudiante sigue con problemas para avanzar después de todo esto, haga preguntas clave para que los estudiantes identifiquen el bug (error) por ellos mismos.

Lección en Code Studio

Acceden a la plataforma para realizar los desafíos en línea.

Transferencia del conocimiento (10 min)

Escribir en la bitácora

El acto de escribir en su bitácora sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para la bitácora:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?
- Haz una lista de los bugs que encontraste en tus programas hoy.
- ¿Cuál fue tu desafío favorito? Dibuja a tu personaje favorito completando desafíos.

Sugerencias de Evaluación

Se sugiere evaluar formativamente los aprendizajes:

- Resuelven problemas
- Asignan roles en los grupos
- Dividen un problema en otros menores

Enséñeles a los estudiantes la manera correcta de trabajar en parejas:

- No usar el teclado del compañero.
- No tocar el mouse del compañero.
- Asegúrate que tu compañero pueda describirte la solución en voz alta antes de que vayas.

Lección 3

Programación de relevos

Propósito

Esta Lección comenzará con una breve Lección de depuración y perseverancia. Luego, rápidamente se transformará en una carrera contra el tiempo, mientras los estudiantes se dividen en grupos y trabajan en equipo para escribir un programa, una instrucción a la vez.

El trabajo en equipo es muy importante en la computación. Los equipos escriben y depuran códigos en conjunto, en lugar de trabajar de forma individual. Los estudiantes aprenderán a trabajar en equipo intentando ser lo más eficientes posible.

Esta Lección también involucra el concepto de urgencia, lo cual enseñará a los estudiantes que deben administrar su tiempo cuidadosamente, evitando cometer errores sin comprometer mucho tiempo. Esta experiencia puede ser estresante (¡y lo será!). Asegúrese de compartir con los estudiantes las herramientas para lidiar con la frustración.

Secuencia para el aprendizaje

Conocimiento inicial (15 min)

Ampliación del conocimiento (20 min)

Transferencia del conocimiento (15 min)

Objetivos

Los estudiantes serán capaces de:

- Definir ideas usando código y símbolos.
- Verificar el trabajo hecho por compañeros.
- Identificar las señales de la frustración.

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- Guía del profesor – [Programación de relevos.](#)
- Guía de depuración para el profesor – [Programación de relevos.](#)
- Video – [Programación de relevos.](#)

Para los estudiantes:

- Guía de trabajo – [Programación de relevos.](#)
- Video – [Programación de relevos.](#)
- Paquete de actividades – [Programación de relevos.](#)

Vocabulario

- **Algoritmo:** lista de pasos para realizar una tarea.
- **Bug (error):** parte de un programa que no funciona correctamente.

Preparación

- Vea el Video – Programación de relevos.
- Encuentre un espacio abierto para esta Lección, como el gimnasio o un área verde.
- Imprima un Paquete de actividades – Programación de relevos para cada grupo.
- Provea a cada grupo hojas y lápices.
- Imprima una Guía de trabajo – Programación de relevos para cada estudiante.
- Asegúrese de que cada estudiante tenga su [Bitácora](#).

- **Depurar (debugging):** encontrar y solucionar los problemas en un algoritmo o programa.
- **Frustración:** sentirse molesto o enojado porque algo no funciona como quieres.
- **Perseverancia:** intentarlo una y otra vez, incluso cuando algo es muy difícil.
- **Programa:** algoritmo que ha sido codificado de forma que pueda ser ejecutado por una máquina.

Estrategia de aprendizaje

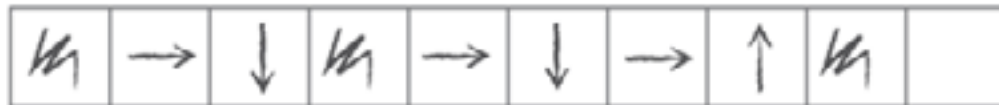
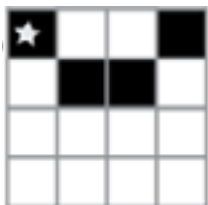
Conocimiento inicial (15 min)

Recordemos que en “programación con papel cuadriculado” guiamos el Sistema de Ejecución Automática (SEA) de nuestros compañeros usando flechas. Esta sección trae de vuelta esas ideas, las cuales serán necesarias en la Ampliación del conocimiento.

¿Dónde me equivoqué?

Objetivo: en esta Lección, queremos ayudar a los estudiantes a identificar y resolver bugs en sus programas. La forma más sencilla de hacerlo es trabajar en un programa que ya contenga bugs, pero que no sean culpa de los estudiantes. Una vez que le hayan ayudado a reparar “su” programa, comparta con ellos lo frustrante que puede llegar a ser cometer errores, y ayúdeles a ver que esos sentimientos son absolutamente normales, no deben sentirse avergonzados por ellos.

Mostrar: Muestre la imagen de la Guía de depuración para el profesor – Programación de relevos.



Análisis: Presente la imagen e indique que tiene un bug ¿pueden identificarlo?

Tómese un momento para repasar las reglas: comenzar en la estrella, seguir las instrucciones paso a paso, terminar cuando se hayan ejecutado todos los cuadrados de la derecha.

Reflexión: ¿pueden descifrar por qué el programa no funciona?

Interacción: pida a los estudiantes que trabajen en equipo para ver si pueden descifrar cómo debería estar escrito el programa.

Exposición: pregunte si alguien encontró una manera de resolver el problema. Cuando le den una respuesta correcta, recuerde la definición de depuración

Sugerencias para lidiar con la frustración:

- Contar hasta 10
- Respirar hondo
- Escribir acerca del problema
- Hablar con algún amigo sobre el problema
- Pedir ayuda

Discusión: En la programación muchas veces nos vamos a ver enfrentados a la frustración, donde quizás te veas tentado a rendirte; sin embargo, la frustración es un sentimiento normal y es un gran indicio de que estás a punto de aprender algo. En lugar de rendirte, intenta ser perseverante. Sigue intentándolo una y otra vez. Después de algunos intentos, ¡empezarás a entender cómo depurar tus problemas!

Distribuir: para asegurarse de que los estudiantes entienden la idea de encontrar y solucionar problemas (depurar), distribuya la Guía de trabajo – programación de relevos y que trabajen en parejas.

Opcional: Si no quiere invertir mucho tiempo en esta Lección, pueden realizarla en conjunto como clase.

Transición: ¡llegó la hora de la acción!

Ampliación del conocimiento (20 min)

Programación de relevos

Con los conceptos de “programación con papel cuadriculado” en mente, ¡es momento de dividirse en equipos y prepararse para la Lección en relevos!

Preparación: prepárese imprimiendo el Paquete de actividades – Programación de relevos para cada equipo, de 4 a 5 estudiantes. Corte o doble cada página por la línea punteada.

Revise las reglas del juego con su clase:

- Dividirse en equipos de 4-5 estudiantes.
- Que cada grupo haga una fila, uno detrás de otro.
- Colocar una de las imágenes guía al otro lado de la sala/gimnasio/área para cada equipo (use la misma imagen para todos los equipos).
- El primer estudiante de la fila debe correr hasta la imagen, revisarla, y escribir el primer símbolo del programa para reproducir esa imagen.
- El primer estudiante debe volver y tocar al siguiente de la fila. Luego, debe ponerse al final de la fila.
- La siguiente persona en la fila debe correr hasta la imagen y revisarla. Luego, debe revisar el programa que se ha escrito y realizar sólo una de estas dos opciones: depurar el programa

Sugerencias para ser perseverante:

- Llevar registro de lo que ya has intentado
- Describir qué es lo que pasa
- Describir qué debería hacer
- ¿Qué te dice eso?
- Haz un cambio y vuelve a intentar

Algunas cosas que deben ser aclaradas y recordadas de vez en cuando:

- Sólo una persona de cada grupo puede estar frente a la imagen a la vez.
- Está permitido discutir el algoritmo con el resto del grupo mientras están en la fila, incluso pueden planificar quién y qué van a escribir en el programa.
- Cuando un estudiante depura el programa tachando una instrucción incorrecta (o un conjunto de instrucciones incorrectas), cuenta como su turno. El siguiente estudiante debe descifrar cómo corregir las instrucciones eliminadas.

tachando los símbolos incorrectos o agregar un símbolo nuevo. El estudiante luego corre de vuelta y toca al siguiente en la fila para que sea su turno.

- Repetir el punto anterior hasta que un equipo termine el programa.

¡El primer grupo que consiga un programa que calce con la imagen es el ganador! Repita esta Lección varias veces, aumentando la dificultad.

Repita el juego tantas veces como pueda, hasta que se agote el tiempo o hasta que los estudiantes se sientan fatigados.

Transición: una vez que el juego acabe, reúna a todos los estudiantes en un círculo para compartir lo aprendido.

Discusión: ¿qué aprendimos hoy?

- ¿Qué tal si cada persona en la fila pudiese hacer 5 flechas a la vez?
- ¿Qué tan importante puede ser depurar nuestro propio trabajo y el trabajo del programador anterior a nosotros?
- ¿Y si pudiesen hacerse 10 flechas?
- ¿Y si fuesen 10.000? ¿Sería más o menos importante?
- ¿Crees que un programa es mejor o peor cuando más de una persona trabaja en él?
- ¿Crees que las personas comenten más o menos errores cuando están apurados?
- Si encuentras un error ¿debes deshacer todo el programa y comenzar desde cero?

Transferencia del conocimiento (15 min)

Escribir en la bitácora

El acto de escribir en su bitácora sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para la bitácora:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?
- ¿Cómo es que el trabajo en equipo influyó en el éxito de escribir el programa de hoy?
- ¿Te sentiste frustrado en algún punto? ¿qué hiciste respecto a eso?

Experiencias de aprendizaje de profundización

Use estos contenidos para ampliar el aprendizaje de los estudiantes. Se pueden usar como Contenidos extras fuera del aula.

Pasa la hoja

- Si no tiene tiempo o espacio para una Lección de relevos al aire libre, puede reunir a los estudiantes en grupos y que pasen la hoja con el programa de un puesto a otro. Que cada estudiante haga una flecha antes de pasar la hoja a un compañero.

Llévalo, muévelo

- Cómo profesor, dibuje una imagen con tantos cuadrados pintados como niños haya en cada grupo.
- Pida a los estudiantes que anoten tantas flechas en el programa como sea necesario para llegar a un cuadrado pintado, incluyendo la acción de pintar ese cuadrado, antes de pasar la hoja al siguiente estudiante.

Depurando juntos

Dibuje una imagen en la pizarra. Pídale a cada estudiante crear un programa para esa imagen. Luego, que intercambien sus programas con sus compañeros de al lado y depuren sus códigos.

- Encierra en un círculo el primer paso incorrecto, luego devuelve el programa.
- Deles a los estudiantes otra oportunidad para revisar y depurar sus propios programas.
- Pida que un voluntario comparta su programa.
- Pregunte:
- ¿Cuántos tienen el mismo programa?
- ¿Alguien tiene algo diferente?

Sugerencias de evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Identifican las partes que componen una situación.
- Elaboran instrucciones.

Lección 4

Depuración con Laurel

Propósito

En esta Lección en línea, los estudiantes practicarán cómo depurar en un ambiente de “recolección”, al practicar cómo leer y editar códigos para reparar desafíos con algoritmos simples, bucles y bucles anidados.

El propósito de esta Lección es enseñar a los estudiantes que fallar es normal cuando se estamos aprendiendo nuevas habilidades. A los estudiantes se les darán programas que NO funcionan correctamente y se les pedirá repararlos. Este proceso, llamado “depuración”, les ayudará a desarrollar habilidades de pensamiento crítico y resolución de problemas, habilidades que los acompañarán mientras avanzan a proyectos de programación cada vez más complejos.

Secuencia para el aprendizaje

Conocimiento inicial (15 min)

Lección puente: depuración (15 min)

Ampliación del conocimiento (30 min)

Transferencia del conocimiento (15 min)

Objetivos

Los estudiantes serán capaces de:

- Leer y comprender un código dado.
- Identificar un bug y los problemas que causa en un programa.
- Describir e implementar un plan para depurar un programa.

Preparación

- Realice los desafíos para encontrar cualquier área potencialmente problemática para su clase.
- Asegúrese de que cada estudiante tenga su [Bitácora](#).

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- Act. Previa. Paquete de actividades – [Programación de relevos](#).
- Manipulativos – [Bloques sin conexión \(cursos C- F\)](#).

Vocabulario

- **Bug (error):** parte de un programa que no funciona correctamente.
- **Depurar (debugging):** encontrar y solucionar los problemas en un algoritmo o programa.

Estrategia de aprendizaje

Conocimiento inicial (15 min)

Introducción

Una de las partes más importantes en el aprendizaje de la programación es aprender a depurar. Pregunte a la clase si alguna vez han aprendido una habilidad nueva y si alguna vez han fallado.

Por ejemplo:

- Aprender a andar en bicicleta y caerse.
- Aprender a hornear y quemar la comida.
- Aprender algún deporte y no ganar ningún juego.

Equivocarse es muy común cuando se aprenden cosas nuevas. Pida a los estudiantes que discutan sobre situaciones en que se hayan equivocado y cómo las superaron.

En la programación, los programadores normalmente encuentran “bugs” en sus códigos.

- Bug (error): parte de un programa que no funciona correctamente.
- Un bug puede realmente arruinar un programa, por lo que es importante aprender a “depurar” el código.
- Depurar (debugging): encontrar y solucionar los problemas en un algoritmo o programa.
- Entretenga la conversación si cree que su clase necesita una mayor introducción, pero no olvide dejar tiempo para una de las actividades puente.

Lección puente: depuración (15 min)

Estas actividades ayudarán a sus estudiantes a traer los conceptos sin conexión de “depuración sin conexión: programación de relevos” al mundo digital al cual se están sumergiendo. Elija una de las siguientes actividades:

Lección sin conexión: bloques de papel

Divida a la clase en equipos de 3 a 5 estudiantes y encuentre un lugar amplio, puede ser un gimnasio o un área verde. Ordene los equipos en fila, similar a como se hizo en la Lección “programación de relevos”. Tome un diseño mediadamente difícil del Paquete de actividades – Programación de relevos. A una distancia considerable, coloque una imagen por cada equipo. Además, entrégueles suficientes bloques de papel de los Manipulativos – Bloques sin conexión (cursos C- F). Cada equipo necesitará muchos bloques llena 1 y mover _____. Los bloques mover _____ pueden ser rellenos con anticipación o durante el mismo juego. De cualquier modo, asegúrese de que los bloques estén bien definidos mientras se desarrolle el juego.

Una vez que todos los equipos estén formados, muestre o lea en voz alta las reglas:

- El primer estudiante en la fila debe correr hacia la imagen, revisarla, y colocar el primer bloque de código en el programa para reproducir esa imagen.
- El primer estudiante debe volver y tocar al siguiente de la fila. Luego, debe ponerse al final de la fila.
- La siguiente persona en la fila debe correr hasta la imagen y revisarla. Luego, debe revisar el programa que se ha escrito y hacer una de dos: depurar el programa quitando los bloques de código incorrectos o agregar un bloque nuevo. Luego, el estudiante debe correr de vuelta y toca al siguiente en la fila para que sea su turno.
- Repetir el punto anterior hasta que un equipo termine el programa.

Asegúrese de que los estudiantes usen sólo los bloques llena 1 y mover ____ y que sólo coloquen un bloque por turno. ¡El primer equipo que escriba correctamente el código de su imagen, gana!

Revisión de desafíos en línea en conjunto

Agrupe a los estudiantes en grupos de 3. Elija un desafío de la Lección, recomendamos el número 5. Pídales que se sienten frente a un computador con el desafío en pantalla. Cada equipo puede tener sólo un computador y un estudiante mirando la pantalla. Muestre o lea las reglas:

- Sólo un estudiante por equipo puede estar mirando la pantalla.
- Esa persona sólo puede añadir o borrar un bloque a la vez. Cuando esa persona haya añadido o eliminado un bloque, puede tocar el hombro de un compañero para que sea su turno.
- El siguiente estudiante puede ponerse frente a la pantalla y jugar su turno.
- Los turnos no pueden repetirse ni saltarse, todos deben jugar la misma cantidad de veces.

¡El primer equipo en resolver el desafío correctamente, gana!

Ampliación del conocimiento (30 min)

Desafíos en línea

Podría ser muy útil que los integrantes de los equipos de la Lección puente se sienten cerca unos de otros. Cada estudiante debe realizar los desafíos de forma individual o en parejas; sin embargo, contar con un grupo de trabajo conocido para hacer y responder preguntas puede ayudarlos a sentir más confianza y a comprender mejor el tema.

Lección en Code Studio

Acceder a Code Studio para realizar los desafíos de depuración.

Transferencia del conocimiento (15 min)Escribir en la bitácora

El acto de escribir en su bitácora sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para la bitácora:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?
- ¿Qué es un bug? ¿cómo sabes que hay un bug en tu programa?
- ¿Qué significa “depurar” un código? ¿cómo depuras un programa?

Sugerencia para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Comprenden y analizan un problema.
- Piensan en la solución.
- Buscan alternativas; es decir, preparan los elementos para que desarrollen el programa y resuelvan el problema.

Contenido 2

Creación de animaciones

Resumen

- **Lección 5:** Programación para el entretenimiento
- **Lección 6:** Trazar formas
- **Lección 7:** Dibujar en Game Lab
- **Lección 8:** Formas y aleatorio
- **Lección 9:** Variables
- **Lección 10:** Sprites
- **Lección 11:** Ciclo de dibujo aleatorio
- **Lección 12:** Patrón de contador desconectado
- **Lección 13:** Movimiento de Sprite
- **Lección 14:** Booleanos desconectados
- **Lección 15:** Booleanos y condicionales
- **Lección 16:** Condicionales y entrada del usuario
- **Lección 17:** Otras formas de entrada
- **Lección 18:** Proyecto- Tarjeta interactiva

Objetivos

- **OA 1.** Aplicar conceptos de Ciencias de la Computación –abstracción, organización lógica de datos, análisis de soluciones alternativas y generalización– al crear el código de una solución computacional.
- **OA a.** Construir y evaluar estrategias de manera colaborativa al resolver problemas no rutinarios.
- **OA d.** Argumentar, utilizando lenguaje simbólico y diferentes representaciones para justificar la veracidad o falsedad de una conjetura, y evaluar el alcance y los límites de los argumentos utilizados.

Referencias

- **Code Studio**- Code.org
- **Cuantrix**- Fundación Televisa
- **CSTA**- Computer Science Teachers Association

Lección 5

Diseñador de Minecraft

Propósito

La hora del código con Minecraft ayuda a que todos los estudiantes puedan aprender las valiosas habilidades adquiridas a través de la enseñanza de las Ciencias de la Computación. Los estudiantes participarán una hora donde experimentarán la programación en un ambiente divertido y libre de estrés, aprenderán conceptos clave de la computadora y los videojuegos.

Los estudiantes aprenderán Ciencias de la Computación a través del taller: La hora del código con Minecraft, dónde diseñarán un juego Minecraft teniendo un acercamiento a conceptos que utilizarán a lo largo del curso.

Secuencia para el aprendizaje

Ampliación del conocimiento (55 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Entender conceptos básicos que conforman el diseño de un videojuego, de una manera divertida: estructura de código por bloques, comandos, ciclos, comandos aleatorios, eventos, agregar elementos y puntuación.

Preparación

- Computadoras con conectividad.
- Realice las actividades para encontrar cualquier área potencialmente problemática para su clase

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

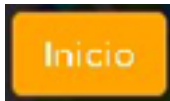
- [Hora del código Minecraft.](#)

Estrategia de aprendizaje

Ampliación del conocimiento (55 min)

Se guiará a los estudiantes al curso, entrando al siguiente link- [hora del código Minecraft](#)- e identificando el curso mostrado en las imágenes a continuación:

Posterior iniciarán el curso dando clic en:



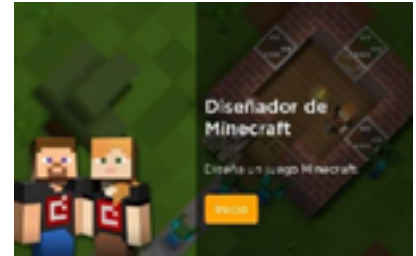
Ahora como clase, podrán explorar el taller.

Diseñador de Minecraft

Grupos: forma equipos de 2-3 personas, según la capacidad del aula.

Explica: se explicará a los estudiantes sobre los elementos que conforman un videojuego, pensarán en sus videojuegos favoritos y se les preguntará ¿qué contienen? ¿qué observan? ¿cómo creen que funcionan?

Como clase observarán el video de introducción:



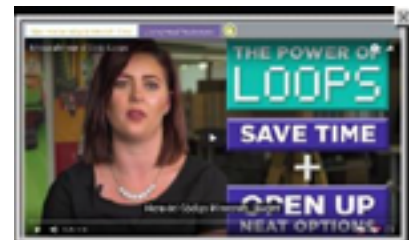
Los equipos deberán seguir avanzando en las diversas etapas y actividades del taller, además podrán seleccionar el personaje que más les guste para la Lección.



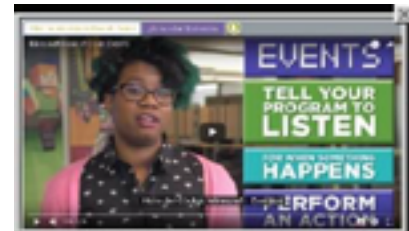
Conocerán bloques de programación que deberán aprender a utilizar para resolver diversos retos.



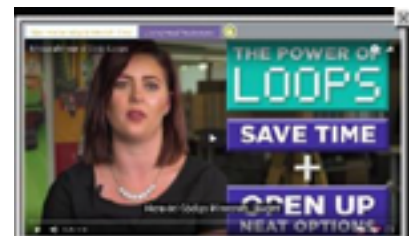
Como clase observarán el video que permitirá a los equipos aprender a utilizar ciclos de programación, que harán sus programas más eficientes, lo aprendido lo aplicarán en los ejercicios de programación.



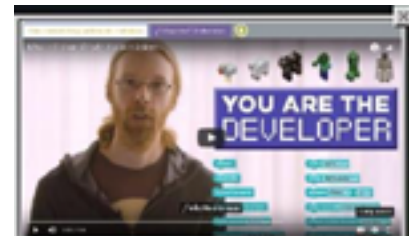
Como clase observarán el video, después los equipos harán uso de eventos que les permitirán incluir eventos dependientes de una acción en su videojuego. Lo aprendido lo aplicarán en los ejercicios de programación.



Como clase observarán el video para generar criaturas, lo aprendido lo aplicarán en los ejercicios de programación.



Como clase observarán el video de felicitaciones, en este momento habrán concluido su primer acercamiento al desarrollo de videojuegos.

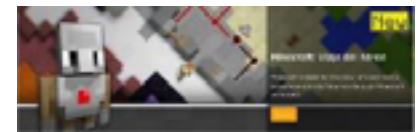


Transferencia del conocimiento (5 min)Reflexión

Pregunta a los estudiantes: ¿qué fue lo que más les gustó? y enlisten en sus cuadernos los elementos que utilizaron a lo largo del taller.

Los conceptos aprendidos los utilizarán a lo largo del curso, a mayor profundidad.

Si cuentas con más tiempo o tu clase está muy interesada en este tema pueden realizar más actividades de la hora del código con Minecraft.

**Sugerencias para evaluación**

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Desarrollan un programa con elementos o fases más pequeñas.
- En una conversación pueden dar un ejemplo de aplicación de elementos como: secuencias, eventos, bucles y condicionales

Lección 6

Trazar formas

Propósito

Los estudiantes exploran los desafíos de comunicar cómo dibujar con formas y usar una herramienta que presente cómo se aborda este problema en Game Lab. La Lección de calentamiento demuestra rápidamente los desafíos de la posición de comunicación sin algún punto de referencia compartido. En la Ampliación del conocimiento, los estudiantes exploran una herramienta de Game Lab que les permite a los estudiantes colocar formas interactivamente en la cuadrícula de 400 por 400 de Game Lab. Luego se turnan para instruir a un compañero sobre cómo dibujar una imagen oculta con esta herramienta, lo que representa muchos desafíos que los estudiantes enfrentarán al programar en Game Lab. Los estudiantes opcionalmente crean su propia imagen para comunicarse antes de una discusión de informe.

El objetivo principal de esta Lección es presentar a los estudiantes el sistema de coordenadas que usarán en Game Lab. Los estudiantes pueden tener experiencia limitada con el uso de una cuadrícula de coordenadas o pueden tener dificultades con el eje Y “invertido” en Game Lab. La herramienta de dibujo también obliga a los estudiantes a pensar en otras características de Game Lab que verán cuando comiencen a programar en la próxima Lección. Estos incluyen la necesidad de tener en cuenta el orden al dibujar, la necesidad de especificar el color y el hecho de que los círculos se colocan por su centro y cuadrados por la esquina superior izquierda. Al final de esta Lección, los estudiantes deben estar listos para transferir lo que han aprendido sobre la comunicación de posición a la programación que harán en la próxima Lección.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (35 min)

Transferencia del conocimiento (5 min)

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- Trazar formas- [Guía de actividades.](#)
- Trazar formas - [Enlace code.org.](#)

Objetivos

Los estudiantes serán capaces de:

- Manejar las ubicaciones en la cuadrícula de coordenadas de Game Lab.
- Comunicar cómo dibujar una imagen en Game Lab, teniendo en cuenta la posición de la forma, el color y el orden.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Comunicar información de dibujo

Observaciones: Vimos muchos programas diferentes ayer, y comenzó a pensar qué tipos de programas querría crear. Cuando creamos un programa, una de las cosas que debemos hacer es dibujar todo en la pantalla. Hoy, vamos a probar eso con una computadora “estudiantil”.

Pida a uno o dos voluntarios que pasen al frente del salón y actúen como “computadoras” para la Lección. Deben sentarse de espaldas para que no puedan ver lo que se proyecta. Entregue a cada voluntario una hoja de papel en blanco.

Visualización: proyecte un dibujo de muestra donde pueda ser visto por la clase.

Observaciones: Deberá explicarle a nuestra “computadora” cómo dibujar la imagen. Al final, compararemos el dibujo con la imagen real. Dé un minuto o dos a los estudiantes para describir el dibujo, mientras que los estudiantes al frente del salón tratan de dibujarlo. Después de un minuto, deténgalos y permítales que los estudiantes comparen ambas imágenes.

Pregunte: ¿cuáles son los diferentes “desafíos” o problemas que vamos a necesitar resolver para comunicar con éxito este tipo de dibujos?

Discuta:

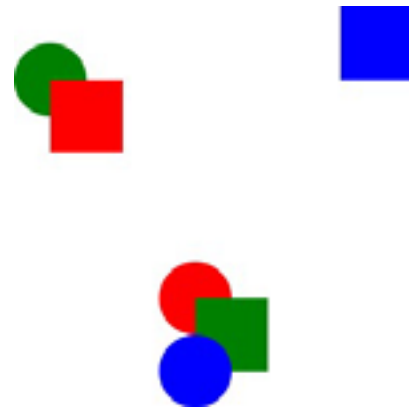
- Los estudiantes deben escribir silenciosamente sus respuestas en sus cuadernos.
- Luego deben hablar con un compañero y finalmente con toda la clase.

Observaciones: hubo varios desafíos que necesitábamos resolver en esta Lección. Necesitamos poder comunicar claramente la posición, el color y el orden de las formas. Vamos a comenzar a explorar cómo resolver este problema.

Ampliación del conocimiento (35 min)

Dibujando con una computadora

Grupo: Agrupe a los estudiantes en parejas.



Meta: Esta discusión tiene la intención de mencionar algunos desafíos que los estudiantes necesitarán abordar en las próximas lecciones, tales como especificar la posición, el orden y el color. Los estudiantes no necesitan decidir cómo especificarán tales cosas, pero reconocen que necesitarán un método para hacerlo.

Meta: En lugar de hacer una demostración en vivo de la herramienta, use esta estrategia para que los estudiantes exploren la herramienta ellos mismos. Luego use la información para asegurarse de que todos los estudiantes conozcan las características clave de la herramienta. Los componentes más importantes son la cuadrícula y el hecho de que las coordenadas del mouse se muestran debajo del espacio de dibujo.

Transición: haga que un miembro de cada grupo tome una computadora y vaya a los contenidos de esta Lección. Hay un solo nivel con una herramienta de Game Lab.

Indicación: trabaje con su pareja toma dos o tres minutos para descubrir cómo funciona esta herramienta. Luego, prepárese para compartir como clase.

Discuta: después de que las parejas hayan tenido la oportunidad de trabajar con la herramienta, ejecute un reparto rápido donde los estudiantes discutan las características que notan.

Dibujando formas

Distribuya [Trazar formas- Guía de actividades](#) a cada par, asegurándose de que un alumno (Estudiante A) reciba las dos primeras páginas y que el otro alumno (Estudiante B) reciba las dos páginas siguientes. Los estudiantes no deben mirar los papeles de los demás.

Configuración: En esta Lección, los estudiantes intentarán recrear las imágenes en función de las indicaciones de un compañero. El estudiante que está ilustrando usará la herramienta de dibujo de formas en Game Lab para realizar las formas. Los estudiantes deben mantener sus dibujos ocultos el uno del otro durante toda la Lección. Al completar un dibujo, el instructor tampoco debería poder ver la pantalla de la computadora.

Dibujo 1: Cada miembro del par debe completar su primer dibujo, turnarse para dar instrucciones y usar la herramienta. Estos dibujos no presentan formas superpuestas, pero los estudiantes pueden tener que lidiar con el hecho de que los círculos se dibujan desde la mitad y los cuadrados desde la esquina superior izquierda. Además, los estudiantes pueden simplemente luchar con la dirección del eje Y.

Discuta: da un par minutos a las parejas para discutir cualquier problema común que estén notando al tratar de completar sus dibujos.

Dibujo 2: Cada miembro del par debe describir su segundo dibujo a su compañero. Estos dibujos tienen formas superpuestas y los estudiantes tendrán que considerar el orden en que se colocan las formas y cuándo deben cambiar el color del lápiz.

Dibuje el suyo: Si el tiempo lo permite, brinde a los estudiantes la oportunidad de crear su propio dibujo para comunicarse con su compañero.

Observaciones: Cuando hacemos imágenes, necesitamos una

Ubicación del origen: El origen de esta cuadrícula, así como el origen en Game Lab, se encuentra en la esquina superior izquierda. Esto refleja el hecho de que los documentos tienden a comenzar en la parte superior izquierda y aseguran que cada punto del avión tenga coordenadas positivas.

forma de comunicar exactamente dónde va cada forma. El plano de coordenadas nos ayuda a hacer eso. Nuestro plano de coordenadas tiene dos coordenadas, “x” y “y”. La coordenada “x” nos dice qué tan lejos está nuestra forma de la izquierda de la cuadrícula. La coordenada “y” nos dice cuán lejos está nuestra forma desde la parte superior de la cuadrícula. Los puntos negros en las formas lo ayudan a ser muy específico sobre cómo se coloca la forma en la cuadrícula.

Transferencia del conocimiento (5 min)

Reflexión

Haga que los estudiantes reflexionen sobre cada uno de los siguientes mensajes

- ¿Qué problema está ayudando a resolver la cuadrícula en Game Lab?
- ¿Has visto formas diferentes de resolver este problema en el pasado? ¿Qué son?

Discuta: haga que las parejas compartan sus respuestas entre sí. Luego abra la discusión a toda la clase.

Observaciones: Al comienzo de la clase vimos que comunicar cómo dibujar incluso formas simples puede ser bastante desafiante. Lo que aprendimos hoy es una solución a este problema, pero hay muchas otras que podrían haber funcionado. De hecho, muchos de ustedes probablemente se dieron cuenta de que la cuadrícula en Game Lab está “volteada”. Las pantallas de las computadoras tienen diferentes formas y tamaños, al igual que el contenido que mostramos en ellas. Tenemos que acordar un punto desde el cual pueda crecer todo el contenido. Como leemos desde la esquina superior izquierda, la cuadrícula en la pantalla de una computadora también comienza en la esquina superior izquierda. También existe el beneficio de no tener que usar números negativos para hablar de ubicaciones en la pantalla. No se preocupe si esta cuadrícula volteada es un poco complicada. Tendremos mucho más tiempo para trabajar en las próximas actividades.

Cuando continuar: determine si esta última Lección vale la pena en su agenda. Darles a los estudiantes la oportunidad de crear y comunicar su propio dibujo puede ayudar a reforzar su conocimiento, pero si los estudiantes obviamente están logrando los objetivos de aprendizaje de la Lección sin él, también puedes pasar al resumen para sintetizar su aprendizaje.

Lección 7

Dibujar en Game Lab

Propósito

Se les presenta a los estudiantes Game Lab, el entorno de programación y comienzan a usarlo para colocar las formas en la pantalla. Aprenden los conceptos básicos de secuenciación y depuración, así como algunos comandos simples. Al final de la Lección, los estudiantes podrán programar imágenes como las que hicieron con la herramienta de dibujo en la Lección anterior.

El objetivo principal de esta Lección es dar a los estudiantes la oportunidad de acostumbrarse al entorno de programación, así como la secuencia básica y la depuración. Los estudiantes comienzan con una introducción al entorno de desarrollo interactivo (IDE) Gamelab, luego aprender las tres instrucciones (rect, ellipse y fill) que van a necesitar para codificar los mismos tipos de imágenes que han creado en el papel en la Lección anterior. Los niveles de desafío ofrecen una oportunidad para que los estudiantes que tienen más experiencia en programación exploren más en Game Lab.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (30 min)

Transferencia del conocimiento (10 min)

Objetivos

Los estudiantes serán capaces de:

- Usar el IDE de Game Lab para trazar diferentes formas de colores en la pantalla.
- Aprender la secuencia de código correctamente para superponer formas.
- Conocer el código de depuración escrito por otros.

Lección en línea
[Ver en Code Studio](#)

Vocabulario

- **Error (bug):** Parte de un programa que no funciona correctamente.
- **Depuración (debugging):** Encontrar y solucionar problemas en un algoritmo o programa.
- **Programa:** Algoritmo que ha sido codificado en algo que puede ser ejecutado por una máquina.

Código

- fill(color)
- ellipse(x, y, w, h)
- rect(x, y, w, h)

Preparación

- Prepara el proyector u otros medios para mostrar videos si desea verlos como una clase.
- [Dibujar en Game Lab- Code.org](#)

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Programación de imágenes

Observaciones

En la última Lección, creamos imágenes en la computadora organizando cuadrados y círculos en una cuadrícula. Para cada imagen que querías crear, tenías que dibujar esas imágenes manualmente, y si querías recrear una imagen era mucho trabajo. Hoy, vamos a programar la computadora para dibujar esas imágenes para nosotros. Según lo que sabe sobre las computadoras, ¿qué cree que será diferente entre decirle a una persona sobre tu imagen y contarle a la computadora tu imagen?

Dé tiempo a los estudiantes para que piensen individualmente y hablen con un compañero, luego junte la clase y escriba sus ideas en el pizarrón.

Para dar instrucciones a una computadora, necesitamos usar un lenguaje que la computadora entienda. Utilizamos HTML, que es ideal para hacer páginas web. Para hacer nuestras animaciones y juegos, usaremos una versión de Javascript que usa bloques. El entorno en el que programaremos se llama Game Lab.

Ampliación del conocimiento (30 min)

Dibujo simple en Game Lab

Grupo: Ubique a los estudiantes en parejas para programar juntos.

Transición: Envía a tus estudiantes a Code Studio- dibujando en Game Lab.

Apoyo: A medida que los estudiantes trabajan en los niveles, puede ayudarlos, pero alíentelos a intentar dedicar algo de tiempo a resolver primero los problemas. Si necesita ayuda para apoyar a los estudiantes, consulte los ejemplos en el visor de respuestas del docente. Cuando los estudiantes alcanzan los niveles de desafío, pueden elegir perseguir uno o más de los desafíos, regresar para mejorar los niveles anteriores o ayudar a un compañero de clase.

Tour de Game Lab

Los estudiantes que son nuevos en la programación a menudo tienen algunos conceptos erróneos comunes con los que se topan. Con el fin de evitar que sigan recordando a los estudiantes sobre las siguientes cosas

- Un comando por línea
- Los comandos se ejecutan en orden de arriba abajo
- El orden de las entradas en los comandos de forma importa
- Cada comando de entrada en forma está separado por comas
- (0,0) está en la esquina superior izquierda de la pantalla
- Todos los valores x- y en la pantalla son positivos

Dependiendo de la edad y el nivel de comodidad de sus estudiantes, puedes optar por utilizar este nivel para recorrer el entorno como una clase completa. Asegúrese de que los estudiantes puedan encontrar las instrucciones de nivel, el área de codificación, el área de visualización y los cajones bloqueados. Esta es también una buena oportunidad para señalar algunos de los recursos útiles, como la documentación y el botón Bloquear texto.

Coloque cuadrados en las esquinas

Una gran parte de usar Game Lab es entender la posición. Recuerda que siempre puede activar la cuadrícula o desplazarse con el mouse para ayudar a encontrar la posición x-y que desea.

Color de relleno

También puede hacer que los rectángulos tengan diferentes colores con fill (rellenar). Establecerá el color para cada forma que viene después en el código.

Hacer esto:

- Mire el código que establece el color.
- Cambie el color de azul a amarillo.
- Agregue un nuevo cuadrado arrastrando un bloque rect al área de código debajo del comando de relleno. El nuevo cuadrado puede ir a cualquier ubicación en la pantalla que desees.

El orden importa

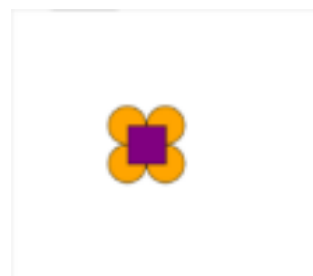
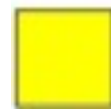
En Game Lab, importa en qué orden se encuentre el código. Se dibujan nuevas formas encima de las anteriores, cubriendo las formas que se dibujan primero. Puedes ver la diferencia cuando usa más de un color en tu código

Elipse

Puedes usar elipse para hacer un círculo.

Depuración

A menudo, el código no funciona la primera vez que se ejecuta y los programadores tienen que depurarlo. Se supone que el siguiente código hace que la imagen aparezca a la derecha, pero



el programador se confundió sobre en qué orden debería estar el código, y cómo colocar un cuadrado en el lugar correcto de la pantalla.

1. `rect(175, 175);`
2. `fill("orange");`
3. `ellipse(150,150);`
4. `ellipse(200,150);`
5. `ellipse(150,200);`
6. `ellipse(200,200);`
7. `fill("purple");`

Formas simplificadas

Los comandos `rect` y `ellipse` que se presentaron a los estudiantes en estas actividades son versiones simplificadas de los comandos completos que verán más adelante. Esto permite que la clase se centre únicamente en la ubicación de las formas en el plano de coordenadas antes de preocuparse también por el ancho y la altura de esas formas.

Transferencia del conocimiento (10 min)

Compartir dibujos

Objetivo: Los estudiantes pueden ver la variedad de cosas diferentes que pueden crear con simples dibujos de formas.

Compartir: Una vez que los estudiantes hayan completado sus dibujos, pide que compartan sus dibujos con la clase.

Ticket de salida

Meta: Los estudiantes comparten los trucos que aprendieron a medida que pasaron por niveles.

Indicación: Hoy aprendieron a dibujar en Game Lab por primera vez. ¿Qué tipo de consejo compartirán con sus amigos que iban a aprender a dibujar en Game Lab para que sea más fácil para ellos? Escríbanlo en un pedazo de papel.

Recopilar: Recopila las respuestas de los estudiantes y selecciona algunas que puedan ser útiles para que los escuchen todos los estudiantes. Comparte esos al comienzo de la próxima clase.

Este es un buen acertijo para usar como una evaluación de si los estudiantes entienden o no los fundamentos de la secuencia y el posicionamiento de formas en Game Lab.

Lección 8

Formas y aleatorio

Propósito

En esta Lección, los estudiantes continúan desarrollando su familiaridad con Game Lab al manipular el ancho y la altura de las formas que usan para dibujar. La Lección comienza con una discusión que conecta la funcionalidad de bloque expandido (por ejemplo, formas de diferentes tamaños) con la necesidad de más entradas de bloque, o “parámetros”. Los estudiantes aprenden a dibujar con versiones de `ellipse()` y `rect()` que incluyen parámetros de ancho y alto. También aprenden a usar el bloque `background()`. Al final del progreso, se les presenta a los estudiantes el bloque `randomNumber()`. Combinando todas estas habilidades, los estudiantes dibujarán una serpiente arcoiris al azar al final de la Lección.

Esta Lección les da a los estudiantes la oportunidad de expandir ligeramente sus habilidades de dibujo mientras continúan desarrollando habilidades de programación de propósito general. Tendrán que razonar sobre el plano de coordenadas x-y, considerar el orden de su código y aumentar ligeramente la complejidad de sus programas. El bloque `randomNumber()` es importante para la próxima clase, donde los estudiantes aprenden a almacenar valores utilizando variables. Esta Lección debe enfocarse principalmente en la construcción de habilidades. Si los estudiantes pueden completar la serpiente arco iris de forma independiente, entonces tienen las habilidades que necesitarán para las próximas actividades

Secuencia para el aprendizaje

Conocimiento inicial (5 min.)

Ampliación del conocimiento (40 min.)

Transferencia del conocimiento (5 min.)

Objetivos

Los estudiantes serán capaces de:

Lección en línea
[Ver en Code Studio](#)

Vocabulario

- **Parámetro:** Dato que se considera como imprescindible y orientativo para lograr evaluar o valorar una determinada situación

Código

- `background(color)`
- `ellipse(x, y, w, h)`
- `rect(x, y, w, h)`
- `randomNumber()`

- Usar y razonar sobre el dibujo de comandos con múltiples parámetros.
- Generar y usar números aleatorios en un programa.
- Usar el IDE de Game Lab para trazar diferentes formas de colores en la pantalla.
- Aprender la secuencia de código correctamente para superponer formas.
- Conocer el código de depuración escrito por otros.

Preparación

- Revisa la secuencia de niveles en [Code Studio- formas y aleatorio](#).

Estrategia de aprendizaje

Conocimiento inicial (5 min)

1. `ellipse()`
2. `randomNumber(5, 10)`
3. `rect()`

Formas de diferentes tamaños

Indicación: Nuestros bloques `ellipse` y `rect` cada uno tienen dos entradas que controlan dónde están dibujadas: la posición x-y. Si quisiera que estos comandos dibujaran una variedad más amplia de rectángulos y elipses, ¿qué entradas adicionales podría necesitar para proporcionar estos bloques? ¿Qué controlaría cada entrada adicional?

Discute: Los estudiantes deben intercambiar ideas en silencio, luego compartir con un compañero, después compartir con toda la clase. Registra ideas mientras los estudiantes las comparten en el pizarrón.

Observaciones: Esta fue una buena lista de ideas. Si queremos que nuestros bloques dibujen formas de diferentes maneras necesitarán más insumos que nos permitan decirles cómo dibujar. Las entradas o aperturas en nuestros bloques tienen un nombre formal, parámetros, y hoy vamos a aprender más sobre cómo usarlas.

Ampliación del conocimiento (40 min)

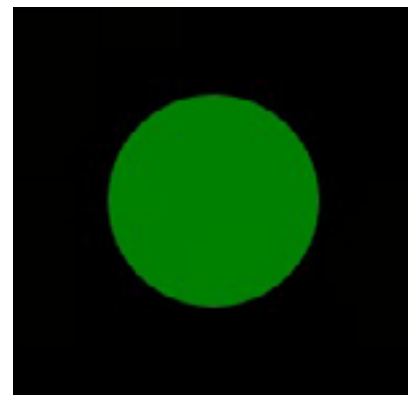
Programación de imágenes

Transición: Mover a los estudiantes a Code Studio y realizar las actividades:

- Resumen de la Lección
- Formas y parámetros
- Números al azar

Compartir: Si algunos estudiantes se toman más tiempo para trabajar en sus proyectos, dales la oportunidad de compartir sus serpientes arcoíris más complejas. Enfoque la conversación sobre

Meta: Esta discusión introduce la palabra de vocabulario “parámetro” y también ayuda a motivar su uso. Los estudiantes verán que las versiones del bloque `ellipse()` y `rect()` en esta Lección tienen parámetros adicionales, así como el bloque `randomNumber()` que tiene dos parámetros. Los estudiantes pueden decir que quieren insumos para el tamaño de las formas, su color, etc. Durante esta conversación, vincule las conductas que los estudiantes desean con las entradas que el bloque necesitaría. Por ejemplo, si desea que los círculos tengan un tamaño diferente, el bloque necesitará una entrada que le permita al programador decidir qué tan grande debe ser.



qué parámetros están manipulando los estudiantes o aleatorizando para crear sus dibujos.

Fondo: A veces querrá llenar toda la pantalla con un color. Para eso, puedes usar el bloque background. Cubra todo en la pantalla con el color que elijas.

El bloque de fondo se dibujará encima de todo lo que ya está en tu dibujo, por lo que el orden en el código es importante.

Números al azar: randomNumber() elija un número aleatorio entre un valor mínimo y máximo. Puede usar este bloque en lugar de escribir en el número específico. Si realiza sus dibujos con números aleatorios, se verá un poco diferente cada vez que ejecute el programa.

Transferencia del conocimiento (5 min)

Reflexión

Preguntar: Pida a los estudiantes que reflexionen sobre el desarrollo de las cinco prácticas de Descubrimientos CS (Resolución de problemas, Persistencia, Creatividad, Colaboración, Comunicación). Elija una de las siguientes indicaciones según lo considere apropiado.

- Elige una de las cinco prácticas en las que crees que demostraste crecimiento en esta Lección. Escribe algo que hiciste que ejemplifica esta práctica.
- Elige una práctica en la que pienses que puede seguir creciendo. ¿Qué te gustaría mejorar?
- Elige una práctica que pensaste que era especialmente importante para la Lección que completamos hoy. ¿Qué lo hizo tan importante?

Lección 9

Variables

Propósito

En esta Lección, los estudiantes aprenden a usar variables para etiquetar un número en su programa o guardar un valor generado aleatoriamente. Los estudiantes comienzan la lección con una descripción muy básica del propósito de una variable. Luego, los estudiantes completan un progreso de nivel que refuerza el modelo de una variable como una forma de etiquetar o nombrar un número. Los estudiantes usan variables para guardar un número aleatorio para ver que las variables realmente almacenan o guardan sus valores, lo que les permite usar el mismo número aleatorio varias veces en sus programas.

En esta lección, los estudiantes verán por primera vez variables, y no se espera que entiendan completamente cómo funcionan para cuando ésta termine. Por lo tanto, los estudiantes deben terminar esta lección sabiendo que las variables son una forma de etiquetar un valor en sus programas para que puedan ser reutilizados o referenciados más adelante. En la siguiente lección, los estudiantes conocerán los sprites, los que deben ser referenciados por una variable.

Usar variables para manipular dibujos es una habilidad sorprendentemente desafiante que requiere una gran cantidad de previsión y planificación. Si bien los estudiantes usarán o modificarán muchos programas en esta lección, no se espera que creen programas que usan variables para modificar las características de un dibujo. En lecciones posteriores, los estudiantes ampliarán su comprensión de las variables y formas más avanzadas.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (30 min)

Transferencia del conocimiento (5 min)

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- Introducción a las variables- [Video](#) (configure los subtítulos en español)

Vocabulario

- **Variable:** Un marcador de posición para una información que puede cambiar.

Código

- Declara variable

Objetivos

Los estudiantes serán capaces de:

- Identificar una variable como una forma de etiquetar y referenciar un valor en un programa.
- Usar variables en un programa para almacenar una información que se usa varias veces.
- Identificar el motivo y solución de errores comunes encontrados al programar con variables

Preparación

- Revisa el progreso de nivel en [Code Studio- Variables](#).

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Etiquetas y valores

Video: Como clase, mire el video que presenta las variables.

Revisión: Los estudiantes pueden revisar los puntos más importantes del video en el nivel 4. Los estudiantes no necesitan comprender todas estas ideas ahora mismo, pero pueden usar este nivel como referencia durante toda la Lección.

Observaciones: Hay mucho que aprender sobre las variables. Hoy vamos a usarlas para ayudarnos a hacer dibujos. Sin embargo, lo más importante es simplemente ver cómo dar a una etiqueta un valor nos ayuda a escribir programas.

Ampliación del conocimiento (30 min)

Programación con variables

Dirige a los estudiantes a los Code Studio y aborden las siguientes actividades:

- Introducción a las variables
- Variables
- Nombrar variables
- Desafío: dibujar una imagen
- Desafío: actualización de variables

Variables

Una variable te permite almacenar un solo valor en la memoria de su computadora con un nombre descriptivo. El uso de variables le permite consultar fácilmente el mismo valor muchas veces en su programa o guardar un número al que le gustaría consultar más adelante.

Creación de variables

El comando `var` creará una nueva variable con la etiqueta que le da. Esta variable tiene la etiqueta `size`.

Evitar la carga frontal: si bien esta Lección comienza con dos recursos que explican cómo funcionan las variables, es probable que sean más significativas para los estudiantes una vez que hayan utilizado variables en los programas de dibujo. Asegúrese de que los estudiantes sepan que estos recursos están disponibles, y luego revíselos -si lo desea- al final de la Lección para ayudarlos a crear sentido.

A purple speech bubble with a white border, containing the text `var size;` in a white monospace font.

Asignación de valores

El operador de asignación = asignará un nuevo valor a su variable. Este comando asignó 100 al tamaño variable. La variable siempre debe estar en el lado izquierdo. Usted leería este comando como “obtiene tamaño 100”, ya que el tamaño obtiene un nuevo valor de 100. Cualquier valor antiguo que podría haber sido asignado se pierde para siempre. A las variables también se les puede asignar un número aleatorio. Esto le permite guardar un único valor aleatorio para que pueda usarlo tantas veces como desee en su programa.

```
size = 100;
```

Transferencia del conocimiento (5 min)

Reflexión

Indicación: Dé a los estudiantes las siguientes instrucciones

- ¿Cuál es tu propia definición de variable?
- ¿Por qué las variables son útiles en los programas?

Discute: Haga que los estudiantes escriban sus ideas de forma independiente, posteriormente, invítalos a compartirlas con los compañeros del grupo.

En sus cuadernos: ¿qué conexiones observas entre las variables y el modelo entrada- almacenamiento- procesamiento- salida de una computadora?

Sugerencias de evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Nombran las variables en relación con su significado (nemotécnico).
- Utilizan las variables para hacer operaciones.

Meta: Utilizar esta discusión para evaluar los modelos mentales de una variable de los estudiantes. Es posible que desee que los estudiantes escriban sus respuestas para que pueda recopilarlas y revisarlas más tarde. Debería ver principalmente que entienden que las variables pueden etiquetar o nombrar un número para que pueda ser utilizado más adelante en sus programas. Si bien hay otras propiedades de una variable que los estudiantes pueden haber aprendido, esta es la más importante antes de pasar a la siguiente Lección.

Lección 10

Sprites

Propósito

Para crear imágenes más interesantes y detalladas, se les presenta a los estudiantes el objeto sprite. A cada sprite se le puede asignar una imagen para mostrar, y los sprites también hacen un seguimiento de múltiples valores acerca de ellos mismos, lo que resultará útil en el camino al hacer animaciones.

Hacer un seguimiento de muchas formas y las diferentes variables que controlan aspectos de esas formas puede ser muy complejo. Habrá muchas variables con diferentes nombres. En cambio, los científicos informáticos crearon algo llamado objeto que permite un nombre de variable para controlar tanto la forma como todos sus aspectos. En Game Lab usamos un cierto tipo de objeto llamado sprite. Un sprite es solo un rectángulo con propiedades para controlar su aspecto. Las propiedades son las variables que están unidas a un sprite. Puede acceder a ellos mediante notación de puntos.

Usando la pestaña Animación, los estudiantes pueden crear o importar imágenes para ser usadas con sus sprites. Más adelante, estos sprites se convertirán en una herramienta útil para crear animaciones, ya que sus propiedades se pueden cambiar y actualizar a lo largo del curso de un programa.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento

Transferencia del conocimiento (5-10 min)

Evaluación

Objetivos

Los estudiantes serán capaces de:

- Asignar un sprite a una variable.

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Actividades Sprite enlace code.org.](#)

Para los estudiantes:

- [Sprite planificación de escena- Guía de actividades.](#)

Vocabulario

- **Propiedad:** Atributos que describen las características de un objeto.
- **Sprite:** Un personaje gráfico en la pantalla con propiedades que describen su ubicación, movimiento y apariencia.

Código

- `drawSprites(group)`
- `var sprite = createSprite(x, y, width, height)`
- `sprite.scale`

- Usar notación de puntos para actualizar las propiedades de un Sprite.
- Crear una escena estática combinando sprites, formas y texto.

Preparación

- Sprite planificación de escena- [Guía de actividades.](#)

Estrategia de aprendizaje

Conocimiento inicial (5 min)

¿Cuánta información?

Piense, empareje, comparta: Hasta ahora solo hemos escrito programas que ponen formas simples en la pantalla. Haz una lista de todas las diferentes piezas de información que has utilizado para controlar cómo se dibujan estas formas.

Preguntar: ¿Qué pasaría si quisiéramos crear programas con imágenes más detalladas, tal vez incluso con personajes con los que pudiéramos interactuar? ¿Qué otras piezas de información.

Observaciones: Hoy aprenderemos cómo crear personajes en nuestras animaciones llamados sprites. Estos sprites se almacenarán en variables, al igual que almacenaron números en el pasado, pero los sprites pueden contener muchos datos, lo que le permitirá crear programas mucho más interesantes (¡y eventualmente animados!).

Ampliación del conocimiento

Introducción a Sprites

Distribuir: Sprite planificación de escena - Guía de actividades. Los estudiantes pueden usar esta hoja para planear la escena de Sprite que crean al final de esta Lección, pero la planificación también se puede completar en papel borrador.

Transición: Enviar estudiantes a Code Studio y que realicen los Niveles de Code Studio. Sprite.

- Creando Sprites
- La pestaña de animación
- Escenas de sprite
- Extiende tu escena

Sprites

Crear Sprites: El bloque crea un nuevo sprite y lo asigna a una variable. El nombre predeterminado es sprite, por lo que querrá cambiarlo a algo más significativo.

El objetivo aquí es hacer que los estudiantes piensen en todos los diferentes valores que forman una única forma en la pantalla, y cuántos valores más pueden necesitar para controlar un carácter más detallado en un programa. Si los estudiantes están luchando para encontrar ideas, puede usar algunas de las siguientes indicaciones: *¿Cómo le dice a una figura a dónde ir a la pantalla? ¿Cómo le dices a una forma qué tamaño necesita ser? ¿Cómo le dices a una forma de qué color debería ser? ¿Qué hay de su esquema? ¿Qué sucede si quiere cambiar alguno de esos valores durante su programa o controlar otros elementos, como la rotación?*

El sprite es un tipo de datos llamado **objeto**. Si bien todavía no estamos introduciendo explícitamente el concepto de objetos, los estudiantes deben comprender que un sprite es un tipo de valor diferente de los que hemos visto anteriormente, uno que puede contener referencias a muchos más valores. Para los estudiantes que sienten curiosidad sobre si hay otros objetos en nuestros programas, pídeles que vean si hay más bloques en la caja de herramientas que siguen la misma notación de puntos (como `World.width` y `World.height`).

Dibujo de Sprites: Los Sprites solo aparecen en la pantalla cuando los dibujas allí. Llamar al comando dibuja todos los sprites creados en la pantalla.

Imágenes: Una vez que haya creado un sprite, puedes usar el `sprite.setAnimation()` comando para cambiar el aspecto de su sprite de un rectángulo a una imagen. Todas las imágenes que ha cargado en la pestaña Animación aparecen en el `sprite.setAnimation()` menú desplegable.

Suba su propia imagen: También puedes usar la pestaña Animación para cargar o dibujar su propia imagen.

- Haga clic y luego cargar una imagen.
- Selecciona el archivo de tu computadora.
- Cambie el nombre de su imagen para que sea fácil de recordar. Para cambiarle el nombre, haga clic en el texto debajo de la imagen.
- De vuelta en el modo de código, agregue un bloque `sprite.setAnimation()` para que su sprite use su nueva animación.

Cambiar el tamaño con escala: En el cajón de Sprites de la caja de herramientas, verá un nuevo bloque llamado `sprite.scale`. Este comando le permite cambiar el tamaño de un sprite en relación con su tamaño original. `sprite.scale = 1` es el tamaño normal. `sprite.scale = 0.5` hace que su sprite sea la mitad de grande, mientras `sprite.scale = 2` lo hace dos veces más grande.

Agregar texto: El bloque `text` le permite colocar texto en cualquier lugar que desee en la pantalla. Cambie este texto en el bloque provisto a otra cosa y agregue un segundo `text` bloque para escribir en una parte diferente de la pantalla.

Consejo: El tamaño de texto predeterminado es bastante pequeño, pero puede usar el bloque `textSize` para cambiar eso. También puede usar el bloque `fill` para cambiar el color de tu texto.

Transferencia del conocimiento (5-10 min)

Compartir

Permite a los estudiantes compartir sus escenas de Sprite. Alienta a los estudiantes a reflexionar sobre sus escenas e identificar las formas en que les gustaría mejorar.

```
var sprite = createSprite();
```

```
drawSprites();
```



Evaluación

Evaluar escenas de sprite

Para evaluar las escenas de Sprite, pida a los estudiantes que hablen de su código. Verifique para asegurarse que los estudiantes sepan por qué secuenciaron su código de la manera en que lo hicieron y, en particular, busquen “código muerto” o código que no afecte a la escena final. En este punto, es probable que los estudiantes sigan dibujando formas antes de dibujar el fondo (que luego no se verá) o que estén llamando `drawSprites()` varias veces (solo se debe llamar una vez).

Lección 11

Ciclo de dibujo aleatorio

Propósito

En esta Lección, se presenta a los estudiantes el ciclo de dibujo, uno de los paradigmas de programación principales en Game Lab. Para comenzar la lección, los estudiantes miran algunos flipbooks físicos para ver que tener muchos marcos con diferentes imágenes crea la impresión de movimiento. Luego, los estudiantes ven un video que explica cómo el ciclo de dibujo en Game Lab ayuda a crear la misma impresión en sus programas. Los estudiantes combinan el ciclo de dibujo con números aleatorios para manipular algunas animaciones simples con puntos y luego con sprites. Al final de la Lección, los estudiantes usan lo que aprendieron para actualizar su escena de sprites de la lección anterior.

El ciclo de dibujo es un componente central de Game Lab. El hecho de que el entorno de Game Lab llama repetidamente a esta función muchas veces por segundo (por defecto 30) es lo que permite a la herramienta crear animaciones. Esta lección tiene dos objetivos, por lo tanto. El primero es que los estudiantes vean cómo la animación en general depende de mostrar muchas imágenes ligeramente diferentes en una secuencia. Para ayudar a los estudiantes a tener flipbooks físicos que pueden usar, un video. El segundo objetivo es que los estudiantes comprendan cómo el ciclo de dibujo les permite crear este comportamiento en Game Lab. Los estudiantes deben dejar la Lección entendiendo que los comandos del ciclo de dibujo se invocan después de todos los demás códigos, pero luego se los llama repetidamente a una velocidad de fotogramas. Los estudiantes tendrán la oportunidad de continuar desarrollando una comprensión de este comportamiento en las próximas dos lecciones.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (60 min)

Transferencia del conocimiento (5 min)

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- Aleatorio- [Manipulador](#).
- Ejemplo de Flipbook- [Video](#).

Vocabulario

- **Animación:** Una serie de imágenes que crean la ilusión de movimiento al mostrarse rápidamente una tras otra.
- **Marco:** Una sola imagen dentro de una animación.
- **Frecuencia de fotogramas:** La velocidad a la que se muestran los fotogramas de una animación, normalmente medida en fotogramas por segundo.

Código

- `function draw() { }`
- `World.frameRate`

Objetivos

Los estudiantes serán capaces de:

- Explicar qué es una animación y cómo crea la ilusión de un movimiento simple.
- Explicar cómo el ciclo de dibujo permite la creación de animaciones en Game Lab.
- Utilizar el ciclo de dibujo en combinación con el comando `randomNumber ()`, las formas y los sprites para realizar animaciones simples.

Preparación

- Imprime [Flipbook no aleatorio- Flipbook Manipulativo](#) y [Aleatorio- Manipulador](#).

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Video: Mostrar [ejemplo de Flipbook- Video](#).

Indicación: Este video muestra un flipbook para hacer animaciones. Con tus propias palabras, explica: ¿cómo está funcionando? ¿Por qué “engaña a nuestros ojos” para que se piense que algo se está moviendo?

Discuta: Haga que los estudiantes escriban sus ideas de forma independiente, luego compártanlo con sus compañeros, luego como un grupo completo.

Observaciones: Vamos a empezar a aprender cómo hacer animaciones, no solo imágenes fijas. Para hacer esto, necesitamos una forma que nuestros programas dibujen muchas imágenes por segundo. Para hacer esto, tendremos que aprender una nueva herramienta importante.

Ampliación del conocimiento (45 min)

Video: Mira el [video de introducción a ciclos de dibujo](#).

Dirige a los estudiantes a CodeStudio y realiza los [Niveles de Code Studio](#).

- Formas y el lazo
- Sprites y Draw Loop
- Sprite Propiedades
- Anima tu escena

Transferencia del conocimiento (10 min)

Compartir: Los estudiantes sólo necesitan hacer pequeños cambios en sus proyectos (p. Ej., Agitando un solo sprite), pero pide que compartan con un compañero o como clase completa.

Preguntar: Haga que los estudiantes respondan a las siguientes instrucciones:

- ¿Qué es una animación?
- ¿Por qué el ciclo de dibujo nos ayuda a hacer animaciones?
- ¿Cuáles son algunos errores o errores comunes que debemos

Objetivo: Esta discusión debe presentar algunas ideas clave sobre la animación. Los estudiantes deben entender que la clave es ver muchas imágenes seguidas que son ligeramente diferentes. Introduzca la palabra de vocabulario “marco” como una de esas imágenes. Luego haga la transición al hecho de que pronto los estudiantes crearán sus propias animaciones.

Meta: Use estos avisos para evaluar si los estudiantes han entendido los principales objetivos de aprendizaje de la Lección.

Conceptos clave: Existen muchos conceptos erróneos comunes con el ciclo de extracción. Asegúrese de que los estudiantes entiendan lo siguiente:

- El ciclo de extracción se ejecuta después de todos los demás códigos en su programa. En realidad, no importa donde se encuentre en su programa.
- The Game Draw es ejecutado por Game Lab a una tasa de cuadros constante de 30 fotogramas por segundo. En realidad, no necesita llamar a la función usted mismo.
- Los “marcos” en Game Lab se pueden considerar hojas de transparencia, A menos que dibujes un fondo, todas tus nuevas formas o sprites simplemente aparecerán encima de tus antiguos.

tener en cuenta a medida que seguimos programando con el ciclo de dibujo?

Revisión: Regrese a los recursos que los estudiantes vieron al comienzo de la lección y aborda los conceptos erróneos que han surgido en la lección.

Sugerencias para la evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Ejecutan ciclos (loops), a partir de un patrón que se repite en una secuencia.
- Evalúan su trabajo de manera colaborativa.
- Utilizan procesos de depuración
- Aprovechan recursos disponibles en las áreas de trabajo.

- - Solo debe de tener un ciclo de extracción en su programa

Lección 12

Patrón de contador desconectado

Propósito

Los estudiantes exploran el comportamiento subyacente de las variables a través de una lección desconectada. Usando tarjetas de notas y cadenas para simular variables dentro de un programa, los estudiantes implementan algunos programas cortos. Una vez que se sienten cómodos con esta sintaxis, los estudiantes usan el mismo proceso con las propiedades de los sprites, haciendo un seguimiento del progreso de un sprite en la pantalla.

El razonamiento sobre las variables puede ser complicado, especialmente para los nuevos programadores. En esta Lección, los estudiantes completan una lección desconectada utilizando manipulativos físicos (tarjetas y cuerdas) para construir un modelo mental de cómo la información puede almacenarse en una variable y ser manipulada por un programa. Este modelo luego se extiende a las propiedades de los sprites, que mantienen los valores de una manera similar. Esta Lección presenta la sintaxis y los conceptos que los estudiantes podrán “conectar” en la siguiente lección.

Secuencia para el aprendizaje

Conocimiento inicial (15 min)

Ampliación del conocimiento (20 min)

Ampliación del conocimiento 2 (30 min)

Transferencia del conocimiento (10 min)

Objetivos

Los estudiantes serán capaces de:

- Describir la conexión entre la actualización de las propiedades de ubicación de un sprite y el movimiento de los sprites en la pantalla.

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Variables desconectada-Guía de actividades.](#)

Vocabulario

- **Expresión:** Cualquier unidad de código válida que resulta en un valor.
- **Variable:** Un marcador de posición para una información que puede cambiar.

- Leer y seguir los pasos de un breve programa escrito en pseudocódigo que manipula los valores de las variables.

Preparación

- Preparar materiales para etiquetas y valores: Fichas, post-its o trozos de papel, etc.
- Preparar materiales para conectores: piezas de cuerda, plumas o limpiapipas, etc.
- [Tablero de variables desconectado- Manipulador](#) para cada grupo o reúne papel para que los estudiantes lo utilicen para hacer sus tableros.
- Revisar las reglas de la Lección Desconectada de Variables para asegurarte de que las comprenda y prepárate para responder preguntas, especialmente si las demostrarás tú mismo.
- Sacar copias de [Variables desconectada- Guía de actividades](#) para cada alumno.

Estrategia de aprendizaje

Conocimiento inicial (15 min)

Pregunta: En la última Lección, usamos el ciclo de dibujo para hacer que nuestros sprites se muevan en la pantalla. ¿Cuáles son otras formas en que podríamos hacer que nuestros sprites se muevan?

Ponga las ideas de los estudiantes en el pizarrón.

Observaciones: En las próximas actividades, vamos a ver muchas formas de mover nuestros sprites. Para hacer eso, necesitamos aprender un poco más sobre las variables y cómo funcionan. Hoy vamos a hacer una Lección con variables y propiedades de sprites que nos ayudarán a hacer estos tipos de movimiento. A medida que avanzamos en la lección, piensen cómo lo que está aprendiendo puede ayudarlo a hacer que sus sprites se muevan de la manera que desee.

Ampliación del conocimiento (20 min)

Variables Lección desconectada

Grupo: Agrupar a los estudiantes en parejas.

Distribuir: Entregar a cada par:

- Un conjunto de etiquetas / valores y conectores
- Una sola hoja de papel para crear su pizarrón
- 2 copias de las Variables desconectada- Guía de actividades.

Pantalla: Muestre las reglas de la página principal de la guía de actividades y escribe los primeros dos programas de la segunda página.

Observaciones: Hoy vamos a trabajar en un mundo de etiquetas, valores y conectores entre ellos. Para simular este mundo, usará los trozos de papel y cuerda que les he dado. Para empezar, tendremos que configurar nuestros tableros, y luego repasaremos cómo funcionan los comandos para este mundo.

Demuestra: Muestre a la clase cómo dividir sus tablas en 3 secciones y etiquételas en secuencia, como se muestra en la primera página de la guía de actividades.

Apoyo: Los estudiantes deben trabajar en los dos primeros programas como un grupo, mientras hace referencia a los pasos

Esta Lección está diseñada para abordar muchos **conceptos erróneos comunes** con variables y memoria.

Conceptos erróneos comunes:

- Las variables pueden tener valores múltiples (no pueden, las variables tienen como máximo un valor)
- Las variables “recuerdan” los valores antiguos (no, por lo tanto, los valores antiguos se eliminan en la Papelera)
- Las variables se conectan después de un comando como “x=y” (Esto puede surgir más tarde en el curso cuando los estudiantes usan sprites y se abordarán en gran detalle.
 - Por ahora, los estudiantes se ven obligados a crear nuevas tarjetas de valor cada vez porque incluso para una declaración como x=y no hay “conexión” entre las variables formadas)
- Las variables tienen expresiones (por ejemplo, 1+5). Las variables solo contienen valores, las expresiones se calculan de antemano. Es por eso que las tarjetas de valores solo se crean una vez que los estudiantes tienen un solo valor. Aplicar esta regla de cerca.

en la guía de actividades. A medida que los equipos se sienten cada vez más cómodos al completar los comandos, alientan a los equipos a ejecutar cada comando de forma independiente antes de comparar sus tableros con un par compañero. El objetivo es solo asegurarnos de que todos tengan la oportunidad de comprender los pasos de la Lección.

Observaciones: Ahora es su turno de intentar ejecutar algunos de estos programas por tu cuenta. En la parte inferior de la página hay dos programas más que puede ejecutar. Para cada uno, debe ejecutar el programa para averiguar el estado final del programa. En otras palabras, debe saber qué etiquetas están conectadas a qué valores. Una vez que llegue al final de cada programa, puedes comparar tus resultados con un compañero. Si no estás de acuerdo, vuelve para ver si puedes encontrar dónde perdió la pista.

Apoyo: Los estudiantes deben trabajar en parejas a través de los dos programas en la guía de actividades. Haga un chequeo para asegurarse de que todos estén de acuerdo con el estado final del programa (qué etiquetas están conectadas con qué valores). Si los estudiantes no están de acuerdo, refuerza la necesidad de depurar al leer el código retrocediendo y rastreando cada paso.

Ampliación del conocimiento 2 (30 min)

Sprite propiedades

Distribuir: Propiedades de Sprite en Variables desconectada-Guía de actividades si no lo ha incluido en la guía de actividades original.

Demuestre: Muestra a los estudiantes las reglas de lección en la guía de actividades. Trabaje en el programa 5 como grupo y muestre cómo crear una nueva tarjeta Sprite y conectarla a tu tarjeta de etiqueta variable y a las tarjetas de propiedad de sprite. Asegúrate de que los estudiantes comprendan que deberían crear una nueva tarjeta sprite cada vez que vean el comando createSprite y que dibujen sus sprites en la cuadrícula cada vez que vean el comando drawSprites.

Apoye: Los estudiantes deben trabajar en los últimos dos programas usando sus materiales manipulables.

Pregunta: ¿Cómo se movió el sprite a través de la red en el Programa 3? ¿Cómo se movió el sprite a través de la red en el Programa 4 de la guía de actividades?

Objetivo: Los estudiantes deben ver que los comandos, como $x=x+1$ mover un sprite de una manera deliberada a través de la pantalla, en posición al movimiento aleatorio que vieron en la Lección anterior

Después de que los estudiantes hayan completado las preguntas de reflexión, deben comparar con otro par, luego discutirlo como clase.

Transferencia del conocimiento (10 min)

Discusión

Indicación: Hoy vimos algunas pistas de cómo podríamos programar los tipos de movimiento que queremos para nuestros sprites. ¿Cuáles son algunos problemas que aún tenemos que resolver para que el sprite parezca moverse en la forma que deseo?

Permita que los estudiantes hagan una lluvia de ideas sobre problemas y anótelos en el pizarrón.

Preguntar: Elija uno o dos de estos problemas y comience a pensar en algunas formas en que podría ser resuelto.

Dé tiempo a los estudiantes para que hagan una lluvia de ideas individualmente antes de compartir sus soluciones.

Observaciones: Estas son buenas ideas. En la próxima lección, vamos a ver cómo podemos usar algunas de las cosas que hemos aprendido hoy para hacer que nuestros sprites se muevan de muchas maneras diferentes.

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Describen de manera explícita, cada paso que compone un proceso secuencial.

El objetivo de la discusión no es hacer que los estudiantes piensen en soluciones a todos los problemas, si no que los identifiquen, preparándolos para la Lección del patrón contrario. Algunos programas que los estudiantes puedan ver son que las caras sonrientes permanecieron en la pantalla, en lugar de moverse, que solo podían trasladarse la izquierda y la derecha o hacia arriba o hacia abajo, o que las imágenes se detienen después de un cierto periodo de tiempo.

Lección 13

Movimiento de sprite

Propósito

Al combinar Dibujo de repetición (Draw loop) y patrón de contador, los estudiantes escriben programas que mueven los sprites a través de la pantalla, así como también animan otras propiedades de los sprites.

Esta lección combina el dibujo con ciclos que los estudiantes conocieron anteriormente, de igual manera que el patrón de contador que aprendieron para crear programas con un movimiento determinado. Al aumentar o disminuir las propiedades de los sprites, como `sprite.x`, puedes escribir programas que muevan los sprites en los patrones esperados, en lugar de uno aleatorio que usamos en el pasado. Las animaciones que los estudiantes aprenden a crear en esta lección sientan las bases para todas las animaciones y juegos que harán en el resto del curso.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Usar el patrón de contador para incrementar o disminuir las propiedades de los sprites.
- Identificar qué propiedades de los sprites deben cambiarse, y de qué manera, para lograr un movimiento específico.

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Revisando las propiedades de Sprites

Preguntar: En una hoja de papel, enumera todas las propiedades de los sprites que puedas imaginar y el aspecto de un sprite que afecten.

Comente: ¿Qué tipo de animaciones podrías hacer al combinar las propiedades de los sprites con el patrón contrario? Considera agregar y restar propiedades, o incluso actualizar varias propiedades al mismo tiempo. Registra ideas mientras los estudiantes las comparten en el pizarrón.

Ampliación del conocimiento (40 min)

Niveles: Sprites e imágenes

Transición: Envía estudiantes a Code Studio- Movimiento Sprite.

Apoyo: Los estudiantes deben progresar a través de los niveles sin detenerse en esta clase. Cuando hayan terminado los niveles de desarrollo de habilidades, se les dará la opción de qué proyecto desean extender.

Niveles de Code Studio:

- El patrón de contador
- Movimiento con el patrón de contador
- Depuración con vigilantes
- Animar a los Sprites
- Crea tu propia animación
- Desafíos
- Juego gratis

Transferencia del conocimiento (5 min)

Preguntar

Pida a los estudiantes que reflexionen sobre el desarrollo de las cinco prácticas del curso DISCOVERIES CS (Resolución de problemas, Persistencia, Creatividad, Colaboración,

El propósito de esta discusión es motivar a los estudiantes a pensar cómo podrían usar las diversas propiedades de sprites que han visto hasta ahora para hacer animaciones con un movimiento determinado. Si los estudiantes luchan para encontrar ideas, pueden limitar la pregunta a propiedades específicas. Por ejemplo:

- ¿Qué le pasaría a un sprite si aumentara constantemente su x propiedad?
 - ¿Qué le pasaría a un sprite si aumentara constantemente su y propiedad?
- ¿Qué pasa con otras propiedades o combinando propiedades múltiples?

Comunicación). Elija una de las siguientes indicaciones según lo considere apropiado.

- Elige una de las cinco prácticas en las que crees que demostraste crecimiento en esta Lección. Escribe algo que hiciste que ejemplifica esta práctica.
- Elige una práctica en la que pienses que puedes seguir creciendo. ¿Qué te gustaría mejorar?
- Elige una práctica que pensaste que era especialmente importante para la Lección que completamos hoy. ¿Qué lo hizo tan importante?

Lección 14

Booleanos desconectados

Propósito

En esta lección, se les presenta a los estudiantes los valores booleanos y la lógica, así como las declaraciones condicionales. La clase comienza jugando un juego simple de levantarse (Stand Up), sentarse (Sit Down) en el que las afirmaciones booleanas (verdadero / falso) describen propiedades personales (color de pelo u ojos, tipo de ropa, edad, etc.). Esto hace que los estudiantes piensen cómo pueden enmarcar una propiedad con múltiples valores potenciales (como la edad) con una pregunta binaria.

A partir de ahí, los estudiantes reciben un grupo de objetos con propiedades físicas similares, pero variables. Con un compañero, agrupan esos objetos basándose en declaraciones booleanas cada vez más complejas, incluidos booleanos compuestos con AND (y) y OR (o).

Finalmente, revelamos que Condicionales es una herramienta para tomar decisiones o afectar el flujo de un programa que utiliza declaraciones booleanas como entrada

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (30 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Organizar objetos basados en declaraciones booleanas simples y compuestas.
- Describir las propiedades de un objeto usando declaraciones booleanas.

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- Propiedades booleanas
- [Guía de actividades](#).

Vocabulario

- **Booleano:** Un valor único de VERDADERO o FALSO.
- **Condicionales:** Declaraciones que solo se ejecutan bajo ciertas condiciones.
- **Expresión:** Cualquier unidad de código válida que se resuelve en un valor.

Preparación

- Propiedades booleanas- [Guía de actividades](#) para cada estudiante.
- (Opcional) Reúne objetos con características similares pero variables para usar en lugar de la hoja de trabajo (los ladrillos LEGO funcionan bien, una bolsa de dulces mixta también puede ser divertida).

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Párate, siéntate

Distribuir:

Entregue a cada alumno una tarjeta y pida que respondan las siguientes preguntas (puedes agregar algunas propias).

1. ¿Cuál es tu color de cabello?
2. ¿Usas lentes o lentes de contacto?
3. ¿Cuál es tu número favorito?
4. ¿Cuál es tu color favorito?
5. ¿En qué mes naciste?
6. ¿Tienes hermanos?
7. ¿Cuál es el último dígito de tu número de teléfono?
8. Escribe algo acerca de ti que las personas no saben y no pueden deducir con solo mirarte.

Luego recoja las cartas. Para jugar, sigue estos pasos:

Para cada tarjeta:

1. Selecciona una tarjeta
2. Di: Voy a leer la respuesta al # 7, pero si eres tú, no digas nada.
3. Lee la respuesta al # 7
4. Di: Ahora todos se ponen de pie y vamos a hacer algunas preguntas con respuestas booleanas para ayudar a determinar quién es esta persona. Voy a decir un montón de declaraciones. Si son ciertos, mantente en pie. Si son falsos, siéntate.
5. Traduce las respuestas del # 1 al # 7 en enunciados que pueden ser verdaderos o falsos, consulta a continuación, en voz alta. La persona que queda en pie es aquella que respondió verdadero a todas las afirmaciones y, por lo tanto, es responsable de la tarjeta.
6. Según cómo se preguntaron los números 3, 4, 5 y 7, es probable que más de una persona siga en pie. Si es el caso, vuelve a preguntarlas, esta vez restringiendo más la información, por ejemplo, "Mi color favorito es el violeta" para la pregunta 4.

Traducción de ejemplo

Pregunta	Responder	Declaración verdadera / falsa
1. ¿Cuál es tu color de pelo?	marrón	"El color de mi cabello es marrón"
2. ¿Usa lentes o lentes de contacto?	sí	"Uso gafas o lentes de contacto"
3. ¿Cuál es tu número favorito?	12	"Mi número favorito es mayor que 10 y menor que 20"
4. ¿Cuál es tu color favorito?	púrpura	"Mi color favorito aparece en una puesta de sol"
5. ¿En qué mes naciste?	mayo	"Nací en la primavera"
6. ¿Tienes hermanos?	sí	"Tengo hermanos"
7. ¿Cuál es el último dígito de su número de teléfono?	5	"El último dígito de mi número de teléfono es excelente"

Juegue esto varias veces cambiando los enunciados “verdadero / falso” que usas. Sea creativo con el uso y recuerde a los estudiantes que la OR (o) significa que una parte de la declaración es verdadera dará como resultado que toda la declaración sea verdadera.

Discute: El juego Pararse/Sentarse con los estudiantes:

- ¿Qué tipo de cuestionamientos realizamos?
- ¿Alguna vez te confundiste sobre si debieses estar parado o sentado? ¿Por qué?
- En cualquier punto del juego, ¿en cuántos estados diferentes podrías estar?

Introduzca el vocabulario booleano como una descripción para los tipos de preguntas que estábamos haciendo. La característica que define a un booleano es que puede tener solo dos estados: en nuestro juego, esos estados eran Verdadero y Falso, o de pie y sentado.

Ampliación del conocimiento (30 min)

Hacer las preguntas correctas

Idea Genial

Preguntar: Haga una lluvia de ideas sobre lugares donde hayan visto valores booleanos anteriormente, ya sea en la clase o en el mundo.

Discute: Haga que los estudiantes compartan sus respuestas. Las posibles respuestas podrían incluir:

- Binario
- Diagramas de flujo
- Interruptores de luz (y otros dispositivos que pueden estar encendidos o apagados)

Ordenando con booleanos

Observaciones: En el juego anterior, las preguntas booleanas que hice se basaron en sus propiedades. Sus propiedades no tienen que existir solo en dos estados (¿cuántos colores de cabello diferentes hay en el salón?), Pero las preguntas que hice tenían que dividirlos en dos estados (¿cuántas personas en el salón tienen el pelo rojo?). Vamos a hacer una clasificación similar usando las propiedades de varias imágenes.

Grupo: Organiza a los estudiantes en parejas.

Configuración: Asigna a cada miembro del par como verdadero o falso.

Distribuir: Distribuye las imágenes recortadas de la hoja de trabajo o proporciona a los estudiantes algunos objetos para ordenar.

Observaciones: Voy a leer un montón de enunciados binarios en forma de “la figura equivale a un cuadrada o tiene más de 4 lados”, y ustedes van a clasificar sus objetos para organizarlos en pilas VERDADERAS y FALSAS.

Si los estudiantes no están de acuerdo con respecto a qué pila debe colocar un objeto, primero deben analizar cuál es la propiedad, cuáles son los dos resultados de la pregunta binaria y luego, si aún no pueden ponerse de acuerdo, deben llevarla a la clase para votar.

Si está utilizando los recortes proporcionados, puede comenzar con las siguientes preguntas:

- Lados es igual a 3
- El relleno es igual a negro
- Las esquinas son menos de 1

Los estudiantes a menudo luchan con la idea de hacer mayor que o igual a ser siempre FALSO- tienden a pensar en esos como declaraciones de verdad en lugar de preguntas de relación.

En esta Lección usamos el lenguaje algo es igual a otra cosa para reflejar el código que verán más adelante (por ejemplo `something == something_ else`), pero para los estudiantes que están luchando por ver esto como una pregunta en lugar de una afirmación, puede alentarlos a que reformulen la afirmación moviendo el ‘es’ al principio, entonces: algo es igual a otra cosa se convierte ¿es algo igual a otra cosa?

Al combinar booleanos con “O”, estamos utilizando lo que se llama lógica o- lo que significa que estamos preguntando si alguno (o ambos) o los booleanos son VERDADEROS. A menudo, los estudiantes piensan en O como exclusivo o- lo que significa que estamos preguntando si solo uno (pero no el otro) es VERDADERO. Asegúrese de hacer preguntas o preguntarles a los estudiantes donde los dos booleanos son VERDADEROS para asegurarse de que pueda obtener esta idea equivocada temprano.

- El ancho es igual a la altura
- El relleno es igual a gris y lados son más que 4
- Lados es mayor que 4 y menos de 7
- Lados es mayor que 4 o menor que 7
- Lados es más que o igual a 5

Condicionales

Meta: Después de acostumbrarnos a ordenar objetos en VERDADERO y FALSO, debemos presentarles a los estudiantes el concepto de que los Booleanos también se pueden usar para controlar el flujo de un programa.

Observaciones: Una condicional nos permite tomar una decisión basada en el resultado de una pregunta (o condición) booleana. De hecho, estábamos usando implícitamente condicionales en la Lección de Pararse/Sentarse porque había una acción relacionada con cada resultado potencial del booleano. Podríamos haber reformulado las instrucciones como si la declaración es verdadera: permanece de pie de lo contrario; siéntate.

Preguntar: Selecciona un objeto de tu montón y sostenlo.

Observaciones: Voy a hacerles una pregunta booleana sobre su objeto y darles una acción relacionada con el resultado del booleano. Averigüe cuál debería ser su respuesta para su forma y haga la respuesta correcta.

Preguntar: Haga a los estudiantes algunas preguntas booleanas sobre ese único objeto y deles algo que hacer si esa pregunta es verdadera. Por ejemplo:

- Si los lados son iguales a 4, haz un baile.
- Si el patrón es igual a rayado, siéntate.
- Si el ancho es igual a la altura, salta sobre un pie.

Transferencia del conocimiento (5 min)

Condición explícita

Objetivo: Los booleanos y condicionales son en realidad algo que utilizamos en nuestra vida cotidiana, simplemente no solemos ser explícitos al respecto.

Modelo: Como una forma de practicar el pensamiento de forma explícita sobre condicionales, considera despedir a tus estudiantes utilizando compuestos booleanos y condicionales. Por ejemplo.

- Si te sientas en la mesa cuatro y tu cabello es marrón, puedes irte.
- Si tu primer nombre comienza con A o B, puedes irte.
- Si tus zapatos son negros, puedes irte.

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Organizan objetos en base a criterios simples
- Describen el proceso de trabajo para la organización de los elementos con los que trabajan.

Lección 15

Booleanos y condicionales

Propósito

Los estudiantes comienzan usando booleanos para comparar el valor actual de una propiedad de sprite con un valor objetivo, usando esa comparación para determinar cuándo un sprite ha alcanzado un punto en la pantalla, crecido a un tamaño determinado o alcanzado un valor usando el patrón contrario. Después de usar booleanos directamente para investigar los valores o las propiedades de sprite, los estudiantes agregan sentencias condicionales “if” (si) para escribir código que respondan a esas comparaciones booleanas.

Esta lección sigue de cerca el modelo booleano que los estudiantes experimentaron por primera vez en la lección Booleanos desconectada. Como antes, comenzamos con el uso de booleanos directamente antes de usar booleanos para desencadenar sentencias if. En la siguiente lección, presentaremos algunos bloques de producción booleana, como `keyDown()`, que se pueden usar en lugar de comparaciones booleanas simples para escribir programas que respondan a la entrada del usuario.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Predecir el resultado de declaraciones booleanas simples.
- Usar condicionales para reaccionar a los cambios en las variables y las propiedades de los sprites

Lección en línea
[Ver en Code Studio](#)

Vocabulario

- **Expresión Booleana:** En la programación, una expresión que se evalúa como Verdadera o Falsa.
- **If-Statement:** La estructura de programación común que implementa “declaraciones condicionales”.

Código

- If statement
- Equality operator
- Inequality operator
- Greater than operator
- Greater than or equal operator
- Less than operator
- Less than or equal operator

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Contestando preguntas Booleanas

Meta: Al final del juego, con las preguntas booleanas de la lección anterior, los estudiantes comenzaron a agregar condiciones a sus preguntas booleanas. Significa que, si la respuesta a la pregunta es verdadera, debería suceder algo. Antes de programar con condicionales, queremos asegurarnos de que los estudiantes tengan un conocimiento sólido de lo que realmente son los booleanos.

Rápido:

- ¿Cuántos números diferentes hay en el mundo?
- ¿Cuántas palabras diferentes o combinación de letras y otros personajes hay?
- ¿Cuántos valores booleanos diferentes hay?

Discutir: Los estudiantes deben darse cuenta de que las dos primeras preguntas (números y cadenas) son esencialmente infinitas, pero que los booleanos están limitados a dos estados.

Observaciones: A medida que comience a programar hoy, estará usando booleanos para hacer programas que cambien su comportamiento dependiendo de la respuesta a esas preguntas booleanas.

Ampliación del conocimiento (40 min)

Booleanos

Transición: Envía a los estudiantes a Code Studio y realice los Niveles de Code Studio- Condicionales.

- Booleanos y operadores de comparación
- Comparación Booleana
- Declaraciones “si”
- Condicionales Básicos
- Juego gratis

Aunque aparentemente es simple, entender como una declaración booleana evaluará puede ser difícil dado que los diferentes lenguajes de programación tienen opiniones diferentes sobre “veracidad” y “falsedad”. De hecho, JavaScript (el lenguaje utilizado en este curso) tiene dos operadores diferentes para probar la igualdad booleana == y ===.

El operador doble igual (==) es bastante generoso para determinar la veracidad, por ejemplo, cada uno de los siguientes se considera true (verdadero) en JavaScript cuando se usa el == operador, pero se usaría el === operador: para false

```
1=="1" verdadero
1==="1" falso
```

```
1 == verdadero;
"1" == verdadero;
5 == "5";
null == undefined; " " == falso;
```

Usamos el == operador en este curso porque es más indulgente, pero es importante tener en cuenta que a veces puede informar la verdad cuando en realidad no tenía la intención de hacerlo (en cuyo caso podría querer usar el === operador más estricto)

Transferencia del conocimiento (5 min)

Agregar condicionales

Reflexión: Piense en todos los programas que ha escrito hasta ahora;

¿Cómo podrías usar condicionales para mejorar uno de tus programas de actividades pasadas?

¿Qué condición verificaría y cómo respondería a ella?

Sugerencias de evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Representan con ejemplos simples la lógica booleana
- Combinan diversos criterios de búsqueda utilizando los operadores booleanos
- Incorporan preguntas condicionales en la construcción de sus soluciones

Lección 16

Condicionales y entrada del usuario

Propósito

Después de la introducción a las declaraciones booleanas y si en la Lección anterior, se introduce a los estudiantes en un nuevo bloque llamado `keyDown()` que devuelve un booleano y se puede usar en sentencias de condicionales para mover sprites alrededor de la pantalla. Al final de esta lección, los estudiantes tendrán programas escritos que tomarán la entrada de teclado del usuario para controlar los sprites en la pantalla.

Una forma común de usar condicionales es verificar los diferentes tipos de entrada del usuario, especialmente las pulsaciones de teclas. Tener una forma para que un usuario interactúe con un programa, lo hace más interesante y dinámico. Sin interacción del usuario, es muy difícil crear un juego. Por lo tanto, la introducción de condicionales y aportes de los usuarios para la toma de decisiones es el primer gran paso hacia la creación de juegos.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Usar condicionales para reaccionar a la entrada del teclado
- Mover los sprites en respuesta a la entrada del teclado.

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Expresiones booleanas- video](#)
- [Contenido de Lección](#)

Código

- `keyDown(code)`

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Tomando entrada

Debate: Hasta ahora, todos los programas que ha escrito se ejecutan sin intervención del usuario. ¿De qué manera la adición de la interacción del usuario puede hacer que sus programas sean más útiles, efectivos o entretenidos? ¿Cómo podría un usuario proporcionar información sobre su programa?

Ampliación del conocimiento (40 min)

Entrada de teclado

Transición: Enviar los estudiantes a Code Studio, contenido de Lección

- Entrada de teclado
- Edición de imágenes
- Desafío: Agregar más entradas

Transferencia del conocimiento (5 min)

Considerando las condiciones

Indicación: Para que los estudiantes continúen pensando en cómo se pueden usar los condicionales en la programación, pida que ideen escenarios en juegos o programas que usan regularmente que pueden ser activados por los condicionales.

Discute: Haga que los estudiantes compartan las respuestas. Las respuestas de los estudiantes pueden incluir:

- Si mi nombre de usuario y contraseña son correctos, conéctate a Facebook
- Si Pacman ha reunido todas las bolas, comienza el siguiente nivel
- Si mi teclado o mouse no se movió en 10 minutos, encienda el protector de pantalla.

El objetivo aquí no es entrar en las especificaciones técnicas de cómo los programas pueden recibir información (los estudiantes entenderán eso en la parte en línea de la Lección), sino más bien hacer que los estudiantes piensen en cómo permitir que los usuarios que ingresen puedan cambiar los programas que ellos 'han hecho'. Anime a los estudiantes a pensar en las diversas entradas y salidas informáticas que existen ¿Qué entradas serían más útiles para los tipos de programas que han estado haciendo?

Sugerencias de evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Usan sentencias condicionales en un proyecto de programación.
- Permitir que los usuarios que interactúen con el proyecto puedan describir la lógica programada.

Lección 17

Otras formas de entrada

Propósito

En esta lección, los estudiantes continúan explorando maneras de usar declaraciones condicionales para tomar la opinión del usuario. Además del comando `keyDown()` simple aprendido ayer, los estudiantes aprenderán sobre varios otros comandos de entrada de teclado, así como formas de tomar la entrada del mouse.

Los estudiantes han aprendido cómo tomar decisiones simples con condicionales. A veces, sin embargo, queremos tomar una decisión en función de si la condición sobre la que preguntamos originalmente era falsa o si queremos tomar una decisión basada en que múltiples condiciones sean ciertas. Ahí es, donde aparecen enunciados y condicionales más complejos. Estas declaraciones son una segunda declaración adjunta a una instrucción `if` (si). Otras declaraciones se ejecutan cuando la sentencia `if` (si) a la que está asociada es falsa. Puedes pensarlo como “si algo es verdadero haz cosa 1 sino (else) haz cosa 2”.

Este concepto se presenta junto con varios comandos nuevos de ingreso de teclas y mouse, lo que permite a los estudiantes crear gradualmente programas que ingresan de diferentes maneras.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (5 min)

Lección en línea
[Ver en Code Studio](#)

Vocabulario

- **Condicionales:** declaraciones que solo se ejecutan bajo ciertas condiciones.

Código

- `keyWentDown(code)`
- `keyWentUp(code)`
- `mouseDidMove()`
- `mouseDown(button)`
- `mouseWentDown(button)`
- `MouseWentUp(button)`
- `sprite.visible`
- `If/else statement`

Objetivos

Los estudiantes serán capaces de:

- Usar una instrucción else (sino) como el caso de respaldo a una sentencia if (si).
- Diferenciar entre las condiciones que son verdaderas una vez por interacción y aquellas que permanecen verdaderas a lo largo de la duración de una interacción.

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Verificar la comprensión

Observaciones: Hoy retomaremos donde lo dejamos en la lección anterior, usando condicionales para escribir programas que respondan a los comentarios de los usuarios. Actualicemos lo que aprendimos.

Rápido:

- ¿Qué es un booleano? (respuesta; un valor verdadero / falso)
- ¿Cuál es la relación entre un booleano y un condicional? (resp; un condicional hace una pregunta booleana y ejecuta el código si la respuesta es verdadera)
- ¿Cuáles son algunos ejemplos de operadores de comparación que dan como resultado un booleano? (resp., >, <, ==)
- ¿Cuál es la diferencia entre = y ==? (resp., = se usa para asignar un valor, == se usa para verificar si dos valores son iguales).

Ampliación del conocimiento (40 min)

Sí / más y más entradas

Video: Mire el video de condicionales (conditional) juntos como una clase. Será una revisión de las declaraciones if (si) e introducirá también algunos conceptos nuevos.

Transición: Enviar estudiantes a Code Studio Contenido de la Lección

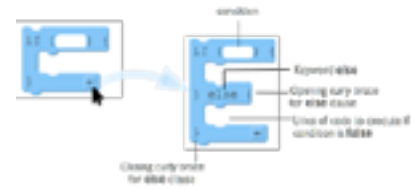
- Sigue al mouse
- Video: Condicionales
- Declaraciones de If-Else (Si-Sino)
- Entrada con If-Else (Si-Sino)
- Desafío: verificar múltiples condiciones

If-Else (Si-Sino) Declaraciones

Cómo funcionan las declaraciones If-Else (Si-Sino)

Con una instrucción if-else, usted está dando un comando cualquiera-or (o): en el cual se ejecutarán las líneas de código dentro de if o las líneas dentro de else. Esas son las opciones.

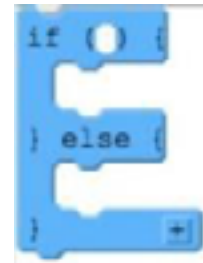
En el video (recuerde activar los subtítulos) se muestra cómo agregar una cláusula else a una instrucción if: pulsa el pequeño símbolo + en la cola de la declaración if.



Dentro de las llaves de la cláusula else, coloque las líneas de código que deseas ejecutar si la condición booleana de la declaración if es falsa. Algunas notas importantes sobre la cláusula else:

- El else debe venir inmediatamente después de la llave de cierre de una declaración if.
- El otro else también tiene su propio conjunto de llaves de apertura y cierre para encapsular líneas de código.
- ¿Qué pasa si mi condición no es verdadera?

A veces queremos decirle a nuestro programa qué hacer si se trata de una condición true, pero también qué hacer si es así false. Al presionar el botón más en la parte inferior de su bloque condicional le dará otra sección llamada else. Esta sección else es una alternativa: se llamará siempre que exista la condición if anterior que sea falsa.



Clics del mouse

Las pulsaciones de teclas son geniales, pero a veces desea que los usuarios interactúen a través de los clics del mouse. Hay un nuevo bloque llamado mouseDown() que, similar a keyDown(), comprueba si se presionan los botones izquierdo o derecho del mouse. Si está usando una computadora con un mouse o un panel táctil que solo tiene un botón, querrá usar siempre mouseDown("left").

mouseDidMove (el mouse se movió)

También podemos usar expresiones booleanas para verificar si el mouse se movió o no. El bloque mouseDidMove devolverá false si el mouse está quieto, pero true si el mouse se ha movido.

Transferencia del conocimiento (5 min)

Compartir

El último programa de esta Lección consiste en pedir a los estudiantes que amplíen un programa a usos condicionales para la participación del usuario de maneras nuevas e interesantes.

Haga que sus estudiantes compartan sus proyectos con la clase, lo que les pedirá que expliquen cómo usaron los condicionales en sus programas.

Sugerencias de evaluación

Se sugieren los siguientes indicadores para evaluar formativamente los aprendizajes:

- Crean variables con nombres claros que expliquen tipos de datos
- Incorporan condicionales y explican la funcionalidad de forma predictiva
- Prueban y perfeccionan sistemáticamente los programas utilizando una serie de casos de prueba

Lección 18

Proyecto – Tarjeta interactiva

Propósito

En este proyecto, los estudiantes planifican y desarrollan una tarjeta de felicitación interactiva usando todas las técnicas de programación que han aprendido hasta este momento.

Esta evaluación es un buen lugar para que los estudiantes junten todas las piezas que aprendieron (dibujo, variables, sprites, imágenes, condicionales, entrada del usuario) en un solo lugar. Los estudiantes todavía deberían estar trabajando con un código que sea fácil de leer y que no sea difícil de entender para ellos. Darles a los estudiantes la oportunidad de ser realmente creativos después de aprender todos estos nuevos conceptos los ayudará a involucrarse más en el contenido.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (2 días)

Transferencia del conocimiento (10 min)

Evaluación

Objetivos

Los estudiantes serán capaces de:

- Usar condicionales para reaccionar a la entrada del teclado o cambios en las variables / propiedades.
- Usar Comandos de secuencia para dibujar en el orden correcto.
- Aplicar un patrón de iterador a las variables o propiedades en un ciclo.

Preparación

- Tarjeta Interactiva- [Guía de actividades.](#)

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Tarjeta interactiva](#)
- [Contenido de la Lección- Code.org](#)

Para los estudiantes:

- [Tarjeta Interactiva - Guía de actividades](#)
- [Tarjeta interactiva- Rúbrica](#)
- [Tarjeta interactiva - Revisión por pares](#)

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Meta: Los estudiantes ven un ejemplo de un proyecto final y discuten los diferentes elementos que se utilizaron para hacerlo.

Demo Ejemplo de proyecto (Nivel 2)

Pantalla: En el proyector, ejecute el ejemplo para los estudiantes. Dado que esto está en el nivel 2 del progreso, si es más fácil para los estudiantes hacerlo en sus propias computadoras, también podría hacer eso.

Niveles de Code Studio

- Ejemplo de tarjeta interactiva CSD U3

Proyecto de ejemplo

Ejecute el programa varias veces y responda las siguientes preguntas:

1. ¿Qué elementos parecen usar comandos de dibujo?
2. ¿Qué elementos parecen ser Sprites?
3. Para cada Sprite, ¿qué propiedades se están actualizando?
4. ¿Dónde ves condicionales que se utilizan?
5. ¿Hay elementos que no entiendes?

Discute: Haga que los estudiantes compartan sus observaciones y análisis del ejemplar.

Alienta a la clase a considerar que existen múltiples enfoques para programar cualquier cosa, pero que puede haber pistas sobre cómo se creó algo. En particular, cuando comparten sus ideas, pida que especifiquen lo siguiente:

- Pistas que sugieren que se utilizó un Sprite
- Se usaron pistas que sugieren un condicional
- Se usaron pistas que sugieren un patrón de iterador

Pantalla: Muestra a los estudiantes la tarjeta interactiva- rúbrica. Revisa los diferentes componentes de la rúbrica con ellos para asegurarse de que entienden los componentes del proyecto.

Ampliación del conocimiento (2 días)

Meta: Los estudiantes deben planear lo que quieren crear antes de dirigirse a la computadora para que, una vez que lleguen a ella, simplemente estén ejecutando el plan.

Desconectado: planificación interactiva de tarjetas

Distribuir: Entregue la [Tarjeta Interactiva - Guía de actividades](#) para los estudiantes. Ésta es la herramienta que los estudiantes usarán para determinar sus proyectos antes de ingresar a las computadoras. Deles tiempo para que hagan una lluvia de ideas sobre el tipo de tarjeta que quieren crear y quién será el destinatario.

Pasos:

1. La primera capa de la tarjeta interactiva es un fondo dibujado con sólo los comandos en el cajón de Dibujo. El anverso de la Guía de actividades proporciona una cuadrícula para que los estudiantes expongan sus fondos, una tabla de referencia de comandos de dibujo y un área para que los estudiantes tomen notas y escriban pseudocódigos.
2. Los siguientes estudiantes piensan en los Sprites que necesitarán, completando una tabla con la etiqueta, las imágenes y las propiedades de cada Sprite.
3. Finalmente, los estudiantes consideran los condicionales que necesitarán para hacer su tarjeta interactiva.

Niveles: implementación de la tarjeta interactiva (nivel 3- 7)

Transición: Una vez que los estudiantes hayan completado su hoja de planificación, es hora de dirigirse a las lecciones de [Code studio](#). La secuencia de nivel corto les pide a los estudiantes completar cada elemento de su proyecto.

Revisión por pares

Distribuir: Entregue a cada alumno una copia de la [Tarjeta interactiva- Revisión por pares](#).

Los estudiantes deben pasar 15 minutos revisando la tarjeta del otro alumno y completando la guía de revisión por pares.

Iterar- Actualizar código

Recorra la sala: Los estudiantes deben completar la guía de la revisión por pares y decidir cómo responder a los comentarios que se les dieron. Luego deberían usar ese comentario para mejorar sus cartas.

Reflexionar

Usando la [Tarjeta interactiva- Rúbrica](#), los estudiantes deben evaluar su propio proyecto antes de enviarlo.

Envíe a los estudiantes a Code Studio para completar su reflexión sobre sus actitudes hacia la informática. Aunque sus respuestas son anónimas, los datos agregados estarán disponibles una vez que al menos cinco estudiantes hayan completado la encuesta.

Transferencia del conocimiento (10 min)

Compartir tarjetas

Meta: Los estudiantes comparten sus creaciones con la clase.

Compartir: Encuentre una forma para que los estudiantes compartan sus tarjetas entre ellos y con el destinatario deseado. Es probable que sea útil utilizar el enlace compartido para el proyecto, de modo que los estudiantes puedan compartir el proyecto con otros estudiantes.

Evaluación

Se proporciona una rúbrica para evaluar los proyectos de los estudiantes en los recursos.

Contenido 3

Creación de videojuegos

Resumen

- **Lección 19:** Velocidad
- **Lección 20:** Detección de colisión
- **Lección 21:** Movimiento complejo de Sprite
- **Lección 22:** Colisiones
- **Lección 23:** Funciones
- **Lección 24:** El proceso de diseño del juego
- **Lección 25:** Uso del proceso de diseño del juego

Objetivos

- **OA 1.** Aplicar conceptos de Ciencias de la Computación –abstracción, organización lógica de datos, análisis de soluciones alternativas y generalización– al crear el código de una solución computacional.
- **OA g.** Elaborar representaciones, tanto en forma manual como digital, y justificar cómo una misma información puede ser utilizada según el tipo de representación.

Referencias

- **Code Studio**- Code.org
- **Cuantrix**- Fundación Televisa
- **CSTA** – Computer Science Teachers Association

Lección 19

Velocidad

Propósito

Después de una breve revisión de cómo utilizaron el patrón de contador para mover los sprites en las lecciones anteriores, los estudiantes reciben las propiedades que establecen la velocidad y rotación. A medida que utilizan estas nuevas propiedades de diferentes maneras, desarrollan las habilidades necesarias para crear un juego básico de desplazamiento lateral.

En esta Lección, se les enseña a usar los bloques de velocidad para simplificar el código para mover un sprite a través de la pantalla. Esto marca un cambio en cómo se introducen los nuevos bloques. Mientras que los bloques anteriores se presentaron como habilitantes de comportamientos completamente nuevos, ahora se presentan como un código simplificador que los estudiantes podrían haber escrito con los bloques disponibles anteriormente. En las siguientes lecciones, los estudiantes verán cómo este método les permite producir comportamientos de sprites más interesantes.

Secuencia para el aprendizaje

Conocimiento inicial (15 min)

Ampliación del conocimiento (75 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Usar los bloques velocidad y rotación para crear y cambiar los movimientos de los sprites.
- Describir las ventajas de simplificar el código mediante el uso de bloques de mayor nivel.

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Contenido de la Lección- Code.org](#)

Código

- `sprite.rotationSpeed`
- `sprite.velocityX`
- `sprite.velocityY`

Estrategia de aprendizaje

Conocimiento inicial (15 min)

Demuestre: Pida que un voluntario pase al frente de la clase y actúe como su sprite. Digamos que le dará instrucciones al sprite como si fuera un programa de Game Lab.

Cuando su estudiante esté listo, organice el lugar para que tengan algo de espacio frente a ellos y pida que “avancen por 1”. Deben dar un paso adelante. Luego repita el comando varias veces, cada vez que esperas que el estudiante avance un paso. Debería apuntar a que la repetitividad de estas instrucciones sea clara. Después de que tu estudiante haya completado esta Lección, pide que vuelvan a donde comenzaron. Esta vez repite la demostración, pero solicita al alumno que “Avance por 2” y haga que el alumno tome 2 pasos cada vez. Una vez que el alumno haya hecho esto varias veces, pida a la clase que dé una ronda de aplausos e invita a los estudiantes volver a su asiento.

Preguntar: Sólo estaba dando instrucciones a mi Sprite, pero parecían ser bastante repetitivas. ¿Cómo podría haber simplificado o simplificado mis instrucciones?

Discute: Dé a los estudiantes un minuto para escribir sus pensamientos antes de invitarlos a compartir con un compañero. Luego haga que la clase comparta sus pensamientos. Puede escribir sus ideas en el pizarrón.

Observaciones: Una forma de simplificar estas instrucciones es decirle a nuestro sprite que siga moviéndose por 1 o 2, o por todos los pasos que queramos. Esto haría las instrucciones más fáciles de entender para los humanos, y como estamos a punto de ver, también hay una manera similar de simplificar nuestro código.

Muestra: Video de Sprite Velocity.

Pantalla: En el pizarrón, escriba las siguientes piezas de código.

- `sprite.velocityX = 4;`
- `sprite.velocityY = -1;`
- `sprite.rotationSpeed = 2;`

Indicación: Acabamos de ver cómo estos nuevos bloques nos

Meta: la demostración anterior debería haber reforzado el hecho de que repetir las mismas instrucciones es algo que nunca harías en la vida real. En su lugar, se le ocurriría una forma de capturar que las instrucciones deberían repetirse, como “seguir avanzando por 1”

Comprobar la comprensión: esto es solo una comprobación rápida para entender después del video. Se les pide rápidamente a los estudiantes que vean si entendieron el punto principal del video y, si no, tienen la oportunidad de reforzarlo antes de pasar a los niveles de Code Studio. Deje la traducción de los bloques de velocidad a si patrón de contador asociado en el tablero para hacer referencia a lo largo de la Lección

ayudan a simplificar el patrón de contador que usamos para mover los sprites. En una hoja de papel, anote el patrón de contador que está reemplazando cada una de estas líneas de código.

Discute: Haga que los estudiantes escriban sus ideas en una hoja de papel antes de hablar sobre sus respuestas como clase. Finalmente, escriba las respuestas correctas en el tablero al lado de cada una, como se muestra a continuación.

- `sprite.x = sprite.x + 4;`
- `sprite.y = sprite.y - 1;`
- `sprite.rotation = sprite.rotation + 2;`

Observaciones: Estos nuevos bloques de “nivel superior” nos ayudan a escribir código que ya sabíamos cómo escribir. Están simplificando nuestro código para nosotros al ocultar algunos de los detalles innecesarios. Sin embargo, como estamos a punto de ver, estos nuevos bloques cambiarán el tipo de programas y juegos que podemos escribir.

Ampliación del conocimiento (75 min)

Aprendiendo a usar los bloques de velocidad

Transición: Los estudiantes se dirigen a Code Studio, [Contenido de la Lección-Code.org](#).

Recorra la sala: Estos niveles introducen las propiedades de velocidad X, velocidad Y velocidad de rotación que acaba de analizar con los estudiantes. Verifique con los estudiantes para ver cómo están y mantenga un registro de cuándo todos han llegado al final del nivel 10.

VelocidadX

Una forma de mover los sprites en Game Lab es con el patrón de contador. Por ejemplo, `sprite1.x = sprite1.x + 1` mueve un sprite por 1 píxel en cada fotograma del ciclo de dibujo. Este patrón es tan común que los sprites tienen una `velocityX`, propiedad que hace esto por ti.

Velocidad de rotación

Ya ha aprendido cómo hacer que su sprite gire usando el `rotation` bloque. Por ejemplo, cuando quería que su sprite girara dos grados cada vez que se dibujaba, colocabas `sprite.rotation = sprite.rotation + 2` dentro del círculo de extracción.

¿Dentro o fuera del lazo de dibujo?: Para los primeros rompecabezas, asegúrese de que los estudiantes estén configurando las velocidades y velocidades de rotación fuera del círculo de dibujo, inmediatamente después de que creen sus sprites. El código funcionará para los primeros rompecabezas, incluso si establecen las velocidades dentro del bucle de extracción, pero causará problemas más adelante.

Ahora puede usar los `rotationSpeed` para hacer que sus sprites giren una cierta cantidad cada vez que se dibujan. Si desea que su sol gire dos grados cada vez que se dibuja, puede usar `sun.rotationSpeed = 2` antes del ciclo de extracción, después de crear su sprite.

Control de velocidad

Usó `rotationSpeed` fuera del lazo de extracción para hacer que tu sprite girara cuando tu programa comienza. También puedes usar `rotationSpeed` dentro del ciclo de dibujo para cambiar la velocidad del sprite durante el juego. Por ejemplo, un Sprite puede comenzar a girar cuando el usuario presiona la barra espaciadora, y seguirá girando hasta que se le indique que se detenga.

Cambiando la velocidad con la posición

Una ventaja del uso de bloques de velocidad dentro de condicionales (if bloques) es que tu sprite se mantiene en movimiento, incluso después de que la condición deje de ser cierta.

Si los estudiantes terminan temprano, anímelos a experimentar con diferentes aspectos del juego de desplazamiento que crearon. Pueden cambiar las animaciones, crear un fondo, etc.

Transferencia del conocimiento (5 min)

Indicación: Hoy aprendieron algunos bloques nuevos. A primera vista, estos bloques hicieron el mismo tipo de cosas que ya habíamos hecho con el patrón de contador, pero nos simplificaron hacerlas. Sin embargo, al pasar por los rompecabezas, comenzaron a hacer algunos movimientos interesantes que no habíamos podido hacer antes.

- Describe uno de esos movimientos y cómo lo hiciste.
- Describe otro movimiento que te gustaría hacer, pero aún no sabes cómo.
- Describe otro bloque que te gustaría tener.
- ¿Tomaría algún argumento?
- ¿Qué haría?
- ¿Qué código escondería dentro?

Observaciones: Todos los movimientos que hicimos hoy son posibles sin los nuevos bloques, pero sería muy complicado codificarlos. Uno de los beneficios de bloques como la velocidad es que cuando no tenemos que preocuparnos por los detalles de los movimientos y acciones simples, podemos usar esa potencia cerebral adicional para resolver problemas más complicados. A medida que desarrolle su juego de desplazamiento lateral, seguiremos buscando nuevos bloques que simplifiquen las cosas, para que podamos construir juegos cada vez más complicados.

Sugerencias para evaluación

Se sugieren los siguientes indicadores para evaluar formativamente los aprendizajes:

- Utilizan bloques de velocidad y rotación, cambiando el movimiento de los sprites.
- Incorporan el código, los medios y las bibliotecas existentes en los programas originales, y le dan atribución.
- Prueban y perfeccionan sistemáticamente los programas.
- Documentan aquellos programas para que sean más fáciles de seguir, probar y depurar.

Lección 20

Detección de colisión

Propósito

Los estudiantes aprenden sobre la detección de colisión en la computadora. Trabajando en parejas, exploran cómo una computadora puede usar la ubicación de sprites y propiedades de tamaño y matemáticas para detectar si dos sprites se están tocando. Luego usan el bloque `isTouching()` para crear diferentes efectos cuando los sprites colisionan, incluidos los sonidos de reproducción. Por último, usan sus nuevas habilidades para mejorar el juego de desplazamiento lateral que comenzaron en la última Lección.

Esta Lección introduce formalmente el uso de abstracciones, formas simples de representar la complejidad subyacente.

En la última Lección, los estudiantes fueron expuestos a la idea de usar un bloque para representar código complejo. Los estudiantes exploran más a fondo esta idea en el contexto del desafío matemático complejo e intencional de determinar si dos sprites se están tocando. Al usar un solo bloque para representar esta complejidad, en este caso, el bloque `isTouching`, se vuelve mucho más fácil escribir y razonar sobre el código, y los estudiantes pueden apreciar el valor de usar abstracciones. En lecciones posteriores, los estudiantes continuarán construyendo sobre la abstracción `isTouching()` para crear interacciones de sprites más complejas.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Detección de colisiones - contenido de la Lección](#)
- [Detección de colisiones - Guía de actividades](#)

Vocabulario

- **Abstracción:** una representación simplificada de algo más complejo. Las abstracciones le permiten ocultar detalles para ayudarlo a manejar la complejidad, enfocarse en conceptos relevantes y razonar sobre problemas en un nivel superior.
- **Depuración:** encontrar y solucionar problemas en un algoritmo o programa.
- **If-Statement:** la estructura de programación común que implementa “declaraciones condicionales”.

- Utilizar el bloque `isTouching` para determinar cuándo se tocan dos sprites.
- Describir cómo las abstracciones ayudan a manejar la complejidad del código.

Preparación

- [Detección de colisiones - Guía de actividades](#) de modo que cada par de estudiantes tenga una parte A y una parte B.

Código

- `sprite.rotationSpeed`
- `sprite.velocityX`
- `sprite.velocityY`

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Revisión: Recuerde brevemente a los estudiantes el juego de desplazamiento lateral que hicieron en la última Lección y pida que compartan cualquier idea que tengan para mejorar su juego en el futuro.

Pantalla: En un proyector o simplemente en una pantalla de computadora, muestre el juego que aparece en el primer nivel en Code Studio para esta lección. El maestro o un alumno puede controlar a la rana para saltar sobre un hongo. Asegúrese de que al menos algunos saltos sean exitosos y que algunos no vean que la detección de colisiones está funcionando.

Inducción: Una mejora interesante en este juego es que el hongo se mueve cuando la rana lo toca. ¿Puedes pensar en alguna forma en que la computadora pueda usar las propiedades de los sprites para averiguar si se están tocando entre sí?

Discute: Permita que los estudiantes hagan una lluvia de ideas sobre cómo la computadora podría determinar si los dos sprites se están tocando. Haga una lista de sus ideas en el pizarrón y dígalas que tendrán la oportunidad de probar sus teorías en un momento.

Ampliación del conocimiento

Colisiones desconectadas

Grupo: Agrupa a los estudiantes en parejas.

Observaciones: Ahora va a tener la oportunidad de probar las estrategias que surgieron como grupo. Cada guía de actividades tiene cuatro hojas de papel. Un compañero debe tomar los papeles con la “A” en la parte superior, y el otro debe tomar los papeles con la “B” en la parte superior. Cada uno dibujará dos sprites secretos en el gráfico, y su compañero tratará de averiguar si están tocando o no según la misma información que la computadora tendrá sobre las propiedades de cada uno de los sprites, así que no permita que su compañero mire lo que estás dibujando.

Distribuir: La [Detección de colisiones - Guía de actividades](#) para cada pareja. Asegúrese de que un compañero haya tomado las páginas con “A” en la parte superior y el otro haya tomado las páginas con “B” en la parte superior.

Mostrar el juego: en este punto, sólo muestre el juego. Tienen una actividad desconectada bastante importante inmediatamente después y importante superarla antes de interactuar con el juego.

Meta: El propósito de esta discusión es solo para obtener algunas ideas en la pizarra para que los estudiantes las utilicen en la próxima Lección. No es necesario evaluarlos o probarlos, porque los estudiantes trabajarán juntos para hacerlo inmediatamente después de la discusión

Cada alumno tendrá una línea para dibujar dos cuadrados. El estudiante elige la ubicación y el tamaño de cada uno de los cuadrados, y luego registra la información sobre los cuadrados en una tabla. A continuación, cambian las tablas (no los dibujos) y tratan de determinar si los dos sprites se están tocando en función del ancho de cada sprite y la distancia entre ellos.

La matemática para determinar si los sprites se están tocando es la siguiente:

- Resta las posiciones x (o y) de los sprites para encontrar la distancia entre sus centros.
- Divide el ancho (o la altura) de cada cuadrado por 2 para obtener la distancia desde el centro hasta el borde.
- Si la distancia entre los centros de los sprites es mayor que la suma de las distancias desde sus centros hasta sus bordes, los sprites no se tocan.
- Si la distancia entre los centros de los sprites es igual a la suma de las distancias desde sus centros hasta sus bordes, los sprites apenas se tocan.
- Si la distancia entre los centros de los sprites es mayor que la suma de las distancias desde sus centros hasta sus bordes, los sprites se superponen.

Recorra la sala: Apoye a los estudiantes mientras completan la hoja de trabajo. Si los estudiantes no están seguros de cómo determinar si los sprites se están tocando, aliéntelos a usar una de las ideas en el pizarrón. Recuerde que no están siendo calificados sobre si tienen razón o no, sino sobre su capacidad para usar el proceso de resolución de problemas. Si alguno de los estudiantes termina antes de tiempo, guíelos para que encuentren un método que funcione para sprites en cualquier lugar de la cuadrícula, no solo en la misma línea.

Compartir: Después de que todos los estudiantes hayan tenido la oportunidad de probar sus soluciones, pida que compartan lo que descubrieron.

Observaciones: La gente puede usar muchas estrategias diferentes para resolver un problema como éste. Debido a que las computadoras no pueden “ver” los dibujos de la misma manera que las personas, necesitan usar las matemáticas para determinar si dos cosas se están tocando. Veamos cómo esto puede funcionar a lo largo de una línea, pero podemos combinar estos métodos para trabajar en cualquier parte de la pantalla del juego.

IsTouching()

Transición: Haga que la clase inicie sesión en la [Unidad 3- Niveles de Code Studio para la Lección 16](#).

Pantalla: Muestre la burbuja 2 y permita que los estudiantes observen y analicen por parejas cómo podrían codificar el comportamiento que ven en el programa.

Niveles de Code Studio

El propósito de este nivel es que los estudiantes vean que pueden probar si dos sprites se están tocando con los bloques que ya están disponibles para ellos. El código es complicado, pero solo usa bloques que los estudiantes ya han aprendido a detectar cuando dos sprites se están tocando.

Discuta: Pregunta a los estudiantes cómo se relaciona el código a continuación con lo que hicieron en la lección desconectada.

Observaciones: Aunque este código funciona, puede ser difícil de leer, y sería fácil cometer un error al escribirlo. También tomaría algo de tiempo escribir cada vez, por lo que un programador guardó el código en un bloque que podemos usar para determinar si dos sprites se están tocando, y no necesitamos escribir todo este código. Debido a que esto es algo que un programador quiere hacer una y otra vez, alguien ha creado un bloque llamado `isTouching()` que ya tiene todas las matemáticas dentro de él. Ocultar los detalles de cómo se hace algo para facilitar la programación se llama abstracción.

IsTouching()

Escribir las matemáticas cada vez que desees comprobar si dos sprites se están tocando puede llevar un tiempo, por lo que un programador creó el bloque `isTouching`, que puede comprobar si un sprite está tocando otro sprite (el objetivo). La computadora sigue haciendo los mismos cálculos que en el programa anterior, pero no tiene que preocuparse porque otro programador ya hizo ese trabajo.

Recorra la sala: Apoye a los estudiantes mientras trabajan en los niveles 3-8. El nivel 8 permite a los estudiantes trabajar en el desplazamiento lateral que crearon en la última lección. Es muy abierto, así que anima a los estudiantes a ser creativos y tratar de codificar las interacciones de los sprites que no han visto antes. Los estudiantes que completan el nivel temprano pueden continuar agregando nuevas interacciones o mejorando sus juegos de otras maneras.

Transferencia del conocimiento (5 min)

Preguntar:

¿Cuáles fueron algunas de las cosas diferentes que tus sprites hicieron cuando interactuaron?

¿Cuál es un tipo de interacción que te gustaría, pero que aún no has visto?

¿Tienes alguna idea de cómo podrías escribir el código para ese tipo de interacción?

Compartir: Permita a los estudiantes compartir el diferente tipo de interacciones que les gustaría ver. Hágales saber que aprenderán otros métodos para que los sprites interactúen más adelante.

Sugerencia para evaluación

Se sugiere los siguientes indicadores para evaluar formativamente los aprendizajes:

- Construyen y evalúan estrategias de manera colaborativa al resolver problemas no rutinarios.
- Incorporan el código, los medios y las bibliotecas existentes en los programas originales, para darles atribución.
- Prueban y perfeccionan sistemáticamente los programas.
- Documentan aquellos programas para que sean más fáciles de seguir, probar y depurar.

Lección 21

Movimiento complejo de Sprite

Propósito

Los estudiantes aprenden a combinar las propiedades de velocidad de los sprites con el patrón de contador para crear movimientos de sprites más complejos. En particular, los estudiantes aprenderán cómo simular la gravedad, hacer un sprite y permitir que un sprite flote hacia la izquierda o hacia la derecha. En los niveles finales de Code Studio, los estudiantes combinan estos movimientos para animar y controlar un solo sprite y crear un juego simple en el que un personaje vuela y recoge una moneda. Se anima a los estudiantes a hacer sus propias adiciones al juego en el nivel final.

Esta lección muestra la combinación de herramientas, en particular las abstracciones aprendidas en lecciones anteriores, les permite construir nuevos comportamientos para sus sprites. Esto resalta el punto más amplio de que las abstracciones no sólo simplifican el código, sino que también pueden usarse como componentes básicos de un comportamiento aún más complejo.

La Lección presenta algunas de las programaciones más desafiantes. Los estudiantes combinarán múltiples construcciones de programación, incluyendo las propiedades de velocidad, el patrón del contador, la interacción del usuario y la detección de colisiones. Los patrones que los estudiantes usan en esta Lección generalmente son útiles para construir juegos y los estudiantes pueden reutilizarlos más adelante.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (60 min)

Transferencia del conocimiento (10 min)

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Contenido de la Lección- code.org](#)

Objetivos

Los estudiantes serán capaces de:

- Usar velocidad de sprite con el patrón de contador para crear diferentes tipos de movimiento de sprite.
- Explicar cómo las construcciones de programación individual pueden combinarse para crear un comportamiento más complejo.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Comentario: Los cuatro conceptos principales que los estudiantes aprenderán en la Lección de hoy son `isTouching()`, `keyDown()`, `sprite.velocityX/ sprite.velocityY`. Pida a los estudiantes que hablen con un compañero y recuerden mutuamente para qué se usa cada una de estas cuatro construcciones y cómo funcionan. Como clase, defina cada uno según lo conversado.

Observaciones: En las últimas actividades aprendimos mucho de construcciones de programación poderosas que nos permitieron crear juegos mucho más interesantes. Hoy vamos a explorar cómo la combinación de estas nos dará aún más control sobre los tipos de juegos que podemos hacer.

Ampliación del conocimiento (60 min)

Transición: Dirija los estudiantes a Code Studio, Contenido de la Lección - `code.org`. Con la excepción de la discusión posterior al primer nivel, los estudiantes trabajarán en Game Lab hasta el final de la lección.

Resumen: La clase aprende a combinar las propiedades de velocidad de los sprites con el patrón de contador para crear un movimiento de sprite más complejo, como simular la gravedad, hacer que un sprite salte y permitir que un sprite flote hacia la izquierda o hacia la derecha. En los niveles finales, la clase combina estos movimientos para animar y controlar un solo sprite y construir un juego simple en el que un personaje vuela y recoge monedas.

Discute: Después de que los estudiantes hayan hecho predicciones sobre cómo se ejecutó rápidamente el código en el primer nivel, analicen por qué vieron que el automóvil comenzó a moverse más rápido. Ten en cuenta que mientras que antes usaban el patrón de contador para aumentar la posición de un automóvil, ahora se usa para aumentar la velocidad del automóvil.

Niveles 2-4: Estos niveles les dan a los estudiantes la práctica de usar la velocidad dentro del patrón del contador.

Este nivel introduce el nuevo patrón de programación principal de esta Lección, combinando el patrón de contador con las propiedades de velocidad de los sprites. Anime a los estudiantes a tomar en serio sus predicciones antes de ejecutar realmente el código.

Velocidad y el patrón contador

Usar una `sprite.velocityX` propiedad con el patrón de contador cambiará la velocidad de un sprite durante el programa. Esto hace

que el sprite se acelere. Practica un poco usando este patrón tú mismo.

Niveles 5 - 6: Estos niveles introducen el uso de la velocidad y el patrón del contador para ralentizar un sprite, eventualmente moviéndolo en la dirección opuesta. Esto lleva a la introducción de cómo este patrón podría usarse para simular la gravedad.

Niveles 7 a 9: Los estudiantes comienzan a trabajar en un juego de volante. En estos niveles usan las construcciones de programación que han aprendido para hacer que su personaje principal se mueva. El personaje responde a gravedad simulada, saltos y flotantes a izquierda y derecha.

Saltando

Aumentar la velocidad de un sprite dentro del patrón de contador puede simular la gravedad. Al agregar interacciones del usuario, puedes hacer que tu sprite parezca que también salte.

Flotante a la derecha

Ahora están usando el patrón de contador con la velocidad Y del sprite para simular la gravedad y saltar. Si usan la velocidad X del sprite en el patrón contrario, pueden hacer que tu sprite flote de lado a lado también.

Niveles 10 a 12: Los estudiantes agregan una moneda al juego para que su personaje la coleccionen. En el último nivel, se les anima a actualizar el juego ellos mismos. Aprovecha esta oportunidad en particular para alentar a los estudiantes a usar otros patrones de programación que hayan aprendido, por ejemplo, crear un marcador.

Transferencia del conocimiento (10 min)

Compartir: Pida a los estudiantes que compartan con sus compañeros de clase qué adiciones hicieron a su último juego. Haga que los estudiantes se centren no sólo en cómo funciona el juego, sino también en cómo se ve el código para crear ese tipo de funcionalidad.

Preguntar: En su papel, hagan dos listas. Primero una lista de cosas nuevas que puedan programar después de la Lección de hoy. En la segunda lista anoten todos los nuevos bloques que aprendieron hoy.

Discute: Haga que los estudiantes compartan sus listas con sus compañeros de clase. Después comparte listas como una clase. Deberían haber enumerado muchos nuevos movimientos de

Objetivo: Esta conversación debe resaltar que los estudiantes no aprendieron ningún bloque nuevo en la Lección de hoy, sino que aprendieron nuevas formas de combinar bloques y patrones que habían aprendido previamente. El punto más amplio aquí es que la programación no siempre se trata de aprender nuevos bloques, sino de ser creativos al combinar las herramientas que ya sabes usar en el lenguaje.

sprites pero los estudiantes no han aprendido ningún bloque nuevo en esta Lección.

Después de avisar que todos los movimientos nuevos que crearon hoy se hicieron combinando bloques y patrones que ya aprendieron, pregunte a los estudiantes qué piensan, qué les puede decir esto sobre cómo los programadores desarrollan el código.

Indicación: Hoy hemos construido muchos movimientos de sprites nuevos, como la gravedad y el salto, pero nada de esto nos obligó a aprender nuevos bloques. ¿Crees que aprender a programar siempre significa aprender nuevos comandos?

Discute: Lidere un seguimiento rápido de su discusión inicial sobre este punto.

Observaciones: Vamos a seguir aprendiendo algunas herramientas más en Game Lab. Para crear nuevos tipos de programas, no siempre es necesario aprender nuevos bloques. La mayoría de las veces, la creatividad de la programación proviene de aprender a combinar cosas que ya conoces de maneras nuevas y creativas.

Sugerencias para evaluación

Se sugieren los siguientes indicadores para evaluar formativamente los aprendizajes:

- Crean nuevos tipos de programas, combinando elementos que ya conocen de manera nueva y creativa.
- Perfeccionan el trabajo involucrando modificaciones y testeos.

Meta: Reforzar el hecho de que aprender a programar no es solo memorizar bloques. Ser creativo con la programación a menudo significa idear maneras inteligentes de combinar los comandos y patrones que ya sabe usar.

Lección 22

Colisiones

Propósito

Los estudiantes programan sus sprites para interactuar de nuevas maneras. Después de una breve revisión de cómo usaron el bloque `isTouching`, los estudiantes intercambian ideas sobre otras formas en que dos sprites podrían interactuar. A continuación, utilizan `isTouching` para hacer que un sprite empuje al otro a través de la pantalla antes de practicar con los cuatro bloques de colisión (`collide`, `displace`, `bounce`, y `bounceOff`).

Esta lección introduce colisiones, otra abstracción útil que permitirá a los estudiantes manipular sus sprites de maneras completamente nuevas. Si bien los estudiantes teóricamente pueden haber escrito sus propios comandos de desplazamiento, colisión o rebote, la capacidad de ignorar los detalles de este código les permite centrar su atención en la estructura de alto nivel de los juegos que desean construir.

Esta Lección también pretende que los estudiantes practiquen utilizando los nuevos comandos que han aprendido. En realidad, es la última vez que aprenderán un nuevo comportamiento de `sprite`, y después de esta lección, los estudiantes pasarán a centrarse más en cómo organizan su código cada vez más complejo.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (10 min)

Lección en línea
[Ver en Code Studio](#)

Vocabulario

- **Abstracción:** una representación simplificada de algo más complejo. Las abstracciones le permiten ocultar detalles para ayudarlo a manejar la complejidad, enfocarse en conceptos relevantes y razonar sobre problemas en un nivel superior.

Código

- `sprite.bounce(target)`
- `sprite.bounceOff(target)`
- `sprite.collide(target)`
- `sprite.displace(target)`
- `setCollider(type, xOffset, yOffset, width/radius, height, rotationOffset)`
- `sprite.bounciness`

Objetivos

Los estudiantes serán capaces de:

- Utilizar los bloques “displace”, “collide”, “rebunce”, y “bounceOff” para producir interacciones de sprites
- Describir cómo se pueden construir las abstracciones para desarrollar aún más abstracciones.

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Pantalla: Si tiene la capacidad, proyecte la animación en el primer nivel en Code Studio para esta lección. De lo contrario, pida a las parejas de estudiantes que lo vean juntos. Muestre dos sprites, uno moviéndose a través de la pantalla hacia el otro y finalmente empujando uno cuando colisionan.

Resumen: La clase programa sus sprites para interactuar de nuevas maneras. Después de una breve revisión de cómo usaron el bloque `isTouching`, la clase intercambia ideas sobre otras formas en que dos sprites podrían interactuar. Luego usan `isTouching` para hacer que un sprite presione otro en la pantalla, antes de practicar con los cuatro bloques de colisión (colisionar, desplazar, rebotar y rebotar).

Indicación: Usando los bloques que ya sabemos cómo usar, ¿cómo podríamos crear la interacción de sprites que podemos ver en este programa?

Observaciones: Tenemos muchas ideas geniales sobre cómo podríamos hacer que un sprite empuje otro en la pantalla. Ahora que ya te has preparado, puedes probar tus ideas en Code Studio. Una gran parte del problema es averiguar cuándo se tocan los dos sprites, pero como ya hemos descubierto cómo hacerlo y ahora podemos usar el bloque `isTouching`, ya no necesitamos pensarlo. Podemos enfocarnos en la nueva parte del problema.

Meta: El objetivo de esta discusión es que los estudiantes piensen en formas de resolver el problema de tener un elemento sprite empujando otro en la pantalla. No es necesario que los estudiantes lleguen a un consenso, ya que cada uno de ellos tendrá la oportunidad de probar una solución en el siguiente nivel en Code Studio. Los estudiantes deben entender que es posible usar bloques para producir el movimiento deseado solo con los bloques que ya han aprendido.

Ampliación del conocimiento

Niveles de Code Studio

Niveles 2-4: En estos niveles, se muestra a los estudiantes una interacción de sprites. Luego implementan sus ideas para crear la interacción de los sprites que observaron. La primera vez pueden implementar las ideas del grupo. La segunda vez desafíalos a implementar ese comportamiento de forma independiente.

Interacciones de Sprite

Hasta ahora han sido capaz de crear interacciones de sprites simples usando el sprite. bloque `isTouching()`. Por ejemplo, han reestablecido una moneda a una ubicación diferente en la pantalla cuando un personaje la toca. Ahora es el momento de comenzar a hacer que los sprites tengan interacciones más complejas.

Hacer esto:

- Ejecuta el programa y observa la interacción entre los dos sprites.
- Discute con un compañero: usando solo los comandos que ya conoces, ¿cómo podrías crear este tipo de interacción? Hay muchas maneras de hacerlo, pero aquí hay algunos bloques a considerar:
 - `sprite.isTouching()`
 - `sprite.velocityX`
 - `sprite.velocityY`
 - `sprite.x`
 - `sprite.y`

Prepárate para compartir tus ideas con tus compañeros de clase.

Inducción: Éste fue un problema desafiante, pero pudimos resolverlo.

¿Qué nos ayudó a resolver este problema?

Observaciones: Todas estas cosas son muy importantes y surgen mucho en Informática. Una cosa que fue particularmente útil fue el `isTouching` bloque, que ocultó el código complicado que nos dice si los dos sprites se están tocando. También hay un bloque que oculta el código que acabamos de escribir, y algunos otros bloques que ocultan el código para otros tipos de interacciones de sprites. Tendrás la oportunidad de probar estos bloques en los siguientes niveles y utilizarlos para mejorar tu juego de vuelo.

Niveles 5-10: Estos niveles introducen cómo usar los 4 nuevos bloques de colisión que los estudiantes aprenderán en esta lección.

Desplazar

`sprite.displace()` hará que un sprite empuje al otro cuando se toquen.

Tipos de colisión

Hay cuatro tipos de colisiones que usamos en Game Lab. Estos bloques causarán un cierto tipo de interacción entre el sprite y su objetivo.

Este es un buen momento para decir cuánto han progresado los estudiantes en sus habilidades desde el comienzo de la unidad. Este problema habría parecido casi imposible a principios de año. Algunas cosas que hicieron que el problema fuera más fácil de resolver fueron:

- Preparación: Los estudiantes intercambiaron ideas y pensaron en soluciones antes de probar su código
- Cooperación: Los estudiantes trabajaron en grupos para llegar a una solución
- Abstracción: los estudiantes pudieron usar los bloques `isTouching` y `velocityY` para ocultar parte de la complejidad de la solución

Desplazar

El bloque `displace` hace que el sprite empuje al objetivo siempre que se toquen entre sí.

El sprite se mueve normalmente.

Chocar

El bloque `collide` hace que el sprite se detenga cuando se encuentra con el objetivo. Si el objetivo se está moviendo, empujará al objeto con él. El objetivo sigue moviéndose normalmente.

Rebotar

El bloque `bounce` hace que el sprite y el objetivo reboten cuando se tocan entre sí. Tanto el sprite como el objetivo cambian cómo se están moviendo. El bloque `bounceOff` hace que el sprite rebote en el objetivo. El objetivo sigue moviéndose normalmente.

Depurar

Hay una `sprite.debug` propiedad especial que puede usar para comprender mejor por qué los sprites interactúan de la manera en que lo hacen.

SetCollider

Los Sprites interactúan en función del tamaño y la forma de su colisionador, no de las imágenes que se les asignan. Solo se puede ver el colisionador cuando se activa el modo de depuración. Puede cambiar la forma del colisionador usando el `sprite.setCollider()` bloque, que le permite elegir entre un “rectángulo” o un “círculo”. Por defecto, todos los colisionadores son “rectangulares”.

Niveles 11-15: Estos niveles guían a los estudiantes al agregar colisiones al juego que comenzaron en la Lección anterior, y eventualmente invitan a los estudiantes a agregar sus propias modificaciones al juego.

Bounciness

Hasta el momento, `bounceOff` ha hecho que los sprites se alejen de otros objetos tan rápido como rebotaban en ellos. En el mundo real, casi todo se ralentiza un poco cuando rebota en otra cosa. Puedes usar el `bounciness` bloque para decirle a tu sprite cuánto desacelerar o acelerar cuando rebota en otra cosa.

Transferencia del conocimiento (10 min)

Compartir y publicar 3-2-1

Permita a los estudiantes tiempo para jugar los juegos de los demás. Pida que se centren no sólo en el nuevo comportamiento que agregaron, sino también en el código que usaron para crearlo.

Revise:

Pida a los estudiantes que escriban y reflexionen sobre las siguientes indicaciones.

- ¿Tres cosas que viste en el juego de otra persona que realmente te gustaron?
- ¿Cuáles son dos mejoras que harías en tu juego si tuvieras más tiempo?
- ¿Cuál es un bloque que te gustaría tener en Game Lab y cómo funcionaría?

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Utilizan bloques para programar en base un diseño predictivo
- Explican la forma en que diseñan su programa.

Lección 23

Funciones

Propósito

Los estudiantes aprenden cómo crear funciones para organizar su código, hacerlo más legible y eliminar bloques repetidos de código. Una actividad desconectada de calentamiento explora cómo las direcciones en diferentes niveles de detalle pueden ser útiles dependiendo del contexto. Los estudiantes aprenden que los pasos de mayor nivel o más abstractos facilitan la comprensión y razonamiento de los pasos. Luego, los estudiantes aprenden a crear funciones en Game Lab. Usarán funciones para eliminar bloques largos de código y para reemplazar piezas repetidas de código con una sola función. Al final de la Lección, los estudiantes usan estas habilidades para organizar y agregar funcionalidad a la versión final de su juego de desplazamiento lateral.

En las primeras cuatro lecciones, los estudiantes han aprendido a utilizar una serie de abstracciones en sus programas, que incluyen las propiedades de velocidad, el toque y las colisiones. Estas abstracciones les han permitido crear programas mucho más complejos sin tener en cuenta los detalles de cómo se crea ese comportamiento. En esta Lección, los estudiantes aprenden a crear abstracciones propias mediante la creación de funciones.

Los estudiantes utilizarán principalmente las funciones para dividir el código en fragmentos lógicos que son más fáciles de razonar. Esto forma parte de la transición de la construcción de habilidades técnicas a los procesos organizacionales utilizados para desarrollar software.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (60 min)

Transferencia del conocimiento (10 min)

Lección en línea
[Ver en Code Studio](#)

Vocabulario

- **Función:** un grupo nombrado de instrucciones de programación. Las funciones son abstracciones reutilizables que reducen la complejidad de escribir y mantener programas.

Código

- Define a function
- Call a function

Objetivos

Los estudiantes serán capaces de:

- Crear y usar funciones para bloques de código que realizan una sola tarea de alto nivel dentro de un programa
- Crear y usar funciones para eliminar bloques repetidos de código de sus programas
- Crear y usar funciones para mejorar la legibilidad de sus programas
- Explicar cómo las abstracciones permiten a los programadores razonar sobre un programa en un nivel superior.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Instrucciones de alto nivel vs. Nivel bajo

Preguntar: Imagina que necesitas escribir en un conjunto de instrucciones de 5 pasos para pasar la mañana. ¿Qué serían? Escribe sus pasos y prepárese para compartir.

Discute: El alumno debe escribir las instrucciones de 5 pasos para la mañana y compartir con un compañero. pide a un par de estudiantes que compartan con la clase.

Inducción: Este conjunto de instrucciones es bastante fácil de seguir y comprender. Están en el nivel en el que podrías pensar en describir tu día a un amigo. Ahora vamos a un nivel más profundo. Elija uno de tus 5 pasos y divídelo en 5 pasos más pequeños que necesitas para completar esa tarea más grande. Prepárese para compartir sus ideas de nuevo.

Discute: Los estudiantes deben compartir sus pasos más pequeños. Nuevamente, solicita a algunos voluntarios que luego compartan su paso original y cómo lo separaron.

Esto se está poniendo interesante. Parece que la primera vez que dimos nuestros pasos, estábamos “ocultando” algunos de los detalles necesarios para completar la tarea. Probemos esto una vez más. Toma uno de sus 5 pasos más pequeños y divídelo de nuevo en 5 pasos aún más pequeños.

Pida a los estudiantes que compartan sus pasos una vez más. Pide nuevamente a algunos voluntarios que compartan cómo dividieron uno de sus pasos en pasos aún más pequeños.

Indicación: Imagina que dividimos cada uno de los primeros 5 pasos en 5, y luego dividimos todos esos pasos nuevamente. Esto significa que tendríamos un conjunto de instrucciones de alto nivel, y en la parte inferior un conjunto de instrucciones realmente bajo o detallado. Prepárate para responder a las siguientes preguntas.

- ¿Cuándo sería más apropiado el conjunto de pasos de alto nivel que acabas de escribir?
- ¿Cuándo sería más apropiado el conjunto de pasos de nivel más bajo?
- ¿Cuál es más fácil de razonar o entender?

Objetivo: Esta conversación demuestra a los estudiantes que a menudo usan un nombre o descripción de alto nivel para un comportamiento mucho más complejo. Está motivando el valor de agrupar o combinar muchos pasos más pequeños con un nombre más grande y proporciona alguna de las justificaciones para crear funciones dentro de los programas. A saber, los pasos de alto nivel hacen que sea más fácil de entender y razonar sobre un proceso.

Proceso de desglose: El orden en el que están desglosando una tarea más grande aquí también refleja cómo se les pedirá que escriban código con funciones. A menudo, los programadores primero escriben el nombre de la función que pretenden escribir en función de lo que debe de hacer antes de entrar y escribir los detalles.

Discutir: Pida a los estudiantes que compartan sus pensamientos y opiniones. Después de un par de minutos cambie la conversación al comentario de transición a continuación.

Observaciones: A veces los detalles son importantes, pero a menudo los pasos de alto nivel son mucho más fáciles de razonar y dejan claro lo que está sucediendo. Hemos aprendido que bloques como velocityX o isTouching en realidad solo contienen código que podríamos haber escrito nosotros mismos. Usar estos comandos, o abstracciones, es realmente útil ya que podemos pensar en el código a un alto nivel. Hoy aprenderemos cómo agrupar muchos comandos para crear un bloque nuevo. En la programación cuando creamos un nuevo bloque como este, lo llamamos función.

Ampliación del conocimiento (60 min)

Transición: Dirija a los estudiantes a Code Studio, [contenido de la Lección](#) donde aprenderán a crear funciones.

Resumen: Esta lección cubre las funciones como una forma de organizar su código, hacerlo más legible y eliminar bloques repetidos de código. La clase aprende que los pasos de mayor nivel o más abstractos facilitan la comprensión y razonamiento de los pasos, luego comienza a crear funciones en Game Lab. Al final de la lección, la clase usa estas habilidades para organizar y agregar funcionalidad a la versión final de su juego de desplazamiento lateral.

Vocabulario

- **Función:** una pieza de código que se puede llamar fácilmente una y otra vez.

Funciones de llamada

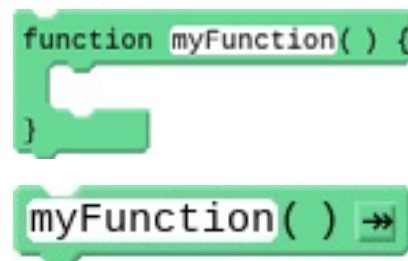
Las funciones permiten construir tus propios bloques y decidir qué código va dentro de ellos. Este es el comando que le permite crear una nueva función.

Usas o “llamas” tu función como cualquier otro bloque usando el nombre que le diste.

Los bloques para crear y llamar a funciones se pueden encontrar en la pestaña “Funciones” de la paleta.

Introduciendo funciones

En estas primeras lecciones a los estudiantes simplemente se les muestra la sintaxis de las funciones y no se les pide que escriban o creen las suyas propias. Puede ser útil explicar la creación de una función como básicamente “crea un nuevo bloque” al igual que otro programador creó los bloques “isTouching” o “velocity” que han visto realmente contienen otro código más complejo.



Reordenando el código

Colocar el código dentro de las funciones facilita la lectura y la realización de cambios. Puedes llamar a tus funciones en un orden diferente para realizar rápidamente cambios significativos en la ejecución de tu programa.

Funciones de llamada en Ciclos de Dibujo

Puede llamar a una función dentro del ciclo de extracción, tal como lo haría en cualquier otro lugar de su código.

Llamar a las funciones varias veces

A menudo querrá usar el mismo código en muchos lugares de su programa. Una vez que haya creado una función, puede llamarla tantas veces como desee.

Hacer cambios en las funciones

Una buena ventaja de utilizar funciones para eliminar código repetido es que ahora puedes realizar fácilmente cambios en múltiples lugares en su código. Simplemente cambia la forma en que crea las funciones, y su programa ahora usará el nuevo código en todas partes donde se llame tu función.

Transferencia del conocimiento (10 min)

Pregunte:

¿Por qué diríamos que las funciones nos permiten “crear nuestros propios bloques”?

¿Por qué es esto algo que quisiéramos hacer?

Discute: Haz que los estudiantes discutan en su mesa antes de hablar como clase.

Preguntar: Escribe tu propia definición de abstracción ¿Por qué una función cuenta como una abstracción?

Discute: Haz que los estudiantes discutan en su mesa antes de hablar como clase.

Observaciones: Las funciones son una herramienta útil para ayudarnos a escribir y

Preguntar: Escribe tu propia definición de abstracción ¿Por qué una función cuenta como una abstracción?

Por qué usar las funciones

Este y los siguientes dos niveles presentan tres usos de funciones, a saber, eliminar la repetición en los programas, permitir que el código se cambie rápidamente en varios puntos y proporcionar organización en el código. Los estudiantes necesitarán escribir más de sus propias funciones en estos niveles.

Meta: Use este primer mensaje para realizar lo que los estudiantes aprendieron hoy. Cuando crean una función, crean su propio bloque que pueden llamar o usar cuando lo deseen. Vieron al menos dos motivaciones principales para crear funciones hoy en día, incluidas.

- Simplificando el código dividiendo en fragmentos lógicamente nombrados
- Evitando el código repetido haciendo un bloque, puedes usarlo varias veces

Discute: Haz que los estudiantes discutan en su mesa antes de hablar como clase.

Observaciones: Las funciones son una herramienta útil para ayudarnos a escribir y organizar fragmentos de código más complejos. Poder mantener tu código organizado será una habilidad importante.

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Agrupan la información en unidades más pequeñas
- Explican cómo las funciones hacen el trabajo más eficiente

Lección 24

El proceso de diseño de juego

Propósito

Esta lección presenta a los estudiantes el proceso que usarán para diseñar juegos. Este proceso se centra en una guía de proyecto que les pide a los estudiantes que definan sus sprites, variables y funciones antes de comenzar a programar su juego. En esta Lección, los estudiantes comienzan jugando un juego en Game Lab donde el código está oculto. Discutan qué creen que deberían hacer los sprites, las variables y las funciones para hacer el juego. Luego reciben una guía de proyecto completa que muestra una forma de implementar el juego. A continuación, los estudiantes pasan por este proceso a través de una serie de niveles. Como parte de esta Lección, los estudiantes también aprenden brevemente a utilizar animaciones de cuadros múltiples en Game Lab. Al final de la Lección, los estudiantes tienen la oportunidad de hacer mejoras en el juego para hacerlo suyo.

Esta Lección presenta animaciones de cuadros múltiples y es la primera de una secuencia centrada en el proceso de creación de software.

Si bien las lecciones anteriores se centraron en el uso de abstracciones para gestionar la complejidad del código, esta lección se centra en la gestión de la complejidad del proceso de desarrollo de software. La lección ancla en gran medida el proceso de desarrollo de software al proporcionar a los estudiantes una guía de proyecto completa, proporcionando código de inicio, y encaminando a los estudiantes a través de su implementación. En las lecciones posteriores, los estudiantes deberán completar una mayor parte de esta guía de forma independiente, y para el proyecto final, seguirán este proceso.

Secuencia para el aprendizaje

Conocimiento inicial (15 min)

Ampliación del conocimiento (60 min)

Transferencia del conocimiento (20 min)

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Defender Game - Guía de proyectos](#)

Objetivos

Los estudiantes serán capaces de:

- Identificar las construcciones de programación central necesarias para construir diferentes componentes de un juego.
- Crear y usar animaciones de multitrama en un programa.
- Implementar diferentes características de un programa siguiendo una guía de proyecto estructurado.

Preparación

- [Contenido de la Lección- code.org](#)

Estrategia de aprendizaje

Conocimiento inicial (15 min)

Jugar Cake Defender

Transición: Esta lección comienza inmediatamente en Code Studio [Contenido de la Lección - code.org](#). En el primer nivel, encontrarán un juego, pero no podrán ver el código. Deben jugar el juego y seguir las instrucciones que les piden que enumeren las variables, los sprites y las funciones que creen que son necesarios para crear este juego.

¡Defiende tu Torta!

Es un ejemplo de un juego que construirá al final de esta Lección.

Detener: revisar la guía de proyectos

Debate: Los estudiantes deberán crear individualmente una lista de variables, sprites y funciones que creen que se utilizan en el juego. Pida a los estudiantes que compartan sus listas con un compañero antes de hablar como clase.

Distribuir: Entregue a cada alumno o pareja de estudiantes una copia del Juego Defender- Guía de proyectos

Preguntar: Compara los componentes del juego que creías que se incluirían con los de esta guía de proyecto. ¿Notan alguna diferencia?

Discute: Como clase, compara la lista que tenías en el pizarrón con la lista de variables, sprites y funciones en la guía del proyecto. Ten en cuenta las similitudes. Donde hay diferencias intenta entender por qué. Evita “correcto” o “incorrecto”.

Observaciones: Por lo general, hay muchas maneras de estructurar un programa para que funcione de la manera que desees. Lo importante al escribir programas complejos o grandes es que comienzas con un plan. Hoy vamos a ver cómo podemos implementar este plan para construir nuestro propio juego. Al final de la Lección, no sólo habrás construido tu juego, sino que sabrás cómo cambiarlo y hacerlo tuyo. ¡Vamos!

Mantener el enfoque: los estudiantes pueden distraerse fácilmente con la diversión de jugar el juego. Déjelos jugar por un tiempo, pero eventualmente anímelos a seguir las instrucciones en pantalla y haga una lista de las variables, los sprites y las funciones que sería necesarias para jugar el juego.

Ejecución de la conversación: puede escribir “variables”, “sprites” y “funciones” en la pizarra y registrar sus ideas debajo de cada una. Pídales a los estudiantes que justifiquen sus decisiones, pero no sientan la necesidad de conformarse con una sola respuesta correcta.

Guía del proyecto: la guía del proyecto se completa de manera exhaustiva para los estudiantes, de modo que pueda experimentar su uso como referencia cuando programe. Esto debería darles más contexto al completar su propia guía de proyectos en las próximas dos lecciones. Puede dar a cada uno su propia copia como referencia, pero también puede optar por imprimir una copia por par, compartir copias digitales

Ampliación del conocimiento (60 min)

Animaciones de multitrama

Transición: Los estudiantes deben regresar a Code Studio. Antes de implementar realmente el plan, los estudiantes deberán revisar rápidamente una nueva habilidad, cómo usar animaciones de multitrama. Los estudiantes aprenderán rápidamente a crear, modificar y cambiar el nombre de las animaciones, así como a elegir las de la biblioteca.

Detener: Antes de seguir adelante, deberá consultar la Guía del proyecto.

Implementar la guía del proyecto

Los estudiantes reciben una gran cantidad de código de inicio en este proyecto. Los sprites, las variables y las funciones ya se les han asignado. El trabajo de este proyecto es escribir el código para las funciones individuales. Estos niveles guían a los estudiantes a través de cómo implementar esas funciones. A medida que los estudiantes avanzan por los niveles, señalan cómo se está utilizando la guía del proyecto.

Las habilidades más desafiantes que usan los estudiantes en estos niveles es reconocer la necesidad de crear nuevas funciones para reemplazar el código repetido. Los estudiantes necesitan desarrollar esta habilidad por su cuenta, pero estos niveles demuestran una instancia donde esto podría suceder.

Realizar los niveles de Code Studio correspondiente a la Lección.

Transferencia del conocimiento (20 min)

Hazlo tuyo

Este último nivel anima a los estudiantes a hacer suyo el juego. Si los estudiantes han llegado a este punto, tienen todas las habilidades que necesitan para progresar a través del plan de estudios, por lo que no hay presión para completar ninguna de las modificaciones sugeridas en este nivel. Sin embargo, si tienen tiempo, obtener una práctica de planificación e implementar nuevas características será una habilidad útil. Incluso modificar las animaciones del juego es una forma sencilla de que los estudiantes puedan apropiarse del juego.

Compartir: Una vez que los estudiantes han completado el proyecto, pueden compartir su trabajo con sus compañeros de clase. Anime a los estudiantes a exhibir el código adicional que

–
o simplemente mostrar la guía en el proyector. Siempre que esté disponible para referencia, cualquier enfoque funcionará bien.

escribieron y explique cómo ha cambiado la forma en que funciona el juego.

Sugerencias para evaluar

Se sugiere los siguientes indicadores para evaluar formativamente los aprendizajes:

- Elaboran una guía de programa
- Explican la finalidad de las diferentes estructuras utilizadas
- Colaboran de forma grupal en el desarrollo del diseño del juego

Lección 25

Uso del proceso de diseño de juego

Propósito

En esta lección de varios días, los estudiantes usan el proceso de resolución de problemas para crear un juego de salto de plataforma. Comienzan por mirar un ejemplo de un saltador de plataforma y luego definen cómo se verán sus juegos. A continuación, utilizan un proceso estructurado para planificar los fondos, las variables, los sprites y las funciones que necesitarán para implementar su juego. Después de escribir el código del juego, los estudiantes reflexionarán sobre cómo se puede mejorar el juego e implementarán esos cambios.

Los estudiantes ya han aprendido todos los constructos de programación que necesitan para hacer un juego. Esta lección repasa muchos de esos conceptos al tiempo que los presenta por medio de un proceso estructurado que los ayudará a administrar el trabajo. Se basa en el uso de la Guía del proyecto de la lección anterior, donde los estudiantes la completan aún más de forma independiente, antes de usarla para construir un juego. Esta lección prepara a los estudiantes a escribir su propio juego desde cero para el proyecto final.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (50 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Identificar las construcciones de programación central necesarias para construir diferentes componentes de un juego.
- Implementar diferentes características de un programa

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Planificación de su plataforma de juego - contenido code.org](#)

Para los estudiantes:

- Planificación de su plataforma de juego- Guía de proyectos (anexo)

siguiendo una guía de proyecto estructurado.

Preparación

- Planificación de su plataforma de juego

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Introducción

El proceso de resolución de problemas nos ayuda a resolver todo tipo de problemas. Piensa en el problema de construir una pieza de software más grande, como el juego que construimos en la última lección. ¿Cómo se veía cada uno de los 4 pasos? ¿Por qué fueron importantes?

Discute: Los estudiantes deben intercambiar ideas en voz baja y anotar cada paso. Después, dirija una discusión de compartir. Puede grabar ideas en el tablero. Las posibles fases incluyen:

- Definir: Averiguar cómo quieres que se vea el juego, cómo debería funcionar y quién lo jugará.
- Preparar: Planificar cómo se verá tu código. Decide una estructura para tu juego.
- Intentar: Escribir el código que sigue a su plan.
- Reflexiona: probar su código, jugar el juego para asegurarse de que funciona, recibir comentarios de otras personas para mejorar el juego.

Observaciones: Cuando se construye un software, el proceso de resolución de problemas puede ser una guía útil. Obviamente, necesitamos escribir el código, pero tener cuidado de definir lo que queremos construir, hacer un buen plan para construirlo y reflexionar luego sobre cómo mejorarlo son parte de la creación de un buen software. Hoy vamos a usar este proceso para hacer un nuevo juego.

Ampliación del conocimiento

Jugar Alien Jumper

Distribuir: Entregue a cada alumno una copia de la planificación de su Guía de plataforma de juego

Observaciones: Vamos a construir un juego de saltos. Tendrá la oportunidad de jugar un juego de muestra y luego planificar cómo crear el juego en su Guía de proyectos.

Dirija a los estudiantes a Code Studio. En el primer nivel, encontrarán un juego, pero no podrán ver el código. Deben jugar el juego y seguir las instrucciones que les piden, que enumeren las

Meta: Los estudiantes deben compartir sus pensamientos, pero si no surgen naturalmente, sugiérales los ejemplos proporcionados. Esta discusión motivará el uso de la guía del proyecto para construir un juego más adelante en la lección.

variables, los sprites y las funciones que creen que son necesarios para crear este juego.

Discutir la guía de proyectos

Recorra la sala: Los estudiantes deben completar la guía del proyecto en el estilo de la que vieron en la lección anterior. Es probable que quieran mantener el juego mientras intentan determinar el comportamiento que tendrán cada uno de los sprites.

Compartir: Los estudiantes comparten sus planes para hacer el juego. Reafírmeles que hay muchas maneras correctas de crear la misma pieza de software, y que tendrán la oportunidad de probar sus ideas en Code Studio.

- Los estudiantes trabajan en parejas para crear el juego.
- Los estudiantes tienen la oportunidad de mejorar su juego después de estar expuestos a otras dos versiones de un saltador de plataforma.
- Los estudiantes comparten sus juegos con sus compañeros de clase.

Reflexión

Preguntar: Pida a los estudiantes que reflexionen sobre el desarrollo de las cinco prácticas de Descubrimientos CS (Resolución de problemas, Persistencia, Creatividad, Colaboración, Comunicación). Elija una de las siguientes indicaciones según lo considere apropiado.

- Elija una de las cinco prácticas en las que cree que demostró crecimiento en esta Lección. Escribe algo que hiciste que ejemplifica esta práctica.
- Elija una práctica en la que piense que puede seguir creciendo. ¿Qué le gustaría mejorar?
- Elija una práctica que pensó que era especialmente importante para la Lección que completamos hoy. ¿Qué lo hizo tan importante?

Hacer el juego tomará al menos dos periodos de clase. Si no hay suficiente tiempo para que todos los estudiantes terminen la clase, los grupos de estudiantes pueden trabajar para codificar diferentes aspectos y luego compartir el código entre ellos. Por ejemplo, un grupo podría trabajar en las plataformas, uno en las estrellas y otro en el jugador. Los estudiantes que terminaron temprano pueden elegir más desafíos de los niveles posteriores.

Sugerencias para evaluación

Se sugiere los siguientes indicadores para evaluar formativamente los aprendizajes:

- Implementan un diseño siguiendo la guía del proyecto
- Explican la finalidad de las diferentes estructuras utilizadas
- Colaboran de forma grupal en el desarrollo del diseño del juego

Lección 6

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Trazar formas A

Resumen

Tu compañero debe tener la herramienta de dibujo abierta en una computadora donde no puedas verla. Trata de explicar cómo dibujar tu imagen. Luego revisa su trabajo pero asegúrate de mantener tus dibujos ocultos.

<p>Dibujo 1A (Color)</p>	<p>Dibujo 1A (Negro y blanco)</p>
<p>Dibujo 2A (Color)</p>	<p>Dibujo 2A (Negro y blanco)</p>

Dibuja el tuyo

Usa el espacio a continuación para dibujar tu propia imagen con las formas. Luego ve si puedes comunicarlo a tu compañero para dibujar usando la herramienta de dibujo de formas en Game Lab. También puedes dar tu dibujo a otro grupo para usarlo como desafío.

Tamaño de la forma
Estas formas son del tamaño correcto

Referencia de patrón
Si no tiene verde, rojo o azul, rellene los patrones con los colores que usará.

Rojo

Verde

Azul

Reflexión

- ¿Qué problema está ayudando a resolver la cuadrícula en Game Lab?
- ¿Has visto formas diferentes de resolver este problema en el pasado? ¿Qué son?

Trazar formas B

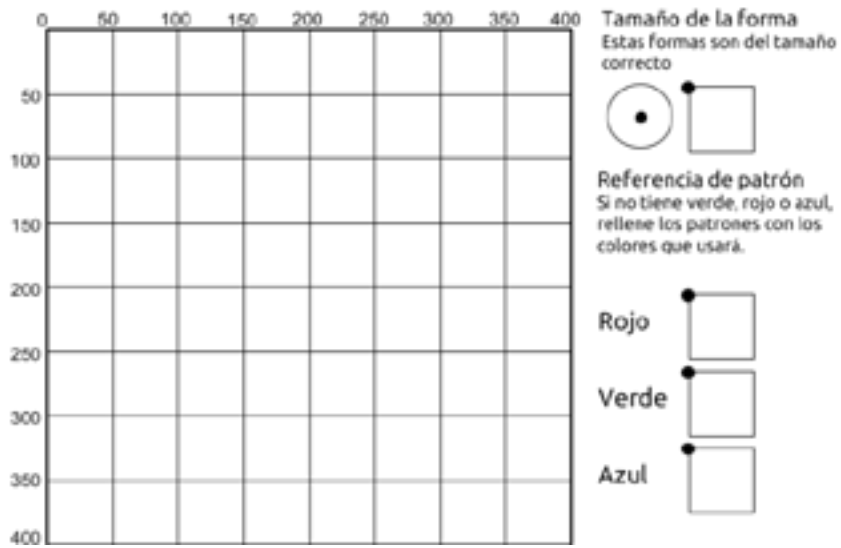
Resumen

Tu compañero debe tener la herramienta de dibujo abierta en una computadora donde no puedas verla. Trata de explicar cómo dibujar su imagen. Luego revisa su trabajo, pero asegúrate de mantener tus dibujos ocultos.

<p>Dibujo 1B (Color)</p>	<p>Dibujo 1B (Negro y blanco)</p>
<p>Dibujo 2B (Color)</p>	<p>Dibujo 2B (Negro y blanco)</p>

Dibuja el tuyo

Usa el espacio a continuación para dibujar tu propia imagen con las formas. Luego vea si puede comunicarlo a su compañero para dibujar usando la herramienta de dibujo de formas en Game Lab. También puede dar su dibujo a otro grupo para usarlo como desafío.



Reflexión

- ¿Qué problema está ayudando a resolver la cuadrícula en Game Lab?
- ¿Has visto formas diferentes de resolver este problema en el pasado? ¿Qué son?

Lección 10

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

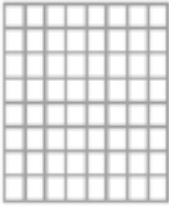
Sprite planificación de escena

Usando formas, sprites y texto, vas a crear una escena simple. Puede pensar en esto como una página en una historia, un panel en un cómic o simplemente una escena independiente.

Dibuja tu escena

Lo primero que debe considerar al diseñar su escena es cómo se verá su fondo. Puedes usar los comandos de dibujo que has usado en el pasado para diseñar un fondo simple sobre el cual colocarás tus sprites.

El espacio de abajo es 50 por 50. Eso significa que cada cuadrado en el papel se asignará a 50 píxeles en la computadora. Dibuje su fondo usando solo los comandos de dibujo (referencia proporcionada a la derecha). Enumera los sprites que usarás a continuación.

	formas:
	background(color)
	rect(x, y, width, height)
	ellipse(x, y, width, height)
	line(x1, y1, x2, y2)
	shape(x1, y1, x2, y2...)
	text(string, x, y, width, height)
	textSize(pixels)
	Color and Style:
	fill('color')
	noFill()
	stroke('color')
	noStroke()
	strokeWeight()

Etiqueta sprite	Descripción

Lección 12

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Tablero variables desconectada

Etiquetas	Valores
Espacio de trabajo	

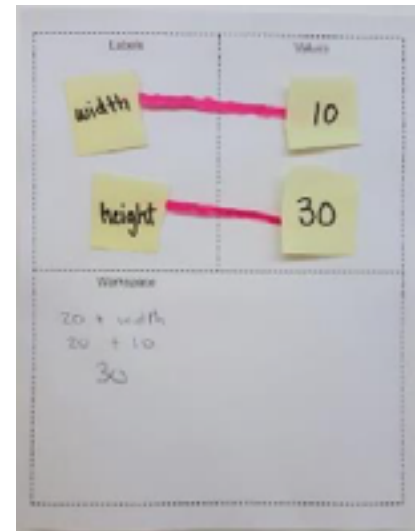
Variables desconectadas

La Lección

Trabajarás con tu compañero para conectar etiquetas de variables y sus valores en función de diferentes programas.

Preparar

- Tablero: un pedazo de papel dividido en 3 secciones:
- etiquetas, valores y espacio de trabajo.
- Etiquetas de variables y tarjetas de valor (Variable Label y Value Cards): pequeños trozos de papel (~ 50).
- Conectores: algo para conectar las etiquetas y los valores.



Reglas

1. Las tarjetas de etiqueta siempre deben tener un solo conector.
2. No puedes crear la misma tarjeta de etiqueta dos veces.
3. Debes crear una nueva tarjeta de valor para cada comando “=”.
4. Las tarjetas de valor sólo pueden tener un número escrito en ellas.
5. Cualquier tarjeta de valor no conectada a una tarjeta de etiqueta al final de un comando se descarta.

Comandos

Ejecuta los programas en orden siguiendo los pasos para cada comando.

Paso 1:

Encuentra o crea la tarjeta de etiqueta y el conector

- A. Si el comando comienza con “var”, crea una nueva tarjeta de etiqueta con la palabra que viene antes del “=”. También debes darle un conector, que usarás más tarde.
- B. Si el comando no comienza con “var”, busca la etiqueta con la palabra que aparece antes del “=”.

Paso 2:

Calcula el número de tu tarjeta de valor

- A. Copia lo que está a la derecha de “=” en tu área de trabajo.
- B. Tacha todas las etiquetas de variables y reemplázalas con los valores (números) conectados a ellas.
- C. Haz cualquier cálculo (suma, resta) en los números.
- D. Escribe el número final en una nueva tarjeta de valor.

Paso 3:

Conecta tu tarjeta de etiqueta y tu tarjeta de valor

- A. Coloca tu nueva tarjeta de valor en la sección “Valores”.
- B. Mueve el conector de la tarjeta de etiqueta del Paso 1 para conectarse a esta nueva tarjeta de valor.

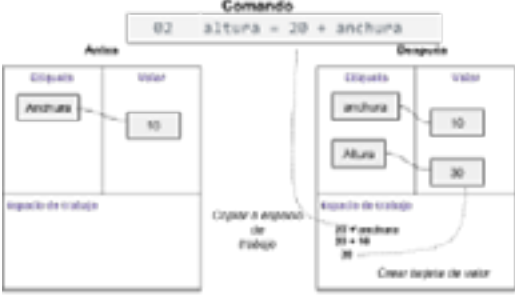
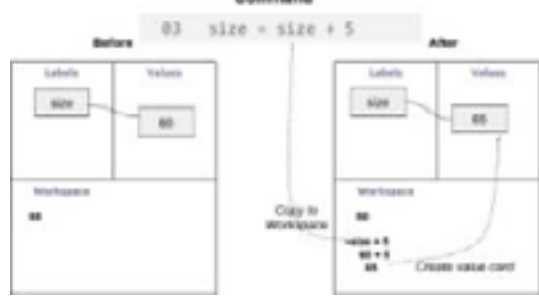
Paso 4:

Eliminar los valores sobrantes

- A. Toma todas las tarjetas de valor que no estén conectadas a las tarjetas de etiquetas y deséchalas.

Pruébalo

Ejecuta los programas a continuación. Para ayudarte en los dos primeros programas, consulta los ejemplos anteriores y posteriores a la derecha que muestran cómo se ejecutan los diferentes comandos.

	<table border="1"> <thead> <tr> <th colspan="2">Programa 1</th> </tr> </thead> <tbody> <tr> <td>01</td> <td>var anchura = 10</td> </tr> <tr> <td>02</td> <td>var altura = 20 + width</td> </tr> <tr> <td colspan="2">Estado final</td> </tr> <tr> <td>anchura:</td> <td></td> </tr> <tr> <td>altura:</td> <td></td> </tr> </tbody> </table>	Programa 1		01	var anchura = 10	02	var altura = 20 + width	Estado final		anchura:		altura:																					
Programa 1																																	
01	var anchura = 10																																
02	var altura = 20 + width																																
Estado final																																	
anchura:																																	
altura:																																	
	<table border="1"> <thead> <tr> <th colspan="2">Programa 2</th> </tr> </thead> <tbody> <tr> <td>01</td> <td>var size = 30</td> </tr> <tr> <td>02</td> <td>size = 60</td> </tr> <tr> <td>03</td> <td>size = size + 5</td> </tr> <tr> <td colspan="2">Estado final</td> </tr> <tr> <td>tamaño:</td> <td></td> </tr> </tbody> </table>	Programa 2		01	var size = 30	02	size = 60	03	size = size + 5	Estado final		tamaño:																					
Programa 2																																	
01	var size = 30																																
02	size = 60																																
03	size = size + 5																																
Estado final																																	
tamaño:																																	
<table border="1"> <thead> <tr> <th colspan="2">Programa 3</th> </tr> </thead> <tbody> <tr> <td>01</td> <td>var age = 11;</td> </tr> <tr> <td>02</td> <td>var height = 60;</td> </tr> <tr> <td>03</td> <td>age = age + 1;</td> </tr> <tr> <td>04</td> <td>height = height + 5;</td> </tr> <tr> <td colspan="2">Estado final</td> </tr> <tr> <td>edad:</td> <td></td> </tr> <tr> <td>altura:</td> <td></td> </tr> </tbody> </table>	Programa 3		01	var age = 11;	02	var height = 60;	03	age = age + 1;	04	height = height + 5;	Estado final		edad:		altura:		<table border="1"> <thead> <tr> <th colspan="2">Programa 4</th> </tr> </thead> <tbody> <tr> <td>01</td> <td>var xPosition = 100</td> </tr> <tr> <td>02</td> <td>var yPosition = xPosition</td> </tr> <tr> <td>03</td> <td>xPosition = yPosition + 30</td> </tr> <tr> <td>04</td> <td>yPosition = yPosition + 50</td> </tr> <tr> <td colspan="2">Estado final</td> </tr> <tr> <td>xPosición:</td> <td></td> </tr> <tr> <td>yPosición:</td> <td></td> </tr> </tbody> </table>	Programa 4		01	var xPosition = 100	02	var yPosition = xPosition	03	xPosition = yPosition + 30	04	yPosition = yPosition + 50	Estado final		xPosición:		yPosición:	
Programa 3																																	
01	var age = 11;																																
02	var height = 60;																																
03	age = age + 1;																																
04	height = height + 5;																																
Estado final																																	
edad:																																	
altura:																																	
Programa 4																																	
01	var xPosition = 100																																
02	var yPosition = xPosition																																
03	xPosition = yPosition + 30																																
04	yPosition = yPosition + 50																																
Estado final																																	
xPosición:																																	
yPosición:																																	

2020 [Imagen]. Recuperada de <http://www.code.org>

Sprite propiedades

Puedes cambiar las propiedades de tus sprites de la misma manera que cambias tus variables, lo que te permite controlar cómo se mueve el sprite en la pantalla.

Para hacer un seguimiento de las propiedades de un sprite, necesitarás una carta de sprite. La tarjeta Sprite va en la sección de valores, conectada a una etiqueta variable. En la parte inferior de la tarjeta, enumera las propiedades de las que desea realizar un seguimiento. Utilizarás un conector

para cada propiedad.

En el ejemplo de la derecha, la tarjeta realiza un seguimiento de las propiedades "x" y "y" del sprite, que le indican a la computadora dónde colocarlo en la pantalla.



Creando una tarjeta Sprite

Cada vez que veas el comando `createSprite`, tendrás que crear una nueva tarjeta sprite con las propiedades que deseas controlar.

Usa las propiedades sprite

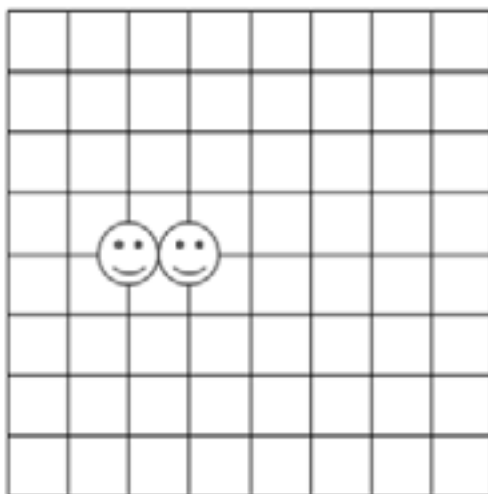
Puedes usar los valores de propiedad de tu sprite de la misma manera que usaste los valores de variable.

Dibuja tu sprite

Cada vez que veas el comando `drawSprites`, dibuja tu sprite en la cuadrícula según tus valores de coordenadas x-y.

Puedes usar una carita sonriente para la animación de tu sprite.

Puedes hacer tus propias tarjetas de sprites.



Programa 5

```

01 var smiley = createSprite(100,200);
02 smiley.setAnimation("smileyFace");
03 drawSprites();
05 smiley.x = smiley.x + 50;
06 drawSprites();
    
```

Pruébalo

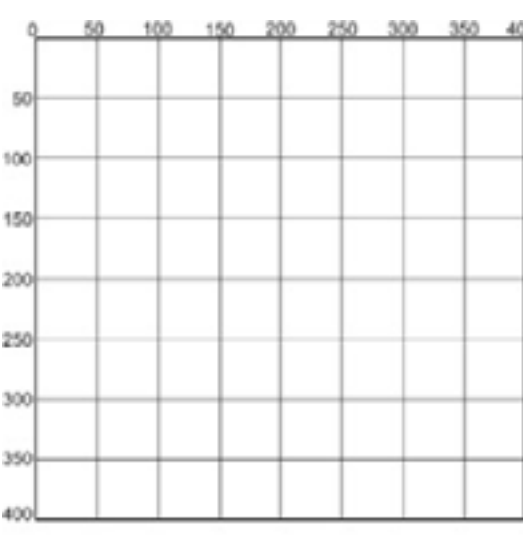
Ejecute estos programas con su compañero. No olvides dibujar tus sprites cuando veas el comando `drawSprites`.

Program 6	
01	<code>var smiley = createSprite();</code>
02	<code>smiley.setAnimation("smiley");</code>
03	<code>smiley.x = 50;</code>
04	<code>smiley.y = 100;</code>
05	<code>drawSprites();</code>
06	<code>smiley.x = smiley.x + 50;</code>
07	<code>drawSprites();</code>
08	<code>smiley.x = smiley.x + 50;</code>
09	<code>drawSprites();</code>
10	<code>smiley.x = smiley.x + 50;</code>
11	<code>drawSprites();</code>
Estado final	
<code>smiley.x</code>	
<code>smiley.y</code>	



¿Cómo se movió el sprite a través de la red en el Programa 6?

Programa 7	
01	<code>var smiley = createSprite();</code>
02	<code>smiley.setAnimation("smiley");</code>
03	<code>smiley.x = 200;</code>
04	<code>smiley.y = 300;</code>
05	<code>drawSprites();</code>
06	<code>smiley.y = smiley.y - 30;</code>
07	<code>drawSprites();</code>
08	<code>smiley.y = smiley.y - 30;</code>
09	<code>drawSprites();</code>
10	<code>smiley.y = smiley.y - 30;</code>
11	<code>drawSprites();</code>
Estado final	
<code>smiley.x</code>	
<code>smiley.y</code>	



¿Cómo se movió el sprite a través de la red en el Programa 7?

Lección 14

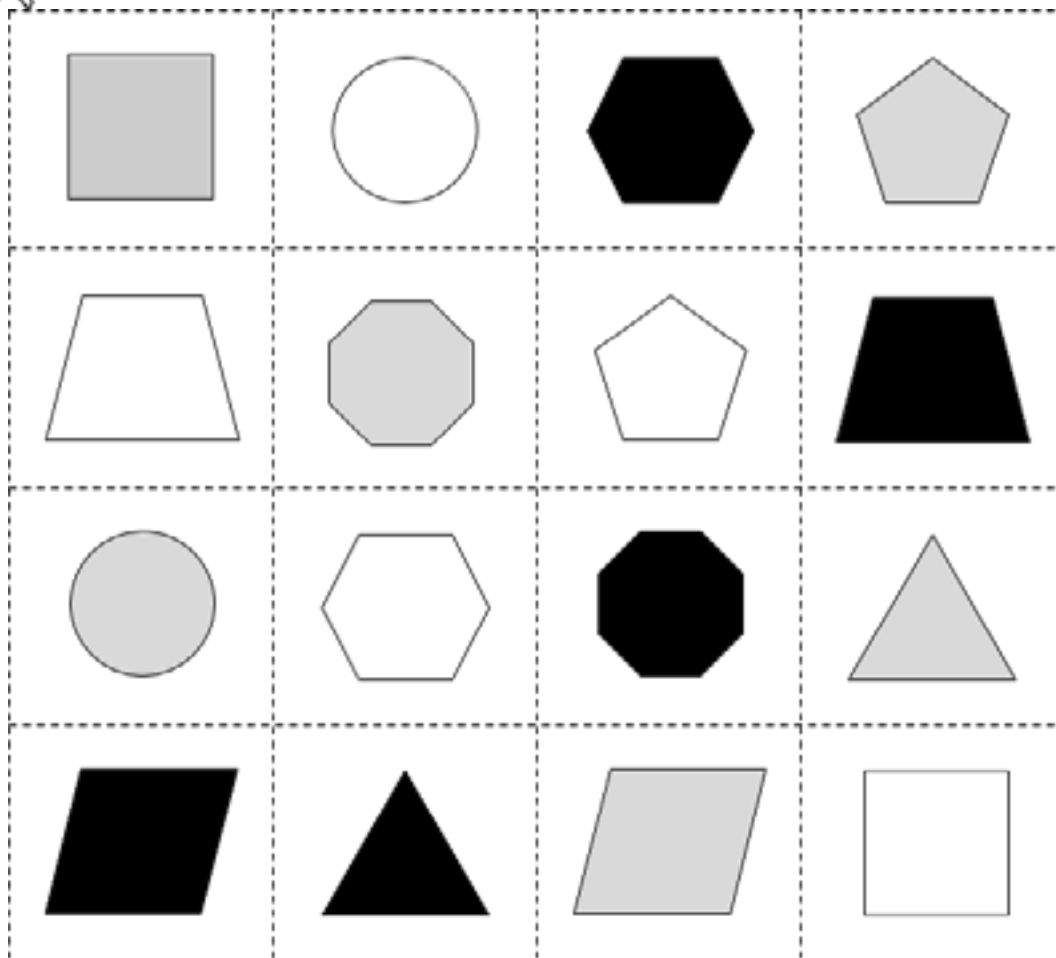
(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Propiedades booleanas

Prepara tus formas

Recorta la siguiente tabla de formas. Estas serán las tarjetas que usarás para la Lección de clasificación de propiedades booleanas.



2020 [Imagen]. Recuperada de <http://www.code.org>

Lección 18

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Guía de proyecto - tarjeta interactiva

Resumen

Vas a desarrollar una tarjeta digital interactiva para compartir con alguien que te importa, pero tendrás que planificar un poco antes de comenzar a programar.

Planificando tus Sprites

Use la tabla a continuación para planificar sus sprites. (¡Necesitas al menos tres, pero puedes usar tantos como quieras!) Junto a cada elemento, diseña la imagen que usará y qué propiedades cambiarán.

Sprite etiqueta	Imagen(es)	Propiedades

Desarrollando interacciones

El elemento final de su tarjeta a considerar es cómo el usuario interactuará con él, y cómo los sprites pueden interactuar entre sí. Deberá incluir los condicionales que responden a la entrada del teclado (como `keyDown ()`), así como los condicionales que responden a variables cambiantes o propiedades de sprites (como `sprite.y > 300`). Use la tabla a continuación para planificar todos sus condicionales y la acción correspondiente.

If / Else if / Else	Condición	Acción

Desarrolla tu tarjeta

Una vez que su maestro haya aprobado su diseño, vaya a Code Studio para programar su tarjeta.

Verifique su tarjeta

Verifique su tarjeta para asegurarse de que tenga todo lo que necesita.

Reflexión

¿De qué parte de tu proyecto estás más orgulloso?

¿Por qué?

Si tuvieras más tiempo, ¿qué mejorarías de tu tarjeta?

Rúbrica - tarjeta interactiva

Rúbrica de tarjeta digital

Evalúe su tarjeta digital de acuerdo con los siguientes criterios. Explique en qué parte de su código se pueden encontrar los criterios en la columna de comentarios.

Criterio	sí/no	comentarios
Utiliza al menos 3 sprites		
Al menos un sprite responde a la entrada del usuario (ejemplo keyDown, mouseDidMove)		
Actualiza al menos 3 propiedades de sprites diferentes en el ciclo de extracción (ejemplo sprite.x, sprite.scale, sprite.visible)		
Utiliza al menos 1 condicional que se desencadena por una propiedad de variable o sprite (ejemplo sprite.y > 300)		
Incrementa a disminuye una variable o propiedad de sprite (eg score = score + 1)		
La tarjeta incluye texto que le dice al usuario cómo usar la tarjeta		

Prácticas de reflexión

Práctica	Cosas para celebrar	Cosas para trabajar
Problema resuelto		
Persistencia		
Creatividad		
Colaboración		
Comunicación		

Revisión por pares - tarjeta interactiva

Pre-Revisión

Nombre de los creadores:

Una cosa sobre la que quiero recibir comentarios es ...

Revisar sección

Nombre del revisor:

Preguntas	Clasificación	Comentarios
Hay al menos un sprite que responde a la entrada del usuario (ejemplo. <code>keyDown</code> , <code>mouseClick</code>)	✓ X	
Actualiza al menos 3 propiedades de sprites diferentes en el ciclo de extracción (ejemplo. <code>sprite.x</code> , <code>sprite.scale</code> , <code>sprite.visible</code>)	✓ X	
Utiliza al menos 1 condicional que se desencadena por una propiedad de variable o sprite (ejemplo. <code>sprite.y > 300</code>)	✓ X	
Incrementa o disminuye una variable o propiedad de sprite (ej. <code>score = score + 1</code>)	✓ X	

Retroalimentación de respuesta

Me gustó

Deseo

Reflexión del creador

1. ¿Qué comentario te fue más útil? ¿Por qué?

•

2. ¿Qué comentario te sorprendió más? ¿Por qué?

•

3. En función de los comentarios, ¿qué cambios harías en tu tarjeta interactiva?

Lección 20

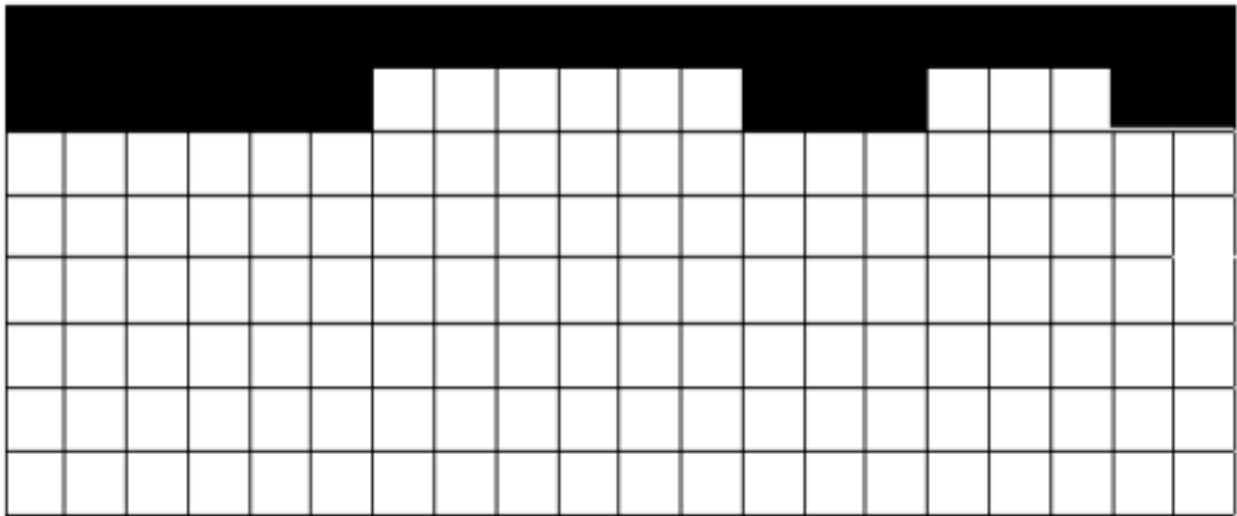
(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Colisiones A

Dibujar con Sprites

En la cuadrícula de abajo, dibuja dos sprites cuadrados en la línea inferior. Los sprites pueden estar tocándose o no. No dejes que tu pareja vea tus sprites.



Sprites propiedades

Usando la cuadrícula, encuentra la posición x del sprite (justo en el medio del cuadrado) y el ancho del sprite. Esta es la única información que debe decirle a su compañero sobre los sprites

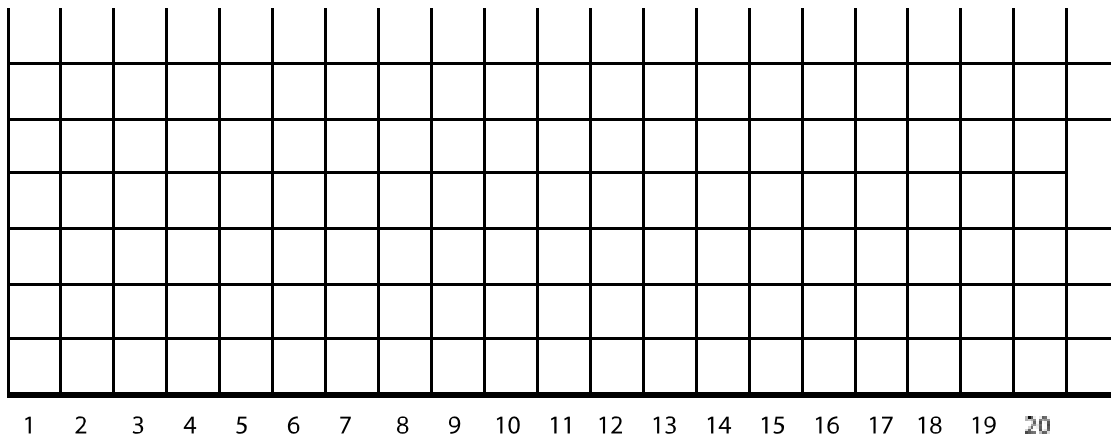
	x	anchura
sprite 1		
sprite 2		

Colisiones B

Dibuja tus Sprites

En la cuadrícula de abajo, dibuja dos sprites cuadrados en la línea de la izquierda. Los sprites pueden

estar tocándose o no. No dejes que tu pareja vea tus sprites.



Sprites propiedades

Usando la cuadrícula, encuentra la posición del sprite (justo en el medio del cuadrado) y la altura del sprite.

	y	height
sprite 1		
sprite 2		

Colisiones A

Detección de colisión

Copie la información de la primera página en este cuadro, luego gire la imagen para que su compañero no pueda echar un vistazo. El siguiente cuadro es la única información que su compañero debería ver sobre sus sprites. Una vez que haya completado el primer cuadro, canjee esta hoja de trabajo con su compañero para que ambos puedan probar sus detectores de colisión.

¡Detente!

Cambia las hojas de trabajo con su compañero antes de continuar.

	x	anchura
sprite1		
sprite2		

Mira la información en el cuadro de arriba. ¿Puedes pensar en alguna forma en que puedas usar estos números para averiguar si los dos sprites se están tocando? Haga una lluvia de ideas a continuación.

¿Crees que los sprites se están tocando? ¿Por qué sí o por qué no?

Verifica el dibujo de su compañero para ver si estaba en lo correcto. ¿Crees que tu estrategia fue efectiva?

Colisiones B

Detector de colisiones

Copie la información de la primera página en este cuadro, luego gire la imagen para que su compañero no pueda echar un vistazo. El siguiente cuadro es la única información que su compañero debería ver sobre sus sprites. Una vez que haya completado el primer cuadro, canjee esta hoja de trabajo con su compañero para que ambos puedan probar sus detectores de colisión.

¡Detente!

Cambia las hojas de trabajo con su compañero antes de continuar.

	y	altura
sprite1		
sprite2		

Mira la información en el cuadro de arriba. ¿Puedes pensar en alguna forma en que puedas usar estos números para averiguar si los dos sprites se están tocando? Haga una lluvia de ideas a continuación.

¿Crees que los sprites se están tocando? ¿Por qué sí o por qué no?

Verifica el dibujo de su compañero para ver si estaba en lo correcto. ¿Crees que tu estrategia fue efectiva?

Lección 24

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Defender game

Visión en conjunto

Crear una pieza de software más grande, como un juego, puede volverse complejo rápidamente. Comenzar con un plan puede ayudarlo a mantenerse organizado e identificar problemas con anticipación. Mucho del trabajo que hace aquí hará que sea mucho más fácil hacer un seguimiento de lo que debe hacer una vez que comience a escribir su código real.

Jugabilidad y visuales

Comienza pensando en lo que realmente hace tu juego.

¿Cómo se ve?

¿Cómo lo juegas?

¿Qué lo hará divertido, interesante o relevante para el jugador?

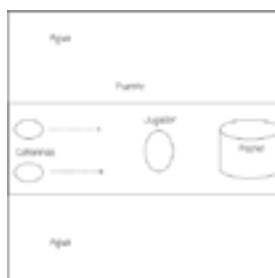
Describe tu juego

En un par de oraciones, describe el juego que vas a construir y cómo funcionará.

Eres un extraterrestre que defiende su pastel de las malvadas catarinas. Las catarinas marchan a través de un puente hacia su pastel. Tendrás que empujar a las catarinas fuera del puente hacia el agua. Obtenga puntos por detener a las catarinas y pierda puntos por dejar que las catarinas terminen.

Dibuja tu juego

Dibuja un boceto rápido de cómo funcionará tu juego ¿Quiénes son los personajes?



¿Cómo se ve el fondo? ¿Cómo se mueven las cosas? Etiqueta las cosas para que sean más claras.

Sprites y variables

Usando la descripción de tu juego anterior, averigua qué información y personajes necesitarás seguir durante tu juego. Complete una descripción para cada uno en el espacio a continuación.

Sprites

En la siguiente tabla, enumera información sobre los diferentes sprites en tu juego.

¿Dónde están ubicados? ¿Cómo se mueven? ¿Cómo interactúan con otros sprites?

Nombre (Etiqueta) y Apariencia	Al inicio del juego (Animación, posición, rotación, velocidad, velocidad de rotación)	Usuario y Sprite e Interacciones (¿El usuario controla este sprite? ¿Cómo se mueve? ¿Alguna vez necesita restablecer su posición? ¿Interactúa con otros sprites? ¿Cómo?)
Jugador - alien	x: 300 y: 200 Sin rotación o velocidad	Las teclas de flecha mueven el sprite en 4 direcciones Desplaza las catarinas
pastel - pastel	x: 350 y: 200 Sin rotación o velocidad	Nunca se mueve Cuando las mariquitas te tocan pierdes puntos
enemigo 1 - catarina	x: 0 y: aleatorio entre 150 y 250 velocidadX: 2 (puede necesitar hacer más rápido)	Se mueve a través de la pantalla de izquierda a derecha El jugador sprite puede desplazar a las catarinas Si presionan el restablecimiento del agua y le dan al jugador un punto. Si alcanzan el reinicio del pastel, quita 5 puntos
enemigo 2 - catarinas	x: 0 y: aleatorio entre 150 y 250 velocidadX: 2 (puede necesitar hacer más rápido)	Igual que el otro

Variables

Piensa en la información que tu juego necesita para realizar un seguimiento. ¿Hay un puntaje? ¿Una cantidad de vidas? Describe cada variable en el espacio a continuación.

Nombre	Lo que hace un seguimiento de	¿Cómo cambia durante el juego? (¿Cuál es el valor inicial, cuándo cambiará?)
Puntuación	Puntuación de jugadores	Comienza en 0. Cuando las mariquitas empujadas en el agua obtienen 1 punto. Cuando las mariquitas golpean el pastel pierden 5 puntos.

Funciones

Tu ciclo de extracción no debería tener un código complejo. En cambio, divide tu programa en los principales pasos que necesitarás para que tu juego funcione. Los diferentes comportamientos que

describiste para sus sprites y variables deberían ayudarte a decidir cuáles deberían ser estos pasos. Luego, describe qué debería hacer el código para esa función

Nombre de la función	Que sucede en esta función ¿Qué comportamientos detallaste para tus sprites incluye esta función? ¿Se puede usar esta función en varios lugares de tu programa?
<i>gameBackground()</i>	<i>Dibuja el fondo del juego</i>
<i>enemiesTouchCake()</i>	<i>Restablece a los enemigos cuando tocan el pastel. Aumenta el puntaje del jugador.</i>
<i>movePlayer()</i>	<i>Mueve al jugador con las teclas de flecha y cambia las animaciones</i>
<i>displaceEnemies()</i>	<i>jugador empuja a las catarinas</i>
<i>enemiesTouchWater()</i>	<i>Si alguna catarina toca el agua, se restablece y la puntuación del jugador aumenta</i>
<i>showScore()</i>	<i>Muestra el puntaje en las pantalla</i>

Lección 25

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Planificación de su plataforma de juego

Visión en conjunto

Crear una pieza de software más grande, como un juego, puede volverse complejo rápidamente. Comenzar con un plan puede ayudar a mantenerse organizado e identificar problemas con anticipación. Mucho del trabajo que hace aquí hará que sea mucho más fácil hacer un seguimiento de lo que debe hacer una vez que comience a escribir su código real.

Jugabilidad y visuales

Comienza pensando en lo que realmente hace tu juego ¿Cómo se ve? ¿Cómo lo juegas? ¿Qué lo hará divertido, interesante o relevante para el jugador?

Describe tu juego

En un par de oraciones, describe el juego que vas a construir y cómo funcionará.

Antecedentes

Dibuja un boceto rápido de cómo quieres que se vea el fondo (s) de tu juego.

Sprites y variables

Usando la descripción de tu juego anterior, averigua qué información y personajes necesitarás seguir durante tu juego. Complete una descripción para cada uno en el espacio a continuación.

Sprites

En la siguiente tabla, enumera información sobre los diferentes sprites en tu juego.

¿Dónde están ubicados? ¿Cómo se mueven? ¿Cómo interactúan con otros sprites?

Variables

Nombre (Etiqueta) y Apariencia	Al inicio del juego (Animación, posición, rotación, velocidad, velocidad de rotación)	Usuario, Sprite e Interacciones (¿El usuario controla este sprite? ¿Cómo se mueve? ¿Alguna vez necesita restablecer su posición? ¿Interactúa con otros sprites? ¿Cómo?)

Piense en la información que su juego necesita para realizar un seguimiento. ¿Hay un puntaje? ¿Una cantidad de vidas? Describe cada variable en el espacio a continuación.

Nombre (Etiqueta)	Lo que hace un seguimiento de	¿Cómo cambia durante el juego? (¿Cuál es el valor inicial, cuándo cambiará?)

Funciones

Su ciclo de extracción no debería tener un código complejo. En cambio, divida su programa en los principales pasos que necesitará para que su juego funcione. Los diferentes comportamientos que describió para sus sprites y variables deberían ayudarlo a decidir cuáles deberían ser estos pasos. Luego, describe qué debería hacer el código para esa función.

Nombre de la función	Qué sucede en esta función (¿Qué comportamientos detallaste para tus sprites incluye esta función? ¿Se puede usar esta función en varios lugares de tu programa?)

Unidad 02

La resolución de problemas y las máquinas

Contenido 1:

La representación de distintos tipos de datos y su uso masivo

Contenido 2:

Diferentes usos de BIG DATA

Contenido 3:

Machine Learning o Aprendizaje Automatizado

Contenido 1

La representación de distintos tipos de datos y su uso masivo

Resumen

- **Lección 26:** La representación importa
- **Lección 27:** Patrones y representación
- **Lección 28:** ASCII y representación binaria
- **Lección 29:** Representar imágenes
- **Lección 30:** Representando números
- **Lección 31:** Mantener los datos secretos
- **Lección 32:** Combinando representaciones
- **Lección 33:** Crear una representación

Objetivos

- **OA 2.** Representar diferente tipo de datos en una variedad de formas que incluya textos, sonidos, imágenes y números.
- **OA g.** Elaborar representaciones, tanto en forma manual como digital, y justificar cómo una misma información puede ser utilizada según el tipo de representación.

Referencias

- **Code Studio-** Code.org
- <https://cuantrix.mx/>. Fundación Televisa

Lección 26

La representación importa

Propósito

Los estudiantes obtienen una visión general de qué son los datos y cómo se usan para resolver problemas. Los estudiantes comienzan con una breve discusión para llegar a una comprensión común de los datos. Luego se dividen en grupos y usan un conjunto de datos para hacer una serie de recomendaciones de comidas para personas con diversos criterios. Cada grupo tiene la opción de comida representada de una manera diferente (imágenes, recetas, menú, nutrición) que da una ventaja para una de las recomendaciones. Luego, los grupos comparan sus respuestas y discuten cómo las diferentes representaciones de los datos de las comidas afectaron la forma en que los estudiantes pudieron resolver los diferentes problemas.

Esta Lección presenta una serie de ideas importantes que los estudiantes explorarán en este capítulo. Verán que los mismos objetos se pueden representar de diferentes maneras. Aprenderán los diferentes sistemas de representación que las computadoras usan para representar información. También verán en esta Lección que la representación importa cuando usamos datos para tomar decisiones. Los estudiantes explorarán más profundamente cómo los humanos y las computadoras pueden usar los datos para tomar decisiones.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Definir datos como información recopilada del mundo para ayudar a hacer una recomendación o resolver un problema.

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Datos de comidas- Recurso](#)
- [La representación importa - guía de actividades](#)

Para los estudiantes:

- [La representación importa - guía de actividades](#)

- Proporcionar ejemplos de cómo representar los datos de diferentes maneras puede afectar su capacidad para resolver diferentes problemas.
- Elegirla mejor manera de representar cierta información en función de cómo se utilizará.

Preparación

- Datos de comidas- [Recurso \(link a documento. idea de ejemplo en inglés\)](#)
- La representación importa- [guía de actividades](#)

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Preguntar: Hoy vamos a comenzar a hablar sobre los datos y cómo se usan en Ciencias de la Computación. Antes de comenzar, nos tomamos unos minutos para pensar qué son los datos.

Pantalla: Muestra las siguientes preguntas y pida a los estudiantes que anoten sus respuestas en silencio.

- ¿Qué son los datos?
- ¿Cómo usas los datos en tu vida?
- ¿Cómo pueden los datos ayudarte a resolver problemas?

Dé a los estudiantes unos minutos para que piensen por sí mismos sobre qué son los datos, y luego permite compartir en silencio con un compañero. Después de que todos los estudiantes hayan tenido la oportunidad de hablarse entre ellos, que comparta toda la clase y escriban las ideas en el pizarrón.

Observaciones: Todas son buenas ideas. Vamos a pasar el resto del curso mirando más de cerca qué son los datos, de dónde provienen y cómo pueden ayudarnos en Ciencias de la Computación. Por ahora, vamos a definir los datos como “información que se ha recopilado para ayudarnos a responder una pregunta o resolver un problema”.

Ampliación del conocimiento (40 min)

Grupo: Organice a los estudiantes en grupos de 3-5.

Distribuir: Entregue a cada grupo La representación importa - guía de actividades y una de las cuatro versiones de Datos de comidas- Recurso. Asegúrese de que al menos un grupo tenga un recurso de imagen, un recurso de menú, un grupo de recursos de nutrición y un grupo de recursos de recetas.

Usando datos: Cada grupo de estudiantes hará una recomendación de comida a cuatro personas diferentes, y deberán justificar su recomendación con sus datos. Debido a que los diferentes grupos tienen conjuntos de datos diferentes, la dificultad de las recomendaciones variará de un grupo a otro.

Después de hacer las recomendaciones, los grupos deben elegir la recomendación que pensaron que era la más fácil de hacer, y explicar su razonamiento.

Promoción de la oferta:

Si a los estudiantes les resulta difícil comenzar, recuérdelos que esto es solo una lluvia de ideas, y estarán trabajando para responder estas preguntas durante toda la unidad. Los datos pueden tener diferentes definiciones según el contexto (plan de teléfono móvil, clase de matemática, etc.) Anime a los estudiantes a pensar en diferentes situaciones en las que hayan usado datos, y recuérdelos que no hay una sola respuesta “correcta”.

Objetivo: Los estudiantes deben entender que los datos son información que se ha recopilado sobre el mundo. Deberían ver que los datos podrían ser cualquier tipo de información, no solo números.

Lidiar con la frustración: debido a que cada grupo solo tendrá la información adecuada para una de las cuatro recomendaciones, los estudiantes pueden sentirse frustrados porque no pueden encontrar la respuesta “correcta”. Asegúreles que no siempre hay una recomendación “correcta” y que la parte más importante del ejercicio es que expliquen por qué hicieron su elección.

Cuando todos los grupos hayan completado la hoja de trabajo, vuelvan a reunirse como una clase y compartan las respuestas y el razonamiento de cada recomendación. A medida que los grupos comparten respuestas y razonamientos, permita ver los conjuntos de datos de los demás.

Persona 1

“Soy alérgico a los huevos”.

Aunque el menú y las imágenes pueden ayudar un poco, el conjunto de datos de recetas es el único que les dice a los estudiantes los ingredientes en cada comida.

Persona 2

“Mi médico dijo que comiera menos sodio”.

Aquellos con los datos de nutrición deberían ver qué comida tiene el menor contenido de sodio.

Persona 3

“Estoy tratando de ahorrar dinero”.

Aquellos con el conjunto de datos del menú deben ver los precios de cada comida.

Persona 4

“Quiero publicar una buena foto en línea”.

Si bien la recomendación para este es más subjetiva, el grupo con el conjunto de datos de imágenes está en la mejor posición para hacer una recomendación informada.

Pregunta: Ahora que has visto todas las formas en que representamos las cuatro comidas, piensa en lo que implica una forma de representar algo bueno o malo. ¿Crees que algunas de las representaciones fueron mejor o peor que otras? ¿Qué los hizo mejores o peores?

Observaciones: Cuando recopilamos información sobre el mundo, debemos tomar decisiones sobre lo que es importante que incluyamos en nuestras representaciones. Las elecciones que hacemos afectan los tipos de problemas que podemos resolver con nuestros datos. En las próximas actividades, hablaremos sobre cómo las computadoras representan datos y cómo usamos esos datos para resolver problemas.

Transferencia del conocimiento (5 min)

Invita a los estudiantes a reflexionar sobre las siguientes preguntas en sus cuadernos.

Hoy, viste cuatro maneras diferentes de representar una comida, y cómo esas diferentes representaciones fueron útiles para resolver diferentes problemas.

- ¿Por qué algunas representaciones fueron más útiles que otras?

- Si tuviera que crear una forma de representar una comida, ¿cuáles serían las cosas más importantes para que piense?

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Representan datos usando varios esquemas de encriptación

Lección 27

Patrones y representación

Propósito

En esta lección, los estudiantes crean su propio sistema para representar información. Comienzan por una lluvia de ideas sobre todos los diferentes sistemas que ya utilizan para representar respuestas de sí o no. Luego trabajan en pequeños grupos para crear un sistema que puede representar cualquier letra en el alfabeto utilizando solo una pila de tarjetas. Las tarjetas utilizadas tienen uno de los 6 posibles dibujos diferentes (6 animales, 6 colores, etc.) y, por lo tanto, para representar todo el alfabeto, los estudiantes deberán usar patrones de varias tarjetas para representar cada letra.

Los estudiantes crean mensajes con sus sistemas e intercambian con otros grupos para asegurarse de que el sistema funcionó según lo previsto. En la discusión de recapitulación, la clase revisa los pros y los contras de los diferentes sistemas. Discuten las similitudes entre los sistemas de trabajo y reconocen que hay muchas soluciones posibles para este problema.

En esta lección, los estudiantes exploran por sí mismos las cualidades de un buen sistema para representar la información. Deberían encontrar a través de la creación y prueba de sus sistemas que: 1) Normalmente hay muchos sistemas posibles para representar información y 2) Las personas deben acordar un conjunto común de reglas para que un sistema funcione.

Las características de los sistemas que los estudiantes crean en esta lección sirven como un punto de referencia en las próximas lecciones cuando los estudiantes exploran los sistemas de representación que realmente se usan en las computadoras.

En la próxima lección, se les presentará el concepto binario y cómo las computadoras usan señales de encendido-apagado o “0’s y 1’s” para representar la información. Aunque se insinúa este concepto en la lección introductoria de sí-no, no se debe presentar hasta la próxima Lección.

Lección sin conexión
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Representando información- Ejemplar](#)
- [Formas de animales - Manipulador](#)

Para los estudiantes:

- Información representativa - [Guía de actividades](#)

Vocabulario

- **Decodificar:** para cambiar la forma en que se representa la información para que pueda leerla una persona.
- **Codificar:** para cambiar la forma en que se representa la información para que pueda leerla una computadora.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (10 min)

Objetivos

Los estudiantes serán capaces de:

- Describir las características necesarias de un sistema para representar información
- Crear, usar y proporcionar comentarios sobre un sistema para representar información.
- Mejorar iterativamente un sistema para representar información probando y respondiendo a los comentarios.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Sistemas para responder Sí-No preguntas

Preguntar: imagina que tu amigo te preguntó si podía pasar un rato más tarde. Esta es una pregunta de sí o no, por lo que una de las formas en que podría responder es diciendo “Sí” o “No”. Este es un par de respuestas que tu amigo podría entender, pero hay muchas más parejas posibles.

¿De cuántas maneras diferentes sabes que podrías escribir, decir o representar la respuesta a una pregunta “Sí-no”? Escribir tantas ideas como puedas.

Discute: Los estudiantes deben intercambiar ideas en silencio, luego compartir con sus tablas, y finalmente compartir como clase. Escribe ideas al frente del salón a medida que se comparten las ideas.

Indicación: Cada fila en nuestra tabla es un “sistema” diferente, pero todos representan la misma información. ¿Por qué tenemos tantos sistemas diferentes para representar la misma información?

Observaciones: Lo que estamos viendo aquí es que hay muchos “sistemas” que podríamos elegir usar para representar la misma información. La información de un “Sí” podría ser un asentimiento con la cabeza o un pulgar hacia arriba, y un “No” podría ser un signo menos o una flecha hacia abajo. Como humanos, hemos creado muchos sistemas, porque son mejores o peores dependiendo de la situación. En una sala ruidosa, un pulgar hacia arriba funciona mejor que gritar. La gente habla muchos idiomas diferentes. Hemos creado muchos sistemas porque queremos comunicar información en muchas situaciones diferentes.

Las computadoras también necesitan sistemas para representar la información, y hoy vamos a empezar a explorar qué tipos de sistemas necesitaría una computadora para representar la información. Hoy nos enfocaremos más en lo que hace que los sistemas sean buenos, y mañana empezaremos a acercarnos a los sistemas específicos que usan las computadoras.

Ampliación del conocimiento (40 min)

Representando el alfabeto

Grupo: Coloca a los estudiantes en grupos de 2 o 3

Objetivo: Los estudiantes deben ver que la misma información se puede representar de muchas maneras diferentes. Se les puede ocurrir que digan “Sí” o “No” en muchos idiomas diferentes, pulgar hacia arriba y pulgar hacia abajo, más o menos, flecha arriba vs, flecha abajo, moviendo la cabeza de arriba a abajo, de un lado a otro, etc.

Objetivo: Esta conversación configura otro punto importante, es que se crean diferentes sistemas para diferentes contextos. Hoy pida a los estudiantes que creen un sistema para lo que parecerá un contexto muy arbitrario, una baraja de cartas de animales. Apóyese en el hecho de que habrán lanzado una lluvia de ideas de “extraños sistemas arbitrarios” en este calentamiento. Eso es porque hay diferentes contextos en los que se usan. En la lección siguiente puede decir que el diseño de sistemas para cables con electricidad es tan extraño como arbitrario.

Anime a los estudiantes a resolver problemas: hay muchas soluciones posibles para esta lección y es intencionalmente muy abierto. Recuérdeles a los estudiantes que la resolución de problemas no

Distribuir: Entrega a cada grupo [Información representativa- Guía de actividades](#) y un conjunto de [formas animales- Manipulativo](#). Puedes cortarlos de antemano o hacer que el grupo corte su propio grupo.

[Guía de actividades](#)

Representación de información

Lección de repaso: Revise las reglas de la lección con la clase.

Desarrolla reglas: Dé a los estudiantes unos minutos para intercambiar ideas sobre sus reglas. Anímelos a probar sus ideas con los miembros de su grupo, ya que saben exactamente cómo se ejecutará la prueba. Pida que elijan un par de palabras cortas y las prueben con los miembros de su grupo antes de compartirlas con otro grupo.

Reglas de prueba: Una vez que los estudiantes hayan terminado de desarrollar sus reglas, pida que elijan una nueva palabra corta para representar. Deben hacer cuidadosamente una sola pila de cartas para representar su palabra, dejar las cartas en la mesa y luego pasar sus reglas y la pila de cartas o rotar alrededor del salón. Los grupos deben voltear la pila de tarjetas en sus mesas y decodificar cuidadosamente la palabra.

Los grupos deben consultar con los creadores de reglas originales para ver si decodificaron correctamente la palabra.

Revisar y revisar las reglas: Con base en la prueba, haga que los grupos decidan si las reglas tienen sentido o si necesitan mejorarse de alguna manera. Pida a los grupos que brinden comentarios constructivos y proporcionen algo de tiempo para revisar las reglas.

Reglas de prueba nuevamente: Haga que los grupos preparen una pila de cartas para representar una palabra nueva antes de repetir la prueba. Deberían cambiar las reglas con un grupo diferente esta vez.

Revisar y revisar las reglas: Brinde a los estudiantes una oportunidad más para proporcionar comentarios y realizar ediciones finales de sus reglas.

Transferencia del conocimiento (10 min)

[Discusión de grupo](#)

Indicaciones: Tómense un minuto en sus mesas y hablen de las siguientes indicaciones. Prepárense para compartir sus pensamientos.

siempre significa hacer que algo funcione la primera vez, y que tendrán que probar iterativamente sus soluciones antes de asegurarse de que sean correctas.

Idea errónea común: los estudiantes pueden inventar patrones de diferentes longitudes. Por ejemplo, "A=1 elefante, B=2 elefantes, C=3 elefantes...". Descubrirán que no sabrán si una palabra tiene 2 A seguidas o una sola B. Permita que los estudiantes lo encuentren por sí mismos probando su sistema (una parte importante del proceso de resolución de problemas).

Cuántas pruebas: esta lección les da a los estudiantes la oportunidad de experimentar por sí mismos los desafíos de crear un sistema para representar información. Después de dos intentos, los estudiantes deben tener suficiente experiencia para participar en las discusiones de recapitulación y ver los puntos importantes de la lección, incluso si tienen algunos problemas con sus propios sistemas.

Comprende la Lección:

El hecho de que haya solo 6 tipos de tarjetas significa que los estudiantes necesitarán hacer patrones de múltiples tarjetas para representar cada letra. El hecho de que todas las cartas están colocadas en una pila ordenada significa que cada patrón necesitará tener la misma longitud. De lo contrario, será difícil

- ¿Qué fue lo mismo y qué fue diferente sobre los diferentes conjuntos de reglas que viste?
- ¿Hay grupos que necesitaron explicar cosas para completar el desafío?

Observaciones: Hoy crearon sus propios sistemas para representar información. Vimos que hay muchos sistemas diferentes que podríamos usar para representar la misma información. Lo importante es que existen reglas claras sobre cómo usar los sistemas, y que todos conocen las reglas.

Los ingenieros informáticos se preocupan por los sistemas para representar información, porque una computadora no “entiende” mágicamente el mundo. Necesita recibir información usando un sistema que tenga en cuenta el hecho de que es solo una caja de cables.

Cuando cambiamos la forma en que representamos la información para que sea más fácil de usar para una computadora, codificamos esa información. Cuando lo cambiamos para que sea más fácil de entender para un humano, decodificamos esa información.

Vocabulario

Introduzca los siguientes términos:

Codificar: Para cambiar la forma en que se representa la información para que pueda leerla una computadora.

Descodificar: Para cambiar la forma en que se representa la información para que pueda ser leída por una persona.

Observaciones: En el resto de este capítulo, vamos a ver formas en que podemos codificar información para que pueda ser utilizada por una computadora.

saber cuándo termina un patrón y comienza otro. Los informáticos se enfrentan a estos mismos desafíos cuando diseñan sus propios sistemas de representación. Las soluciones más simples a la Lección serán asignar a cada letra un patrón único de dos formas (por ejemplo, A= “Cerdo Cerdo”, B= “Cerdo Elefante”, etc.). Sin embargo, todavía hay una enorme cantidad de sistemas posibles con estas cualidades y es probable que no haya dos sistemas en su clase iguales. Ninguno es “mejor” que el otro y lo que importa es que todos decidamos arbitrariamente usar el mismo. De nuevo, esto es cierto de los sistemas de representación que los estudiantes verán en las próximas lecciones.

Objetivo: Esta discusión tiene como objetivo resaltar el hecho de que hubo muchas soluciones posibles a este problema. Puede darse el caso de que algunos grupos encuentren problemas con sus reglas como resultado de probarlos. Señale, sin embargo, que incluso conjuntos muy diferentes de sistemas para representar esta información podrían funcionar. Al concluir el segundo aviso, intente establecer similitudes entre reglas, por ejemplo

- Cada letra necesita un patrón separado
- Cada carta debe usar el mismo número de tarjetas (probablemente dos) o puede comenzar a tener problemas de no saber dónde termina una letra y comienza la siguiente.

Lección 28

ASCII y representación binaria

Propósito

En esta lección los estudiantes aprenden a usar su primer sistema binario para codificar información: el sistema Estadounidense Estándar para el Intercambio de Información (ASCII, por sus siglas en inglés) para representar letras y otros caracteres. Al comienzo de la lección, el docente presenta el hecho de que las computadoras deben representar la información usando “encendido” o “apagado”. Luego se introduce a los estudiantes en el sistema ASCII para representar texto usando símbolos binarios. Los estudiantes practican usando este sistema, antes de codificar sus propios mensajes usando ASCII. Al final de la lección, una conversación informativa ayuda a sintetizar los principales objetivos de aprendizaje de la lección.

Esta lección es una oportunidad para hacer algunas conexiones del mundo real de la lección anterior y revisar algunos de los conceptos que los estudiantes vieron en esa lección. A los estudiantes se les presenta formalmente el concepto de binario, aprenden que es un poco de información y practican usando un sistema binario. Los estudiantes revisan el hecho de que los patrones de bits se pueden usar para representar información y consideran por qué tener patrones que tienen la misma longitud (en este caso, 7 bits) hace que sea más fácil usar un sistema.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (10 min)

Objetivos

Los estudiantes serán capaces de:

- Definir un sistema binario como uno que usa solo dos estados posibles para representar información

Lección sin conexión
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [ASCII- Ejemplar](#)

Para los estudiantes:

- [ASCII- Guía de actividades](#)

Vocabulario

- **ASCII:** Código estándar para el intercambio de información; el formato de texto en bruto universalmente reconocido que cualquier computadora puede entender
- **Binario:** Una forma de representar información utilizando solo dos opciones.
- **Bit:** Una contracción de “Binary Digit”; la única unidad de información en una computadora, típicamente representada como 0 o 1.

- Definir un bit como una sola pieza de información binaria
- Usar el sistema ASCII para codificar y decodificar información de texto en formato binario.

Preparación

- [ASCII- Guía de actividades](#)

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Revisión de la Lección de las tarjetas binarias

Observaciones: Las computadoras modernas son dispositivos electrónicos llenos de pequeños cables. Estos cables transportan electricidad y en cualquier momento un cable puede estar encendido (alto voltaje) o apagado (bajo voltaje). De hecho, toda la información que alguna vez ha usado en una computadora, como documentos, videos e imágenes, finalmente debe traducirse en estas señales de “encendido” y “apagado”. Este es un gran desafío y lo vamos a explorar para las próximas actividades.

Ampliación del conocimiento (40 min)

Usando ASCII

Grupo: Coloca a los estudiantes en parejas

Distribuir: [ASCII- Guía de actividades](#) para cada par de estudiantes

Guía de actividades de texto ASCII

¿Por qué Binario?

Como clase lee esta sección.

Vocabulario

Revisa la definición de los términos.

Binario: Una forma de representar información usando solo dos opciones

ASCII: Un sistema popular para representar texto en binario

ASCII

Revisa la tabla ASCII. Informa a los estudiantes que ASCII es el sistema que probablemente todas las computadoras que usaron alguna vez usan para representar letras. Hoy van a practicar con el uso de este sistema.

Desafíos: Haga que los estudiantes decodifiquen los tres mensajes.

¿Por qué no dividir los símbolos? Al enviar información por un cable con electricidad, no hay forma de poner un “espacio” entre las señales. En cualquier momento, estás enviando electricidad. Esta es la razón principal por la que los caracteres tienen una longitud acordada, en este caso de 7 bits de longitud.

Objetivo: 0 y 1 es solo otro par binario que se puede usar para indicar on-off. Los estudiantes han visto muchos pares binarios en esta lección que pueden usarse para el sistema de codificación ASCII. En el futuro, sin embargo, usarán 1 y 0. Dado que toda la información en una computadora debe convertirse en señales de encendido y apagado en un cable, decimos que toda la información es 0s y 1s.

Haga los suyos: Haga que los estudiantes diseñen su propio sistema binario simple diseñando lo que serán el símbolo “Encendido” y el símbolo “Apagado”. Luego pida que escriban sus mensajes usando ASCII, intercambien con un compañero y decodifiquen. Si hay tiempo, los estudiantes intercambian con múltiples grupos.

Transferencia del conocimiento (10 min)

Compartir

¿Qué símbolos usaron para su sistema binario? Haga que la clase comparta sus ideas.

Pregunta: ¿Por qué crees que a veces escuchamos que la gente dice que la informática es “0 y 1”? ¿Necesita ser 0 y 1?

Discute: Haga que los estudiantes desarrollen respuestas individualmente, luego compártanlas en grupos pequeños, después con todo el grupo.

Vocabulario

Presenta el vocabulario de:

- **Bit:** Una sola pieza de información binaria

Observaciones: Vimos partes de la información representadas de muchas maneras hoy en día como perros / gatos o manzanas / plátanos. Todas estas son formas igualmente válidas de representar On y Off. En el futuro usaremos 1 y 0, pero esto solo significa On y Off.

Ignorar números por ahora: el sistema ASCII como se presenta aquí es solo un sistema para representar caracteres con patrones de bits. En realidad, hay un paso intermedio en el que se usa el sistema de números binarios para representar cada letra. Si esto ocurre, reconozca que hay mucha información adicional que debe conocer sobre ese sistema, pero adhiérase a esa definición.

Símbolo de encendido-apagado: Es posible que los estudiantes hayan visto antes este icono de encendido y apagado. ¡Es solo un 0 y un 1 combinado!

Investigación: si tiene más tiempo en línea, por ejemplo, explore la página de Wikipedia para ASCII y haga que los estudiantes examinen que otros símbolos están incluidos en el sistema ASCII. ¿Les sorprende algo que ven allí? ¿Hay símbolos que no pensaron que necesitarían representar?

Lección 29

Representar imágenes

Propósito

En esta lección, los estudiantes aprenden cómo las computadoras representan imágenes. Para comenzar la lección, consideran el desafío de entender toda la complejidad de la visión y representarla en un patrón binario. A través de una serie de imágenes que muestran cómo se realiza esta transformación, a los estudiantes se les presenta el concepto de dividir imágenes en cuadrados o “píxeles” que luego pueden activarse o desactivarse (encender o apagar) individualmente para crear la imagen completa. Luego, los estudiantes hacen un pequeño conjunto de desafíos usando el widget Pixelation para dibujar imágenes en blanco y negro. Los niveles están diseñados para completar desafíos de representar imágenes de esta manera. En el resumen, los estudiantes establecen conexiones entre el sistema para representar imágenes y el sistema para representar el texto que aprendieron en la lección anterior.

Esta Lección presenta otro sistema comúnmente utilizado para representar información en binario. Las conclusiones más obvias del Pixelation Widget obviamente serán que un 1 significa un “Encendido” y un 0 significa un “Apagado”. Sin embargo, lo que podría ser menos obvio es que es necesario acordar el ancho y el alto de la imagen para interpretar también la información binaria de la imagen. Finalmente, esta es una oportunidad para explorar cómo se crean los sistemas para representar la información. Una imagen “en bruto” es realmente difícil de representar. La idea de convertir una imagen en una cuadrícula de cuadrados en blanco y negro simplifica significativamente este desafío y lo convierte en el tipo de problema que los ingenieros saben cómo resolver.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (10 min)

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Ejemplos de imágenes: diapositivas](#)
- [Niveles de code studio](#)

Vocabulario

- **Pixel:** abreviatura de “elemento de imagen”, la unidad fundamental de una imagen digital, generalmente un pequeño cuadrado o punto que contiene un único punto de color de una imagen más grande.

Objetivos

Los estudiantes serán capaces de:

- Crear y manipular patrones binarios para representar imágenes en blanco y negro.
- Describir las características comunes de los sistemas utilizados para representar información en binario.

Preparación

- Practica usando el [widget Pixelation](#) durante ~ 10 minutos para prepararse para responder preguntas.

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Pantalla: Muestre las imágenes de la bicicleta y el árbol a los estudiantes desde la primera diapositiva de [Ejemplos de imágenes: diapositivas](#).

Indicación: Mire estas dos imágenes. ¿Qué sería un desafío para representarlos en una computadora? ¿Cree que es más fácil o difícil que representar letras?

Discuta: Haga que los estudiantes realicen una lluvia de ideas en silencio, luego compártanlo con un compañero y luego discutan como una clase completa.

Pantalla: Muestra la segunda diapositiva que muestra diferentes versiones de la misma imagen en más formas pixeladas.

Preguntar: ¿Crees que ahora es más fácil? ¿Cuál es la “gran idea” que podría ayudarnos a representar imágenes?

Discute: Puede optar por ejecutar esto como una segunda lluvia de ideas silenciosa o, de lo contrario, discutir inmediatamente como una clase completa.

Observaciones: Resolver un gran problema generalmente significa dividirlo en otros más pequeños. Un panorama general puede ser difícil de representar, pero si lo dividimos en partes más pequeñas, de repente es menos intimidante. Hoy vamos a ver más de cerca cómo funciona este sistema.

Ampliación del conocimiento (40 min)

Demostrar: Puede ser difícil ver los píxeles en las pantallas modernas, ya que son muy pequeños. A veces, las pantallas del proyector funcionan bien, usando una lupa, o simplemente colocando una gota de agua o una pantalla de computadora para crear un efecto de lente. Elija lo que elija, busque la forma de mostrarles a los estudiantes que todas las imágenes en sus computadoras se están dividiendo en pequeños cuadrados. Luego, introduzca el vocabulario a continuación, que es el nombre de estas casillas.

Vocabulario:

Presente brevemente el siguiente vocabulario:

- **Píxel:** Un pequeño cuadrado o punto que contiene un único

Objetivo: presentar a los estudiantes el reto de representar imágenes en código binario.

Objetivo: Estas imágenes tienen la intención de demostrar visualmente la idea de dividir imágenes en partes más pequeñas. Los estudiantes deben reconocer que lo que están mirando es un sistema binario (blanco y negro) y así, con el sistema correcto, deberían poder representar las últimas imágenes. Debe estar preparado para reconocer que, si bien la última imagen es de menor calidad, las mismas ideas probablemente podrían usarse para hacer el original. ¡Está en su pantalla después de todo!

No se puede romper: los widgets están diseñados para imponer reglas para que los estudiantes puedan explorar libremente los conceptos. Anime a los estudiantes a experimentar y no se preocupe por romperlo.

Darle espacio a los estudiantes para resolver problemas: como docente, también es probable que los estudiantes aprendan más jugando solo con la herramienta que teniendo que explicarla. Intente configurar un temporizador durante 5-10 minutos después de comenzar la

punto de color de una imagen más grande.

Dirija a los estudiantes a los Niveles de code studio [Pixelation Widget](#)

Rompecabezas 2: Los estudiantes aprenden a escribir 0s y 1s para activar los píxeles de un patrón X de 10 por 10

Rompecabezas 3: Los estudiantes hacen una sonrisa de 10 por 10.

- Muestra a los estudiantes los botones “Formato limpio” frente a “Formato sin formato” si no lo ven. Esta es una forma muy útil de hacer que la herramienta sea más fácil de usar.

Rompecabezas 4: Los estudiantes cambian el deslizador de ancho para “encontrar” la imagen de un conjunto de bits.

Rompecabezas 5: Los estudiantes “editan” una imagen para agregar cejas a una cara sonriente.

- Esta lección demuestra que los bits se leen en secuencia. Si elimina o agrega uno anteriormente en la computadora, no “sabe” que el resto no debe moverse.

Rompecabezas 6: Juego libre. Los estudiantes deben crear al menos una imagen de 8 por 8 (aunque idealmente más grande). Probablemente desees marcar el tiempo de esta lección y hacer saber a los estudiantes por adelantado cuánto tiempo tendrán, 10-15 minutos antes de hacer la discusión en clase, probablemente esté bien. Si tienen más tiempo, los estudiantes siempre pueden volver para terminar sus dibujos.

Transferencia del conocimiento (10 min)

Compartir: Haga que los estudiantes compartan las imágenes que crearon el uno con el otro.

Rápido:

- Piensen en el sistema ASCII que aprendimos y en el sistema de representación de imágenes que aprendimos hoy. ¿Cómo son ambos ejemplos de descomponer grandes problemas en pequeños que podemos resolver?
- Qué información ADEMÁS de los 0s y 1s necesitan para decodificar un mensaje binario.

Discute: Dé a los estudiantes un minuto para escribir sus ideas antes de compartirlas con un compañero. Eventualmente ejecuta una discusión de toda la clase.

lección en el computador y deje tiempo para analizar lo que están viendo. Al principio se debe esperar un poco de resistencia, pero generalmente conduce a que los estudiantes resuelvan los problemas por sí mismos.

Conocer el sistema

utilizado: este nivel demuestra la importancia de conocer el “sistema” exacto que se utiliza para interpretar una cadena de bits. Leer una imagen de 8 bits de ancho como si tuviera 10 bits de ancho conducirá a una imagen distorsionada. Este es un lugar donde los estudiantes ven claramente este fenómeno, pero pueden perderlo si no se lo señala. Si bien no se explora en este plan de estudios, el ancho y el alto de la imagen también podrían representarse en binario como parte de un sistema más flexible para representar imágenes.

Tenga las ideas listas:

muchos estudiantes pueden necesitar ayuda para decidir qué dibujar. Tenga un par de ideas listas para compartir como su hobby, su animal favorito, un logotipo personal, etc.

Observaciones: Los ingenieros informáticos resuelven problemas. Necesitan trabajar con las herramientas disponibles. Representar algo tan complejo como una imagen usando solo on y off puede parecer realmente desafiante. Sin embargo, al dividir un problema en piezas más pequeñas, puede encontrar soluciones. La próxima vez vamos a ver un sistema más para representar un nuevo tipo de información.

Sugerencias para evaluación

Se sugieren los siguientes indicadores para evaluar formativamente los aprendizajes:

- Representan datos usando varios esquemas de encriptación.
- Elaboran diversas representaciones, identificando el tipo de dato.

Lección 30

Representando números

Propósito

En esta lección, los estudiantes aprenden sobre el sistema de números binarios. Con un conjunto de cartas que representan los valores de posición en un sistema numérico binario (base-2) mediante una colección de puntos, los estudiantes encienden o apagan fichas girando las cartas boca arriba y boca abajo, luego observan los números que resultan de estos diferentes patrones. Eventualmente, los estudiantes extienden el patrón a un sistema genérico de 4 bits.

Los estudiantes aprenden a usar el sistema binario para representar números enteros. A diferencia de ASCII, que se presenta como un mapeo arbitrario de patrones de bits a caracteres, los números binarios se introducen dentro del contexto de patrones de números. Los estudiantes usan los patrones para determinar las representaciones binarias de diferentes números, en lugar de simplemente buscarlos en una tabla.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (10 min)

Objetivos

Los estudiantes serán capaces de:

- Usar un sistema binario para representar números.
- Extender un sistema de representación basado en patrones.

Lección sin conexión
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Representando números- Ejemplar](#)

Para los estudiantes:

- [Representando números - Guía de actividades](#)
- [Tarjetas de números](#)

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Pregunta: Cree una lista de toda la información que desee representar en una computadora como un número. Aquí hay algunas ideas para comenzar

- Una tienda en línea (¿a qué tipo de números hace seguimiento una tienda?)
- Un perfil de redes sociales (¿qué cosas sobre usted o sus amigos son números?)

Discute: Los estudiantes deben intercambiar ideas de forma independiente, luego compartir con un compañero, y finalmente discutir como una clase completa.

Observaciones: Los números son una forma realmente útil e importante de representar todo tipo de información. Si queremos representar números en una computadora, vamos a tener que aprender un nuevo sistema que nos permita hacerlo.

Ampliación del conocimiento (40 min)

Grupo: Agrupa a los estudiantes en parejas

Distribuir: Entrega a cada pareja [Representando números- Guía de actividades](#) y un conjunto de [Tarjetas de números](#).

Tarjetas binarias

Modelo: Revise el problema de ejemplo como clase. Pide a los estudiantes que coloquen las tarjetas boca arriba y boca abajo para que se vean exactamente trece puntos. (Es importante que cada grupo use solo un juego de cartas.) Después de un poco de prueba y error, los estudiantes deben ver que las cartas 8, 4 y 1 deben estar boca arriba, y las 2 deben estar boca abajo.

Permita que los estudiantes completen el resto de la página por parejas. Cuando todos los estudiantes hayan terminado la página, pide que vuelvan a estar juntos como clase y que compartan sus respuestas.

Pregunta: ¿Hubo más de una respuesta posible para alguno de los problemas?

Permita que los estudiantes verifiquen con otras parejas y traten de encontrar otra forma de representar los números.

Propósito: En ésta y las siguientes lecciones, los estudiantes aprenderán un nuevo sistema para representar números usando señales de encendido y apagado. Motive esta lección pensando en los tipos de información que los estudiantes eventualmente querrán representar con este sistema. Si los estudiantes necesitan ayuda para intercambiar ideas, dales un par de ejemplos: edad, altura, fecha de nacimiento, número de amigos que tienen, costo de los artículos, calificación de un artículo, etc.

Esta Lección modela un sistema numérico de base 2 para el alumno. Sin embargo, no es necesario que los estudiantes comprendan las matemáticas detrás de los patrones que están creando. Los estudiantes deben enfocarse en el hecho de que están usando un sistema binario (boca arriba / boca abajo) para representar la información.

Objetivo: Después de un debate, los estudiantes deben tener en cuenta que solo hay una forma de representar un número en particular en este sistema. Este es un punto importante para resaltar, porque sería confuso si dos patrones

Pregunta: ¿Cuál es el número más pequeño que puede hacer? ¿Y el más grande?

Usando el patrón

Permita que los estudiantes completen la segunda mitad de la hoja de trabajo. Para la segunda mitad de la página, las respuestas de los estudiantes pueden variar. Puedes usar la [siguiente guía](#) para verificar su trabajo.

Preguntar: Cuando aprendimos ASCII, tenían que usar una tabla para buscar cada letra. Para números binarios, pudieron representar sus números sin una tabla. ¿Cómo hicieron eso? ¿Podrían averiguar qué podría ser el binario para un número superior a 15?

Permita que los estudiantes discutan sus ideas por parejas antes de devolverlas al grupo.

Observaciones: Con estas tarjetas, creamos un sistema binario para representar números. Debido a que utilizamos un patrón que podemos seguir a medida que aumentan nuestros números, nuestro sistema puede funcionar tan alto como podamos contar. Por supuesto, nuestras tarjetas eventualmente se quedarán sin espacio para poner los puntos, entonces vamos a ver una herramienta que nos ayudará a usar números binarios en cientos.

Dirija a los estudiantes a los [Niveles de Code Studio](#) correspondientes a la lección

Transferencia del conocimiento (10 min)

Indicación: Hasta ahora, hemos visto cómo podemos representar texto, imágenes y números en binario. ¿Hay algún otro tipo de datos que use en su computadora que necesiten ser representados?

- Todos los estudiantes para llamar a diferentes tipos de datos, como música, videos, etc.

Observaciones: Esos son todos los tipos de datos que deben representarse en binario. Vamos a ver un video que explica un poco más sobre cómo funciona esto.

Pantalla: Reproduce el video [“Binario y datos”](#).

– significaran lo mismo. El número más pequeño que los estudiantes pueden hacer con las tarjetas es 0 (no 1). Esto será referenciado en una lección posterior.

Objetivo: Los estudiantes deben reconocer que los patrones de los números y las reglas que siguieron para obtenerlos les permitieron descubrir cómo se representaría un número, sin utilizar una tabla, como lo hicieron en ASCII. Si amplían los patrones, deberían poder obtener números mucho más altos.

0		8	
1		9	
2		10	
3		11	
4		12	
5		13	
6		14	
7		15	

Lección 31

Mantener los datos secretos

Propósito

Los estudiantes tienen una discusión sobre los diferentes niveles de seguridad que desearían para los datos personales. Una vez que la clase ha desarrollado una comprensión de la importancia de la privacidad, aprenden sobre el proceso de cifrado de información codificando una nota para un compañero y descifrando la nota del compañero. La clase concluye con una discusión sobre la importancia de la seguridad física y digital.

Como los estudiantes han estado codificando y decodificando con datos, no han estado preocupados por la seguridad de los datos que están usando. En esta lección, comienzan a pensar en cómo pueden garantizar que solo el destinatario pueda leer los datos que envía. Usarán un cifrado simple para codificar un mensaje. Los estudiantes deben comprender que, para encriptar un mensaje, necesitan tanto un algoritmo como una clave, y que es importante que la clave se mantenga en secreto.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Aplicar un método de encriptación para garantizar la transmisión segura de datos.
- Usar medidas de seguridad físicas y digitales para proteger los datos.

Lección sin conexión
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- Mantener el secreto de los datos- guía de actividades – [Versión en español](#).
- Mantener el secreto de los datos- [ejemplar](#)

Vocabulario

- **Descifrar:** para cambiar la información para que se muestre su significado oculto.
- **Cifrar:** para cambiar la información de modo que su significado quede oculto.

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Muestre la cadena binaria de [Code Studio nivel 2](#).

Dirija a los estudiantes a los Niveles de Code Studio correspondientes a la lección

Preguntar: Aquí hay un ejemplo de un mensaje que alguien podría enviar por Internet a un amigo. ¿Qué necesitas para decodificar este mensaje?

Observaciones: En las últimas actividades, hemos visto muchos tipos de datos diferentes que se pueden codificar en binario. Sin embargo, todos nuestros sistemas de codificación tienen un par de cosas en común.

Revisión: Revise brevemente las características de un sistema de codificación:

1. Necesita ser inequívoco.
2. Todos deben estar de acuerdo.

Muestra la cadena binaria y la clave de decodificación en [Code Studio nivel 3](#).

Niveles de Code Studio

Dé tiempo a los estudiantes para decodificar la secuencia binaria y discutan lo que creen que significa.

Preguntar: ¿Qué pasaría si quisiera enviar un mensaje secreto, para que sólo mi amigo lo pudiera entender?

Hoy vamos a ver un sistema que nos permitirá mantener todo tipo de mensajes en secreto.

Ampliación del conocimiento (40 min)

Grupo: Pon a los estudiantes en parejas.

Distribuye Mantener el secreto de los datos- [guía de actividades \(Versión en español\)](#) para cada alumno.

Codificación y decodificación

Observaciones: Este es el mismo sistema que vimos antes, pero

El mensaje sirve para iniciar la revisión de lo que alguien necesita en un sistema de codificación. Los estudiantes deben entender que, sin conocer el sistema utilizado para codificar la secuencia en binario, no tienen forma de saber lo que significa.

Permita que los estudiantes intercambien ideas para mantener el mensaje en secreto. En la mayoría de los casos, los estudiantes pueden pensar en usar un código secreto en lugar de uno públicamente disponible. Esto podría funcionar, pero es mucho trabajo hacer un nuevo código para cada tipo de datos, y tendrías que repetir ese trabajo si su código fuera revelado accidentalmente.

Reducir el papel: Esta lección se puede hacer en línea. En lugar de colorear en los cuadros, los estudiantes pueden escribir una "X" en cada cuadro que sería de color negro.

con más emojis. Con tu compañero, decodifica la secuencia binaria y decide qué crees que significa el mensaje.

Permita que los estudiantes compartan lo que han encontrado y sus propias interpretaciones de los emojis.

Cifrado y descifrado

Observaciones: La siguiente cadena binaria utiliza el mismo código emoji que vio antes, pero el remitente no quería que otras personas lo leyeran. Con su pareja, intente descifrar el mensaje.

Permite que los estudiantes intenten descifrar el mensaje. Pueden notar que los códigos binarios no están en el sistema de codificación que se les ha dado.

Observaciones: Este mensaje ha sido encriptado. Eso significa que alguien lo cambió para que no podamos leerlo. Para leer el mensaje, primero debemos descifrarlo.

Vocabulario: Introduce los siguientes términos:

Encriptar: Para cambiar la información de modo que su significado esté oculto

- Alguien ha cifrado el mensaje, por lo que no podemos entenderlo.

Descifrar: Para cambiar la información para que se muestre su significado oculto

- Necesitamos descifrar el mensaje antes de poder leerlo.

Modela el descifrado del primer mensaje.

1. Copie el resto de la cadena binaria en la primera fila del cuadro.
2. Continúe repitiendo la tecla hasta que hayas llegado al final del cuadro. (La última repetición solo tendrá dos bits).
3. Para cada bit en la tercera fila del gráfico, colorea en el cuadrado si y sólo si los dos bits encima son iguales. Por ejemplo, si los dos bits superiores son blancos o ambos negros, colorea en el cuadrado. No colorea en el cuadrado si los dos bits son diferentes.

Una vez que el mensaje ha sido descifrado, permite que los estudiantes lo decodifiquen y hablen sobre lo que piensan que significa.

Recorrer la sala: Apoye a los estudiantes mientras descifran el mensaje en la segunda página.

Los estudiantes pueden no entender la diferencia entre la codificación / decodificación y el cifrado / descifrado. Lo principal para que los estudiantes entiendan es que las intenciones entre los dos son muy diferentes.

La codificación se usa para cambiar la forma de los datos, no para ocultar su significado a los demás. Por ejemplo, ASCII se utiliza para codificar caracteres en binario, pero la intención es que todos puedan decodificar la información. El objetivo es facilitar la tarea de almacenar y procesar información. El cifrado se utiliza para garantizar que solo el destinatario de la información pueda leerlo, se usa para seguridad y privacidad.

En este sistema de encriptación, el método de cifrado y descifrado es idéntico y utiliza la misma clave. En la mayoría de los sistemas de encriptación, ese no es el caso. Esta vez ha sido elegido por el mismo de simplicidad.

Para compartir el mensaje en línea, los estudiantes pueden escribir "B" para cada cuadrado negro y "W" para cada cuadrado blanco en la fila.

Encripta tu propio mensaje

Los estudiantes crean sus propios mensajes emoji y claves, luego los codifican y los encriptan.

Los estudiantes deben publicar sus mensajes encriptados públicamente, mientras mantienen en secreto sus mensajes y claves no encriptados. Permita que los estudiantes miren los mensajes de sus compañeros de clase e intenten descifrarlos sin una clave.

Indicación: Todos han compartido sus datos encriptados. ¿Crees que podrías descifrar los datos de alguien sin la clave? ¿Cómo? ¿Qué lo hace difícil?

Junte diferentes grupos y pida que intercambien claves secretamente. Deberían poder descifrar el mensaje del otro grupo. Permita que cada uno de los grupos comparta el mensaje emoji que descifraron y lo que creen que significa.

Reflexión

Indicación: ¿Cómo mantuviste tu llave segura al dársela a tu pareja?

El alumno debería compartir cómo se aseguraron de que nadie más pudiera ver la clave que usaban para encriptar sus datos.

Observaciones: Utilizaste la seguridad física para mantener tu llave segura, asegurándote de que nadie pudiera acceder físicamente a ella.

Inducir: ¿De qué manera utiliza la seguridad física para mantener seguros sus datos en línea?

Permita que los estudiantes discutan las preguntas con sus compañeros antes de compartirlas con la clase.

Observaciones: Para mantener nuestros datos seguros, debemos prestar atención a la seguridad digital y física. La seguridad digital incluye el uso de encriptación o protección de cosas con contraseñas. La seguridad física mantiene nuestros dispositivos y contraseñas físicamente seguros.

Transferencia del conocimiento (5 min)

Indicación: Cuando las personas se comunican en Internet, no pueden confiar en la seguridad física para mantener sus claves seguras. Trata de pensar en una forma en que todavía puedan comunicarse de forma segura, incluso si alguien pudiera leer todo lo que le enviaron.

El objetivo de esta discusión es resaltar el valor de cifrado, y que incluso con el conocimiento de algoritmo, es difícil descifrar un código sin la clave. Permita a los estudiantes la oportunidad de compartir sus ideas sobre cómo descifrar este código y probarlo. Luego, pídale a los estudiantes que digan qué es lo que lo dificulta.

El objetivo de la primera discusión es introducir la idea de la seguridad física, que incluye todo lo que hacemos en el mundo físico, como la prevención del acceso físico a computadoras y contraseñas. Los estudiantes deben entender que la seguridad física es tan importante como la seguridad digital cuando salvaguardan la información. Algunas cosas que pueden hacer los estudiantes para mantener su información segura puede ser guardar las contraseñas en un lugar seguro (o evitar escribirlas por completo) y mantener los dispositivos bloqueados cuando no estén en uso.

Permita que los estudiantes realicen una lluvia de ideas, luego compartan sus ideas con un compañero de clase.

Sugerencias para evaluar

Se sugieren los siguientes indicadores para evaluar formativamente los aprendizajes:

- Aplican varios métodos de encriptación para modelar la transmisión segura de información
- Identifican los derechos propios como los de los otros, y aplican estrategias de protección de la información en ambientes digitales.

Lección 32

Combinando representaciones

Propósito

En esta Lección, los estudiantes usan los tres tipos de sistemas de representación binarios (caracteres ASCII, números binarios e imágenes) para decodificar información en un registro. Después de ver una serie de bits y de que se les pida que los decodifiquen, se les presenta a los estudiantes la idea de que, para comprender la información binaria, deben comprender tanto el sistema que se está utilizando como el significado de la información codificada. Luego decodifican un registro que representa una mascota basada en una estructura determinada.

En lecciones anteriores, los estudiantes usaron diferentes sistemas de representación de forma aislada. Aquí, ven varios sistemas de representación usados juntos dentro del mismo registro. Deben ver que es importante no solo conocer los distintos códigos, sino también crear reglas sobre qué códigos se utilizan y en qué puntos del registro. Sin esta estructura, es imposible decodificar una serie de bits.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Usar múltiples sistemas binarios para decodificar información.
- Determinar el sistema de codificación más apropiado para una determinada información.

Lección sin conexión
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Combinando representaciones- Ejemplar](#)
- [Representación de mascotas- Ejemplar](#)

Para los estudiantes:

- [Registros de mascotas - Guía de actividades](#)
- [Tabla ASCII a Binaria- Referencia](#)

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Revisión: Revise rápidamente los tipos de sistemas de codificación binarios que se han cubierto: ASCII, imágenes y números binarios.

Indicación: Muestre a los estudiantes el código binario en Code Studio Bubble 2.

Pida que usen lo que saben sobre los códigos binarios para descubrir lo que significa.

Dé a los estudiantes la oportunidad de pensar por sí mismos, luego hablen en grupos de mesa antes de abrirse a una discusión completa en clase.

Dirija a los estudiantes a los [Niveles de Code Studio](#) correspondientes a la lección

[Representación de mascotas- Ejemplar](#)

Preguntar: Luego, diga a los estudiantes que el código es un número y pregúnteles si saben lo que significa. Pida que superen la respuesta de que el número es 67. Si un extraño se acercara a ellos y dijera “67” de la nada, ¿sabrían de qué estaba hablando esa persona?

Observaciones: Cuando usamos códigos, necesitamos saber dos cosas: el sistema que se está utilizando y el significado de la información en sí. Eso significa que cuando almacenamos algo en una computadora, tenemos que organizarlo para que sepamos estas cosas para cada parte de nuestro código binario.

Ampliación del conocimiento (40 min)

Preguntar: Imagine que está creando un sistema para guardar información sobre los estudiantes de la clase. ¿Qué tipo de información le gustaría saber sobre cada estudiante? ¿Qué sistema usaría para codificar esos tipos de información en binario?

Pantalla: Vaya al [tercer nivel en Code Studio](#), titulada “Student Record”, y revise la información con viñetas en el lado izquierdo de la pantalla.

Modelo: Modele la decodificación del registro del alumno con la clase.

Grupo: Agrupe a los estudiantes en parejas.

Objetivo: los estudiantes deben tener en cuenta que sin saber qué código se está utilizando, no pueden saber con certeza qué significa el binario. Específicamente, pueden ver que el código podría ser el carácter “C” en ASCII o el número 67 como un número binario. De hecho, podría ser un código que no habían visto antes, en cuyo caso podría significar cualquier cosa.

Objetivo: los estudiantes deberían encontrar diferentes significados posibles del número como la edad, la temperatura, etc. Nuevamente, sin contexto, el número podrá significar muchas cosas.

ASCII de 7 contra 8 bits: en lecciones anteriores, ASCII ha usado 7 bits, y los números han usado 8 bits. En esta lección, comenzamos a organizar la información en conjuntos de 8 bits, llamados byte. Esto nos facilita la organización de la información para que podamos leerla. Como ASCII solo usa 7 bits, agregamos un 0 al principio del código para llenar el espacio adicional.

Comenzando en 0: en nuestros registros, comenzamos en la fila 0 en lugar de la fila 1. Comenzar a contar desde el 0 en lugar del 1 es una forma común

Distribuir: Entregue Registros de mascotas- [Guía de actividades](#) y Tabla ASCII a Binaria- [Referencia](#) a cada par. Asegúrese de que cada grupo tenga acceso al widget binario.

Los estudiantes deben usar el widget binario para decodificar los números y sus tablas ASCII a Binarias para decodificar las letras. No es necesario decodificar la imagen porque ya está en el formato correcto para verla directamente.

En la segunda página, la imagen se reemplaza con un nuevo número entero y caracteres, porque la imagen en sí misma no contenía suficiente información para ser muy útil.

Indicación: En la hoja de trabajo, también se les pide a los estudiantes que piensen en una nueva información y cómo se codificará. Pida a los estudiantes que compartan sus ideas y las escriban en el pizarrón.

Observaciones: Cuando representamos algo en la computadora, tenemos que tomar decisiones sobre qué aspectos queremos registrar y cómo se registrarán esos aspectos. En la próxima lección, tendrá la oportunidad de crear su propia representación de algo y codificarlo en binario.

Transferencia del conocimiento (5 min)

Indicación del cuaderno

- ¿Qué tipo de información (número, personaje, imagen) le pareció más útil?
- ¿Qué lo hizo tan útil?

Sugerencias para evaluar

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Representan datos usando varios esquemas de encriptación con números binarios.

–
de hacerlo en Ciencias de la Computación.

Debido a que el nombre no ocupa todo el espacio disponible, el estudiante puede preguntarse por qué existen los bytes 06 y 07. Recuérdeles que, sin la estructura del registro, la computadora no sabrá dónde buscar. Si la imagen comenzara en Byte 06, la computadora lo interpretaría como un personaje en el nombre.

Lección 33

Crear una representación

Propósito

En esta lección, los estudiantes diseñan una estructura para representar su día perfecto utilizando los sistemas de representación binarios que han aprendido en este capítulo. Los estudiantes primero escribirán una breve descripción de su día perfecto y luego revisarán con un compañero para identificar las piezas clave de información que creen que una computadora podría capturar. Como clase, los estudiantes decidirán cómo se interpretará una tarjeta perforada de bytes de información para representar esas piezas de información. Luego, los estudiantes usarán el ASCII, el número binario y los formatos de imagen que han aprendido para representar sus días perfectos. A continuación, intercambian tarjetas perforadas e intentan descifrar cómo es el día perfecto del otro alumno.

Esta lección obliga a los estudiantes a lidiar con algunos de los desafíos de representar información en una computadora. Las computadoras son buenas para representar algunos tipos de información, como números o caracteres, pero esto obliga a las personas a representar información de maneras que no siempre capturan la experiencia analógica completa de un objeto o evento. El proyecto también sirve como un proyecto acumulativo para el capítulo, que reúne los diferentes sistemas de representación que los estudiantes han aprendido. A medida que los estudiantes pasen al siguiente capítulo, verán más de cerca cómo se usa la información para tomar decisiones una vez que está en un formato que se puede ingresar en una computadora.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (5 min)

Lección sin conexión
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Crear una representación: ejemplar](#)

Para los estudiantes:

- [Crear una representación - Guía de proyectos](#)
- [Tarjeta perforada](#)
- [Resuelve un problema de datos- rúbrica](#)

Objetivos

Los estudiantes serán capaces de:

- Elegir y justificar el uso de diferentes sistemas de representación binarios según la información que se represente.
- Codificar y decodificar información representada en números binarios y texto ASCII.
- Crear un sistema de representación generalizado para muchas instancias de un tipo complejo de información.

Datos usando varios esquemas de encriptación.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Revisión: Es posible que desee comenzar esta lección con una revisión rápida de los sistemas de representación que los estudiantes han aprendido en este capítulo. Es posible que desee señalar, en particular, que un byte de información (8 bits) tiene significado si conoces:

1. El sistema que se está utilizando.
2. El significado de la información en sí.

Sin 1 (el sistema que se está utilizando) no podrías distinguir un número de un personaje de una línea en una imagen. Incluso si sabes cuál es la información (por ejemplo, el número 8), no lo sabes necesariamente si ese número significa un tamaño de calzado, una edad o una calificación de película. Estos deberían ser puntos que puedes hacer basados en la Lección anterior.

Ampliación del conocimiento (40 min)

Grupo: Los estudiantes deberán trabajar en colaboración en este proyecto.

Distribuir: Entregue a los estudiantes Guía del proyecto.

Guía de proyectos: crea una representación

Indicaciones: Cierre los ojos y piensa en tu día perfecto. ¿Qué tiempo hace? ¿Qué hará? ¿Quién está con usted? Pase un minuto pensando en todos los detalles que pueda.

Paso 1:

Describe tu día perfecto: Pida a los estudiantes que escriban todos los detalles clave de su día perfecto en sus guías de actividades. Está bien que los estudiantes alternen entre enumerar y completar oraciones.

Paso 2:

Compartir con un compañero: Haga que los estudiantes elijan a un compañero para compartir su día perfecto. Deben continuar actualizando su información del Paso 1 a medida que brindan

Salte al proyecto: evalúe usted mismo, basándose en la comprensión de los estudiantes de la Lección anterior, si cree que estos puntos de la Lección anterior deben ser revisados. Si no, simplemente salta al proyecto.

Elegir categorías: probablemente sea un paso difícil para algunos estudiantes. Es posible que desee modelar cómo asignar información diferente a distintas categorías o como clase asignar información de los días perfectos de algunos voluntarios.

Sistemas posibles: Aquí hay un posible sistema para las tarjetas perforadas

- Número 1: Temperatura ese día
- Número 2: ¿Cuántas personas hay?
- Número 3: ¿Cuánto tiempo estás allí en horas?
- Caracteres 1-10: nombre de la Lección o ubicación
- Imagen: Imagen de la Lección

Facilitando la Discusión: Es probable que los estudiantes quieran crear sistemas que funcionen muy bien para su información, pero no para sus compañeros de clase. Haga hincapié en que los sistemas solo son útiles si son universales.

más información sobre su día perfecto.

Paso 3:

Identificar información: Haga que los estudiantes asignen la información sobre su día perfecto a una de tres categorías: Números, Caracteres o Imágenes.

Paso 4:

Revise las opciones de información: Los estudiantes deben volver a reunirse con su compañero y comparar las notas sobre cómo asignaron diferentes piezas de información a diferentes categorías.

Paso 5:

Lección completa de la clase - Acordar una tarjeta perforada para toda la clase: Muestre la tarjeta perforada de la guía del proyecto en algún lugar donde toda la clase pueda verla. Haga que los estudiantes compartan las categorías de información que desean representar. Explique que la clase tendrá que acordar cómo van a usar cada fila de los números, los personajes y las secciones de imágenes de la tarjeta perforada.

Paso 6:

Completar la tarjeta perforada: Muestre el sistema en algún lugar donde todos los estudiantes puedan verlo. Haga que los estudiantes completen sus tarjetas perforadas usando este sistema de clase.

Paso 7:

Intercambiar y decodificar tarjetas perforadas: Los estudiantes deben encontrar un nuevo compañero que no esté familiarizado con su día perfecto. Deberían intercambiar tarjetas perforadas y decodificar la información codificada. Hay espacio en la parte posterior de la guía de actividades para escribir

- La información del número y su significado
- La información del personaje y su significado
- Lo que ellos creen que la imagen está mostrando

Después, los estudiantes deben escribir lo que creen que sucede en el día perfecto de su compañero de clase en el espacio provisto.

Un sistema que solo funciona para una persona no es muy útil y todos los sistemas implicarían algunas compensaciones.

Paso 8:

Compartir y reflexionar: Los estudiantes deben reunirse con una copia de seguridad y comparar su descripción del día perfecto de su compañero de clase con la intención original. ¿Qué fueron capaces de capturar? ¿Qué se perdió?

Transferencia del conocimiento (5 min)

Recolectar: Crear una representación- [Guía de proyectos](#) y [Tarjeta perforada](#). Asegúrese de hacer un seguimiento con quienes se han asociado los estudiantes para que pueda calificar tanto su codificación como el trabajo de decodificación.

Preguntar: Si lo desea, puedes usar las siguientes instrucciones para informar el proyecto.

- Si le hubiesen dado otro byte de información (otra línea en la tarjeta perforada) ¿cómo habrías querido usarlo?
- Si pudiera mover un byte de una categoría (números, imágenes, caracteres) a otra, ¿cuál cambiarías?

Discute: Haga que los estudiantes escriban sus ideas en silencio antes de compartirlas con un compañero y, finalmente, con toda la clase.

Dirija a los estudiantes a los Niveles de Code Studio correspondientes a la Lección. [Crear una representación: ejemplar.](#)

Objetivo: Estas preguntas deberían ayudarlo a evaluar en un nivel alto si los estudiantes han entendido los desafíos de esta Lección. En sus justificaciones para la primera pregunta, deberían describir qué información querrían capturar y cómo querrían representarla. En la segunda pregunta, debe presionar sobre por qué está bien perder el byte de una categoría y por qué esa pérdida de información será compensada por el nuevo byte en otra categoría.

Contenido 2

Cómo funciona el BIG DATA

Resumen

- **Lección 34:** Dispositivos del futuro
- **Lección 35:** Anatomía de un Smartphone
- **Lección 36:** Smart cities
- **Lección 37:** Smart buildings
- **Lección 38:** Creando un anuncio en Facebook
- **Lección 39:** El boom de las redes sociales
- **Lección 40:** El poder de las redes sociales
- **Lección 41:** Open data para un mundo más transparente
- **Lección 42:** Periodismo de datos y data scraping
- **Lección 43:** La cuarta revolución industrial

Objetivos

- **OA 3.** Desarrollar y programar algoritmos para ejecutar procedimientos matemáticos, realizar cálculos y obtener términos definidos por una regla o patrón.
- **OA i.** Buscar, seleccionar, manejar y producir información matemática/cuantitativa confiable a través de la web.

Referencias

- <https://educaixa.org/es/programa-bigdata>

¹ **Licencia de uso de los recursos EduCaixa:** La licencia básica de uso de los recursos EduCaixa será de tipo Creative Commons, concretamente en la modalidad **by-nc-nd** (no permite realizar un uso comercial de la obra original ni la generación de las obras derivadas).

Lección 34

Dispositivos del futuro

Propósito

Los dispositivos del futuro ya son nuestro presente. Por lo tanto, se espera que los estudiantes sean capaces de identificar dispositivos ya existentes, e idear nuevos dispositivos que potencien el desarrollo de ciudades inteligentes, sostenibles y amigables con el medio ambiente.

Secuencia para el aprendizaje

Conocimiento inicial (20 min)

Ampliación del conocimiento (15 min)

Transferencia del conocimiento (20 min)

Objetivos

Los estudiantes serán capaces de:

- Identificar que, gracias a los sensores, a la conexión a internet y al análisis de los datos, aparecen nuevos dispositivos con múltiples funcionalidades que cambian nuestra cotidianidad
- Idear wearables o dispositivos vinculados al internet de las cosas.

Preparación

- (Opcional) vea el Video [El internet de las cosas](#)
- Imprima o mantenga lista y disponible la Infografía [Tu “yo” cuantificado](#) para cada estudiante y Guía de trabajo Dispositivos del futuro
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los Profesores:

- Video (respaldado en pendrive o computador) – [El internet de las cosas](#)
- Tener copia impresa de Infografía – [Tu “yo” cuantificado](#) y Guía de trabajo – Dispositivos del futuro

Para los estudiantes:

- Video – [El internet de las cosas](#)
- Infografía – [Tu “yo” cuantificado](#)
- Píldora informativa – [La mujer más “datificada” del mundo](#)
- Guía de trabajo – Dispositivos del futuro

Estrategia de aprendizaje

Conocimiento inicial (20 min)

Pregunta disruptiva inicial + video

Se desarrolla en grupos usando como guía la guía de trabajo Dispositivos del futuro e iniciándose con la siguiente pregunta disruptiva:

“ideen cómo podría el armario de su dormitorio enviar un mensaje a sus celulares, recomendándoles que ropa debieran usar cada mañana”

Para contestarla, los estudiantes deben consultar el video el [Internet de las cosas](#) y la píldora informativa un [Hogar inteligente como asistente personal](#).

Ampliación del conocimiento (15 min)

Idear un wearable

Siguiendo con la guía de trabajo Dispositivos del futuro, los estudiantes deben idear un nuevo wearable intentando que sea lo más original posible e identificando qué tipo de dispositivo sería el más adecuado para obtener y procesar la información (unas gafas, una pulsera, un colgante, una cámara acoplada a nuestra chaqueta, etc.)

Los estudiantes pueden consultar la infografía [Tu “yo” cuantificado](#) y la píldora informativa [La mujer más datificada del mundo](#).

Algunos ejemplos de wearables originales pueden ser dispositivos que nos indiquen cuánto azúcar hemos comido, cuánto rato hemos hablado durante el día, a cuántos amigos hemos visto hoy, etc.

Transferencia del conocimiento (20 min)

Puesta en común

Una vez completada la ficha de trabajo, se comparten con toda la clase los diversos objetos ideados por los grupos.

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes

Vocabulario

- **Dispositivo:** elemento conectado a una unidad de procesamiento computacional.
- **Wearable:** dispositivo electrónico inteligente usado por un individuo.

a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?
- ¿Reconoces que dispositivos forman parte del internet de las cosas?
- ¿Puedes definir que es un wearable y cómo funciona?

Dispositivos del futuro

Los dispositivos conectados a internet son la fuente principal de generación de datos, siendo el smartphone el gran protagonista gracias a todos los sensores capaces de recoger información que lo componen.

Revisen el video [El internet de las cosas](#) y la píldora informativa [Un hogar inteligente como asistente personal](#) e ideen cómo podría el armario de sus piezas enviarle un mensaje al celular recomendándole qué ropa deberían usar cada mañana.

Piensen en qué sensores necesitaría para recoger datos, cómo los procesaría y qué tipo de app se necesitaría para recibir toda esa información.

El armario inteligente

Ahora ideen un wearable que aún no exista. Intenten que sea lo más original posible y expliquen el tipo de dispositivo que sería, los sensores que necesitaría y qué tipo de información nos proporcionaría. Consulten la infografía [Tu “yo” cuantificado](#) y la píldora informativa [La mujer más datificada del mundo](#) para estimular su capacidad creativa.

El wearable del futuro

Sugerencias para evaluar

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Recomiendan mejoras en el diseño de dispositivos informáticos, basándose en un análisis de cómo los usuarios interactúan con los dispositivos.

Lección 35

Anatomía de un smartphone

Propósito

Los teléfonos inteligentes o smartphones (en inglés) son dispositivos tecnológicos que hacen la vida del usuario mucho más fácil. Sin embargo, el usuario debe hacerse consciente del dilema que surge entre privacidad funcionalidad, por ejemplo. Es por esto, que el estudiante debe ser capaz de reconocer un smartphone y sus principales partes y funcionalidades.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (15 min)

Transferencia del conocimiento (25 min)

Objetivos

Los estudiantes serán capaces de:

- Identificar diversos sensores que generan datos presentes en un smartphone.
- Ser conscientes de la cantidad de datos sobre nuestras acciones que recogen los smartphones.
- Ser conscientes de los riesgos para la privacidad que puede tener el uso de los datos generados por los sensores.

Preparación

- Tener preparada y lista para presentar Infografía interactiva [Anatomía de un Smartphone](#)
- Imprima o mantenga lista y disponible las guías de la Lección Anatomía de un smartphone
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#)

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los Profesores:

- Copia impresa de la Guía de la Lección – [Anatomía de un Smartphone](#)
- El debate ping-pong (para el docente)

Para los estudiantes:

- Guía de la Lección impresa – [Anatomía de un Smartphone](#)

Vocabulario

- **Smartphone:** teléfono inteligente.
- **Privacidad:** cualidad de lo que es privado.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Anatomía de un smartphone

Los estudiantes revisan de forma individual o por equipos la infografía [Anatomía de un smartphone](#). Durante unos minutos, de forma individual o en equipo, los estudiantes investigan cuáles son los sensores activos en su smartphone, con el objetivo de averiguar cuál es el uso que hace el celular de ellos para funcionar y qué aplicaciones los utilizan. Deben rellenar la tabla de la ficha de trabajo, completándola con una estimación de los riesgos para su privacidad.

Ampliación del conocimiento (15 min)

Preparando el debate

Planteamos en clase de la tensión que existe entre privacidad y funcionalidad en el uso de los celulares. Por un lado, está la comodidad y eficacia de las funcionalidades que, gracias a los sensores, ofrecen el smartphone y sus aplicaciones. Por el otro, el riesgo para la privacidad que puede representar el uso por parte de estas mismas aplicaciones de la información recogida por los sensores.

Cada equipo elabora cinco argumentos a favor y cinco argumentos en contra relacionados con la siguiente afirmación:

Si no comparto mis resultados o no vinculo mi Lección con el celular a mis redes sociales, me divierto de forma totalmente anónima.

Transferencia del conocimiento (25 min)

Debate ping-pong

Se lleva a cabo el debate según el formato del Debate ping-pong, que se explica en el documento para los profesores. Conviene finalizar el debate explicitando de manera resumida los argumentos a favor y en contra y, si fuera el caso, las conclusiones a las que se ha llegado.

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?
- ¿Eres consciente de la cantidad de sensores que usa tu smartphone y cómo captan los datos de

tus actividades?

En adición a lo anterior, podría proponerse a los estudiantes que revisen los permisos de acceso a sensores y datos que tienen autorizados en sus propios dispositivos (micrófono, cámara, ubicación, etc.)

Debate ping-pong – Privacidad vs funcionalidad (solo para el docente)

Preparar argumentos a favor y en contra

Inicialmente presentamos a los estudiantes la cuestión a debate, que en nuestro caso será:

“Si no comparto mis resultados o no vinculo mi Lección con el celular a mis redes sociales, me divierto de forma totalmente anónima”.

Cada equipo elabora cinco argumentos a favor y cinco argumentos en contra de la afirmación.

Estos serán los argumentos que expondrán en el debate. Para prepararlos, parten del análisis que han efectuado sobre el uso que hacen el smartphone y las aplicaciones de los datos que generan los sensores.

Para preparar sus argumentos necesitarán tener presente que al utilizar un smartphone generamos diferentes tipos de datos, entre ellos:

- Los que necesitan de la participación del usuario para llegar al dispositivo (correos, nombres de usuario, contraseñas, teléfonos celulares, textos en aplicaciones, fotografías, videos, mensajes de voz, búsquedas de páginas web, likes, etc.).
- Los metadatos que el smartphone asocia automáticamente a muchos de los datos que proporcionamos con nuestra Lección (por ejemplo, la fecha, la hora, la aplicación que hemos utilizado, etc.).
- Los datos que se recopilan de forma automática a partir de los distintos sensores que el dispositivo lleva incorporados.

Al analizar los riesgos para la privacidad, los estudiantes suelen poner poco énfasis en los datos generados por los sensores recopilados automáticamente por los smartphones.

Los sensores son imprescindibles para los smartphones ofrezcan determinadas funcionalidades y servicios (por ejemplo, el ajuste de la luminosidad de la pantalla a la luz ambiente requiere de un sensor de luminosidad que mida la intensidad de la luz ambiente y, con ese dato, un programa provocará el ajuste). Pero también hay muchas aplicaciones que utilizan estos datos para dar servicios que pueden poner en riesgo la privacidad del usuario (por ejemplo, los sensores GPS permiten determinar la ubicación del celular, lo que puede servir para averiguar dónde está en caso de pérdida, pero también para rastrear el trayecto que ha hecho la persona que lo utiliza).

La cuestión que se plantea tiene la finalidad de provocar la reflexión sobre la información recopilada automáticamente y de propiciar la toma de conciencia de su existencia y de la necesidad de hacer un buen uso de ella.

Tanto en la preparación como en las conclusiones del debate, es recomendable que el docente refuerce positivamente los argumentos que inciden en la necesidad de controlar los ajustes de

privacidad de los smartphones, de ser cuidadosos con los permisos de acceso que se dan a las aplicaciones, de la idea que las aplicaciones gratuitas a menudo tienen un modelo de negocio basado en la recopilación de datos de los usuarios y, en general, de aquellos argumentos que incidan en la capacidad de cada persona de ser propietario y gestionar los datos que origina.

Debate ping-pong

El debate se realizará con la dinámica del debate ping-pong, en el que los argumentos a favor y en contra pasan de un equipo a otro.

Estructura del debate: Preparación. Analizar el tema desde distintos ángulos. Los equipos piensan en los pros y contras de cada opción y los escriben. Tiempo_ alrededor de 10-15 minutos. Cada equipo elige a un portavoz o deciden turnarse el papel de portavoz en cada turno.

Empieza el debate:

Primer equipo. Argumento a favor: el primer portavoz abre el debate ping-pong con un discurso. Resume el tema brevemente y presenta un argumento a favor. El portavoz solamente puede presentar un argumento. Tiempo 1-2 minutos.

Siguiente equipo. Escuchar, repetir, argumento en contra: el segundo portavoz repite el argumento expuesto por el primer portavoz y presenta un argumento en contra. De ser posible, debería intentar refutar el argumento del primer portavoz.

Siguiente equipo. Escuchar, repetir, argumento a favor: hasta ahora han hablado dos estudiantes de dos equipos diferentes. El siguiente portavoz repite el argumento en contra y añade otro argumento a favor. De esta manera, los pros y los contras vuelan de un lugar a otro como una bola en una partida de ping-pong. Es importante que cada portavoz repita el argumento que se acaba de plantear para practicar la escucha activa y para vincular de manera explícita su argumento con el grupo anterior. Otras rondas. Resumen: cada portavoz presenta un nuevo argumento. EL debate ping-pong puede durar hasta 20 minutos y debe realizarse rápidamente.

Una alternativa a la dinámica expuesta es que el turno de un equipo a otro lo decide la persona que modera, de modo que todos los equipos deben estar atentos a las aportaciones de sus compañeros y dispuestos a aportar argumentos para ambas opciones.

Otra posibilidad es dejar que cada equipo pueda pedir un turno, levantando la mano, si cree que dispone de un contraargumento convincente al que se acaba de exponer. En este caso quien modera puede asignar los turnos según crea conveniente, siempre velando por la participación equitativa de todos los equipos.

El papel de moderador puede llevarlo a cabo el docente, pero merece la pena contemplar la posibilidad de que este rol lo desarrolle algún estudiante.

Sugerencias para evaluar

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Modelan cómo el hardware y el software de un dispositivo funcionan juntos como un sistema para realizar las tareas

Lección 36

Smart cities

Propósito

La incorporación de dispositivos tecnológicos a nuestras vidas hace que todo se más fácil y cómodo. Sin embargo, ¿estamos conscientes de cómo la tecnología repercute en nuestras ciudades? ¿Cuáles son los efectos positivos o negativos de la incorporación de tecnologías inteligentes para nuestras ciudades?

En esta Lección, los estudiantes identificarán las potencialidades que las tecnologías inteligentes ofrecen para el desarrollo de ciudades más participativas y sostenibles.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (30 min)

Transferencia del conocimiento (10 min)

Objetivos

Los estudiantes serán capaces de:

- Conocer las características de una Smart city.
- Identificar en qué aspectos podría mejorar su localidad.
- Idear soluciones recurriendo a las posibilidades tecnológicas, la participación ciudadana y el uso de datos masivos que mejoren la gestión de la energía, de los recursos y del espacio urbano de su localidad.

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- Video – [Smart cities](#)
- Infografía – [Una ciudad construida en datos](#)
- Infografía interactiva – [A qué se enfrentan las grandes ciudades](#)
- Píldora informativa – [En el interior de una “Smart city”](#)
- Guía de trabajo – Nuestra “smart city”

Vocabulario

- **Smart cities:** Ciudades inteligentes.

Preparación

- (Opcional) vea el video [Smart cities](#)
- Imprima o mantenga lista y disponible la Infografía [Una ciudad construida en datos](#), Píldora informativa [En el interior de una “Smart city”](#), Guía de trabajo Nuestra “smart city”
- Tenga preparadas Infografía interactiva [A qué se enfrentan las grandes ciudades](#)
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Identificar retos de su localidad

La Lección se desarrolla grupalmente. A partir de la guía de trabajo Nuestra “smart city”, los estudiantes deberán escoger los tres retos de mejora prioritarios a los que se enfrenta su ciudad entre ocho posibles propuestas.

Ampliación del conocimiento (30 min)

Ver el video (5 min)

Visualización de forma grupal del video [Smart cities](#).

Ideando soluciones (25 min)

Cada grupo debe idear en conjunto posibles soluciones o mejoras para los tres retos de su localidad que marcó al inicio de la sesión. Las soluciones o mejoras deben usar las posibilidades tecnológicas actuales y la gestión de datos masivos, siempre teniendo en cuenta la participación ciudadana.

Para obtener referencias y ejemplos, los grupos pueden consultar la infografía [Una ciudad construida en datos](#), la infografía interactiva [A qué se enfrentan las grandes ciudades](#), la píldora informativa [En el interior de una “smart city”](#) o realizar una búsqueda en internet usando el concepto clave “smart city” o “ciudad inteligente”.

Transferencia del conocimiento (10 min)

Puesta en común

Cada grupo expondrá ante toda la clase las ideas que ha desarrollado. Y de manera individual, los estudiantes elaborarán su propia definición de smart city.

Una vez finalizada la etapa de Transferencia del conocimiento, se dará paso a la siguiente Lección de metacognición.

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

- ¿Serías capaz de identificar soluciones tecnológicas que incorporan las smart cities para mejorar la calidad de vida y sostenibilidad?

Sugerencias para evaluar

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Diseñan proyectos que combinen componentes de hardware y software para recopilar e intercambiar datos.

Lección 37

Smart buildings

Propósito

La incorporación de dispositivos tecnológicos a nuestras vidas hace que todo sea más fácil y cómodo. Sin embargo, ¿estamos conscientes de cómo la tecnología repercute en nuestras ciudades, específicamente en las edificaciones? ¿Cuáles son los efectos positivos o negativos de la incorporación de tecnologías inteligentes para este tipo de edificaciones?

En esta Lección, los estudiantes identificarán las potencialidades que las tecnologías ofrecen para el desarrollo de ciudades y edificios inteligentes más participativos y sostenibles.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (20 min)

Transferencia del conocimiento (20 min)

Objetivos

Los estudiantes serán capaces de:

- Conocer qué es un Smart building.
- Identificar soluciones tecnológicas que permitan una gestión inteligente de edificios e industrias.

Preparación

- Imprima o mantenga lista y disponible la Píldora informativa [Ciudades inteligentes, edificios inteligentes](#), y Guía de trabajo “Smart institute”
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- Píldora informativa – [Ciudades inteligentes, edificios inteligentes](#)
- Guía de trabajo – “Smart institute”

Vocabulario

- **Smart building:** Edificio inteligente.
- **Smart institute:** Instituto inteligente.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Smart buildings

Lección individual en la que los estudiantes leerán de manera individual la píldora informativa [Ciudades inteligentes, edificios inteligentes](#) y realizarán sus propias definiciones de smart building.

Ampliación del conocimiento (20 min)

Smart institute

La guía de trabajo Smart institute se desarrolla en grupos. A partir de la información obtenida en el paso anterior y mediante una búsqueda en internet usando los términos “smart buildings” o “edificios inteligentes”, cada grupo debe anotar mejoras o soluciones tecnológicas que se podrían incorporar en el instituto para convertirse en un smart institute. Los ejes de las mejoras serán:

- Iluminación
- Consumo de agua
- Climatización
- Obtención de energía verde

Transferencia del conocimiento (20 min)

Análisis de las propuestas y puesta en común

Una vez encontradas varias soluciones, cada grupo selecciona una de ellas y analiza sus debilidades y fortalezas para exponerlas al resto de la clase.

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?
- ¿Serías capaz de identificar soluciones tecnológicas que incorporan los smart buildings para mejorar la calidad de vida y sostenibilidad?

Sugerencias para evaluar

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Recomiendan y elaboran una serie de medidas y actuaciones que conducen a la mejora en el diseño de dispositivos informáticos, basándose en un análisis de cómo los usuarios interactúan con los dispositivos.

Lección 38

Creando un anuncio en Facebook

Propósito

Las redes sociales son herramientas que están presentes en nuestro día a día, ya sea para ver una foto, comentar un video, charlar con familia y amigos, entre otras funciones.

En esta Lección, se busca que los estudiantes sean capaces de crear un anuncio con la ayuda de diferentes herramientas de edición que la red social Facebook ofrece, para luego compartirlo con los contactos.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (70 min)

Transferencia del conocimiento (20 min)

Objetivos

Los estudiantes serán capaces de:

- Ser conscientes de que las redes sociales pueden almacenar muchos datos de sus usuarios.
- Experimentar con el diseño y prototipo de un anuncio de redes sociales.
- Identificar la publicidad como una de las principales vías de ingresos de las redes sociales gratuitas.

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- Video – [El origen de las redes](#)
- Infografía – [Lo que no sabes de las redes](#)
- Píldora informativa – [Facebook, la red que todo lo ve](#)

Vocabulario

- **Redes sociales:** Aplicaciones o sitios web que permiten el intercambio de información entre personas.

Preparación

- (Opcional) vea el video [El origen de las redes](#)
- Imprima o mantenga lista y disponible la Infografía [Lo que no sabes de las redes](#), y Píldora informativa [Facebook, la red que todo lo ve](#)
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Visualización del video inicial e infografía

Toda la clase visiona el video [El origen de las redes sociales](#) y la infografía [Lo que no sabes de las redes](#).

Ampliación del conocimiento (70 min)

Presentación de la Lección (15 min)

Presentación de la guía de trabajo con la que los estudiantes crearán un anuncio en Facebook de las propias cuentas de YouTube o Instagram (u otras que elijan) destinado a obtener más seguidores. Creación de las parejas de trabajo y entrega de la ficha que guiará el trabajo para realizar en la guía de trabajo Nuestro anuncio en Facebook.

Creación de un anuncio en Facebook (55 min)

Lectura de la píldora informativa [Facebook, la red que todo lo ve](#) y creación del anuncio, que incluye:

- Identificar el contenido clave de la cuenta que se quiere promocionar.
- Definir el público objetivo de la cuenta.
- Delimitar el target del anuncio usando las herramientas de Facebook (si no disponen de una cuenta, deberán crearla).
- Escoger imagen y texto del anuncio.
- Rellenar la plantilla de presentación.

Transferencia del conocimiento (20 min)

Presentación del anuncio y reflexión final

Presentación de los anuncios creados al resto de compañeros de la clase.

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?

- ¿Cómo te sentiste durante la Lección?
- ¿Eran conscientes de que las redes sociales usan sus datos y preferencias para mostrarnos anuncios personalizados?

Nuestro anuncio en Facebook

A todos nos gusta hacer cosas diferentes, como escuchar música, salir con los amigos, cocinar, ir de compras, los autos, el fútbol, etc. Cada vez más, nos gusta mostrar en las redes sociales todos estos hobbies, gustos, actividades o relaciones que tenemos, ya sea a través de likes, fotografías, videos o comentarios. Toda esta información, si es pública, puede ser utilizada también para generar campañas publicitarias.

En esta guía de trabajo les proponemos utilizar la información que podemos obtener de Facebook para promocionar una cuenta de Instagram o YouTube (puede ser suya u otra de su preferencia).

Paso 0 – Facebook y los datos

Comiencen el ejercicio leyendo individualmente la píldora informativa [Facebook, la red que todo lo ve](#) para conocer qué datos suele almacenar y usar Facebook.

Paso 1 – Definan el contenido de la cuenta de Instagram o canal de YouTube

Escojan qué cuenta de Instagram o YouTube quieren promocionar mediante un anuncio. Es recomendable que la cuenta tenga una orientación pública y una temática concreta. Pueden escoger alguna que sea suya u otra que les llame la atención.

Cuenta de Instagram o YouTube que queremos promocionar:

Tipo de contenido de la cuenta seleccionada, como, por ejemplo: música de rap, baloncesto, paisajes, amigos, consejos de estilismo o moda, videojuegos...

Paso 2 – Definan el perfil del público objetivo

Público que podría estar interesado en seguir la cuenta seleccionada.

Sexo	
Edad	
Localidad	
Idioma	
Aficiones	
Gustos	

Paso 3 – Encuentren cuánta gente en Facebook coincide con ese perfil

Accedan a una cuenta personal de Facebook.

Localicen el enlace PUBLICIDAD, ubicado en la parte inferior de la columna derecha de anuncios que aparece en nuestros muros.

Mediante este enlace inicien el proceso de creación de un anuncio para Facebook.

Hagan clic en el botón CREA UN ANUNCIO, situado en la esquina superior derecha de la pantalla.

Accederán al panel de control para generar anuncios. Para conocer la cantidad de perfiles de Facebook que coinciden con el público que han definido previamente en el punto 2, rellenen la información del apartado PÚBLICO que pueden ver en la columna izquierda de nuestro panel de control.

Introduzcan los datos del público elegido. Existen tres tipos de anuncio en función del objetivo: reconocimiento, consideración o conversión. Para este caso, es mejor simplificar, por eso elijan RECONOCIMIENTO DE MARCA en el apartado Reconocimiento.

Asígnenle un nombre a la campaña para poder avanzar.

Creen una cuenta publicitaria: indiquen el país, la moneda y la zona horaria.

Llegados a este punto aparece el formulario para establecer unas determinadas características que definen al grupo al que quieran impactar con el anuncio. Asignen los valores que vean oportunos y que lo representen en la medida de lo posible.

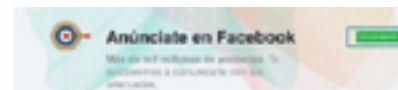
Definan la población, la edad, el sexo y los idiomas, e introduzcan características específicas en el apartado SEGMENTACIÓN DETALLADA.

Una vez rellenados estos datos, se mostrará una gráfica con las características del público seleccionado y el número de perfiles que coinciden con el mismo. Copien esa imagen y péguenla en un documento.

Paso 4 – Piensen un anuncio dirigido a ese público para promocionar la cuenta

Hagan una búsqueda individual de imágenes usando el buscador de Google.

Cada miembro de la pareja tiene que buscar y seleccionar una imagen clara y representativa de la cuenta que quieran promocionar que llame la atención del público objetivo. Peguen



aquí las propuestas y escojan la que más les guste.

Escriban un breve texto que acompañará a la imagen seleccionada. El texto no puede tener más de 15 palabras y debe ser claro, directo y original.

Paso 5 – Presentación del anuncio

Coloquen en su documento la imagen y el texto seleccionados para presentar el anuncio al resto de compañeros de clase.

Lección 39

El boom de las redes sociales

Lección en línea

Propósito

Las redes sociales han llamado completamente nuestra atención con diferentes características. Y cada una de estas redes sociales, existente actualmente, sacan lo mejor de sí para poder atraer nuevos usuarios y adaptarse a las necesidades de éstos. Por esto la presente Lección busca que los estudiantes identifiquen en las actuales redes sociales los temas y características principales de los perfiles de los usuarios.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (10 min)

Transferencia del conocimiento (30 min)

Objetivos

Los estudiantes serán capaces de:

- Conocer las características de las redes sociales.
- Ser conscientes de la cantidad de datos personales y registro de acciones que compartimos en las redes sociales.
- Identificar que el beneficio de las redes sociales puede basarse en usar y vender nuestros datos a terceros.

Preparación

- Imprima o mantenga lista y disponible la Infografía [El negocio de las redes](#), Píldora informativa [Nuestro “yo” digital](#), Píldora informativa [Snapchat: una aplicación irresistible](#), Píldora informativa [Cambiar las cosas con un clic](#), y Guía de trabajo El boom de las redes sociales
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- Infografía – [El negocio de las redes](#)
- Píldora informativa – [Nuestro “yo” digital](#)
- Píldora informativa – [Snapchat: una aplicación irresistible](#)
- Píldora informativa – [Cambiar las cosas con un clic](#)
- Guía de trabajo – El boom de las redes sociales

Vocabulario

- **Datos:** secuencia de elementos que en informática se transforma en información.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Presentación de la Lección

Presentación de la Lección, en la que los estudiantes analizarán 10 redes sociales completando una tabla de datos. Creación de los grupos de trabajo y presentación de la guía El boom de las redes sociales.

Ampliación del conocimiento (10 min)

Selección de 10 redes sociales

Se desarrollará grupalmente con la guía de la Lección. Los estudiantes deberán identificar y anotar las dos redes sociales que más usen y anotar dos redes sociales más que le llamen la atención (LinkedIn, Vimeo, YouTube, Pinterest, Periscope, Snapchat, etc). Pueden encontrar información y características de otras redes consultando la infografía interactiva [El negocio de las redes](#) o cualquiera de las píldoras informativas.

Transferencia del conocimiento (30 min)

Búsqueda de información para el análisis de las redes seleccionadas

Consultando la información anterior y realizando búsquedas en internet, los estudiantes completarán la tabla de análisis, donde reflejarán de cada red:

Tipo, beneficios que aportan en el terreno individual y social, personas conectadas, datos que se comparten en esas redes, canales para obtener beneficios económicos y finalmente, una lista de posibles riesgos derivados de un uso no responsable.

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?
- ¿Serías capaz de enumerar beneficios individuales y sociales que aportan las redes sociales?

Lección 40

El poder de las redes sociales

Lección en línea

Propósito

¿Somos conscientes del poder e influencia que tienen las redes sociales en la vida de las personas, y cómo nuestra privacidad puede ser vulnerada? Si bien las redes sociales son una herramienta importante en la vida de las personas, su uso siempre estará cuestionado.

En esta Lección, se espera que los estudiantes puedan reconocer las potencialidades y amenazas que las redes sociales pueden representar cuando el uso de éstas no es debido o bien dirigido.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (25 min)

Transferencia del conocimiento (20 min)

Objetivos

Los estudiantes serán capaces de:

- Identificar formas de mantener la privacidad de nuestros datos en las redes sociales.
- Reconocer los riesgos y beneficios de las redes sociales.

Preparación

- (Opcional) vea el video [El poder de las redes sociales](#)
- Imprima una guía de trabajo para cada estudiante.
- Imprima o mantenga lista y disponible la píldora informativa [Mi privacidad y datos ¿qué hacer?](#)
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- Video – [El poder de las redes sociales](#)
- Píldora informativa – [Mi privacidad y datos ¿qué hacer?](#)
- Guía de trabajo – El poder de las redes sociales

Vocabulario

- **Privacidad:** cualidad de lo que es privado.

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Visualización del video

Visualización grupal de video [El poder de las redes sociales](#).

Ampliación del conocimiento (25 min)

Reflexión sobre el uso de las redes sociales

A partir de la guía de trabajo, los estudiantes deben marcar qué acciones del listado que encontrará en la píldora informativa [Mi privacidad y datos ¿qué hacer?](#) llevan a cabo para proteger su privacidad en las redes sociales. Después debe seleccionar las tres redes que más utiliza a partir del análisis realizado en la anterior sesión y contestar a las siguientes dos preguntas de forma argumentada a partir de todas las informaciones vistas durante el desarrollo de las diversas sesiones del programa educativo BigData:

- ¿Las implicaciones que tiene el uso de estas redes compensan los beneficios que ofrecen?
- ¿Pueden las redes sociales hacer un mundo más justo?

Transferencia del conocimiento (20 min)

Debate y contraste de opiniones

Puesta en común de las acciones que realizan los estudiantes para proteger su privacidad y debate abierto de toda la clase a partir de las respuestas dadas en las preguntas de reflexión.

El debate se ha de moderar para que las intervenciones de los estudiantes ejemplifiquen cómo las redes sociales modifican nuestras rutinas diarias y cómo su uso ha de ir acompañado de una reflexión responsable que pondere las implicaciones que conlleva.

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?
- ¿Serías capaz de enumerar acciones que podemos tomar para asegurar que nuestra seguridad no sea vulnerada durante el uso de una red social?

El poder de las redes sociales

El uso de las redes sociales contribuye a crear nuestra identidad, huella y reputación digital y, por tanto, genera la necesidad de ser responsables con la información que publicamos en ellas. Consulten el listado de acciones que ayuda a garantizar la privacidad de la píldora informativa [Mi privacidad y datos ¿qué hacer?](#).

Marquen de forma individual las que conozcan o no y cuales hacen:

	¿Lo conocía?	¿Lo hago?
Leer las políticas de privacidad que tienen los servicios que usamos		
Proteger nuestros dispositivos móviles con una clave		
Administrar las contraseñas de un modo responsable e inteligente		
Limpiar el historial y las cookies de nuestros dispositivos y establecer la navegación privada		
Revisar la configuración de privacidad		
Agregar como contactos solo a personas que conozcamos		
No proporcionar el correo, teléfono o código postal de forma continuada		
Tomar precauciones cuando transferimos datos financieros a través de internet		
Poner una alerta en google con tu nombre		
Desactivar la geolocalización cuando no se precise su uso		
Cerrar todas las sesiones a las que accedas		
Cifrar nuestros dispositivos		
Actualizar siempre todos los programas con las últimas versiones		

Teniendo en cuenta lo visto en el video [El poder de las redes sociales](#) y lo que han aprendido en las anteriores sesiones, contesten de forma argumentada a las siguientes preguntas para realizar un breve debate entre toda la clase.

¿Las implicaciones y riesgos que tiene el uso de estas redes compensan los beneficios que ofrecen?

¿Pueden las redes sociales hacer un mundo más justo?

Sugerencias para evaluar

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Analizan y evalúan el impacto de las tecnologías digitales en contextos sociales, económicos y culturales.

Lección 41

Open data para un mundo más transparente

Propósito

Conocer las implicaciones en el mundo laboral que tiene la cuarta revolución industrial y saber que el acceso a datos públicos y abiertos open data puede generar oportunidades para innovar y crear nuevos servicios para la ciudadanía y la empresa.

Dado lo anterior, se busca que los estudiantes, en esta Lección, practiquen la habilidad destreza del pensamiento capturando datos abiertos para luego tratarlos y analizarlos.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (30 min)

Transferencia del conocimiento (15 min)

Objetivos

Los estudiantes serán capaces de:

- Conocer qué es el open data y sus características.
- Identificar las posibilidades que ofrece el open data para la gestión de las ciudades y la participación ciudadana.

Preparación

- (Opcional) vea el video [La datificación del mundo](#)
- Tener abierta y preparada Infografía interactiva – [El empoderamiento ciudadano](#)
- Imprima una Píldora informativa – [Datos a tu alcance](#) y Guía de trabajo – Open data para cada estudiante.
- Asegúrese de que cada estudiante tenga su [Bitácora](#).

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- Video – [La datificación del mundo](#)
- Infografía interactiva – [El empoderamiento ciudadano](#)
- Píldora informativa – [Datos a tu alcance](#)
- Guía de trabajo – Open data

Vocabulario

- **Open data:** datos abiertos públicamente.

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Visualización del video inicial

Toda la clase ve el video [La datificación del mundo](#).

Ampliación del conocimiento (30 min)

Destreza del pensamiento: veo, pienso, me pregunto

Esta destreza de pensamiento se desarrollará individualmente usando la guía de trabajo Open data, que pauta una observación meticulosa de las informaciones que aporta el video visto anteriormente y la construcción de interpretaciones propias a partir de tres preguntas base:

¿Qué has visto en el video?

¿Qué piensas de lo que has visto?

¿Qué otras preguntas te surgen sobre el open data?

Para completar la destreza del pensamiento, los estudiantes pueden visionar de nuevo el video y consultar la infografía interactiva [El empoderamiento ciudadano](#) y la píldora informativa [Datos a tu alcance](#).

Transferencia del conocimiento (15 min)

Puesta en común

Una vez completada la guía de trabajo, toda la clase comparte las reflexiones anotadas durante la destreza de pensamiento.

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?
- ¿Serías capaz de definir qué es el open data?

Sugerencias para evaluar

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Analizan y Evalúan las formas en que la informática afecta las prácticas personales, éticas, sociales, económicas y culturales.
- Reflexionan sobre sus puntos de vista y temas de interés relacionados con los datos.

Lección 42

Periodismo de datos y data scraping

Lección en línea

Propósito

En la presente Lección, se espera que los estudiantes pongan en práctica la metodología del periodismo de datos capturando datos abiertos de la Wikipedia que, una vez tratados, analizados y sectorizados, permitan al estudiante redactar una noticia a partir de su interpretación.

Secuencia para el aprendizaje

Conocimiento inicial (15 min)

Ampliación del conocimiento (60 min)

Transferencia del conocimiento (25 min)

Objetivos

Los estudiantes serán capaces de:

- Experimentar el proceso de análisis e interpretación de datos.
- Conocer y experimentar herramientas digitales de creación de gráficos on-line.
- Experimentar con herramientas digitales de raspado de datos.

Preparación

- Imprima una Infografía [Datos para un nuevo periodismo](#), Píldora informativa [Datos que son noticia](#), Guía de trabajo Wikipedia y el data scraping para cada estudiante.
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- Infografía – [Datos para un nuevo periodismo](#)
- Píldora informativa – [Datos que son noticia](#)
- Guía de trabajo – Wikipedia y el data scraping

Vocabulario

- **Data scraping:** Extracción o captura de datos.
- **Wikipedia:** Enciclopedia libre y políglota disponible en la red.

Estrategia de aprendizaje

Conocimiento inicial (15 min)

Presentación de la Lección e infografía

Presentación de la Lección de creación de una noticia periodística basada en el raspado de datos o data scraping. Creación de grupos de trabajo y se presenta la guía de trabajo Wikipedia y el “data scraping”. Visualización en grupo de la infografía Datos para un nuevo periodismo.

Ampliación del conocimiento (60 min)

Generación de una noticia a partir del data scraping

Trabajo grupal de creación y redacción de una noticia basada en la recopilación y análisis de datos abiertos (open data). Para hacerlo, los estudiantes deben seguir los siguientes pasos:

- Selección de un tema y de los datos que usaremos para hacer una breve noticia ficticia.
- Uso de las hojas de cálculo de Google para hacer data scraping de los datos escogidos (se debe tener una cuenta de Gmail).
- Uso de la herramienta on-line <https://infogr.am/> para crear un gráfico de los datos recogidos.
- Redactar la noticia final que incluya la gráfica obtenida, un titular y varias hipótesis NO VERIFICADAS que expliquen los resultados de los datos mostrados.

Transferencia del conocimiento (25 min)

Presentación de las noticias

Presentación de las noticias generadas al resto de la clase.

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?
- ¿Sabrías utilizar el data scraping para capturar datos desde internet?

Wikipedia y el data scraping

Wikipedia es la gran enciclopedia de internet y consultarla se ha convertido en un acto natural cuando se quiere saber algo, sin importar el tema ni el usuario. Desde estudiantes hasta periodistas, todos consultamos Wikipedia. Es la fuente de información digital más popular.

A pesar de que algunos ponen en duda la fiabilidad de su información por el hecho de que son los mismos usuarios quienes crean los contenidos, es una de las fuentes de información más visitadas de internet, donde podemos encontrar desde la evolución de la población mundial o los resultados de las últimas elecciones, hasta los goles que se hicieron en la liga, entre otros ejemplos. Pero es importante tener en cuenta la verificación de la información.

<https://es.wikipedia.org/wiki/Wikipedia:Verificabilidad>

Les proponemos experimentar con el periodismo de datos utilizando técnicas para capturar datos de páginas web (en este caso Wikipedia) y la herramienta infogr.am, para crear gráficos de forma rápida y sencilla, a fin de redactar una noticia corta a partir de ellos.

Cosas que deben saber

1. El data scraping (raspado de datos) con hojas de cálculo de Google

es una técnica que sirve para obtener los datos de una tabla de una página web para poder trabajar con ellos: cruzarlos, analizarlos, visualizarlos... las hojas de cálculo de Google Drive permiten aplicar esta técnica de un modo muy sencillo:

<https://support.google.com/docs/answer/3093339?hl=es&ref>

2. Infogr.am

Es una herramienta on-line para crear gráficos e infografías:

<https://infogram.com/>

Nuestra noticia

Paso 1 – Busquen información sobre un tema en Wikipedia

Deberán escoger un tema para redactar su noticia que esté basado en datos que se puedan encontrar en Wikipedia. Por ejemplo, si quieren analizar la evolución del público de la serie Lost (Perdidos), podrían acceder a:

<https://es.wikipedia.org/wiki/Lost>

Allí podrán ver las audiencias de la serie y cómo fueron evolucionando de una temporada a otra.

Temporada	Eva de emisión (R1)	Episodios	Estadísticas			Ranking	Disponibilidad en streaming
			Revisión	Revisión por episodio (R1)	Revisión por episodio (R2)		
Temporada 1	Septiembre, 2005	24	100.000.000	4.166.666	100.000.000	100	Netflix
Temporada 2	Septiembre, 2005	24	100.000.000	4.166.666	100.000.000	100	Netflix
Temporada 3	Septiembre, 2005	24	100.000.000	4.166.666	100.000.000	100	Netflix
Temporada 4	Septiembre, 2005	12	100.000.000	8.333.333	100.000.000	100	Netflix
Temporada 5	Septiembre, 2005	12	100.000.000	8.333.333	100.000.000	100	Netflix
Temporada 6	Septiembre, 2005	12	100.000.000	8.333.333	100.000.000	100	Netflix

Otros ejemplos de temas para analizar podrían ser: la evolución de medallas olímpicas de Chile en los últimos 50 años, cuáles han sido las películas más rentables de Pixar, los países con más Grammys latinos de la última década...

Paso 2 – Extraigan los datos en una hoja de cálculo de Google

Para capturar los datos en los que basaremos nuestra noticia (audiencias de la serie por años, medallas según la Olimpiada, recaudación por película, Grammys latinos por países...) vamos a usar el data scraping, una técnica que sirve para obtener los datos de una página web y poder trabajar con ellos: cruzarlos, analizarlos, visualizarlos... Para hacerlo usaremos una hoja de cálculo de Google, por lo que debemos iniciar sesión en una cuenta de Gmail o crear una nueva para todo el grupo.

Para capturar los datos de una tabla deben escribir en el apartado Escribir fórmula de una hoja de cálculo de Google la siguiente sintaxis:

=IMPORHTML (“página web” ; “table” ; número tabla)

Donde dice “página web” deben escribir el enlace completo y donde dice “número de tabla”, deben poner el número identificativo de la tabla que quieren capturar (1, 2, 3...), ya que en una misma página de Wikipedia puede haber diversas tablas. Si quisiéramos capturar las audiencias de la serie Lost por temporadas, la sintaxis completa sería:

=IMPORHTML (“https://es.wikipedia.org/wiki/Lost”; “table” ;3)

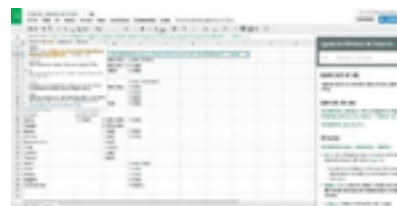
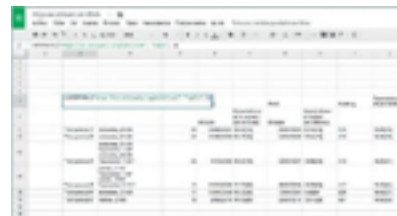
Si tienen dudas o dificultades, consulten el sistema de ayuda de las hojas de cálculo de Google o consulten el siguiente documento:

<https://support.google.com/docs/answer/3093339?hl=es&reftopic=3105411>

Una vez tengan los datos en su hoja de cálculo, limpien la tabla y dejen solamente los datos relevantes, eliminando aquellas columnas que necesiten. Para hacerlo, peguen los datos en una nueva tabla evitando repetir la sintaxis de captura de datos (ya que, si no lo hacen así, la tabla volverá a su estado original, aunque la modifiquen).

Paso 3 – Creen un gráfico en Infogr.am

Para crear un gráfico de los datos que han capturado, hagan clic en este enlace para acceder a la herramienta on-line <https://infogr.am>.



Una vez registrados, escojan el gráfico que mejor se adapte al contenido de lo que quieren mostrar. Peguen los datos de su hoja de cálculo en la tabla de Infogr.am y personalicen el gráfico resultante.

Paso 4 – Redactar un titular que resuman el gráfico

Escriban un titular para el gráfico que han creado insertando un bloque de texto en Infogr.am. El titular ha de ser breve, directo y conciso. En nuestro ejemplo de las audiencias de la serie Lost podríamos escribir:

La audiencia bajó en las últimas temporadas de “Lost”

Finalmente, pueden descargar el gráfico generado en JPG.

Paso 5 – Planteen hipótesis para explicar los hechos

A continuación, planteen diversas hipótesis que puedan explicar los hechos del gráfico que han generado. En nuestro ejemplo, podríamos argumentar que las audiencias de la serie Lost cayeron a medida que avanzaban las temporadas porque:

- Empezaron otras series de éxito
- La trama perdió interés
- Los actores populares dejaron la serie
- ...

Paso 6 – Redacten una breve noticia

Redacten una breve noticia que recoja el gráfico, el titular y una de las hipótesis planteadas en el paso anterior para explicar los hechos en un formato que les permita presentarla al resto de grupos de trabajo.

Recuerden que el periodismo real, todas las hipótesis que hicieron deberían citar las páginas y autores en los que se basan para hacer sus afirmaciones y poder comprobar así su veracidad.

Lección 43

La cuarta revolución industrial

Propósito

Si les dijeran que estamos en la cuarta revolución industrial, ¿qué pensarían? Pues sí, actualmente nos encontramos en dicha revolución industrial y quizás sin saberlo. Esta revolución industrial resulta tan vital puesto que está cambiando a pasos agigantados todo el mercado laboral y lo relacionado a esto.

Por esta razón, en esta Lección, los estudiantes conocerán las implicancias que tiene y tendrá la cuarta revolución industrial en el mundo laboral.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (30 min)

Transferencia del conocimiento (15 min)

Objetivos

Los estudiantes serán capaces de:

- Conocer qué es la cuarta revolución industrial.
- Identificar profesiones futuras que se desarrollarán gracias al Big data.

Preparación

- (Opcional) vea el video [La cuarta revolución industrial](#).
- Imprima Infografía [Las nuevas profesiones de la era digital](#) y Guía de trabajo Profesiones del mañana para cada estudiante.
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- Video – [La cuarta revolución industrial](#)
- Infografía – [Las nuevas profesiones de la era digital](#)
- Guía de trabajo – Profesiones del mañana

Vocabulario

- **Revolución industrial:** proceso de transformación tecnológica, social y económica.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Visualización del video

Visualización en grupo del video La cuarta revolución industrial.

Ampliación del conocimiento (30 min)

Destreza de pensamiento: conectar, desafiar, conceptos y cambios

Destreza de pensamiento: Conectar, desafiar, conceptos y cambios, donde a través de preguntas clave los estudiantes identifican las ideas clave de un video o un texto, las contrastan con los conocimientos que ya tienen y establecen conexiones con su realidad.

A partir de las informaciones observadas en el video y de la consulta de la infografía Las nuevas profesiones de la era digital, los estudiantes, de forma individual, deben completar la guía de trabajo Profesiones del mañana.

Gracias a ella puede identificar profesiones de su entorno que podrían desaparecer tras la cuarta revolución industrial y profesiones que podrían surgir gracias a estos cambios.

La guía de trabajo de la destreza se completa con una reflexión sobre las consecuencias e impacto de estos cambios y cómo afecta a su elección de profesión o estudios futuros.

Transferencia del conocimiento (15 min)

Puesta en común y reflexión final

Puesta en común de toda la clase donde cada estudiante expone los argumentos e ideas que ha elaborado a partir de las informaciones consultadas en la destreza de pensamiento.

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?
- ¿Sabrías identificar que profesiones desaparecerán tal como las conocemos ahora tras la cuarta revolución industrial y por qué piensas que será así?

Sugerencias para evaluación

Se sugiere los siguientes indicadores para evaluar formativamente los aprendizajes:

- Producen argumentos sobre cómo las tecnologías informáticas impactan las actividades cotidianas e influyen opciones profesionales de las personas.
- Analizan y evalúan el impacto de las tecnologías digitales en contextos sociales, económicos y culturales.

Contenido 3

Qué es el Machine Learning o Aprendizaje Automatizado

Resumen

- **Lección 44:** ¿Qué sabes sobre Machine Learning?
- **Lección 45:** Entrenar modelo de aprendizaje automático
- **Lección 46:** Datos y sesgos del muestreo
- **Lección 47:** Programación con aprendizaje automático
- **Lección 48:** Impacto social de utilizar tecnologías de IA en aplicaciones de la vida diaria

Objetivos

- **OA 3.** Desarrollar y programar algoritmos para ejecutar procedimientos matemáticos, realizar cálculos y obtener términos definidos por una regla o patrón.
- **OA i.** Buscar, seleccionar, manejar y producir información matemática/cuantitativa confiable a través de la web.

Referencias

- <https://www.ibm.org/activities/machine-learning-for-kids> (disponible en español) desarrollado por <https://www.iste.org/>

Plan de las actividades

Programación con aprendizaje automatizado

Propósito

Lo sepan o no los alumnos, el aprendizaje automático está integrado en muchas de las aplicaciones que utilizan todos los días. Al potenciar herramientas desde Netflix hasta la autocorrección, la tecnología de IA es utilizada para procesar datos rápidamente, personalizar la experiencia del usuario y hacer más fáciles ciertas tareas.

En este proyecto, los alumnos aprenderán qué es el aprendizaje automático y cómo funciona. Los alumnos aplicarán este conocimiento para desarrollar un programa que utilice un modelo de aprendizaje automático que hayan entrenado. En el proceso, verán lo útil que puede ser el aprendizaje automático para desarrollar las soluciones de programas de cómputo más eficaces en la actualidad.

Duración

6-8 horas.

Objetivos

Los estudiantes serán capaces de:

- Entrenar un modelo de aprendizaje automático.
- Entender los recursos y las implicaciones de un muestreo sesgado en los conjuntos de datos.
- Usar un modelo de aprendizaje automático en el desarrollo de un programa de cómputo

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- Computadora(s) o tablets con conexión a internet para acceder a herramientas y recursos en línea.
 - Sitio web: Aprendizaje automático para niños.

Para los profesores:

- Aplicación: **“Explicación del aprendizaje automático con árboles de decisión”**.

Vocabulario

- inteligencia artificial
- sesgo
- modelo de clasificación
- grado de confianza
- datos
- conjunto de datos
- características del árbol de decisión
- etiquetas
- aprendizaje automático
- modelo
- comprensión del lenguaje natural

Preparación de la clase

- Prepare a su grupo para el sitio [web Aprendizaje automático para niños \(ML4K\)](#). Para las instrucciones completas para crear su cuenta gratuita en IBM Cloud y la cuenta de maestro de ML4K. Es recomendable que los alumnos inicien sesión en sus cuentas el día anterior al comienzo de la Lección 45, para confirmar que pueden entrar a la plataforma sin problemas.
- Prepare al grupo completo para el proyecto ML4K de la Lección 47 “Hazme feliz”. Le sugerimos que utilice la función “Proyecto de todo el grupo” del ML4K para este proyecto de modo que todo el grupo pueda aportar al modelo y colaborar con los datos. Cuando cree un proyecto, seleccione la opción “¿Proyecto de todo el grupo?” o bien, después de crear el proyecto, haga clic en el botón “Compartir” para compartirlo con el grupo. Un video tutorial está disponible aquí: [link](#).
- Evalúe los recursos de apoyo para los proyectos de ML4K “Sobrevivientes del Titanic” y “Hazme feliz”.
 - **“Sobrevivientes del Titanic”** documento de orientación para el maestro
 - **“Sobrevivientes del Titanic”** hoja de trabajo para los alumnos
 - **“Hazme feliz”** documento de orientación para el maestro
 - **“Hazme feliz”** hoja de trabajo para los alumnos

Seleccione un proyecto de ML4K para la demostración culminante de sus alumnos.

- sesgo de muestreo
- aprendizaje supervisado
- datos de prueba
- datos de entrenamiento

Lección 44

¿Qué sabes sobre machine learning?

Propósito

El uso del aprendizaje automático vendrá dado por el tipo de dato que el usuario quiera procesar. Pero ¿qué es y cómo se aplica el aprendizaje automático en la vida cotidiana? Frente a estas preguntas, se espera que, en esta Lección, los estudiantes reconozcan el uso del aprendizaje automático con diferentes tipos de datos.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (30 min)

Transferencia del conocimiento (15 min)

Objetivos

Los estudiantes serán capaces de:

- Conocer qué es el aprendizaje automático (machine learning) y cómo funciona.

Preparación

- Sitio web Aprendizaje automático para niños listo para visualizarlo.
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los Profesores:

- Computadora(s) o tablets con conexión a internet para acceder a herramientas y recursos en línea.
 - Sitio web: Aprendizaje automático para niños

Para los estudiantes:

- Computadora(s) o tablets con conexión a internet para acceder a herramientas y recursos en línea.
 - Sitio web: Aprendizaje automático para niños

Vocabulario

- **Aprendizaje automático:** método que permite que las computadoras aprendan.
- **Inteligencia artificial (IA):** simulación de inteligencia humana, expresada por máquinas.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

En esta Lección, los alumnos activarán sus conocimientos previos sobre el aprendizaje automático al examinar funciones de aplicaciones con las que están familiarizados, aunque quizás no hayan considerado cómo funcionan las herramientas. Los ejemplos proporcionados exploran situaciones de la vida real en que se usa el aprendizaje automático para procesar diversos tipos de datos: textos, cifras, imágenes y sonido. Si es necesario, los alumnos pueden realizar una investigación rápida para aprender más de la tecnología de IA que los asiste.

1. Proporcione a los alumnos una introducción de alto nivel al aprendizaje automático. Haga hincapié en que los modelos de aprendizaje automático pueden funcionar con muchos tipos de datos. Se pueden encontrar recursos de apoyo en el Apéndice A: Fundamentos de la inteligencia artificial.
2. Para cada ejemplo de la vida real que figura a continuación, muestre y genere un debate entre los alumnos de las siguientes preguntas en equipos pequeños; luego deben compartir sus experiencias con todo el grupo.
 - ¿Cuál es la finalidad de la tecnología de aprendizaje automático?
 - ¿Qué tipo de datos necesitaría analizar la tecnología para aprender a realizar esa tarea?
 - ¿Qué tipo de datos necesitaría recabar la tecnología para realizar la tarea para el usuario?
 - ¿Cómo creen que funciona la tecnología del aprendizaje automático?

Ampliación del conocimiento (30 min)

A continuación, se presentan ejemplos de aprendizaje automático en la vida real para el debate en grupos pequeños. También se incluyen preguntas opcionales de extensión para el debate de todo el grupo.

Datos de texto. Texto predictivo en correos electrónicos, en dispositivos inteligentes o en solicitudes de búsqueda. (Ampliar el razonamiento: ¿Cómo sabe el texto predictivo qué es lo que sigue? ¿Cómo aprende tu estilo de escritura? ¿Cómo sabe deletrear tu apellido?).

Datos numéricos. Aplicaciones de mapas que determinan la ruta más corta en función de la distancia o el tiempo. (Ampliar el razonamiento: ¿Cómo utilizan las aplicaciones de mapas los valores numéricos para calcular tiempos de viaje? ¿Cómo determinan agregar más tiempo cuando hay un tráfico intenso?).

Datos de imágenes. Los programas de cómputo de reconocimiento que se utiliza en las redes sociales para etiquetar o desbloquear teléfonos inteligentes.

(Ampliar el razonamiento: ¿Qué tipo de características faciales utiliza el programa de cómputo para reconocer a una persona específica?).

Datos de sonido. Una aplicación de reconocimiento de música, como Shazam, que informa el

título y el artista de la canción que se está reproduciendo. (Ampliar el razonamiento: ¿Qué tipos de características está identificando la aplicación en la canción? Si varios artistas han grabado la misma canción, ¿cómo sabe la IA cuál es la diferencia?).

Transferencia del conocimiento (15 min)

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Piensan en formas de mejorar la accesibilidad y la usabilidad de los productos tecnológicos para las diversas necesidades y deseos de los usuarios.

Lección 45

Entrenar modelo de aprendizaje automatizado

Propósito

Una vez ya conocido el modelo de aprendizaje automático, se espera que los estudiantes, en esta Lección, pongan a prueba el modelo de aprendizaje automático con diferentes datos.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (30 min)

Transferencia del conocimiento (15 min)

Objetivos

Los estudiantes serán capaces de:

- Entrenar un modelo de aprendizaje automático.

Preparación

- Imprima los documentos Sobrevivientes del Titanic y Hazme feliz para cada estudiante.
- Sitio web Aprendizaje automático para niños listo para visualizarlo.
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los Profesores:

- Documento Sobrevivientes del Titanic.
- Documento Hazme feliz.
- Aplicación- “Explicación del aprendizaje automático con árboles de decisión”.
- Computadora(s) o tablets con conexión a internet para acceder a herramientas y recursos en línea.
 - Sitio web: Aprendizaje automático para niños.

Para los estudiantes:

- Guía de trabajo Sobrevivientes del Titanic.
- Guía de trabajo Hazme feliz
- Computadora(s) o tablets con conexión a internet para acceder a herramientas y recursos en línea.
 - Sitio web: Aprendizaje automático para niños.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Aprendizaje automático para niños (ML4K) es una herramienta gratuita, basada en las API de IBM Watson, que permite a los alumnos entrenar y usar modelos de aprendizaje automático. En esta Lección, los alumnos trabajarán con todo el grupo para crear y debatir dos modelos, uno que use un conjunto de datos preexistente y otro que use un **conjunto de datos** que crearán los alumnos. Los alumnos no necesitan desarrollar un proyecto de programación para esta Lección, en lugar de eso se centrarán en aprender cómo alimentar datos a un modelo de aprendizaje automático, entrenar al modelo y probarlo.

NOTA: La siguiente demostración pide a los alumnos considerar las razones por las que hubo personas que pudieron o no pudieron sobrevivir el hundimiento del Titanic. Si considera que este tema podría ser molesto para los alumnos de su grupo, puede omitir este ejemplo e impartir los conceptos de aprendizaje supervisado, etiquetas, características, modelos, árboles de decisión y nivel de confianza, solo con el segundo modelo, “Hazme feliz”.

Ampliación del conocimiento (30 min)

Trabaje en el proyecto “Sobrevivientes del Titanic” con todo el grupo. Este proyecto utiliza un conjunto de datos ya provisto para desarrollar un modelo de aprendizaje automático predictivo basado en un árbol de decisión.

- Distribuya o muestre los primeros 13 pasos de la hoja de trabajo de los alumnos para el proyecto “**Sobrevivientes del Titanic**”.
- Describa los primeros nueve pasos trabajando con todo el grupo. Con estos pasos prepare una plantilla para el proyecto ML4K “Sobrevivientes del Titanic” y muestre el conjunto de datos del entrenamiento. Una vez que las dos etiquetas (survived y did_not_survive) y los datos estén desplegados, use el material visual para enseñar los siguientes conceptos:
 - Esta Lección utiliza el aprendizaje automático supervisado para entrenar un modelo de aprendizaje automático para clasificar datos. **El aprendizaje supervisado** es una forma de aprendizaje automático en la que el entrenador le proporciona a la IA etiquetas para cada elemento de los datos de entrenamiento. La IA a su vez analiza los datos de

Vocabulario

- **Conjunto de datos:** Colección de datos normalmente usado para propósitos estadísticos.
- **Árbol de decisión:** herramienta gráfica que muestra posibles resultados según una serie de decisiones y su probabilidad de ocurrencia.

cada grupo de etiquetas para identificar patrones en las características (es decir, atributos definitorios) y crea un modelo.

- En este caso, el modelo será representado por un **árbol de decisión** que busca cada una de las características definitorias que identifique en los datos. Por ejemplo, un modelo para clasificar datos de animales con la etiqueta “elefante” puede identificar características como cuatro patas, grande, orejas grandes y una trompa.

- Pida a los alumnos que analicen los datos de entrenamiento provistos e identifiquen patrones. Luego plantee un debate en el grupo utilizando las preguntas provistas en el paso 10 de la hoja de trabajo de los alumnos.

- Siga los pasos 11 y 12 para entrenar y probar el modelo en la página “Learn & Test”. Los resultados pronostican si una persona con determinado conjunto de características habría sobrevivido o no al hundimiento del Titanic además del nivel de confianza de la IA en ese pronóstico. Explique que el **nivel de confianza** son las probabilidades de que el elemento concuerde con la etiqueta correcta.

- Haga clic en el botón “¡Describe your model!” para ver el árbol de decisión del modelo de aprendizaje automático. Pregunte a los alumnos: Basándose en la información del árbol de decisión, ¿cuáles son las características del modelo que están asociadas con la etiqueta “survived”? ¿Qué características están asociadas a la etiqueta “did_not_survive”?

Trabaje el proyecto **“Hazme feliz”** con todo el grupo. El proyecto “Hazme feliz” recopila datos creados por el usuario para crear un modelo clasificador que analiza si un enunciado es un cumplido o un insulto.

- Antes de la Lección, asegúrese de preparar y compartir una versión de “Proyecto de todo el grupo” para “Hazme feliz” de modo que los alumnos colaboren con datos de cada una de sus cuentas individuales.

- Recorra los pasos del 7 al 10 de la hoja de trabajo “Hazme feliz” de los alumnos junto con todo el grupo. Estos pasos abren el proyecto y agregan las etiquetas “kind_things” y “mean_things” al proyecto del grupo. Explique que esta vez usted creará un modelo de clasificación que usará la comprensión del lenguaje natural para identificar características para clasificar enunciados como cumplidos o insultos.

- Pida a los alumnos que entren a sus cuentas, abran el proyecto del grupo “Hazme feliz” de su lista de proyectos, y hagan clic en Train para acceder a los cestos de etiquetas. Haga que cada alumno agregue un enunciado con un cumplido muy agradable al cesto “kind_things” y un enunciado con un insulto apropiado para la escuela al cesto “mean_things”. Estos serán los datos de entrenamiento utilizados para enseñar el modelo de IA. Haga que cada alumno agregue otro cumplido e insulto al documento colaborativo (digital, papel o pizarra), que serán utilizados como datos de prueba. Una vez recopilados todos los datos, entrene al modelo, luego pruébelo con la lista de datos de prueba para ver cómo actúa. Comente a los alumnos que se discutirán algunas razones por las que el modelo podría funcionar bien o mal en la tarea de la siguiente Lección.

Transferencia del conocimiento (15 min)

Concluya esta Lección haciendo que los alumnos expliquen cómo funciona el modelo “Hazme feliz” usando los términos de aprendizaje supervisado, etiquetas, características, modelos, árboles de decisión, nivel de confianza, modelo de clasificación, datos de entrenamiento y datos de prueba.

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Desarrollan y programan algoritmos para ejecutar y obtener términos definidos por una regla o patrón.

Lección 46

Datos y sesgos del muestreo

Propósito

En esta Lección, los estudiantes conocerán distintos factores que pueden influenciar el resultado de un modelo de aprendizaje automático, y cómo esta práctica puede ser corregida oportunamente cuando se conocen los errores.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (30 min)

Transferencia del conocimiento (15 min)

Objetivos

Los estudiantes serán capaces de:

- Entender los recursos y las implicaciones de un muestreo sesgado en los conjuntos de datos.

Preparación

- Sitio web Aprendizaje automático para niños listo para visualizarlo.
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los Profesores:

- Computadora(s) o tablets con conexión a internet para acceder a herramientas y recursos en línea.
 - Sitio web: Aprendizaje automático para niños

Para los estudiantes:

- Computadora(s) o tablets con conexión a internet para acceder a herramientas y recursos en línea.
 - Sitio web: Aprendizaje automático para niños

Vocabulario

- **Sesgo de muestreo:** distorsión que sufre un análisis estadístico.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

En esta Lección, los alumnos considerarán las formas en que el muestreo de datos para el entrenamiento y la evaluación de datos puede afectar el resultado de un modelo de aprendizaje automático. Luego, volverán a los conjuntos de datos para su modelo "Hazme feliz" y realizarán mejoras para refinar los conjuntos de datos.

Ampliación del conocimiento (30 min)

Pregunte a los alumnos: ¿Les gustó la forma en que funcionó el modelo "Hazme feliz"? ¿Fue siempre correcta? ¿Podría haber funcionado de mejor manera?

Comente a los estudiantes que una IA no tiene opiniones ni pensamientos propios, sino que solo puede tomar decisiones con base en los datos de los que aprende. Debido a que los modelos de aprendizaje automático aprenden de los datos de entrenamiento, la calidad de la muestra de los datos de entrenamiento determina de forma directa la calidad del modelo. El sesgo en el muestreo, que daría lugar a un modelo inexacto, es causado por tener un conjunto de datos que no representa con precisión las etiquetas. Un conjunto de datos de calidad tiene las siguientes características:

- Suficientes datos. La IA necesita suficientes ejemplos para poder identificar patrones en las características de los datos. La cantidad de datos necesaria depende del objetivo de desempeño específico para la precisión en la tarea. A mayor necesidad de precisión, será mayor cantidad de datos de entrenamiento necesarios.
- Datos correctos: La IA necesita recibir suficientes tipos correctos de ejemplos para comprender todas las características correctas de los elementos a los que se les debe dar una etiqueta particular con precisión. Esto significa que, si faltan ejemplos de esa etiqueta o son confusos, la IA probablemente no pueda identificarlos correctamente en la evaluación de los datos. Algunos ejemplos de sesgo de muestreo incluyen lo siguiente:
 - Si uno entrena a la IA para identificar insectos, pero solo es entrenada con ejemplos de hormigas y escarabajos, es probable que no pueda identificar una mantis religiosa como insecto porque algunas características de los insectos faltarían en los datos. Del mismo modo, si el modelo es utilizado por muchas personas en muchos países, pero solo los insectos de un país están representados, es posible que la IA no pueda reconocer insectos de otros lugares.
 - Si se entrenó a la IA usando un conjunto de datos en el que todas las imágenes de insectos se tomaron entre la hierba, pero las imágenes de los que no eran insectos se tomaron en una variedad de lugares, la IA podría identificar la hierba como una característica de la etiqueta del insecto. Si el modelo se probó con una imagen de un perro en el césped, podría etiquetarlo como un insecto.
 - Si hay una característica que puede aplicarse a ambas etiquetas, pero solo se muestra en una, esto podría confundir a la IA. Por ejemplo, si todas las imágenes de entrenamiento de no-insectos tampoco fue de no-animales, la IA podría confundir a cualquier animal con patas, cabeza, etc. con un insecto cuando se pruebe el modelo.
 - Si los datos de entrenamiento incluyen significativamente más ejemplos de una etiqueta

que de otra, entonces la IA podría aprender que la primera etiqueta es más común y, por lo tanto, seleccionará incorrectamente esa etiqueta con más frecuencia.

Transferencia del conocimiento (15 min)

Invite a los alumnos a reflexionar sobre la experiencia colaborativa de crear el modelo “Hazme feliz”. Pídales que identifiquen los elementos que funcionaron bien de los conjuntos de datos de entrenamiento y prueba que fueron creados. Luego, pregúnteles qué se podría hacer para mejorar los resultados del modelo. Registre sus respuestas y exhíbalas al grupo. Haga que los alumnos regresen a su proyecto “Hazme feliz”, realicen las mejoras del conjunto de datos que identificaron y prueben el modelo para ver si los resultados mejoran. Haga que los alumnos continúen refinando el conjunto de datos hasta que el modelo sea capaz de identificar correcta y constantemente oraciones con un alto grado de confianza.

Opcional: Amplíe esta Lección de aprendizaje examinando los cuatro ejemplos de herramientas de aprendizaje automático en la Lección 44 y analizando las posibles fuentes e impactos del sesgo de muestreo. Quizás sea recomendable que los alumnos vayan más allá e investiguen acontecimientos de la actualidad que describan incidentes en los que el sesgo de muestreo condujo a resultados negativos en el uso de modelos de aprendizaje automático en aplicaciones. Si bien esta extensión del proyecto revelará una desventaja de usar el aprendizaje automático en las aplicaciones, los alumnos deben centrarse en la importancia de evitar el sesgo de muestreo y el papel que juegan las personas en el resultado del modelo de aprendizaje automático.

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Debaten sobre la importancia de evitar el sesgo de muestreo y el papel que juegan las personas en el resultado del modelo de aprendizaje automático.

Lección 47

Programación con aprendizaje automatizado

Propósito

A medida que los estudiantes vayan adquiriendo mayor conocimiento teórico sobre el aprendizaje automático, también se espera que este conocimiento vaya acompañado de uno práctico. Por esta razón, en esta Lección, los estudiantes practicarán la programación bajo el modelo de aprendizaje automático.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (30 min)

Transferencia del conocimiento (15 min)

Objetivos

Los estudiantes serán capaces de:

- Usar un modelo de aprendizaje automático en el desarrollo de un programa.

Preparación

- Sitio web Aprendizaje automático para niños listo para visualizarlo.
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los Profesores:

- Computadora(s) o tablets con conexión a internet para acceder a herramientas y recursos en línea.
 - Sitio web: Aprendizaje automático para niños

Para los estudiantes:

- Computadora(s) o tablets con conexión a internet para acceder a herramientas y recursos en línea.
 - Sitio web: Aprendizaje automático para niños

Vocabulario

- **Programación:** proceso de diseño y codificación para la creación de programas.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Para sintetizar su aprendizaje de este proyecto con otros aprendizajes de su curso de informática, los alumnos practicarán la programación en pares para completar un proyecto ML4K en el que entrenarán e integrarán un modelo de aprendizaje automático en Scratch, App Inventor o Python. Para que las correcciones sean más sencillas, le sugerimos que todos sus estudiantes utilicen la misma plataforma.

Ampliación del conocimiento (30 min)

ML4K proporciona una variedad de hojas de trabajo de proyectos con revisiones sobre la capacitación y programación, así como los conjuntos de datos correspondientes. Seleccione y asigne una o más opciones de proyectos de aprendizaje supervisados de ML4K para que los alumnos la lleven a cabo. Las recomendaciones para proyectos para principiantes de ML4K basadas en algunos temas comunes de cursos de informática incluyen: Ciberseguridad—Face Lock; ciencia de datos—Journey to School; internet de las cosas—Smart Classroom y Desarrollo de juegos—Snap!

Cuando los alumnos hayan completado la programación de esta Lección, pídeles que respondan las siguientes preguntas con su compañero. Luego debata cada una de las preguntas con todo el grupo.

- Al probar sus modelos, ¿qué tan bien creen que funcionó la aplicación? ¿Por qué creen que funcionó bien o no tan bien?
- ¿Qué sesgos de muestreo identificaron en sus conjuntos de datos? ¿Qué hicieron para mejorar su conjunto de datos y el funcionamiento del modelo?
- ¿Creen que usar el modelo de aprendizaje automático hizo a su programa más útil o efectivo que los programas que no usan aprendizaje automático? ¿Por qué?
- ¿Cuál sería otro tipo de programa o solución que podrían crear usando el mismo modelo que ya entrenaron? ¿Cómo podrían expandir su modelo para trabajar con varios tipos de programas o soluciones?

Transferencia del conocimiento (15 min)

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Debaten sobre la importancia de evitar el sesgo de muestreo y el papel que juegan las personas en el resultado del modelo de aprendizaje automático.

Lección 48

Impacto social de utilizar tecnologías de IA en aplicaciones de la vida diaria

Propósito

Cómo parte del ciclo de conocimiento sobre el aprendizaje automático, se espera que esta Lección los estudiantes sean capaces de desarrollar un pensamiento crítico con respecto al impacto que las nuevas tendencias tecnológicas ejercen sobre las personas y su entorno.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (30 min)

Transferencia del conocimiento (15 min)

Objetivos

Los estudiantes serán capaces de:

- Reflexionar sobre su aprendizaje y considerar el impacto social de utilizar tecnologías de IA en aplicaciones de la vida diaria.

Preparación

- Sitio web Aprendizaje automático para niños listo para visualizarlo.
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los Profesores:

- Computadora(s) o tablets con conexión a internet para acceder a herramientas y recursos en línea.
 - Sitio web: Aprendizaje automático para niños

Para los estudiantes:

- Computadora(s) o tablets con conexión a internet para acceder a herramientas y recursos en línea.
 - Sitio web: Aprendizaje automático para niños

Vocabulario

- **Inteligencia artificial (IA):** simulación de inteligencia humana, expresada por máquinas.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Lluvia de ideas sobre los contenidos y conceptos aprendidos a lo largo de las cuatro primeras actividades.

Ampliación del conocimiento (30 min)

En esta Lección, los alumnos deben comentar las siguientes preguntas para reflexionar sobre su aprendizaje, y considerar el impacto social de utilizar tecnologías de IA en aplicaciones de la vida diaria.

- ¿Cuáles son algunas de las consecuencias de usar aprendizaje automático para hacer funcionar las aplicaciones?

Ahora que conocen más de cerca el aprendizaje automático y el sesgo de muestreo, ¿qué preguntas se harían antes de confiar en herramientas de aprendizaje automático, tales como texto predictivo, reconocimiento facial o un sistema de recomendación de productos.

Transferencia del conocimiento (15 min)

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Debaten, sobre el impacto en que la informática afecta las prácticas personales, éticas, sociales, económicas y culturales.

Lección 26

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

La representación importa

Recomendar una comida

Tú y tu grupo recomendarán comidas para varias personas según un conjunto de datos. Cada persona te dirá algo para ayudarte a decidir qué comida es la mejor. Tendrás que trabajar con su grupo para recomendar una comida y luego explicar su elección en función de los datos que le hayan proporcionado. Sus datos están en una página separada.

"Soy alérgico a los huevos"	"Mi doctor dijo que comiera menos sodio"
¿Qué comida recomienda tu equipo? Explica tu elección usando tus datos	¿Qué comida recomienda tu equipo? Explica tu elección usando tus datos
"Estoy tratando de ahorrar dinero"	"Quiero publicar una buena foto de ella en línea"
¿Qué comida recomienda tu equipo? Explica tu elección usando tus datos	¿Qué comida recomienda tu equipo? Explica tu elección usando tus datos

¿Qué recomendación fue la más fácil de hacer?













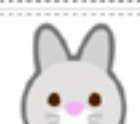



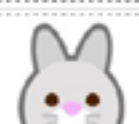








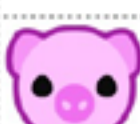










¿Qué pasa con los datos lo hicieron más fácil?

Lección 27

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Formas de animales

 Mono	 Mono	 Mono	 Mono	 Mono	 Mono
 Hipopótamo	 Hipopótamo	 Hipopótamo	 Hipopótamo	 Hipopótamo	 Hipopótamo
 Conejo	 Conejo	 Conejo	 Conejo	 Conejo	 Conejo
 Jirafa	 Jirafa	 Jirafa	 Jirafa	 Jirafa	 Jirafa
 Cerdo	 Cerdo	 Cerdo	 Cerdo	 Cerdo	 Cerdo
 Elefante	 Elefante	 Elefante	 Elefante	 Elefante	 Elefante

Representación de información

Reto

Necesita crear reglas que le permitan representar cualquier palabra que desee utilizando una sola pila de tarjetas. Otra persona debería ser capaz de usar las reglas que escribe en esta guía de actividades para leer las palabras representadas por su pila de tarjetas sin hablar con usted.

Cómo hacer una pila

Coloque las cartas en una sola pila ordenada, todas boca arriba y en la misma dirección. La primera carta debe estar en la parte superior.

Representando letras

A:	J:	S:
_____	_____	_____
B:	K:	T:
_____	_____	_____
C:	L:	U:
_____	_____	_____
D:	M:	V:
_____	_____	_____
E:	N:	W:
_____	_____	_____
F:	O:	X:
_____	_____	_____
G:	P:	Y:
_____	_____	_____
H:	Q:	Z:
_____	_____	_____
I:	R:	
_____	_____	

Escribe cómo se representará cada letra a continuación.

Tus reglas:

Haz tu mensaje

Elige una palabra y haz una pila de tarjetas que usen tus reglas para representar esa palabra. Déjalo junto a esta guía de actividades y luego cambia con otro grupo para ver si puede decodificar su mensaje.

Lección 28

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Texto ASCII

¿Por qué Binario?

Para poder ingresar, exportar, almacenar y procesar información, las computadoras necesitan representarla primero. Los ingenieros usaron las herramientas que tenían a mano, que eran cables que transportaban electricidad. Es fácil saber si un cable tiene o no electricidad que fluye a través de él, por lo que los ingenieros necesitan un sistema basado en solo dos estados de electricidad, o no electricidad, o en otras palabras, encendido / apagado.

ASCII

Hay muchos sistemas binarios posibles para representar las letras en información binaria, pero el más popular se llama ASCII. Puede ver la forma en que los patrones de on y off representan cada letra en la tabla a continuación.

Binario:

cualquier cosa que solo puede ser una de dos opciones.

ASCII:

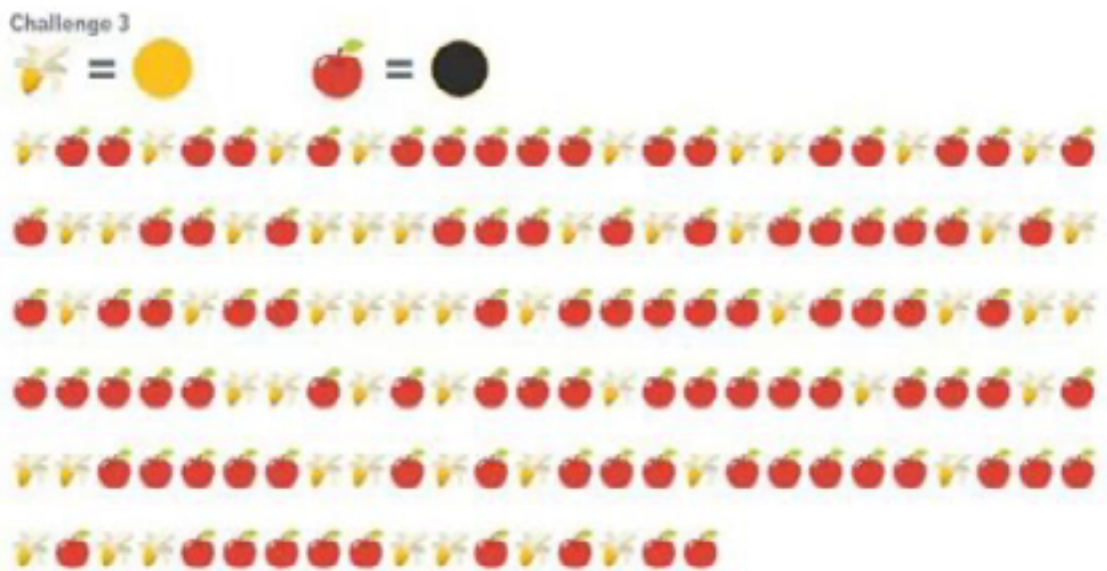
un sistema popular para representar texto en binario.

● = On ● = Off

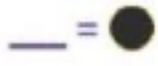
A	● ● ● ● ● ● ● ●	N	● ● ● ● ● ● ● ●
B	● ● ● ● ● ● ● ●	O	● ● ● ● ● ● ● ●
C	● ● ● ● ● ● ● ●	P	● ● ● ● ● ● ● ●
D	● ● ● ● ● ● ● ●	Q	● ● ● ● ● ● ● ●
E	● ● ● ● ● ● ● ●	R	● ● ● ● ● ● ● ●
F	● ● ● ● ● ● ● ●	S	● ● ● ● ● ● ● ●
G	● ● ● ● ● ● ● ●	T	● ● ● ● ● ● ● ●
H	● ● ● ● ● ● ● ●	U	● ● ● ● ● ● ● ●
I	● ● ● ● ● ● ● ●	V	● ● ● ● ● ● ● ●
J	● ● ● ● ● ● ● ●	W	● ● ● ● ● ● ● ●
K	● ● ● ● ● ● ● ●	X	● ● ● ● ● ● ● ●
L	● ● ● ● ● ● ● ●	Y	● ● ● ● ● ● ● ●
M	● ● ● ● ● ● ● ●	Z	● ● ● ● ● ● ● ●
		[space]	● ● ● ● ● ● ● ●

Reto

En cada uno de los desafíos a continuación, el sistema ASCII se ha utilizado para representar un mensaje secreto. Afortunadamente, ahora conoce el sistema ASCII, por lo que podrá decodificar los mensajes. Cada desafío representa las letras de una manera diferente.



Crea un rompecabezas



--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Lección 30





(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Representación de números

Tarjetas de números binarios

Trabaja con tu grupo para descubrir qué cartas deben estar boca arriba o boca abajo para obtener la cantidad correcta de puntos. Ponga una "U" debajo de una carta que debe estar boca arriba y una "D" debajo de una carta que debe estar boca abajo. El primero ha sido hecho por ti.


How many dots?					¿Cuál es el número más bajo que puedes hacer? ¿Cuál es el número más alto que puedes hacer?
13	U	U	D	U	
5					
10					
2					

Usando el patrón

Puedes usar el mismo patrón incluso sin las cartas. Sombrea las tarjetas que NO quieres usar para hacer tu número. El primero ha sido hecho por ti.

¡Puedes usar el mismo patrón incluso sin los puntos! Responda las preguntas sombreando en los cuadrados.

7					
12					
3					
6					

1. ¿Cuántas patas tiene un insecto?		<input type="text"/>
2. ¿En qué grado estás?		<input type="text"/>
3. ¿Cuántas hermanas y hermanos tienes?		<input type="text"/>
4. Haga una pregunta ...		<input type="text"/>

Usa el widget en línea para ayudarte a responder las siguientes preguntas

Números de 5 bits

Puedes usar el mismo patrón incluso sin las cartas. Sombrea las tarjetas que NO quieres usar para hacer tu número. El primero ha sido hecho por ti

En este sistema de 5 bits para números binarios, ¿qué número debe ir en el cuadro de la izquierda?

?	8	4	2	1
---	---	---	---	---

¿cómo harías?

16?	<input type="text"/>
27?	<input type="text"/>
31?	<input type="text"/>

Números de 6 bits

1. En un sistema numérico de 6 bits, ¿cuál es el número más grande que puedes hacer?

El registro de la mayoría de los saltos de un perro y una persona en un minuto es 59



--	--	--	--	--	--	--

¿Cómo codificarías 59 en binario?

Descifrando múltiples números

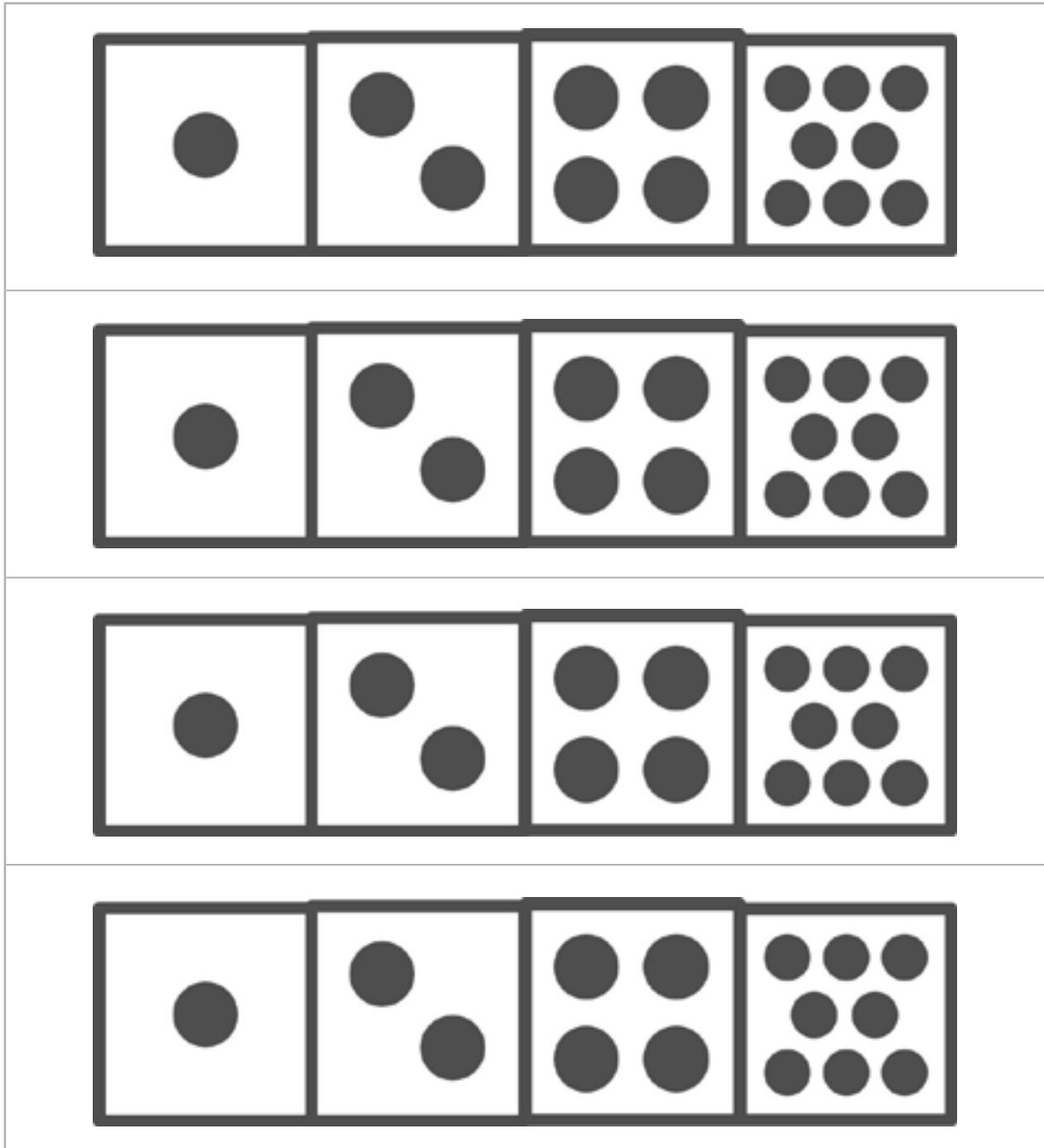
¿Cuál es el mayor número de personas levantadas y arrojadas en dos minutos?	
¿Cuál es la mayor cantidad de bebidas enlatadas abiertas por un loro en un minuto?	
¿Cuántas pulgadas es la cola más larga de un perro?	
¿Cuál es la mayor cantidad de cucharas balanceadas en la cara de alguien?	
¿Cuántos pies de largo es el timbre de viento más largo?	
¿Cuál es el vuelo más lejano, en pies, de una flecha disparada con los pies de alguien?	

Reflexión

¿Por qué es importante tener una longitud de bit establecida para sus números cuando envía información en binario?									
Usa el sistema binario de 8 bits para enviar una pregunta y una respuesta binaria a un compañero de clase.									
Pregunta									
Respuesta	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> </tr> </table>								

Tarjetas de números

Cada grupo necesitará un juego de tarjetas de números.



Lección 32

(reproducir e imprimir copia para cada estudiante)

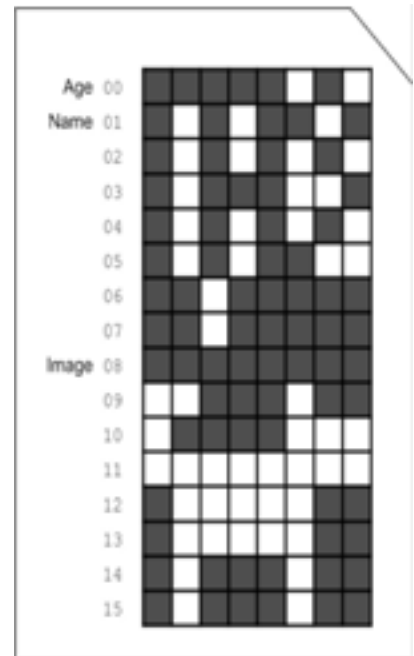
Nombre(s) Periodo Fecha

Registros de mascotas

Registros de mascotas

Un refugio para mascotas está llevando registros de todas las diferentes mascotas disponibles para adopción. Cada registro incluye la edad de la mascota, su nombre y una representación de 8x8 de la mascota. El registro se almacena en binario y está organizado en secciones de 8 bits. Cada sección de 8 bits se llama byte.

- El primer byte (00) mantiene la edad de la mascota como un número binario.
- Los siguientes siete bytes (01-07) tienen el nombre de la mascota. Cada byte es un personaje ASCII.
- Los siguientes ocho bytes (08- 15) contienen la imagen 8x8 de la mascota.



Decodifica el registro

Usa lo que sabes sobre los números binarios y ASCII para decodificar la edad y el nombre de la mascota que se describe en el registro a la derecha.



Completa la edad y el nombre de la mascota en la tabla a continuación.

¿Qué información puedes encontrar en la imagen de 8x8

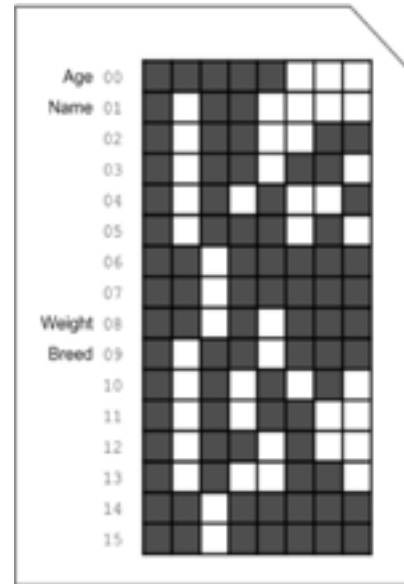
Edad	Nombre						

Nuevo registro

El refugio para mascotas decidió que la imagen no era muy útil, por lo que decidió usar el espacio para guardar el peso y la raza del perro.

El nuevo registro está organizado de la siguiente manera:

- El primer byte (00) mantiene la edad de la mascota como un número binario.
- Los siguientes siete bytes (01-07) tienen el nombre de la mascota. Cada byte es un personaje ASCII.
- El siguiente byte (08) mantiene el peso de la mascota en libras como un número binario.
- Los siguientes siete bytes (09-15) tienen la raza de la mascota. Cada byte es un personaje ASCII.



Decodifica

Usa lo que sabes sobre los números binarios y ASCII para decodificar la edad, el nombre, el peso y la raza de la mascota que se describe en el registro a la derecha



Completa la edad, el nombre, el peso y la raza de la mascota en la tabla a continuación.

Año s	nombre							peso	raza						

¿Qué otra información podría alguien querer poner en el registro?

¿Cómo se codificaría esta información?

Carácter ASCII en la tabla binaria

A	1 0 0 0 0 0	J	1 0 0 1 0 1	S	1 0 1 0 0 1
B	1 0 0 0 0 1	K	1 0 0 1 0 1	T	1 0 1 0 1 0
C	1 0 0 0 0 1	L	1 0 0 1 1 0	U	1 0 1 0 1 0
D	1 0 0 0 1 0	M	1 0 0 1 1 0	V	1 0 1 0 1 1
E	1 0 0 0 1 0	N	1 0 0 1 1 1	W	1 0 1 0 1 1
F	1 0 0 0 1 1	O	1 0 0 1 1 1	X	1 0 1 1 0 0
G	1 0 0 0 1 1	P	1 0 1 0 0 0	Y	1 0 1 1 0 0
H	1 0 0 1 0 0	Q	1 0 1 0 0 0	Z	1 0 1 1 0 1
I	1 0 0 1 0 0	R	1 0 1 0 0 1	(space)	0 1 0 0 0 0

Lección 33

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Crea una representación

Resumen

Representar objetos complejos o ideas que le importan a una computadora es una tarea desafiante. Requiere que comprendas cómo las computadoras representan información y siempre implicará compensaciones. Para este proyecto, diseñarás un sistema binario para representar tu día perfecto. El desafío será encontrar un sistema de clase para capturar todos los diferentes elementos que te importan.

Vas a...

- Escribir una breve descripción de tu día perfecto
- Identificar elementos clave que podrían representarse en una computadora
- Colaborar con tus compañeros para diseñar un sistema de representación para toda la clase
- Representar tu día perfecto en ese sistema en una tarjeta perforada
- Decodificar la tarjeta perforada del día perfecto de un compañero de clase

Enviarás...

- Esta guía de proyecto completa
- Tu tarjeta perforada rellena
- Tarjeta perforada decodificada de tus compañeros de clase y comentarios

Pasos del proyecto

Paso 1: Describe tu día perfecto

Escribe una descripción de un párrafo de tu día perfecto en el espacio a continuación

Paso 2: Compartir con un compañero

Cambia tu descripción de un día perfecto con un compañero de clase. Lee la descripción de tu compañero.

Paso 3: Elige representaciones

En la siguiente tabla, escribe la información que deseas representar en la computadora. Colócalos en las categorías apropiadas en la tabla a continuación.

Números (temperatura, # personas presentes, cuando te levantas, etc.)	Caracteres (nombre de la ubicación, tipo de clima, tipo de actividad, etc.)	Imágenes (a qué se parece la ubicación / actividad / personas / etc.)

Paso 4: Revisa las opciones de información

Reúnete con tu compañero y comparte cómo identificaste los diferentes tipos de información que te gustaría capturar. Discutir:

- Si crees que perdieron alguna información importante
- ¿Qué tipo de información tienen ambos en su historia? Prepárate para compartir con la clase

Paso 5: Lección de clase completa: Acuerda la tarjeta perforada para toda la clase. Mira la tarjeta perforada en la página siguiente. Tu clase tendrá que decidir cómo usar los 3 números, 10 caracteres y una imagen individual de 8 por 8 para representar su día perfecto. Prepárate para analizar qué información será representada por cada fila.

Paso 6: Completar la tarjeta perforada

Usando la estructura de la tarjeta perforada de tu clase, llena tu tarjeta perforada.

Paso 7: Intercambia y decodifica tarjetas perforadas

Dale tu tarjeta perforada a un compañero con el que no has trabajado hasta ahora. No les digas nada acerca de tu día perfecto y haz que decodifiquen tu tarjeta perforada.

Paso 8: Reflexión

Habla con tu pareja sobre la cantidad de días perfectos que pudiste comprender a partir de tu tarjeta perforada. ¿Qué se perdió en el proceso de representar esta información en binario?

Tu tarjeta perforada

Number 1	00								
2	01								
3	02								
Character 1	03								
2	04								
3	05								
4	06								
5	07								
6	08								
7	09								
8	10								
9	11								
10	12								
Image 1	13								
2	14								
3	15								
4	16								
5	17								
6	18								
7	19								
8	20								

Nombre del decodificador: _____

Información del número

Escribe qué números decodificaste y qué representan dado el sistema de representación de toda la clase

Información del personaje

Escribe qué personajes decodificaste y qué representan dado el sistema de representación de toda la clase

Información de la imagen

Escribe lo que crees que muestra la imagen

Describe el día perfecto de tu compañero de clase

Usando la información que decodificaste, escribe la mejor descripción de un día perfecto que puedas en el espacio a continuación.

Reflexión

Discute tu día perfecto con tu pareja.

¿Qué información se perdió al representar el día perfecto?

¿Por qué no pudiste representar esa información?

Rúbrica - resuelve un problema de datos

Rúbrica específica del proyecto

Criterio	Si/No	Comentarios
La información del día perfecto se ha categorizado en números, caracteres (palabras) e imágenes.		
La representación usa los números, los personajes y las imágenes según lo acordado por la clase.		
Los datos representados se conectan nuevamente a la descripción del día perfecto.		
Los números y caracteres en la representación han sido codificados correctamente.		
Los números y caracteres de la representación del compañero de clase se han descodificado correctamente.		
La descripción del día perfecto del compañero de clase se relaciona con la información dada.		

Reflexión de las prácticas

Práctica	Cosas para celebrar	Cosas para trabajar
Resolución de problemas		
Persistencia		
Creatividad		
Colaboración		
Comunicación		

Lección 35

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Anatomía de un smartphone

Consulta la infografía interactiva [Anatomía de un smartphone](#). Elige cinco sensores de tu smartphone según la importancia que les das a los datos que recopilan. Investiga sobre cada uno de los sensores y completa la tabla siguiente, que relaciona los **sensores de tu smartphone con los riesgos para tu privacidad que puede llevar asociados**.

SENSOR	USO PARA EL FUNCIONAMIENTO DEL SMARTPHONE	APLIACIONES QUE UTILIZAN EL SENSOR	RIESGOS PARA TU PRIVACIDAD

Preparando el debate “privacidad vs funcionalidad”

Anota en la tabla adjunta los cinco argumentos a favor y cinco argumentos en contra de la cuestión a debate.

ARGUMENTOS A FAVOR	ARGUMENTOS EN CONTRA

Lección 36

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Nuestra “smart city”

Escojan, a partir de lo que conocen de su localidad, los tres aspectos en los que creen que debería mejorar.

Mejora en la movilidad de las personas y los transportes públicos	
Planes anticontaminación	
Gestión de los residuos	
Gestión eficiente de la energía (alumbrado público, consumo de agua en las viviendas..)	
Participación ciudadana en la toma de decisiones	
Accesibilidad y calidad de servicios públicos	
Información sobre servicios y actividades	
Cuidado de las personas mayores, seguridad e inclusión	

Los retos de mi localidad

Como han visto en el video [Smart cities](#), las ciudades están siendo conectadas a internet para convertirse en smart cities y así resolver los grandes retos a los que se enfrentan a través de la recogida y análisis de datos en tiempo real.

Ideen soluciones o mejoras para los tres aspectos de su localidad que marcaron anteriormente que usen las posibilidades tecnológicas actuales, la gestión de datos masivos y la participación ciudadana.

Para obtener referencias y ejemplos pueden consultar la infografía [Una ciudad construida en datos](#), la infografía interactiva [A qué se enfrentan las grandes ciudades](#), la píldora informativa [En el interior de una “smart city”](#) o realizar una búsqueda en internet usando el concepto clave “smart city” o “ciudad inteligente”.

Lección 37

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Guía de trabajo del estudiante

Smart institute

Individualmente, consulta la píldora informativa Ciudades inteligentes, edificios inteligentes y escribe tu propia definición de “smart building”.

“Smart building”

En grupo, piensen ahora en su colegio y busquen en internet usando los términos “smart building” o “edificios inteligentes” posibles soluciones tecnológicas que podrían convertir el centro escolar en un smart institute y mejorar los siguientes cuatro aspectos:

ILUMINACIÓN	CONSUMO DE AGUA	CLIMATIZACIÓN	OBTENCIÓN DE ENERGÍA VERDE

Seleccionen una de las soluciones encontradas y escriban qué fortalezas o beneficios aporta y qué inconvenientes o debilidades tiene.

MEJORA SMART BUILDING	FORTALEZAS Y BENEFICIOS	INCONVENIENTES O DEBILIDADES

Lección 39

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Período Fecha

El boom de las redes sociales

La irrupción y el crecimiento de las redes sociales han transformado nuestros hábitos y estilo de vida, empoderando a la ciudadanía como nunca antes en la historia. Consulten la infografía interactiva y las píldoras informativas de la Lección.

Anoten las dos redes sociales que más usen y otras dos que les llamen la atención entre las que han visto:

Completen la siguiente tabla de datos de cada una para analizar las redes que más usamos. Busquen en internet la información que desconozcan:

RED				
TIPO				
BENEFICIOS INDIVIDUALES				
BENEFICIOS SOCIALES				
PERSONAS CONECTADAS				
DATOS QUE SE COMPARTEN				
CÓMO SE OBTIENE BENEFICIO ECONÓMICO				
RIESGOS DERIVADOS DE UN MAL USO				

Lección 41

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Open data

El open data, o los datos abiertos, permite a los ciudadanos, empresas y gobiernos estar mejor informados, reinventar productos y servicios y mejorar la toma de decisiones. También pone de relieve la capacidad educativa y de empoderamiento que ofrecen las nuevas tecnologías.

Vuelve a ver con detalle el video [La datificación del mundo](#), consulta la infografía [El empoderamiento ciudadano](#) y la píldora informativa [Datos a tu alcance](#) y contesta las siguientes preguntas:

Veo	¿Qué has visto en el video, la infografía y la píldora informativa?
Pienso	¿Qué piensas de lo que has visto?
Me pregunto	¿Qué otras cosas te gustaría saber del open data?

Lección 42

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Profesiones del mañana

Estamos viviendo la cuarta revolución industrial y, con ella, el impacto que están empezando a tener las nuevas tecnologías en los puestos de trabajo actuales. La irrupción del big data está generando la destrucción de algunos puestos de trabajo y la demanda de otros nuevos.

Consulten la infografía Las nuevas profesiones de la era digital y contesten las siguientes preguntas:

Conexiones con mi vida cotidiana

¿Qué profesiones de gente de tu alrededor crees que desaparecerán en 20 años?

Desafiar y argumentar

¿Crees que los cambios que trae la cuarta revolución industrial son positivos? ¿por qué?

Cambios de actitud/pensamiento/acción

¿Las informaciones que acabas de ver han cambiado algo de tu pensamiento o tu visión profesional futura?

Lección 44

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Titanic

Aprendizaje automático para niños: Notas para profesores	
Ficha	Titanic
Actividad	Entrena la computadora para poder predecir quién sobrevivió al hundimiento del Titanic.
Objetivo	Enseña a una computadora a predecir resultados La analítica predictiva se puede utilizar para identificar patrones en datos estructurados.
Nivel dificultad	Principiante
Tiempo estimado	45 minutos
Resumen	Los estudiantes entrenarán un modelo predictivo basado en datos históricos.
Tema	Modelo predictivo
Configuración	
Cada estudiante necesitará:	
Copias impresas	Hoja de trabajo del proyecto, descargar desde https://machinelearningforkids.co.uk/worksheets Los scripts de Bloques en Scratch están codificados por colores, por lo que imprimir en color se lo facilitará a los estudiantes.
Acceso	Nombre de usuario y contraseña para machinelearningforkids.co.uk
Otro	Una forma de crear y ejecutar programas Python
La cuenta de la clase necesitará:	
Claves API	Ninguno

Ayuda	
Situaciones potenciales	<ul style="list-style-type: none"> • Este proyecto les pide a los estudiantes que piensen en las razones por las que la gente sobreviviría y no sobreviviría después de que un barco se hundiera. Este tema puede resultar incómodo para algunos de ellos, así es que considere si este proyecto es apropiado para su clase antes de usarlo. • El código Python que los estudiantes ejecutarán utiliza las "solicitudes" de la biblioteca. Hay un enlace en la página del estudiante para instalar solicitudes, pero recomiendo hacerlo antes de la clase. • Ver https://3.pythonrequests.org/user/install/#install para más información. • Si tiene tiempo, haga que sus alumnos encuentren información sobre los personajes de la película. Descargue el documento original de Word de la hoja de trabajo desde https://github.com/IBM/taxinomitis-docs/tree/master/project-worksheets/msword • Elimine la información sobre Jack y Rose y escriba una parte para que ellos encuentren más información. • Revisar y comprender los datos de entrenamiento es la parte más importante de este proyecto. Deje mucho tiempo para esto. Podría invitarlos a especular sobre posibles patrones primero (Por ejemplo, los hombres tenían más probabilidades de sobrevivir si tenían una esposa e hijos con ellos, ya que las familias podrían haberse mantenido unidas. ¿O era más probable que los hombres sobrevivieran si estuvieran solos, ya que podrían haber sido más egoístas?) y luego mira si los datos coinciden con eso. Puede invitarlos a encontrar patrones en los datos y luego teorizar sobre las razones posteriores. O puede dejar que hagan un poco de ambas cosas. Deben dedicar tiempo a buscar y pensar en patrones en los datos. • Deje en claro que la computadora buscará patrones en los datos (pero no interpretará, especulará o teorizará sobre esos patrones). • Después de la sesión, anime a los estudiantes a pensar en otras aplicaciones con un modelo predictivo. ¿Qué otros conjuntos de datos numéricos y categóricos (opción múltiple) pueden pensar que podrían tener patrones que una computadora podría aprender? <p>Solución de problemas generales y ayuda en https://machinelearningforkids.co.uk/help</p>

Lección 44

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Hazme Feliz

Aprendizaje automático para niños: Notas para profesores	
Ficha	Hazme feliz
Actividad	Crea un personaje en Scratch que sonría si le dices cosas agradables y llora si le dices cosas malas.
Objetivo	<p>Enséñele a una computadora a reconocer los cumplidos y los insultos</p> <ul style="list-style-type: none"> • Cómo se pueden entrenar las computadoras para reconocer el tono emocional • Cómo el aprendizaje supervisado construye sistemas que pueden lidiar con entradas inesperadas
Nivel dificultad	Principiante
Tiempo estimado	45 minutos
Resumen	Los estudiantes entrenarán un modelo de aprendizaje automático para reconocer cumplidos e insultos escribiendo ejemplos de declaraciones amables y declaraciones malas. Usarán Scratch para crear un personaje que reaccione a los mensajes en función de sus sentimientos.
Tema	Análisis de sentimientos, aprendizaje supervisado
Configuración	
Cada estudiante necesitará:	
Copias impresas	Hoja de trabajo del proyecto (descargar desde: https://machinelearningforkids.co.uk/worksheets) Los scripts de Bloques en Scratch están codificados por colores, por lo que imprimir en color se lo facilitará a los estudiantes.
Acceso	Nombre de usuario y contraseña de machinelearningforkids.co.uk

La cuenta de la clase necesitará:	
Claves API	<p>Asistente Watson</p> <p>1 espacio de trabajo por alumno</p> <p>Una clave de API "Lite" es gratuita, pero solo se puede usar para crear 5 espacios de trabajo.</p> <p>Se puede usar una clave de API "Estándar" para crear 20 espacios de trabajo.</p> <p>Más información en: https://github.com/IBM/taxinomis-docs/raw/master/docs/pdf/machinelearningforkids-apikeys.pdf</p>
Ayudar	
Situaciones potenciales	<ul style="list-style-type: none"> • Los estudiantes más jóvenes pueden dejarse llevar al escribir insultos para entrenar el modelo de aprendizaje automático. Puede ser útil establecer límites del lenguaje apropiado. • La gestión del tiempo es importante para este proyecto. Los estudiantes a menudo pierden la noción del tiempo dibujando su rostro y no dejan suficiente tiempo para entrenar o codificar. • "https://machinelearningforkids.co.uk" es una URL larga que pueden escribir algunos niños. Puede que le resulte más fácil configurar un marcador en el que puedan hacer clic. • Las capturas de pantalla de la hoja de trabajo se basan en Scratch 3. Es posible que prefiera utilizar Scratch 2 en su lugar. Sin embargo, a los estudiantes puede resultarles más difícil encontrar algunos bloques. <p>Solución de problemas generales y ayuda en: https://machinelearningforkids.co.uk/help</p>

Lección 44

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Hazme Feliz

En este proyecto harás un personaje que reacciona a lo que dices.

Si le haces un cumplido, se pondrá feliz. Si lo insultas, se pondrá triste.

Enseñarás a la computadora a reconocer los mensajes amables y los mensajes malos dándole ejemplos de cada uno.



1. Ir a <https://machinelearningforkids.co.uk/> en un navegador web
2. Haz clic en "Empezar"
3. Haz clic en "Pruébalo ahora"
4. Haz clic en el botón "+ Agregar un nuevo proyecto"
5. Nombra tu proyecto "Hazme feliz" y configúralo para aprender a reconocer "texto". Haga clic en el botón "Crear".

Start a new machine learning project

Project name
make me happy

Requesting
text

Language
English

CREATE CANCEL

What type of text do you want to teach the computer to recognise?
For words, sentences or paragraphs, choose 'text'
For photos, diagrams and pictures, choose 'images'
For lists of numbers or multiple choices, choose 'numbers'
For games and events, choose 'events'

6. Ahora debería ver "Hazme feliz" en la lista de sus proyectos. Haz clic en éste.
7. Necesitas ejemplos para entrenar al programa. Haz clic en el botón "Train".

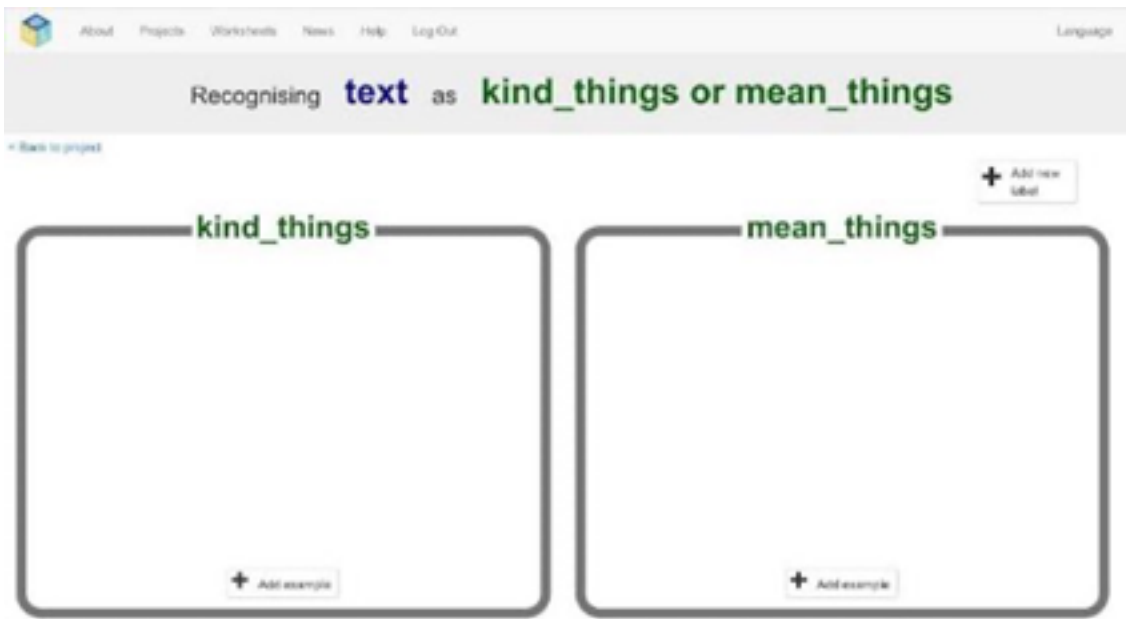
"make me happy"

Train
Collect examples of what you want the computer to recognise
Train

Learn & Test
Use examples to train the computer to recognise text
Learn & Test

Make
Use the machine learning model you've trained to make a game or app, in Scratch or in Python
Make

8. Haz clic en "+ Agregar nueva etiqueta" y llámalo "cosas amables". Hazlo de nuevo y crea un segundo grupo llamado "cosas malas".

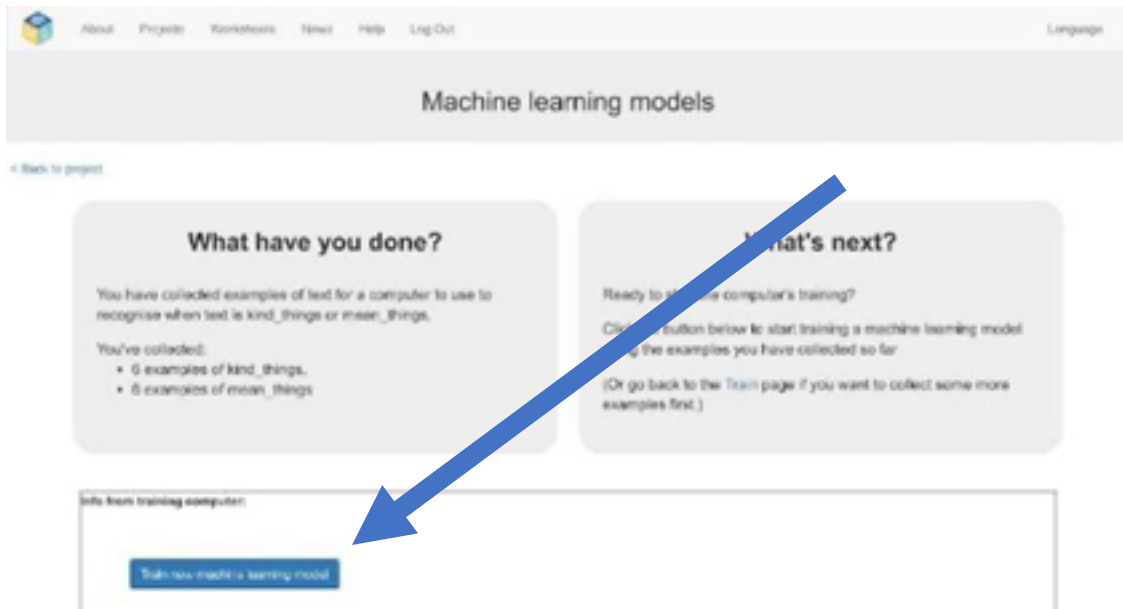


9. Haz clic en "Agregar ejemplo" en el cuadro de "cosas amables" y escribe el cumplido más amable que se le ocurra.
10. Haz clic en "Agregar ejemplo" en el cuadro de "cosas malas" y escribe el insulto más cruel y mezquino que se le ocurra.
11. Repita los pasos 9 y 10 hasta que tenga al menos seis ejemplos de cada uno.

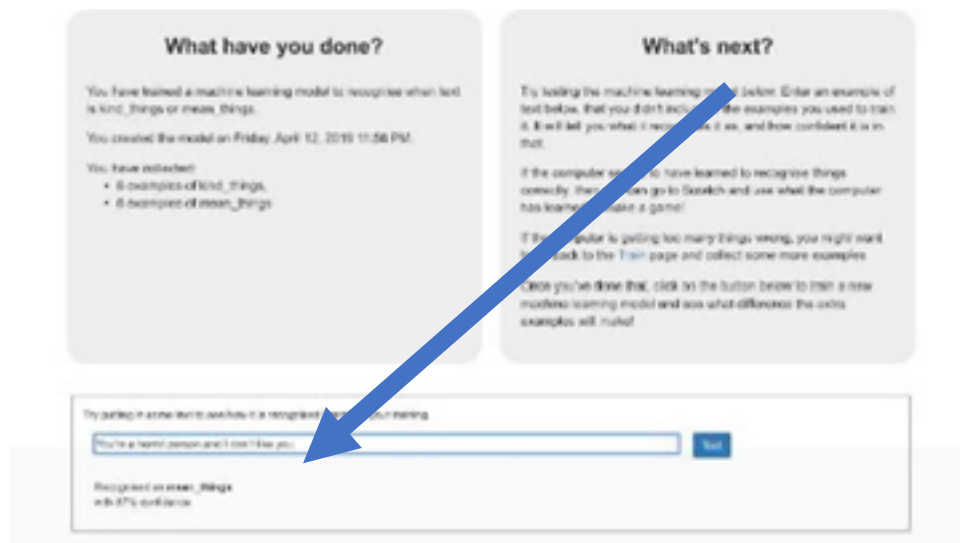


12. Haz clic en el enlace "Volver al proyecto".
 13. Haz clic en el botón "Aprender y probar".
 14. Haz clic en el botón "Entrene un nuevo modelo de aprendizaje automático".
- Siempre que haya recopilado suficientes ejemplos, la computadora debería comenzar a aprender

a reconocer mensajes a partir de los ejemplos que le haya dado.



15. Espere a que finalice el entrenamiento. Esto puede tardar unos minutos.
16. Una vez finalizada la formación, se mostrará un cuadro de prueba.



Intenta probar tu modelo de aprendizaje automático para ver qué haz aprendido la computadora. Escribe algo y presiona Enter. Debe reconocerse como amable. Escribe algo malo y presiona Enter. Debe reconocerse como cruel.

Pruébalo con ejemplos que no hayas usas antes en la computadora. Si no estás satisfecho con la forma en que la computadora reconoce los mensajes, vuelve al paso 11 y agrega algunos ejemplos más. ¡Asegúrate de repetir el paso 13 para entrenar con los nuevos ejemplos!

17. Haz clic en el enlace "Volver al proyecto"
18. Haz clic en el botón "Hacer"

19. Haz clic en el botón "Scratch 3".
Esta página tiene instrucciones sobre cómo usar los nuevos bloques en Scratch. Mantén la página abierta si necesitas verificar cómo usarlos.

¿Qué has hecho hasta ahora?

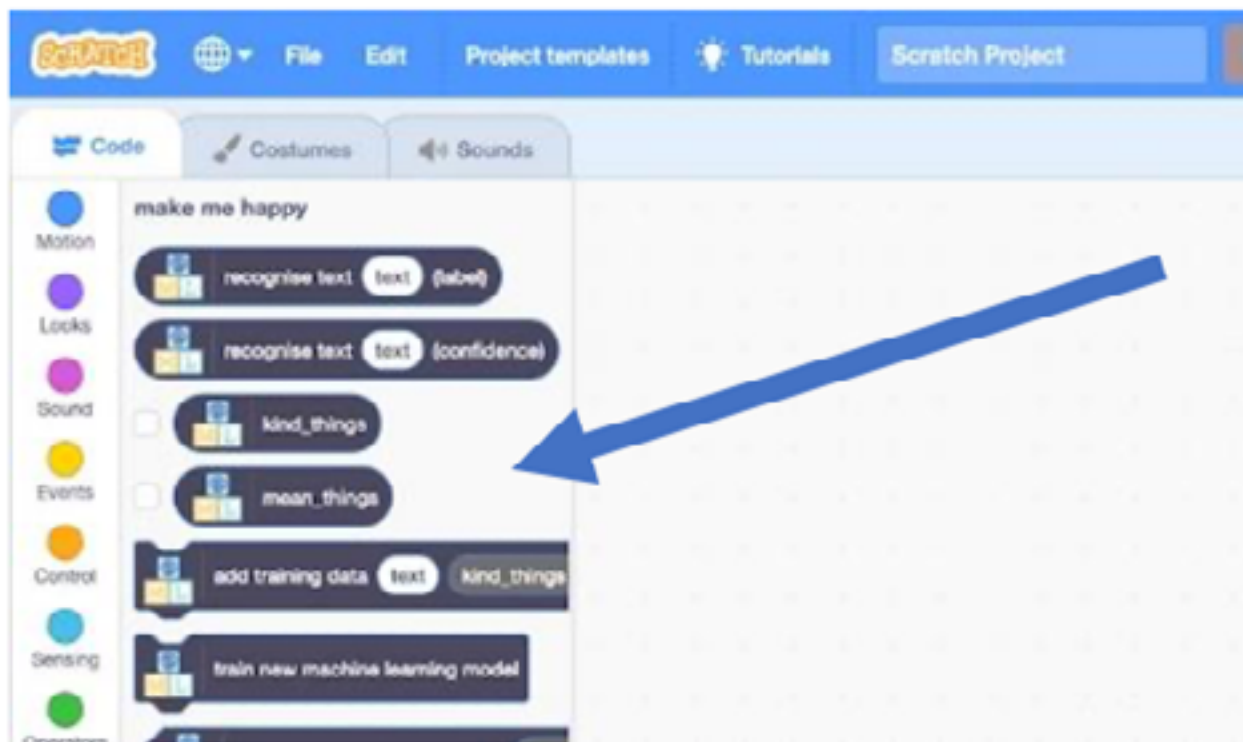
Haz comenzado a entrenar a una computadora para que reconozca el texto como amable o cruel. En lugar de intentar escribir reglas para poder hacer esto, lo está haciendo por medio de recopilar ejemplos. Estos ejemplos se utilizan para entrenar un "modelo" de aprendizaje automático.

Esto se denomina "aprendizaje supervisado" debido a la forma en que supervisa el entrenamiento de la computadora.

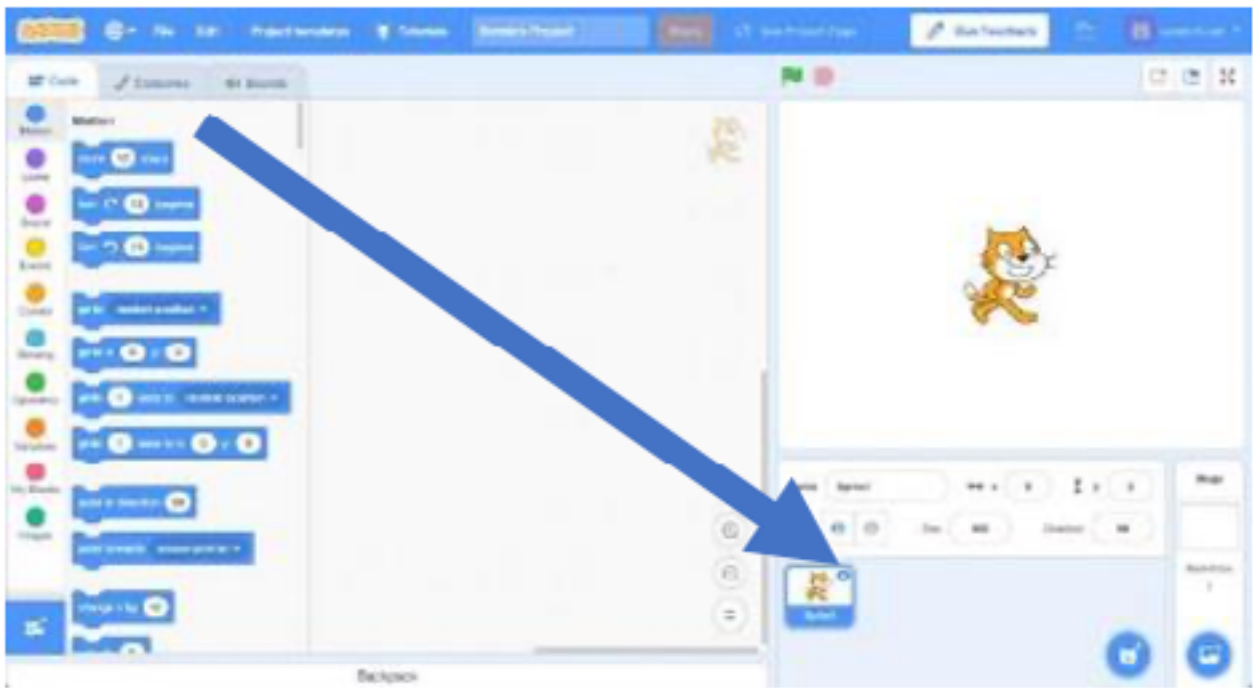
La computadora aprenderá de los patrones en los ejemplos que le ha dado, como la elección de palabras y la forma en que se estructuran las oraciones. Estos se utilizarán para poder reconocer nuevos mensajes.

20. Haz clic en el botón "Abrir en Scratch".

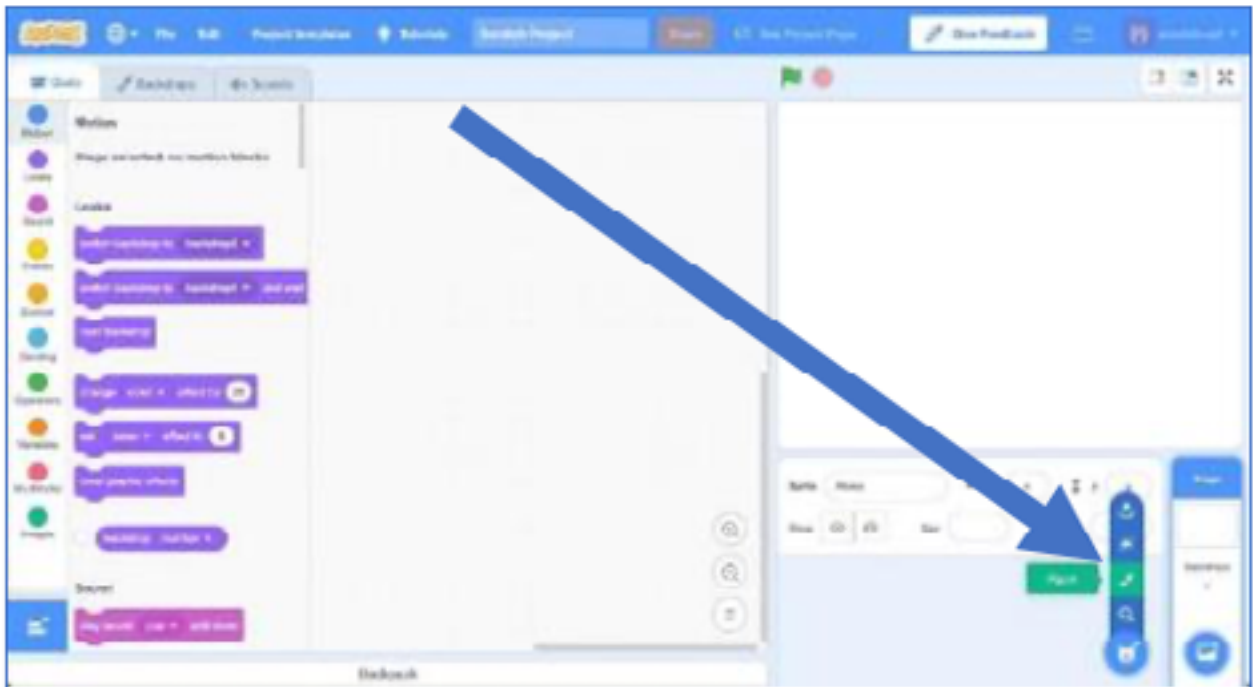
Deberías ver nuevos bloques de su proyecto al final de la lista.



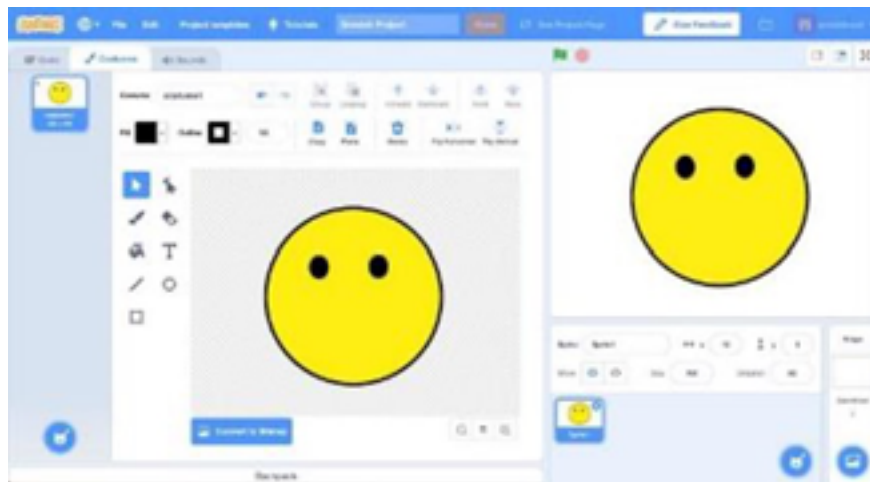
21. Paso siguiente



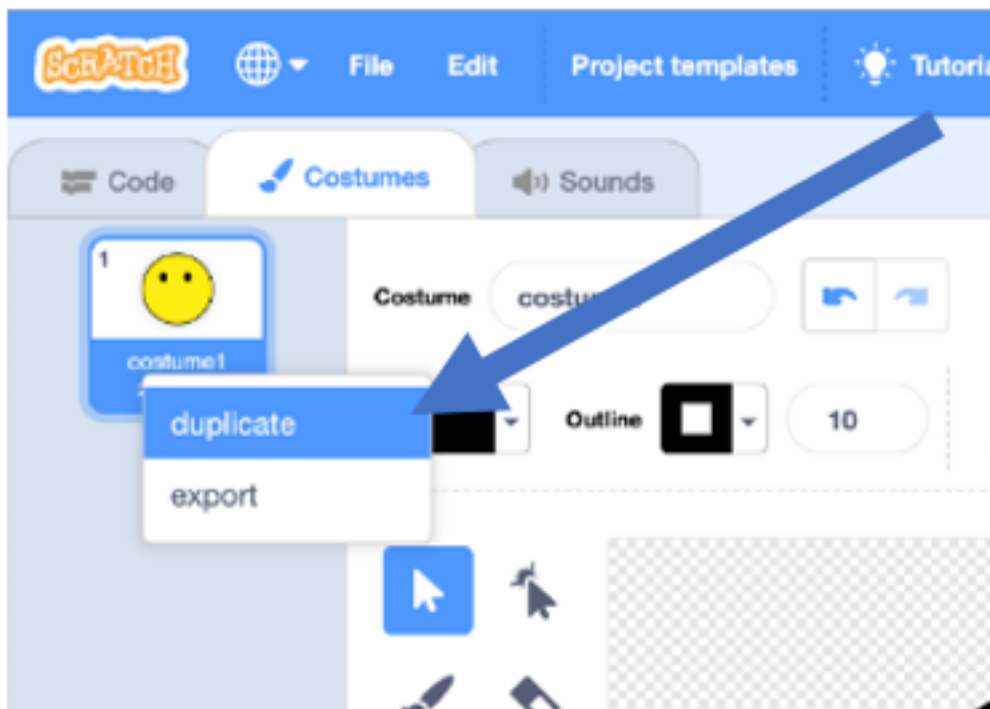
22. Crea un nuevo objeto haciendo clic en el ícono Pincel.



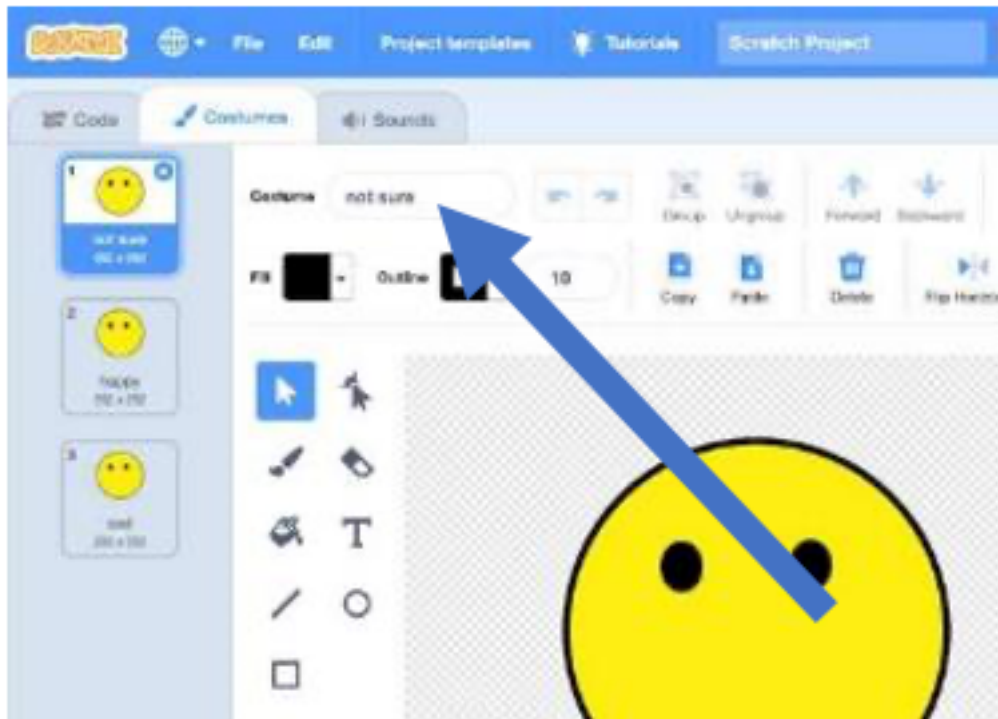
23. Dibuja una cara sin boca en la pestaña Disfraces.



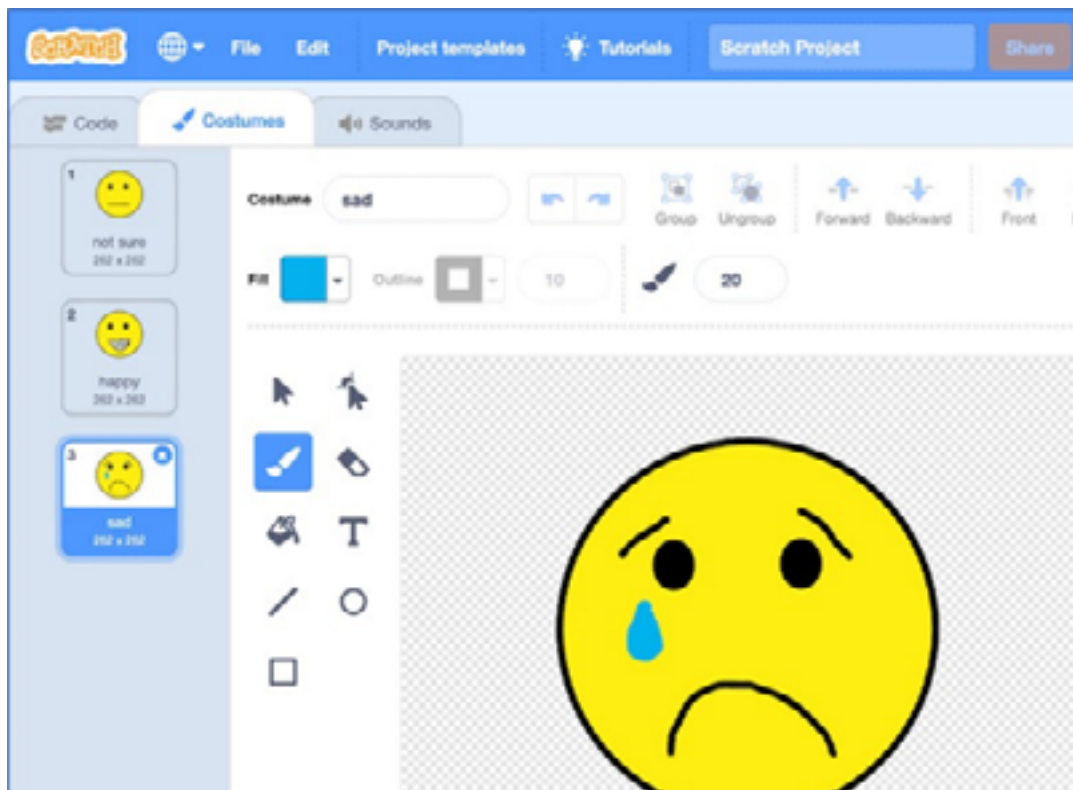
24. Haz clic con el botón derecho en el disfraz y haz clic en "duplicar". Haz eso de nuevo para que tengas Tres copias del disfraz. Asegúrate de duplicar el disfraz, no el sprite



25. Nombra los tres disfraces "no estoy seguro", "feliz" y "triste" Escribe los nombres en el cuadro blanco que se muestra con la flecha a continuación.



26. Dibuja una boca en cada uno de los disfraces. La cara de "no estoy seguro" debe ser una línea recta. La cara "feliz" debe tener una sonrisa. La cara "triste" debe verse triste.

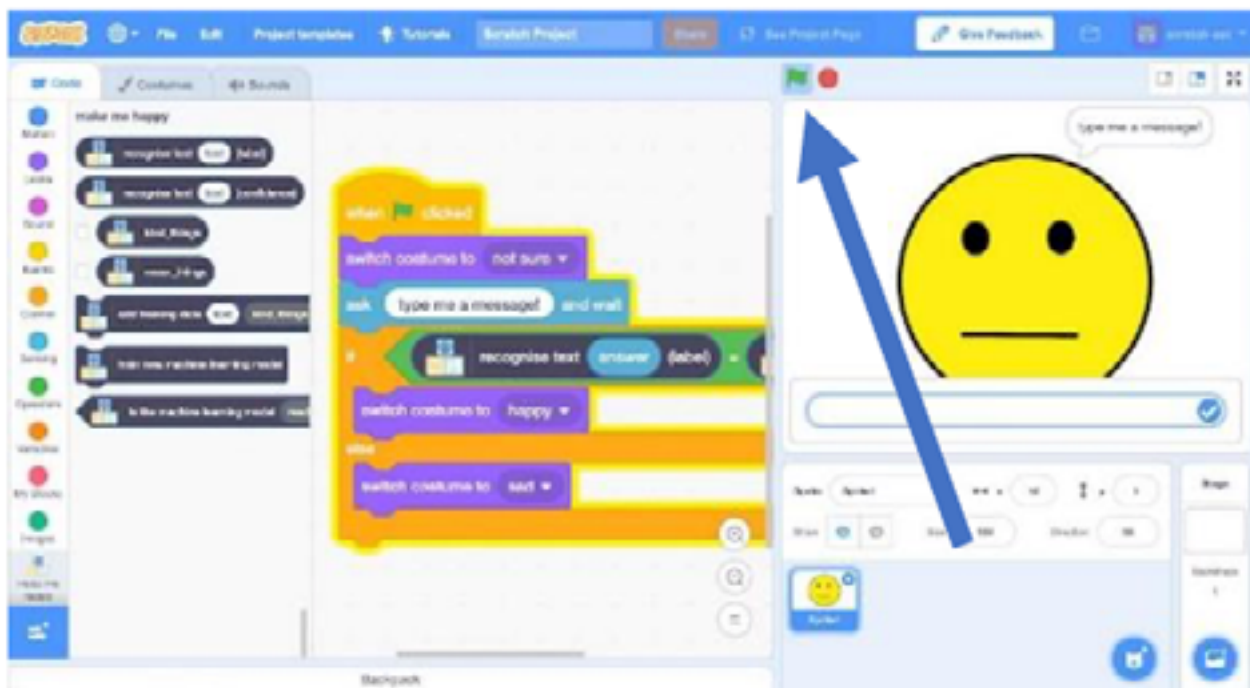


27. Haz clic en el "Código", E ingresa la siguiente secuencia de comandos.

El bloque "reconocer texto ... (etiqueta)" es un nuevo bloque agregado por su proyecto. Si le da un texto, lo reconocerá como "cosas amables" o "cosas malas" según la capacitación que le ha dado a la computadora.



28. Haga clic en la bandera verde Probar.



29. ¡Escribe un mensaje y observa cómo reacciona!

Escribe un mensaje amable y presiona enter. El personaje debe sonreír. Vuelve a hacer clic en la bandera verde. Escribe un mensaje cruel y cruel y presiona Enter. El personaje debe verse triste.

Esto debería funcionar para mensajes que no incluiste en el entrenamiento de tu máquina.

¿Qué has hecho?

Has comenzado a entrenar a una computadora para que reconozca el texto como amable o cruel. En lugar de intentar escribir reglas para poder hacer esto, lo está haciendo, recopilando ejemplos. Estos ejemplos se utilizan para entrenar un "modelo" de aprendizaje automático.

Esto se denomina "aprendizaje supervisado" debido a la forma en que supervisa el entrenamiento de la computadora.

La computadora aprenderá de los patrones en los ejemplos que le ha dado, como la elección de palabras y la forma en que se estructuran las oraciones. Estos se utilizarán para poder reconocer nuevos mensajes.

Ideas y ampliaciones

Ahora que has terminado, ¿por qué no pruebas una de estas ideas?

¿O inventa uno propio?

Escribe una respuesta

En lugar de simplemente cambiar su apariencia, haz que tu personaje responda, ¡según lo que reconoce en el mensaje!

Prueba con un personaje diferente

En lugar de la cara de una persona, ¿por qué no probar algo diferente, como un animal?

Por ejemplo, ¿podrías hacer que un perro mueva la cola si le dices algo amable!

Utiliza la puntuación de confianza

El otro bloque de Scratch nuevo de su proyecto da la puntuación de qué tan seguro está su modelo de aprendizaje automático de que ha reconocido el mensaje. ¿Cómo puedes usar esto para mensajes que no sean cumplidos o insultos?

Análisis de sentimientos del mundo real

¿Puedes pensar en ejemplos en los que sea útil poder entrenar una computadora para que reconozca la emoción por escrito?

Lección 44

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

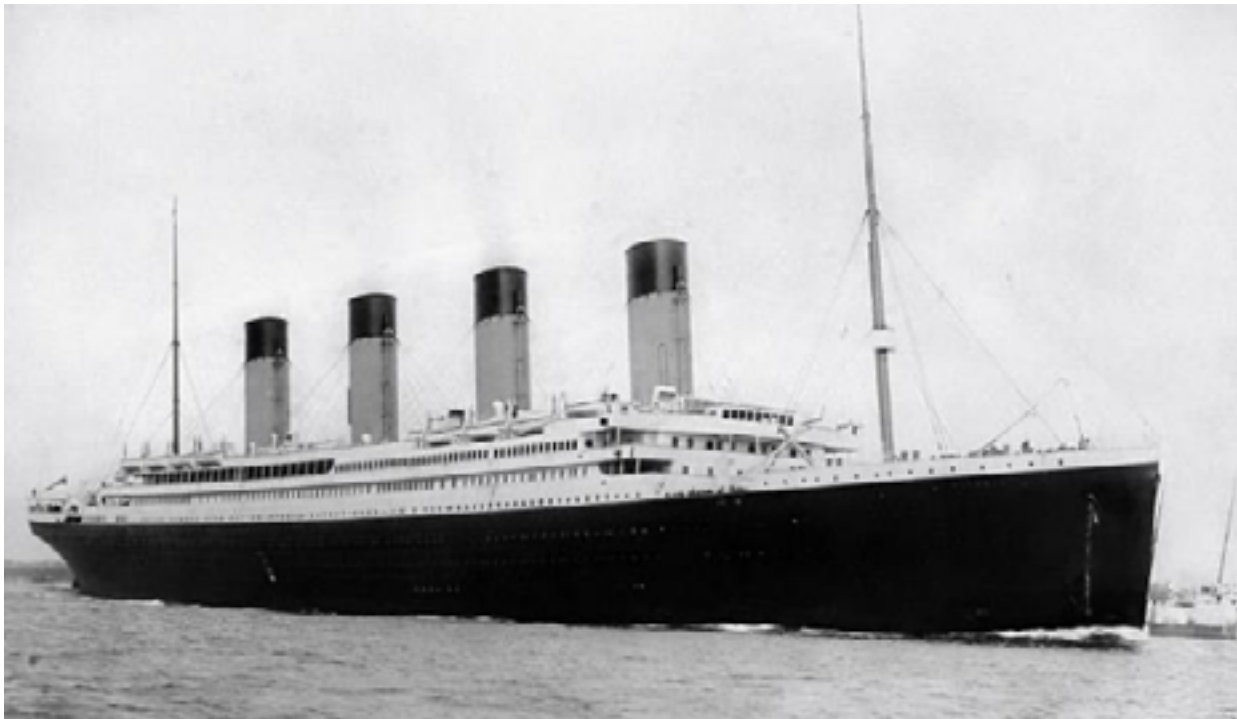
Fecha

Titanic

El RMS Titanic se hundió en el norte del Océano Atlántico en 1912 durante su viaje inaugural que iba desde Southampton hasta New York. Más de 1.500 pasajeros y tripulación de los 2.220 que iban a bordo murieron cuando se hundió el barco.

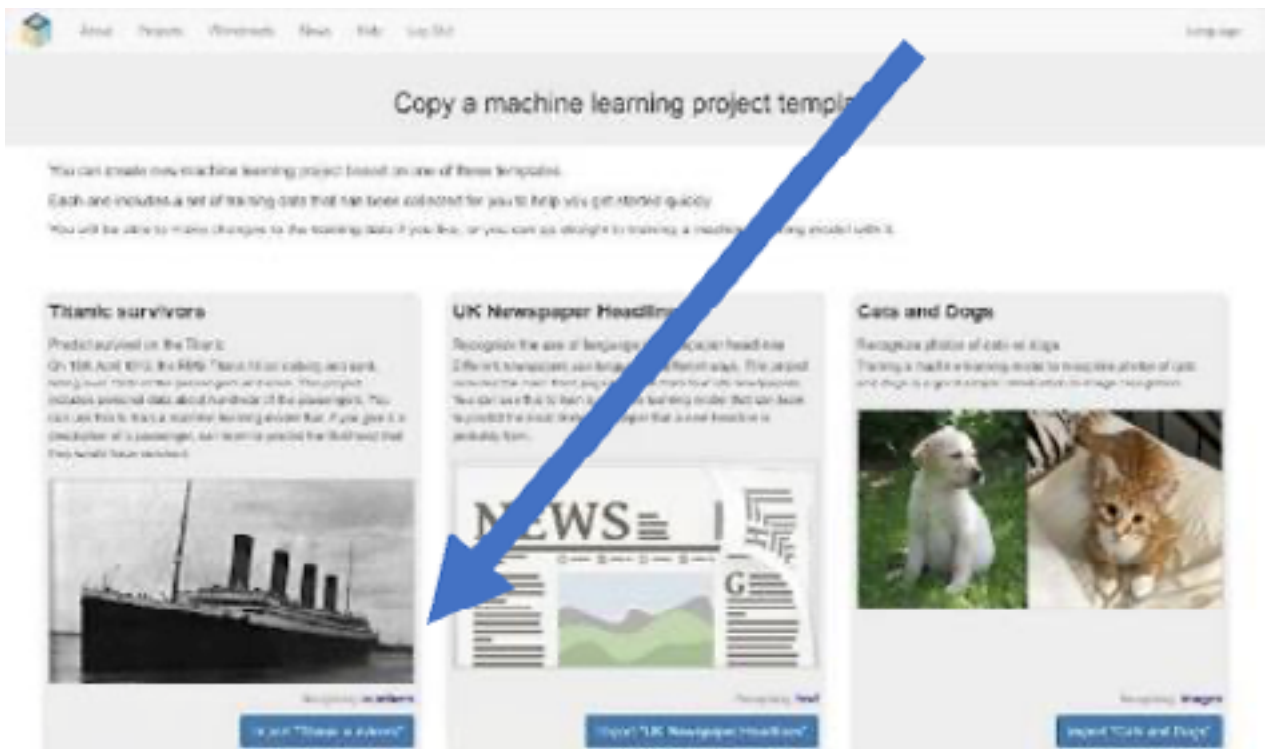
En este proyecto harás un programa con Python que pueda predecir si un pasajero específico sobrevivió.

Usarás machine learning para entrenar un modelo predictivo usando información sobre ejemplos de cientos de pasajeros.

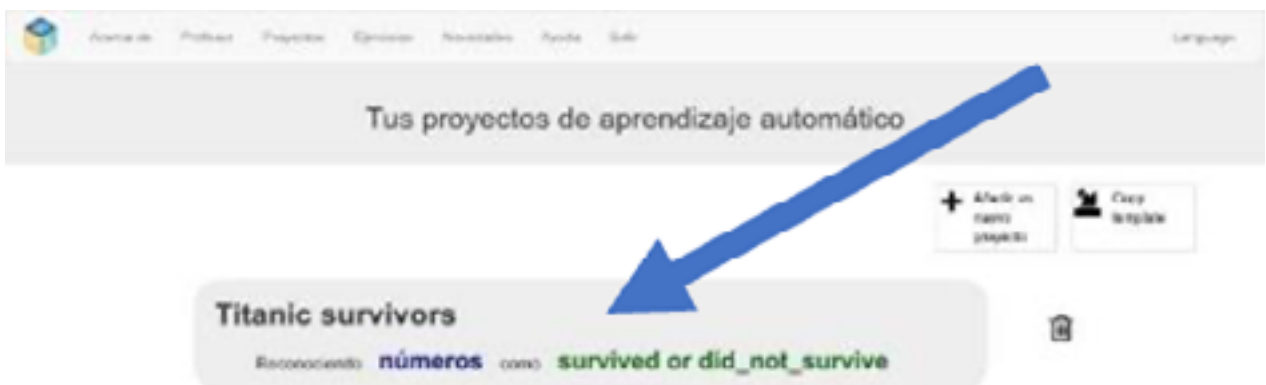


1. Ve a la página <https://machinelearningforkids.co.uk/> en un navegador web
2. Pulsa en **“Empezar”**
3. Pulsa en **“Inicia Sesión”** e introduce tu usuario y contraseña. Si no tienes un usuario, pide a tu profesor que te cree uno. Si no recuerdas tu usuario o contraseña, pide a tu profesor o líder de grupo que te la restablezca.
4. Pulsa en **“Proyectos”** en la barra superior del menú.

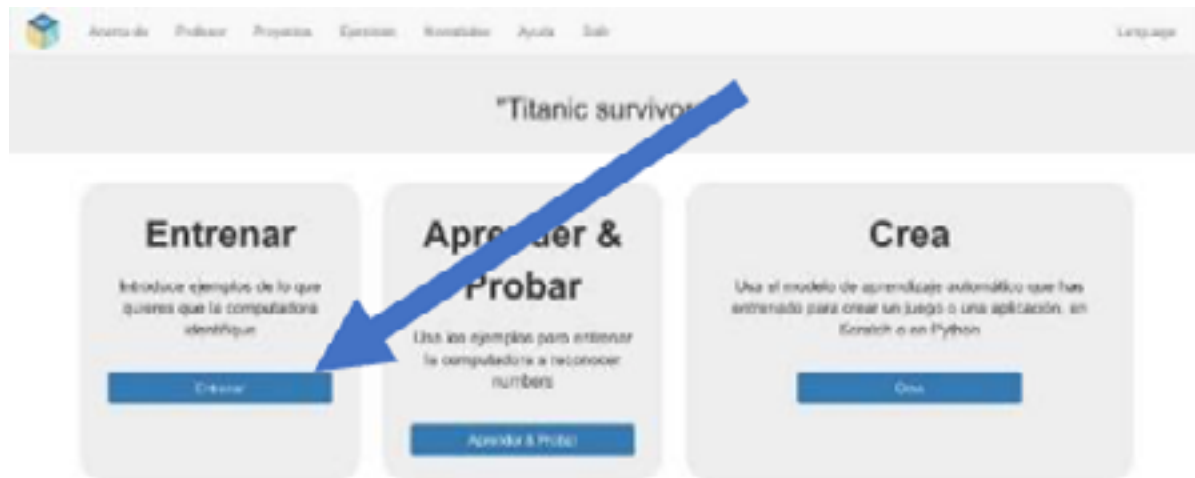
5. Pulsa el botón “Copy template”.
 6. Busca el modelo del proyecto Titanic survivors y pulsa el botón “Import Titanic survivors”.
- Aparecerá una descripción de los datos de los pasajeros. Pulsa en “Import”.



7. Pulsa en tu nuevo proyecto “Titanic survivors”.



8. Empieza desplegando los datos de entrenamiento que acabas de copiar. Pulsa el botón “Entrenar”



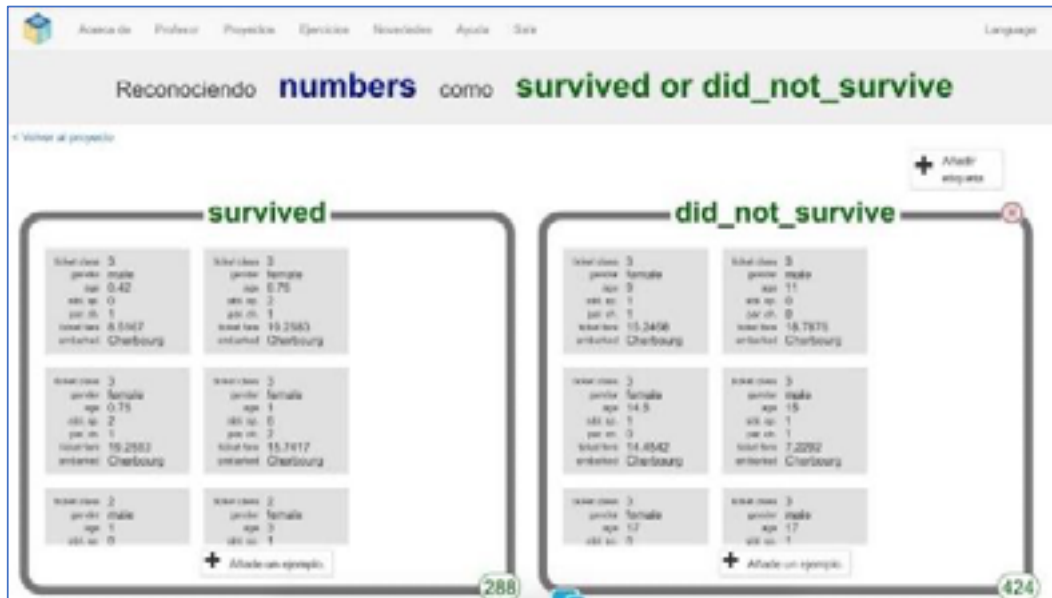
9. Hay dos zonas de entrenamiento:

“survived” contiene ejemplos de pasajeros/tripulación que sobrevivió.

“did_not_survive” contiene ejemplos de personas que no sobrevivieron.

Estos ejemplos te dan información sobre la persona:

ticket class	1, 2, 3	¿Tenían billetes de primera, segunda o tercera clase?
gender	male, female	¿Eran hombre o mujer?
age	Edad en numero	¿Qué edad tenían?
sibl. sp.	Hermanos o cón-yuges	Si eran niños, ¿Cuántos hermanos y hermanas tenían a bordo? Si eran adultos, 1 si tenía cónyuge con ellos o 0 si no.
par. ch.	Padres o hijos	Si eran niños, cuantos padres tenían a bordo con ellos. Si eran adultos, cuantos hijos tenían a bordo.
ticket fare	Coste en libras	¿Cuánto costo su billete?
embarked	Cherbourg, Southampton, Queenstown	¿Dónde subieron a bordo? Cherbourg (Francia), Southampton (Inglaterra) o Queenstown (Irlanda)?



10. Revisa los ejemplos de entrenamiento. ¿Puedes ver algún patrón en los datos? ¿Existen diferencias comunes entre los pasajeros que sobrevivieron y los que no? Por ejemplo, ¿has oído la frase "¡Las mujeres y los niños primero!"? Si fue así como los pasajeros y la tripulación llegaron a los botes salvavidas, ¿qué tipo de patrón esperarías ver en los ejemplos de entrenamiento?

¿Puedes encontrar este patrón en los datos del proyecto?

¿Qué otros patrones se te ocurren? Por ejemplo, ¿piensas que los pasajeros con billetes más caros tendrían una oportunidad mejor o peor de sobrevivir el hundimiento?

Intenta predecir qué tipo de patrones intentara aprender a reconocer tu modelo de machine learning. Después mira a ver si puedes encontrar ese tipo de patrón en tus datos de entrenamiento.

11. Pulsa en el enlace de "**< Volver al proyecto**".

12. Pulsa en el botón "**Aprender & Probar**".

13. Pulsa en el botón "**Entrena un nuevo modelo**".

Los datos de ejemplo de los pasajeros se usarán para entrenar un modelo predictivo.



Probaremos tu modelo usando los personajes de la película de 1997, Titanic.

Jack Dawson (Interpretado por Leonardo DiCaprio)

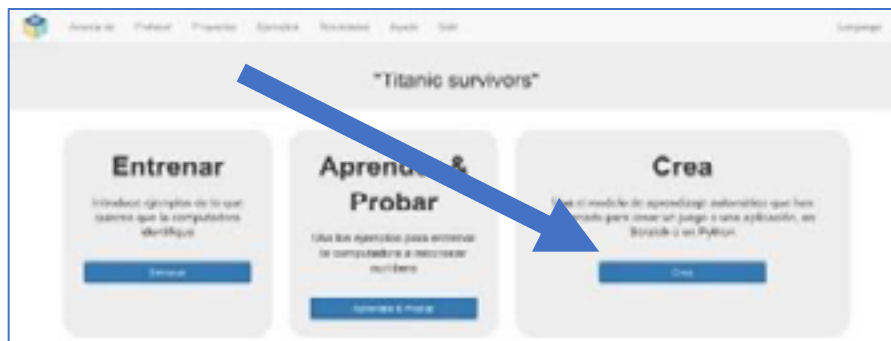
- Jack era un hombre de 20 años.
- No pago nada por su billete de tercera clase, lo gano jugando al póker.
- Subió a bordo del RMS Titanic en Southampton, Inglaterra.
- Se quedó huérfano a los 15 años, no tenía familia a bordo.

Rose DeWitt-Bukater (Interpretado por Kate Winslet)

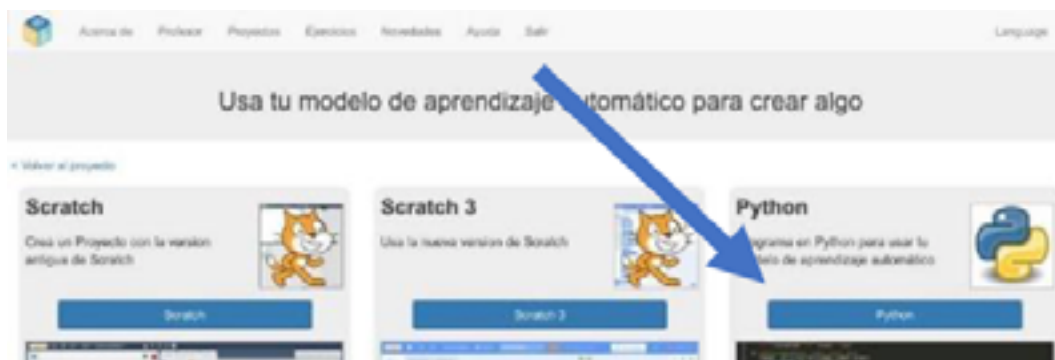
- Rose era una mujer de 17 años.
- Tenía un billete de primera clase. La película no menciona cuanto pago por él, basándonos en la habitación que tenía, calculamos que costo £450.
- Subió a bordo del barco en Southampton, Inglaterra.
- Subió al barco con su madre y su prometido.
- No tenía ningún hermano o hermana en el barco.

14. Pulsa en el enlace de “< Volver al proyecto”.

15. Ahora, usaremos Python. Pulsa el botón “Crea”



16. Pulsa en el botón de Python.



17. Rellena con la información del personaje de Jack Dawson en los campos de la izquierda.

Esto actualizará el código de la derecha.

18. Copia el código actualizado en un editor de texto. Guárdalo como un archivo llamado Jack.py.

19. Encuentra la última línea del código: `print ("result: '%s' with %d%% confidence" % (label, confidence))`

Cámbiala por la siguiente: `print ("%d%% sure that Jack %s" % (confidence, label))`

20. Ejecuta el programa con el comando: `python jack.py`

Tu modelo de machine learning mostrará su predicción sobre si Jack sobrevivió al hundimiento del Titanic.

¿La predicción concuerda con lo que pasa en la película?

21. Intenta hacer lo mismo para el personaje de la película de Rose. Rellena los campos de la página web con la información del personaje de Rose.

Guarda el código actualizado a un archivo llamado **rose.py**. Actualiza la última línea del código: `print ("%d%% sure that Rose %s" % (confidence, label))`

Ejecuta el programa con el comando: `python rose.py`

Tu modelo de machine learning mostrará su predicción sobre si Rose sobrevivió al hundimiento del Titanic.

¿La predicción concuerda con lo que pasa en la película?

¿Qué has hecho?

Has usado información histórica sobre los pasajeros y la tripulación del RMS Titanic para entrenar un modelo predictivo. El modelo predictivo usa los patrones de los datos de entrenamiento históricos para aprender a reconocer los patrones de que tipo de pasajeros sobreviven y cuáles no.

Cuando buscaste patrones en los datos de entrenamiento pensaste en posibles razones para explicarlo. Por ejemplo, los pasajeros más jóvenes tenían más probabilidad de sobrevivir porque los niños tuvieron prioridad al subir a los botes salvavidas. Los ordenadores no habrían hecho esto último. El ordenador habrá visto el patrón entre la edad y si sobrevivieron, pero no intenta explicarlo. Un modelo de machine learning aprende a reconocer patrones en los datos, pero no le importará el por qué.

Finalmente, has hecho un programa de Python para probar el modelo predictivo, usando información sobre personajes ficticios de la película de Titanic

Ideas y ampliaciones

Ahora que has acabado ¿por qué no pruebas una de estas ideas?

¿o ahora probar la tuya?

Prueba con otros pasajeros

¿Puedes encontrar información sobre otros pasajeros o tripulación el Titanic con los que puedas probar tu modelo predictivo?

Inventa tu propio modelo predictivo

¿Puedes pensar en otro tema en el que puedas recolectar valores numéricos o de opción múltiple?

Unidad 03

¿Cómo nos ayuda el computador para resolver problemas matemáticos?

Contenido 1:

Cálculos matemáticos en la resolución de problemas

Contenido 2:

Datos por todas partes

Contenido 3:

Geometría en la naturaleza

Contenido 1

Cálculos matemáticos para resolución de problemas

Resumen

- **Lección 49:** Bloques de Evaluación y expresiones aritméticas
- **Lección 50:** Cadenas (“strings”) e imágenes
- **Lección 51:** Contratos (definición de funciones), dominio y rango
- **Lección 52:** Escribiendo contratos (definición de funciones)
- **Lección 53:** Definiendo variables y substitución
- **Lección 54:** Funciones rápidas
- **Lección 55:** Funciones compuestas
- **Lección 56:** La Receta de Diseño
- **Lección 57:** Resolver problemas con la Receta de Diseño
- **Lección 58:** La altura del cohete

Objetivos

- **OA 4.** Crear aplicaciones y realizar análisis mediante procesadores simbólicos, de geometría dinámica y de análisis estadístico.
- **OA g.** Elaborar representaciones, tanto en forma manual como digital, y justificar cómo una misma información puede ser utilizada según el tipo de representación.

Referencias

- <https://studio.code.org>
- <https://cuantrix.mx/>
- **CSTA-** Computer Science Teachers Association

Lección 49

Bloques de evaluación y expresiones aritméticas

Lección en línea

Propósito

Para comenzar, los estudiantes usarán los Bloques de Evaluación para explorar el concepto de las matemáticas como un idioma, específicamente un lenguaje de programación. Al componer expresiones aritméticas con Bloques de Evaluación, los estudiantes podrán visualizar cómo es que las expresiones siguen el orden de las operaciones.

En esta Lección, los estudiantes experimentarán el primer acercamiento con el lenguaje de programación y con el ambiente en el que se desarrollarán durante este curso. A pesar de que usaremos un contexto y un lenguaje basado en bloques muy similar al visto en Scratch o en Hour of Code, es importante entender que el tipo de programación es un tanto diferente. Con el objetivo de que el contenido esté bien alineado con las reglas del álgebra, seguiremos un paradigma de programación conocido [como programación funcional](#). Esto puede que no llame la atención de todos los estudiantes; sin embargo, a aquellos que ya tengan experiencia con la programación podría pedirles investigar sobre este paradigma en particular y rescatar las similitudes y diferencias.

Secuencia para el aprendizaje

Conocimiento inicial

Lección: bloques de Evaluación

Objetivos

Los estudiantes serán capaces de:

- Convertir expresiones aritméticas a y desde código.
- Usar los Bloques de Evaluación para reflejar el orden correcto de las operaciones en una expresión.

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Guía de trabajo – Bloques de Evaluación](#)
- [Diapositivas – Lección 2 CS y álgebra](#)

Vocabulario

- **Evaluar:** producir una respuesta u observación realizando el cálculo con una expresión.
- **Bloque de Evaluación:** un bloque de código que representa la estructura de una expresión.
- **Expresión:** cualquier unidad válida de código que resuelva un valor.
- **Función:** un objeto matemático que toma ciertas entradas y produce un resultado.
- **Valor:** una porción específica de datos, como “hola” o “5”.

Estrategia de aprendizaje

Conocimiento inicial

¡Bienvenido a las Ciencias de la Computación en el álgebra, de Code.org! Además de las utilidades de programación que ya hemos visto, la programación nos puede servir para resolver problemas matemáticos o de geometría, ¡de manera más simple!

En esta unidad aplicaremos la programación a las matemáticas.

Aprender un idioma

Discutan como clase la naturaleza de un idioma. Por ejemplo: ¿qué es lo que conforma un idioma? ¿hay alguien que hable un segundo o tercer idioma? ¿hablas un idioma diferente al que hablan tus padres y tus abuelos? ¿hay idiomas que tengan características similares, como la raíz (romana, germánica) o un alfabeto similar (latín, cirílico, árabe, kanji)? ¿hay idiomas que estén diseñados para propósitos específicos o con ciertas limitaciones (lenguaje de signos, esperanto)? Bien, ¡pues las matemáticas son un idioma, igual que el español, el inglés o cualquier otro idioma!

En el español, usamos “sustantivos”, como “pan”, “tomate”, “mostaza”, y “queso” para describir objetos físicos. Las matemáticas cuentan con valores, como los números 1, 2 o 3 para describir cantidades. El español cuenta con verbos, como “tostar”, “cortar”, “cocinar” y “derretir” para describir operaciones en los sustantivos. Por otro lado, las matemáticas tienen funciones, como sumar y restar, las cuales son operaciones ejecutadas en números. Así como puedes “tostar un pan”, también se puede “sumar cuatro más cinco”.

Una expresión matemática es como una oración: es una instrucción para realizar una acción. La expresión $4+5$ nos dice que debemos sumar cuatro y cinco. Para evaluar una expresión, seguimos las instrucciones de la expresión. La expresión $4+5$ se evalúa en 9.

A veces, necesitamos varias expresiones para completar una tarea. Para preparar un sándwich, el orden de las acciones es muy importante: derretir el queso, cortar el pan, echar el ketchup, etc. De esta manera, el orden de las funciones también es importante en las matemáticas. Si alguien dice “cuatro menos dos más uno”, puede significar varias cosas.

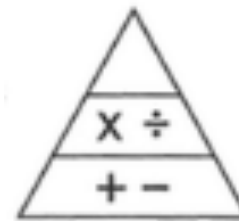
- Restar dos menos cuatro, luego sumar uno: $(2-4) + 1$
- Sumar dos más uno, y el resultado restarlo a cuatro: $4 - (2+1)$

Dependiendo de cómo leamos la expresión, podríamos tener diferentes resultados. Esto es un problema, ya que a menudo usamos las matemáticas para compartir cálculos entre nosotros. Por ejemplo, tú y tu compañía de teléfono deben llegar a un acuerdo sobre cuánto te costará enviar mensajes y hacer llamadas. Obtener resultados diferentes, en este caso, significaría que tu boleta llegaría con un monto diferente al que tú creías. Para evitar estos problemas, podemos acordar un orden para realizar las diferentes operaciones en una expresión. Hay dos formas de hacer esto:

1. Podemos acordar un orden universal

2. Podemos agregar instrucciones que indiquen el orden de las expresiones

Los matemáticos no siempre estaban de acuerdo con el orden de las operaciones. Sin embargo, actualmente tenemos una serie de reglas para evaluar expresiones. Cuando evaluamos una expresión, comenzamos resolviendo las multiplicaciones y divisiones. Sólo después de haber resuelto todas estas operaciones, podemos avanzar al siguiente nivel (suma y resta). Si están presentes las dos operaciones (como en $4 + 2 - 1$), leeremos la expresión de izquierda a derecha, resolviendo las operaciones en el orden en que aparezcan.



Los bloques de Evaluación proporcionan un método visual para indicar el orden de las operaciones en las expresiones.

Todos los Bloques de Evaluación siguen tres reglas:

- **Regla 1:** cada bloque debe tener una función, la cual será mostrada en la parte superior del bloque.
- **Regla 2:** ubicar los valores debajo de la función, de izquierda a derecha.
- **Regla 3:** si un bloque contiene otro bloque como valor, se debe evaluar primero bloque interno y luego el bloque que lo contiene.

Antes de que los estudiantes comiencen a interactuar con los computadores, puede hacerlos trabajar con la [Guía de trabajo – Bloques de Evaluación](#).

Lección – Bloques de evaluación

Desafíos en línea

El lenguaje de programación que aprenderán usa de Bloques de Evaluación para representar visualmente funciones matemáticas. Cada bloque de código es o una función o un valor.

Diríjase al [Curso A Lección 1](#) en Code Studio para comenzar a programar.

Lección 50

Cadenas (“strings”) e imágenes

Lección en línea

Propósito

Para poder calcular más que sólo números, es necesario que los estudiantes aprendan sobre dos nuevos tipos de datos: cadenas (cualquier cadena de los caracteres alfanuméricos) e imágenes. Con estos dos nuevos tipos de datos podremos componer programas que produzcan y editen imágenes.

El lenguaje de programación que usaremos en este curso es un lenguaje [fuertemente tipado](#). Esto significa que debemos identificar explícitamente los tipos de datos que usamos.

Secuencia para el aprendizaje

Conocimiento inicial

Lección: cadenas e imágenes

Objetivo

Los estudiantes serán capaces de:

- Escribir y evaluar expresiones para generar cadenas e imágenes.

Vocabulario

- **Tipo de datos:** en un lenguaje de programación, todos los valores tienen un “tipo” (como un booleano, un número o una cadena) el cual define cómo lo interpretará un computador. Por ejemplo, $7+5$ se interpreta de una forma y “7”+“5”.
- **Imagen:** un tipo de dato usado para representar gráficos e imágenes.
- **Cadena (“strings”):** cualquier secuencia de caracteres que esté entre comillas (por ejemplo, “hola”, “42”, “esto es una cadena”). En general, es utilizada para almacenar texto.

Estrategia de aprendizaje

Conocimiento inicial

Introducción

En la Lección anterior, los estudiantes trabajaron sólo con un único tipo de valor: números. En esta nueva Lección, tendrán la oportunidad de desarrollar programas con nuevos tipos de datos, de manera que obtendrán texto (cadena) e imágenes (imagen).



Muestre a los estudiantes la función “estrella” y pídale reflexionar sobre las siguientes preguntas:

- ¿Cuál es el nombre de la función?
- ¿Cuántos argumentos se dan en esta función?
- ¿Qué creen que hará esta función?

No esperamos que los estudiantes conozcan las respuestas de todas estas preguntas, el objetivo es más bien que puedan aplicar lo que ya saben sobre Bloques de Evaluación en una nueva expresión y reflexionar sobre qué creen que pueda hacer. Pídale que justifiquen sus respuestas y que expliquen por qué creen que están correctas. Anime a los estudiantes a buscar patrones en estos nuevos bloques (como el color o las comillas de las palabras “sólido” y “morado”) ¿qué podrían significar estos patrones?

Lección: cadenas e imágenes

Desafíos en línea

En esta Lección usarás los nuevos tipos de datos, cadenas e imágenes, para crear arte con los Bloques de Evaluación.

Diríjanse al [Curso A Lección 2](#) en Code Studio para empezar a programar.

Lección 51

Contratos (definición de funciones), dominio y rango

Propósito

Los contratos son un método para que los estudiantes puedan comprender y discutir sobre funciones con mayor facilidad. En esta Lección, los estudiantes analizarán funciones ya conocidas y elaborarán los contratos que describan dichas funciones.

Secuencia para el aprendizaje

Conocimiento inicial

Transferencia del conocimiento

Ampliación del conocimiento

Objetivo

Los estudiantes serán capaces de:

- Describir una función respecto a su nombre, dominio y rango.
- Crear contratos para funciones aritméticas y gráficas.

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Diapositivas – CS en algebra Lección 3](#)

Para los estudiantes:

- [Guía de trabajo – Registro de contratos](#)

Vocabulario

- **Contrato:** una declaración del nombre, dominio y rango de una función.
- **Dominio:** el tipo de datos que necesita una función.
- **Rango:** el tipo de datos que produce una función.

Estrategia de aprendizaje

Conocimiento inicial

¿Qué hay en una función?

Ya has visto varias funciones, como $+$ y $-$, que necesitan dos números. Por otro lado, la función “estrella” necesita un número y dos cadenas. Cada función necesita diferentes tipos de entradas, y nosotros necesitamos una forma de llevar registro de los requerimientos de cada función.

- ¿Qué hace la función '+'?
 - ¿Qué entradas necesita?
 - ¿Qué resultado entrega?
- ¿Y qué hay sobre la función 'triángulo'?
- ¿Qué tienen en común todas estas funciones?

Veamos una forma sencilla de describir cualquier función: los “contratos”.

- ¿Qué es un contrato?
 - Un acuerdo formal
 - La descripción de un comportamiento esperado
- ¿Qué nos dicen los contratos?
 - Lo que debe hacer una función
 - Las entradas que necesita una función
 - El resultado que nos dará una función

Anime a los estudiantes a pensar sobre cómo funcionan los contratos en el mundo real. ¿Para qué sirven? Si se firma un contrato, ¿se supone que debemos seguirlo?

Las 3 partes de un contrato

1. Nombre
2. Dominio
3. Rango

El **nombre** de una función describe brevemente lo que hace dicha función.

El **dominio** de una función son las entradas que necesita la función.

El **rango** de una función son los datos que produce la función.

Al mantener un listado de todas las funciones existentes en un lenguaje y sus dominios, los programadores pueden revisar fácilmente cómo funciona cada una. Sin embargo, también es importante llevar registro de qué produce cada función. Por ejemplo, un programa no debería usar la función “estrella” si lo que se intenta producir es un número, ya que “estrella” sólo produce imágenes.

Los dominios y rangos ayudan a los programadores a escribir códigos mejores, ya que previenen errores absurdos y les dan pistas sobre qué hacer después. Un programador que quiera usar “estrella” puede revisar el dominio y saber inmediatamente que la primera entrada debe ser un número (como 100), sin tener que recordarlo siempre. En lugar de escribir un valor simple, un programador podría escribir una expresión completa, como $(25 * 4)$. Sabremos que este código resultará en un valor acorde (en un número) al revisar el rango de $*$. Por ende, el resultado de $*$ se puede usar en lugar de cualquier valor numérico.

Cuando los programadores escriben los dominios y los rangos de cada función, escriben lo que llamamos contratos, los cuales sirven para llevar registro de lo que necesita cada función.

Transferencia del conocimiento

Mantén tus contratos

Mientras programas, asegúrate de registrar un contrato para toda nueva función que inventes o encuentres. En la siguiente Lección, aprenderás cómo crear tus propias funciones, para que puedas ahorrar tiempo escribiendo expresiones (esto resultará ser una parte esencial de un juego de escritura). Además, comenzarás a personalizar tu juego con imágenes de los elementos en tu diseño de juego.

Ampliación del conocimiento

Leer contratos

Vamos a ver algunos ejemplos de contratos: para cada contrato vamos a definir el nombre, el dominio y el rango.

- $+$: Número Número \rightarrow Número



- triángulo: Número Cadena Cadena \rightarrow Imagen



- rotar: Número Imagen \rightarrow Imagen

Algunos de los errores más comunes a la hora de escribir contratos por primera vez son: escribir valores (como “15”) en lugar de tipos (como “número”) y olvidar los argumentos. Revise cuidadosamente los contratos de los estudiantes, ya que es común que indiquen conceptos

Escribir contratos

Veamos si podemos crear contratos para algunas de las funciones que ya conoces. Deberás asegurarte de tener tu registro de contratos siempre contigo, ya que es aquí en donde mantendrás un registro actualizado de todos los contratos que escribas, tanto para las funciones existentes como para las que crees tú mismo.

- Vamos a comenzar con contratos para funciones simples de aritmética
- $+, -, *, /$

Dado que las funciones aritméticas sólo aceptan números, estos contratos son bastante fáciles. Pero cuando se trata de escribir funciones que puedan interactuar con varios tipos de datos, un vistazo a los Bloques de **Evaluación** nos podría dar algunas pistas muy útiles.

- El nombre de la función estará arriba
- Habrá un espacio para cada elemento del dominio
- El color de cada espacio nos dirá el tipo de dominio
- El color del Bloque nos dará el rango
- Códigos de color: Número Cadena Imagen

Muestre los siguientes Bloques de Evaluación y pregunte:

- ¿Cuál es el nombre de esta función?
- ¿Cuál es el dominio de esta función?
- ¿Cuál es el rango de esta función?
- Añade el contrato de esta función a tu registro de contratos



Sugerencias para evaluación

Se sugiere evaluar formativamente los aprendizajes:

- Utilizan algoritmos para ejecutar procedimientos matemáticos, realizar cálculos y obtener términos definidos por una regla o patrón.
- Pueden describir o leer cada uno de los ejemplos de contrato o función que utilizan.

Lección 52

Escribiendo contratos (definición de funciones)

Propósito

Los estudiantes avanzarán por esta Lección trabajando en varias funciones nuevas. Primero, usarán cada una para resolver un problema y luego escribirán un contrato que la describa.

Secuencia para el aprendizaje

Escribir contratos

Conocimiento inicial

Objetivo

Los estudiantes serán capaces de:

- Descomponer funciones ya existentes.
- Escribir contratos que describan funciones.
- Experimentar con transformación geométricas básicas.

Lección en línea

Vocabulario

- **Rotar:** girar una figura alrededor de un eje central.
- **Escalar:** aumentar las dimensiones de una figura por el mismo factor hacia todas las direcciones.
- **Trasladar:** mover una figura desde una ubicación a otra.

Estrategia de aprendizaje

Escribir contratos

Desafíos en línea

En esta Lección verás algunas funciones, algunas que ya conoces y otras que son completamente nuevas. Para cada función, primero tendrás la oportunidad de probarla y luego tendrás que escribir su respectivo contrato. Asegúrate de registrar todas las funciones nuevas en tu Registro de contratos.

Diríjanse al [Curso A – Lección 4](#) en Code Studio para empezar a programar.

Conocimiento inicial

Introducción

Repasen el propósito de los contratos:

- Describir tres elementos de una función
 - Nombre (como se llama la función)
 - Dominio (qué entradas necesita)
 - Rango (qué da como resultado)
- Como clase, describan los contratos de algunas operaciones matemáticas básicas:
 - Adición (nombre: +, dominio: Número Número, rango: Número)
 - Substracción (nombre: -, dominio: Número Número, rango: Número)
 - Multiplicación (nombre: *, dominio: Número Número, rango: Número)
 - Potencia de dos (nombre: sqr, dominio: Número, rango: Número)

Sugerencias para evaluación

Se sugiere evaluar formativamente los aprendizajes:

- Desarrollan y programan algoritmos para ejecutar procedimientos matemáticos, realizar cálculos y obtener términos definidos por una regla o patrón.

Lección 53

Definiendo variables y substitución

Propósito

En esta Lección, los estudiantes aprenderán a definir variables que puedan ser usadas para referirse a valores y expresiones. Una vez que estén definidas, los estudiantes podrán usarlas repetidamente a lo largo de un programa, a modo de sustituto de los valores o expresiones originales.

Como se estableció en lecciones anteriores, en este curso los estudiantes están aprendiendo un paradigma de programación llamado programación funcional, el cual podría variar respecto a otros paradigmas de programación que pudiesen haber aprendido anteriormente. Uno de los conceptos principales de la programación funcional es la inmutabilidad. Una vez que han sido creados, los objetos inmutables no cambian de estado, lo que permite que las variables funcionen como se espera que lo hagan en el álgebra.

Secuencia para el aprendizaje

Conocimiento inicial

Definir variables y substitución

Objetivo

Los estudiantes serán capaces de:

- Definir variables dándoles un nombre y asignándoles un valor o una expresión.
- Usar variables en Bloques de Evaluación.
- Describir una situación en donde usar variables para substituir valores o expresiones sea más eficiente.

Lección sin conexión

Vocabulario

- **Definir:** asociar un valor con un nombre descriptivo.
- **Variable:** una referencia a un valor o a una expresión que puede ser usada repetidamente a lo largo de un programa

Estrategia de aprendizaje

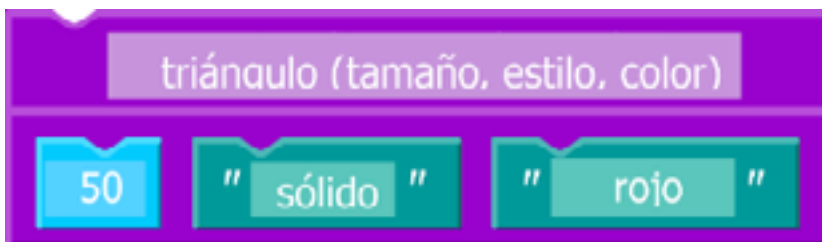
Conocimiento inicial

Introducción

Supongamos que queremos crear una imagen compuesta por setenta y cinco triángulos rojos idénticos. ¡Para eso, tendrías que crear este Bloque de Evaluación setenta y cinco veces!

Aún peor, si de pronto decides que quieres que los setenta y cinco triángulos sean azules, tendrías que revisarlos todos y cambiarlos uno por uno. ¡Debe haber una forma más sencilla!

Podemos almacenar este Bloque de Evaluación en una variable, la cual llamaremos “triángulo rojo”. Este nombre se convertirá en un atajo para llegar al bloque almacenado en esa variable, y podemos usar este atajo cuantas veces queramos a lo largo de nuestro programa. Si de pronto queremos que el triángulo mida 100 píxeles en lugar de 50, lo único que debemos hacer es cambiar la definición de la variable.



Definir variables y sustitución

Desafíos en línea

En esta Lección, usarás variables para referenciar diferentes valores y expresiones.

Diríjase al [Curso A – Lección 5](#) en Code Studio para empezar a programar.

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Desarrollan y programan algoritmos para ejecutar procedimientos matemáticos, realizar cálculos y obtener términos definidos por una regla o patrón.

Si los estudiantes han utilizado variables en otros lenguajes de programación, es esencial tener en cuenta que, en la programación funcional, al igual que en las matemáticas, las variables se consideran inmutables, lo que significa que el valor no puede cambiarse durante la ejecución de un programa. Piénsalo de esta manera: decir $x = 50$, y luego $x = x + 1$ puede tener sentido en Javascript, pero es imposible en álgebra.

Lección 54

Funciones rápidas

Propósito

En esta Lección, reforzaremos la comprensión de las variables desarrollando funciones que reduzcan la cantidad de entradas necesarias. Con estas “funciones rápidas”, los estudiantes podrán practicar cómo usar la Receta de Diseño para desarrollar funciones simples, evitando todas las complicaciones de los problemas verbales.

Secuencia para el aprendizaje

Conocimiento inicial

Lección en grupo (15 minutos)

Lección en grupo pequeño (20 min)

Lección sin conexión

Vocabulario

- **Función:** un objeto matemático que toma ciertas entradas y produce un resultado.
- **Variable independiente:** una variable de entrada controlada por el usuario.
- **Parámetro:** un valor o una expresión perteneciente al dominio de una función.
- **Variable:** una referencia a un valor o a una expresión que puede ser usada repetidamente a lo largo de un programa.

Estrategia de aprendizaje

Conocimiento inicial

Introducción

En este lenguaje de programación, las variables son inmutables. Esto significa que, en cuanto se definen, no se pueden cambiar nunca más. Esto nos permite utilizar las variables para recurrir a un valor exacto las veces que queramos. Sin embargo, ¿qué tal si queremos aplicar un ligero cambio en dicho valor, cada vez que se use?

Supongamos que queremos crear setenta y cinco triángulos rojos, pero de diferentes tamaños. Entonces, deberíamos crear setenta y cinco funciones diferentes, una para cada triángulo. Frustrante, ¿no? ¡Debe existir una forma más sencilla de hacerlo!

Podemos almacenar el Bloque de Evaluación del triángulo en una función, la cual llamaremos triángulo-rojo. Ahora, triángulo-rojo le preguntará al usuario de qué tamaño quiere crear los triángulos, manteniendo siempre la forma triangular y el color rojo. De esta manera, se reducirá la cantidad de entradas que requiere la función.

Esto es muy similar a una variable, ya que estamos substituyendo una función por otra. La mayor diferencia es que una función puede recibir información del usuario. Las variables sólo se repiten, tal cual son, y cualquier cambio hecho dentro del código afectará todas las instancias en que se use dicha variable. Ahora tenemos la capacidad de preguntarle al usuario por los aspectos que queremos modificar y traspasar esa información a tantas funciones como queramos.

Lección en grupo (15 min)

En esta Lección, necesitará crear algunas historias para hacer que los estudiantes se involucren más y más con las razones para usar funciones rápidas.

En conjunto como clase

Defina la función círculo-azul, la cual toma un tamaño y produce un círculo sólido de color azul, en la guía de trabajo de funciones rápidas.

Pregunte:

- ¿Sabes cuál es mi forma favorita? ¡me encantan los círculos azules! Me gustan los círculos pequeños, los grandes, del tamaño que sean. Me gustaría crear una polera que tenga cientos de círculos azules. ¿Podrían ayudarme?
- ¿Podemos usar lo que ya sabemos sobre las variables? Pero... ¿qué tal si quiero que los tamaños varíen? La función del círculo es demasiado trabajo, nadie tiene tanto tiempo como para ingresar

tantas entradas. ¿Podríamos crear una función?...

Los estudiantes han estado trabajando en contratos y en definiciones básicas. Sin embargo, ahora les pedimos algo un poco más complejo y abstracto. Asegúrese de bajar la intensidad del proceso y centrarse en “lo que DEBE hacer la función”. Crear EJEMPLOS es muy importante: fuerza a los programadores a ir más lento, pero les permite asegurarse de que la función haga lo que se supone que deba hacer.

Cómo crear EJEMPLOS:

- Extraiga el nombre de la función del contrato y pregunte POR QUÉ ese debería ser el nombre de la función.
- Pregunte qué tipo de entradas requiere esta función. Asegúrese de pedirles que justifiquen sus respuestas con datos del contrato.
- Pregunte qué tipo de resultado debería PRODUCIR esta función. Muchos estudiantes van a preferir hacer el cálculo ellos mismos en lugar de dejar que la función haga lo suyo.
 - o Para círculo-azul, recuerde a los estudiantes que una función no puede recurrir a si misma; en efecto, eso sería tan absurdo como llamarse por teléfono a uno mismo. Círculo-azul necesita recurrir a otra función que le ayude a construirse (necesita a círculo, la cual está en el Registro de Contratos).
- Cree el segundo EJEMPLO de la misma manera.
- Revíselos pieza por pieza y encierre en un círculo los CAMBIOS, marcándolos como variables.

Cómo crear una DEFINICIÓN:

- Revise y descomponga cada parte que conforma el ejemplo.
- Anote el nombre de la variable encontrada en el ejemplo (el cambio que fue encerrado en un círculo).
- Anote la función a la que recurre círculo-azul y las entradas que requiere, desde el ejemplo. Asegúrese de reemplazar con una variable cualquier elemento que cambie.

Lección en grupos pequeños (20 min)

Grupos de trabajo

Divida a los estudiantes en grupos de tres. Cada miembro del grupo va a representar un paso de las funciones rápidas.

1. Contrato
2. Ejemplos
3. Función

La tarea será crear las siguientes funciones rápidas:

Definir una función punto, la cual solicita un color y produce un círculo sólido con radio 50, relleno con el color solicitado.

Definir una función promedio, la cual solicita dos números y produce un promedio de dichos números (quizás necesite recordarles que, para conseguir un promedio, se debe sumar ambos números y luego dividirlos por dos).

Supongamos que un logotipo de una compañía es una palabra escrita con letras grandes y rojas, girado una cierta cantidad de grados. Definir una función logotipo, la cual solicita el nombre de una compañía y la rotación, y produce un logotipo para dicha compañía.

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Crean variables con nombres claros que representen diferentes tipos de datos y realice operaciones

Los estudiantes están fortaleciendo sus conocimientos, hasta que logren dominar la [Receta de Diseño](#). La diferencia entre las funciones rápidas y la Receta de Diseño es sutil; sin embargo, es importante que, en esta Lección, los estudiantes se concentren y dominen el cómo crear ejemplos.

Lección 55

Funciones compuestas

Propósito

En las lecciones anteriores, los estudiantes han definido variables y desarrollado funciones rápidas. En esta ocasión, seguirán explorando las funciones añadiéndoles cada vez más complejidad.

Secuencia para el aprendizaje

Conocimiento inicial

Funciones compuestas

Objetivos

Los estudiantes serán capaces de:

- Analizar y utilizar funciones existentes.
- Modificar funciones existentes.
- Crear nuevas funciones.
- Crear formas similares, cambiando los parámetros de tamaño en las funciones.

Lección en línea

Vocabulario

- **Parámetro:** un valor o expresión perteneciente al dominio de una función.

Estrategia de aprendizaje

Conocimiento inicial

Introducción

En las actividades anteriores, los estudiantes comenzaron a componer funciones para hacer que las cosas fuesen más sencillas. En esta Lección, los estudiantes podrán profundizar más en sus conocimientos.

En los últimos niveles encontrará aspectos evaluativos, por lo que asegúrese de hacer preguntas formativas para que los estudiantes reflexionen más a fondo sobre su trabajo.

Desde el nivel 1 hasta el nivel 5: se trata de funciones para componer imágenes. Atención con estrella-agua, ya que aquí es donde la mayoría de los estudiantes tienen problemas para traspasar sus conocimientos. Haga una pausa en el nivel 4 para asegurarse de que todos los estudiantes hayan comprendido con certeza.

Desde el nivel 6 hasta el nivel 9: volvemos a las matemáticas. Es crucial que los estudiantes sigan usando el proceso de función rápida en estos problemas, ya que esto les ayudará a que esta Receta de Diseño sea más aplicable a problemas algebraicos en el futuro.

Niveles evaluativos, desde el nivel 10 hasta el nivel 15: esta es una gran oportunidad para que los estudiantes pongan en práctica sus conocimientos. Estos niveles no son obligatorios.

Esta Lección es total y plenamente flexible respecto a los niveles que usted, como profesor, desee realizar con sus estudiantes.

Funciones compuestas

Desafíos en línea

En esta Lección, vas a definir funciones simples.

Diríjase al Curso A – Lección 7 en Code Studio para empezar a programar.

Asegúrese de que los estudiantes tengan suficiente espacio libre en sus guías de funciones rápidas (creadas en la Lección anterior), de manera que puedan construir el contrato, los ejemplos y la definición de estas funciones nuevas. ¡Un gran código comienza primero en papel!

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Crean ecuaciones, utilizando e interpretando funciones
- Utilizan Modelos lineales, cuadráticos y exponenciales

Lección 56

La Receta de Diseño

Propósito

En la última Lección, los estudiantes escribieron algunas funciones muy simples. Sin embargo, funciones más sofisticadas requieren un enfoque mucho más minucioso. La Receta de Diseño es un enfoque estructurado para escribir funciones, el cual implica escribir una declaración de propósito y hacer pruebas para asegurarse de que la función dé el resultado deseado. Una vez que los estudiantes hayan dominado el proceso de la Receta de Diseño, podrán aplicarlo a cualquier problema matemático con el que se encuentren.

El proceso de la Receta de Diseño para escribir funciones les ofrece a los estudiantes las mejores técnicas en diseño de software, técnicas que no son enseñadas hasta la universidad. Al escribir ejemplos (o como se dice en el mundo de la programación, “probar casos”) antes de escribir el cuerpo de la función, los estudiantes están siguiendo una práctica llamada desarrollo guiado por pruebas (TDD, por sus siglas en inglés). En TDD, los desarrolladores de softwares escriben casos de prueba fallidos antes de añadir características nuevas, y luego trabajan en su código hasta que el caso de prueba sea exitoso. De esta manera, los desarrolladores se aseguran de escribir códigos que hagan lo que se propusieron, a la vez que realizan pruebas automatizadas para garantizar que el programa siga funcionando como se espera, mientras se añaden más funciones.

Secuencia para el aprendizaje

Conocimiento inicial

Ampliación del conocimiento

Objetivos

Los estudiantes serán capaces de:

- Usar la Receta de Diseño para identificar variables dependientes, independientes y constantes.

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Diapositivas – Lección 8 – La Receta de Diseño](#)

Para los estudiantes:

- [Guía de trabajo – Funciones rápidas](#)
- [Guía de trabajo – Formulario de la Receta de Diseño](#)

Vocabulario

- **Constante:** un número fijo en una relación.
- **Variable dependiente:** el valor que cambia en base a la variable independiente. El resultado.
- **Receta de Diseño:** una secuencia de pasos para documentar, probar y escribir funciones.
- **Variable independiente:** el valor que controla el usuario. Las entradas.

Estrategia de aprendizaje

Conocimiento inicial

¿Qué es la Receta de Diseño?

La receta de diseño es un mapa para definir funciones, el cual usan los programadores para asegurarse de que el código que escriban haga lo que ellos quieren que haga. Cada paso se basa en el anterior, por lo que es posible detectar cualquier error muy temprano en el proceso. Este mapa tiene varios pasos:

1. Escribir un contrato que describa el problema verbal.
 - Escribir una declaración de propósito para descomponer el problema y aislar la función.
2. Escribir ejemplos basados en el contrato.
3. Definir una función que calce con los ejemplos.

Vamos a comenzar a aplicar la Receta de Diseño juntos. Revisemos el siguiente problema:

Define la función “estrella-violeta”, que requiera el tamaño de la estrella y reproduzca el contorno de una estrella violeta del tamaño dado.

Paso 1 – El contrato

Estrella-violeta: Número-> Imagen

Asegúrate de incluir un buen nombre para cada función, y recuerda que el dominio y el rango sólo pueden incluir tipos como números, imágenes, cadenas, etc.

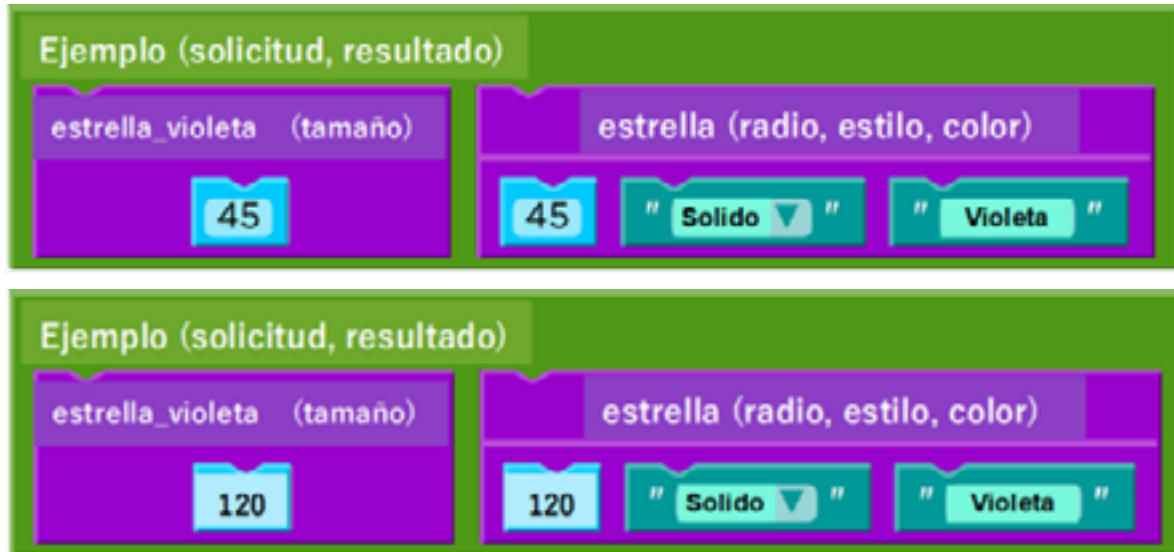
Un contrato contiene las bases de una función, aquello que les da a los programadores la información necesaria para usarla: el nombre de la función, los tipos de datos que recibe y los tipos de datos que entrega.

La declaración de propósito les da a los estudiantes una idea de lo que la función deba hacer en el siguiente paso, por ejemplo.

Haga una pausa para analizar las declaraciones de propósito para una mayor comprensión. En este paso, es muy importante que los estudiantes tengan muy claro qué debe hacer la función, incluso si aún no tienen idea de qué función necesitarán usar.

- **Declaración de propósito:** una descripción breve de lo que hace una función.

Paso 2 – Ejemplos



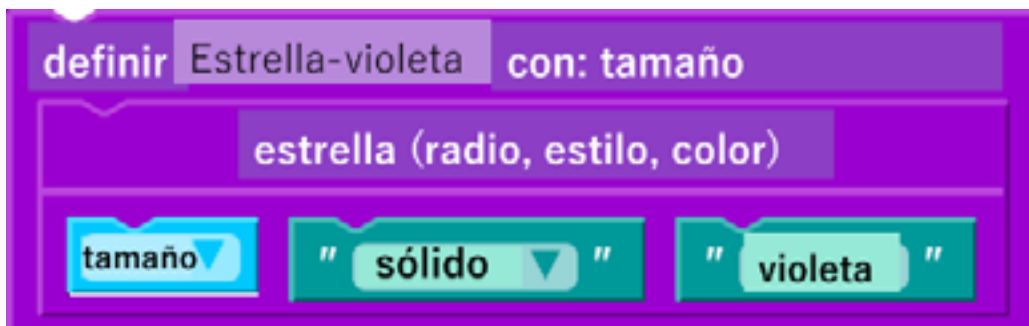
- Cada ejemplo debe comenzar con el nombre de la función. ¿Dónde puedes encontrar el nombre de la función? (en el contrato).
- Cada ejemplo debe incluir entradas de ejemplo. ¿Dónde puedes encontrar cuántas entradas necesita esta función y de qué tipo? (en el contrato).
- Cada ejemplo debe incluir una expresión que establezca qué debe hacer la función una vez que se le aplique una entrada. ¿Dónde puedes encontrar qué hace la función? (en la declaración de propósito)

Una vez que tengas dos o más ejemplos, será sencillo identificar qué es lo que ha cambiado entre ellos. En efecto, la cantidad de elementos que hayan cambiado debe calzar con la cantidad de elementos en el dominio de la función: si el dominio tiene un número y una cadena en él, entonces estos dos valores son los que deben diferir entre tus ejemplos.

Respecto al cambio que hayas encerrado en un círculo, asegúrate de marcarlo como una variable que tenga sentido con este problema antes de pasar al siguiente paso, la definición.

Paso 3 – Definición de función

Al identificar qué es lo que ha cambiado entre estos dos ejemplos, podemos definir la función.



Recuerda que se pueden usar el contrato y la declaración de propósito para escribir los ejemplos, incluso si el programador no tiene claro cómo comenzar. Luego, se usan los ejemplos para desarrollar la función, incluso si el programador no está seguro de qué hacer.

Ampliación del conocimiento

La batalla de la Receta de Diseño

En esta Lección, los estudiantes se reunirán en parejas y analizarán la Receta de Diseño para resolver los siguientes problemas.

Se espera que los estudiantes puedan defender cada decisión que tomen en la Receta de Diseño, usando exclusivamente la evidencia de la sección anterior.

La batalla de grupos

- Usted, como profesor, tendrá que modelar cómo funcionan las batallas. Complete una Receta de Diseño frente a toda la clase, una sección a la vez.
- Deles a los estudiantes 2 minutos para completar el contrato y la declaración de propósito.
- Cuestione cada aspecto del contrato y de la declaración de propósito, durante 1 minuto, siendo tan específico como sea posible. Los estudiantes deberían ser capaces de responder a todas sus preguntas sólo con la información de la descripción o con el problema verbal del problema original.
 - Ejemplo: ¿en dónde encontramos el nombre de la función? ¿en dónde encontramos número para el dominio? ¿cómo supieron que número es el supuesto rango? ¿en dónde encontraron la función en su declaración de propósito?
- Permita que los estudiantes repasen la descripción o el problema verbal en donde surgió el problema original. Ahora, deben crear sus ejemplos usando sólo el contrato y la declaración de propósito.
- Deles 2 minutos para completen ambos ejemplos.
- Durante 1 minuto, cuestione cada aspecto de los ejemplos, siendo tan específico como sea posible.
 - Ejemplo: ¿en dónde encontraron el nombre de la función? ¿por qué escogieron ## como su entrada? ¿Cómo llego a ese resultado? ¿quién hizo el trabajo para encontrar el resultado, ustedes o el computador? ¿por qué hicieron que sus ejemplos fuesen diferentes? ¿cómo va a saber el computador en dónde poner la variable?
- Permita que los estudiantes repasen el contrato y la declaración de propósito.
- Deles a los estudiantes 1 minuto para definir sus funciones.
- Durante 1 minuto, cuestione cada aspecto de la definición, siendo tan específico como sea posible.
 - Ejemplo: ¿en dónde encontraron la variable? ¿cómo sabe el computador qué hacer con las entradas? ¿cómo saben que la función se comportará como ustedes esperan que se comporte?

- Han luchado exitosamente ¡Felicidades!
- Para incentivar más a los estudiantes, puede imprimir una lista de preguntas y pedirles a los estudiantes que las respondan y que anoten cuántas respuestas correctas tiene cada uno.

Conductor – Batalla de navegación

Ahora que los estudiantes han visto el tipo y la intensidad de sus preguntas, están listos para luchar entre ellos.

Es de gran ayuda darles a los conductores (estudiantes A) un problema, y a los navegadores (estudiantes B) un problema diferente.

Como profesor, su prioridad será mantenerlos motivados y con energía, los estudiantes harán el resto:

- 2 minutos para completar el contrato y la declaración de propósito.
- 1 minuto (o 30 segundos, dependiendo de los estudiantes) para que los conductores luchen (hacerse preguntas)
- 1 minuto (o 30 segundos, dependiendo de los estudiantes) para que los navegadores luchen (hacerse preguntas)

Permita que la clase tome un respiro. Vuelvan a concentrarse y repasen el contrato y la declaración de propósito.

- 2 minutos para definir la función.
- 1 minuto (o 30 segundos, dependiendo de los estudiantes) para que los conductores luchen (hacerse preguntas)
- 1 minuto (o 30 segundos, dependiendo de los estudiantes) para que los navegadores luchen (hacerse preguntas)

Es importante tener en cuenta que los estudiantes cometerán muchos errores y que tendrán que hacer varias revisiones a lo largo de las batallas. Una buena idea sería realizar 2 a 3 batallas de diferentes problemas y luego rotar las parejas (al menos por el día).

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Elaboran representaciones, tanto en forma manual como digital, y justifican cómo una misma información puede ser utilizada según el tipo de representación.

Lección 57

Resolver problemas con la Receta de Diseño

Lección en línea

Propósito

En esta Lección, los estudiantes practicarán cómo usar la Receta de Diseño para escribir funciones que puedan resolver problemas. Para el final de la Lección, los estudiantes deberían estar listos para aplicar la Receta de Diseño en problemas de las clases de matemáticas.

Secuencia para el aprendizaje

Conocimiento inicial

Ampliación del conocimiento (45 min)

Objetivos

Los estudiantes serán capaces de:

- Diseñar funciones para resolver problemas verbales.
- Continuar practicando la escritura de contratos en escenarios más complejos.

Preparación

- Imprima copias de la Guía de trabajo – Formulario de la Receta de Diseño.
- Revise la Plantilla – Receta de Diseño – Problema verbal

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Plantilla – Receta de Diseño – Problema verbal](#)

Para los estudiantes:

- [Guía de trabajo – Formulario de la Receta de Diseño](#)

Estrategia de aprendizaje

Conocimiento inicial

Introducción

Los estudiantes deberán volverse expertos en el análisis para encontrar el error en estas Recetas de Diseño incompletas o “buggeadas” (que presentan bugs).

Se debe tomar en cuenta que se deben llenar los ejemplos por completo. Cuando un ejemplo este incompleto, aparecerá un mensaje de error que dirá “tienes un bloque con una entrada sin llenar”.

Ampliación del conocimiento (45 min)

Desafíos en línea: depuración

En esta Lección, usarán la Receta de Diseño para analizar y depurar funciones rotas.

Diríjense al [Curso A – Lección 9](#) en Code Studio para empezar a programar.

Antes de que los estudiantes se sumerjan en esta Lección, el profesor debe plantear cómo usar las habilidades en el uso de la Receta de Diseño para depurar. Es recomendable realizar el primer desafío en conjunto, mediante el uso del proyector, un monitor o de la pizarra. Luego, los estudiantes pueden intentar completar los desafíos 2, 3, 4 y 5.

Con Formularios en blanco de la Receta de Diseño:

- Los estudiantes abrirán el [Nivel 1 de la Lección 9](#) y completarán el formulario de la Receta de Diseño que se les proporcionó, tanto como puedan (pueden rellenar un formulario por pareja).
- Mientras los estudiantes completan sus formularios, pregunte "¿Qué falta?" y "¿Qué necesitaríamos para que esto funcione?".
- Una vez que los estudiantes creen haber resuelto el error en el papel, anímelos a enfrentar a su compañero para pulir su trabajo.
- Una vez que se hayan enfrentado entre sí, pídeles que suban su trabajo a Code Studio y que vean cómo funcionan sus funciones.

Asegúrese de hacer una pausa en la [lección 9, nivel 5](#) y reagrupar

Cuidado:

Es probable que los estudiantes quieran sólo arrastrar y soltar para rellenar los espacios en blanco, buscando soluciones rápidas sin analizar nada. Aliente a los estudiantes a desafiar a sus compañeros y a producir una versión completa de cada receta “buggeada” en el formulario.

a los estudiantes. Pídales que compartan su proceso de depuración y destaque los trucos o consejos que los estudiantes utilizaron para ayudarles a encontrar todos los errores.

El [nivel 6](#) pasa de las funciones rotas de los problemas de palabras a las funciones en blanco y desconocidas. Este paso es muy útil, pero si se pasa por alto, los estudiantes podrían perderse.

En este punto de transición, reitere a los estudiantes que, aunque los problemas parezcan diferentes, el mismo proceso que han usado funcionará de todas formas. Haga que los estudiantes creen sus funciones utilizando los formularios de la Receta de Diseño en blanco, uno por pareja.

Experiencias de aprendizaje de profundización

Use estos Contenidos para ampliar el aprendizaje de los estudiantes. Se pueden usar como Contenidos extras fuera del aula.

En esta Lección, se deben plantear problemas a partir del libro de texto o de los materiales del aula, a fin de poder utilizar la Receta de Diseño. La creación de funciones es un criterio importante en Álgebra 1, pero hay una cantidad limitada de herramientas para ayudar a los estudiantes a tomar una descripción, una tabla, puntos o una gráfica y crear la función. Muchas veces se asume que "Crear una ecuación" es obvio, pero, lamentablemente, para muchos estudiantes no lo es. Utilizando esta plantilla y sus propios problemas, ayudará a los estudiantes a transferir su pensamiento analítico de las Ciencias de la Computación a sólo Álgebra.

En el juego libre, Nivel 10, haga que los estudiantes usen la Receta de Diseño para construir funciones que resuelvan problemas de su libro de texto. Puede crear una hoja de trabajo con los problemas para los que quiera crear una función o que, simplemente, intenten usar una.

La Plantilla – Receta de Diseño – Problema verbal cuenta con 5 problemas de ejemplo, los cuales fueron modificados a partir del libro del profesor o de un libro de texto, para darles a los estudiantes la oportunidad de crear sus propias funciones.

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Elaboran representaciones, tanto en forma manual como digital, y justificar cómo una misma información puede ser utilizada según el tipo de representación. (OA g)

Lección 58

La altura del cohete

Propósito

Aplicando la Receta de Diseño, los estudiantes resolverán una serie de problemas relacionados a calcular la altura de un cohete tras una cantidad de segundos dada desde el lanzamiento. Las funciones que escriban se usarán para animar el despegue del cohete.

Secuencia para el aprendizaje

Conocimiento inicial

Ampliación del conocimiento

Objetivos

Los estudiantes serán capaces de:

- Diseñar funciones para resolver problemas.
- Usar la Receta de Diseño para escribir contratos, probar casos y definir funciones.

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Guía de trabajo – La altura del cohete](#)

Estrategia de aprendizaje

Conocimiento inicial

Introducción

Las funciones son una pieza clave en los programas de animación. Por ejemplo, un programa que haga un dibujo estático de un murciélago puede colocar dicho murciélago en cualquier parte, dependiendo de la entrada que se ingrese. Cuando esa entrada cambie ligeramente a causa del tiempo o de la interacción del usuario, parecerá que el murciélago se mueve. Funciona de una forma similar a las animaciones de folioscopio, en donde cada página tiene dibujada una imagen estática que presenta un leve cambio respecto a la anterior. Al mostrar las páginas rápidamente, una detrás de otra, parecerá que la imagen se mueve suavemente.

Si ponemos todas estas imágenes juntas, conseguiremos una animación del murciélago dando una vuelta.



En los desafíos en línea, los estudiantes se encontrarán con una caja de funciones que no tienen ningún parámetro de entrada, lo que representa la función como un tipo de datos. Esto le permite pasar su función a la función de inicio, donde se puede utilizar para controlar la animación del cohete. Al pasar la función de altura del cohete al motor podemos ejecutarla repetidamente con nuevos números a medida que pasa cada segundo en lugar de ejecutarla con un valor específico. La ejecución periódica crea el efecto de animación. A medida que pasa cada segundo, la función altura del cohete se ejecuta de nuevo, se calcula la nueva ubicación y se vuelve a dibujar el cohete en su nueva ubicación. Este dibujo y rediseño en diferentes ubicaciones da la apariencia de movimiento.

Ampliación del conocimiento

Desafíos en línea

En esta Lección, escribirán funciones que manipulen imágenes para crear animaciones.

Diríjense al [Curso A – Lección 10](#) en Code Studio para empezar a programar.

Animación no lineal

El último desafío de esta Lección es un desafío libre que les permitirá a usted y a sus estudiantes experimentar con otras variaciones de la fórmula del cohete. Los estudiantes encuentran particularmente interesante (y a menudo desafiante) la Lección de escribir funciones que produzcan aceleración no lineal. Si los estudiantes están familiarizados con las cuadráticas, puede denominar a esta Lección como tal. Sin embargo, quienes no estén familiarizados con ellas también pueden disfrutar de este desafío.

Coloque las siguientes tablas de entrada/salida en la pizarra y vea si los estudiantes pueden crear las funciones que produzcan la animación apropiada.

Después de crear un movimiento lineal simple, se les pedirá a los estudiantes que escriban funciones para animar una aceleración simple. Se les dará una tabla de entrada/salida para que escriban su nueva función. Es recomendable que realiza esta Lección en conjunto como clase, de manera que los estudiantes puedan ver cómo analiza una tabla de entrada/salida, a fin de entender la relación entre los valores de entrada y de salida.

Desafío 1	
Entrada	Salida
1	10
2	40
3	90

Desafío 2	
Entrada	Salida
1	15
2	45
3	95

Una vez que los estudiantes hayan calculado las tablas de entrada y salida, anímelos a idear sus propias funciones de animación no lineal.

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Desarrollan y programan algoritmos para ejecutar procedimientos matemáticos, realizar cálculos y obtener términos definidos por una regla o patrón.

Contenido 2

Datos por todas partes

Resumen

- **Lección 59:** Big data
- **Lección 60:** Mi video en Youtube
- **Lección 61:** Infografía de la clase I
- **Lección 62:** Dónde viven y viajan los datos
- **Lección 63:** Infografía de la clase II

Objetivos

- **OA 4.** Crear aplicaciones y realizar análisis mediante procesadores simbólicos, de geometría dinámica y de análisis estadístico.
- **OA g.** Elaborar representaciones, tanto en forma manual como digital, y justificar cómo una misma información puede ser utilizada según el tipo de representación.

Lección 59

Big data

Propósito

La gran cantidad de datos que se encuentra en la red es tan vasta que no es posible cuantificarla. Incluso, en ocasiones, tampoco es posible procesar toda esta información con programas procesadores de datos tradicionales.

En esta Lección, los estudiantes conocerán más sobre qué es el big data y cómo es posible lidiar con todos estos datos disponibles en la red, cuyo volumen crece a pasos agigantados sin ser conscientes de esto.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (10 min)

Transferencia del conocimiento (30 min)

Objetivos

Los estudiantes serán capaces de:

- Ser conscientes de la multitud de datos que generamos diariamente cuando usamos dispositivos e internet.
- Conocer qué es el big data o la revolución de los datos masivos.
- Identificar las ideas clave de videos e infografías.
- Trabajar en grupo de forma cooperativa.

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- Video – [Datos por todas partes](#)
- Infografía – [Las increíbles cifras de internet](#)
- Infografía interactiva – [Los datos también pesan](#)
- Guía de trabajo – Mi titular

Vocabulario

- **Big data:** conjunto grande de datos.

Preparación

- (Opcional) vea el video [Datos por todas partes](#)
- Tener preparada y lista para presentar [infografía interactiva](#).
- Imprima [una infografía](#) y guía de trabajo para cada estudiante.
- Asegúrese de que cada estudiante tenga su [Diario de apuntes](#).

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Presentación del programa educativo

¿Qué sabemos del big data? ¿Es importante para nuestro futuro personal y laboral? ¿Tiene algo que ver con nosotros? ¿Nos gusta, no nos gusta, nos encanta? Presentación del programa BigData y de la guía de trabajo Producto final.

Ampliación del conocimiento (10 min)

Video

Visualización con toda la clase junta del video Datos por todas partes, que sirve como introducción a la temática del programa.

Transferencia del conocimiento (30 min)

Desarrollen la guía de trabajo Mi titular.

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Debaten sobre la importancia de evitar el sesgo de muestreo y el papel que juegan las personas en el resultado del modelo de aprendizaje automático.

Lección 60

Mi video en YouTube

Propósito

El sitio web YouTube es por definición uno de los sitios web que está presente en la vida de las personas, ya sea de forma consciente o inconsciente, puesto que visualizamos videos en YouTube para entretenernos, aprender, hacer deporte, entre otras actividades.

En esta Lección, los estudiantes recrearan la acción de subir un video en YouTube para poder determinar el impacto que este podría tener.

Secuencia para el aprendizaje

Conocimiento inicial (25 min)

Ampliación del conocimiento (10 min)

Transferencia del conocimiento (15 min)

Objetivos

Los estudiantes serán capaces de:

- Identificar los tipos de datos que generamos.
- Identificar cómo los datos que generamos pueden usarse para obtener otras informaciones y servicios.
- Trabajar en grupo de forma cooperativa.

Preparación

- Imprima una copia de cada píldora informativa y la guía de trabajo para cada estudiante.
- Asegúrese de que cada estudiante tenga su Bitácora de trabajo.

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- Píldora informativa – [¿Qué son los datos?](#)
- Píldora informativa – [¿Qué son los metadatos?](#)
- Píldora informativa – [Los datos contra el virus H1N1](#)
- Guía de trabajo – Mi video en YouTube

Vocabulario

- **YouTube:** sitio web usado para compartir videos.
- **Streaming:** conexión directa de contenido de video y audio de un dispositivo a otro.

Estrategia de aprendizaje

Conocimiento inicial (25 min)

Dinámica – Guía de trabajo Mi video en YouTube

Se desarrolla grupalmente usando la guía Mi video en YouTube y consultando las píldoras de información.

Cada grupo se pone en el supuesto de tener que subir un video a YouTube y, a partir de esa acción, deben identificar:

- ¿Qué datos cuantitativos podría obtener una empresa del video que acaban de subir?
- ¿Qué datos cualitativos podrían obtener?
- ¿Qué metadatos del video necesitaría si una empresa quisiera crear listas de reproducción de videos de YouTube a partir de las preferencias marcadas por el usuario?

Ampliación del conocimiento (10 min)

Puesta en común

Puesta en común de las respuestas de cada grupo de trabajo.

Transferencia del conocimiento (15 min)

Ideando nuevos servicios

Brainstorming de toda la clase junta, donde los estudiantes idean y comparten otras posibilidades de servicios que se podrían crear a partir de los datos que genera subir un video a YouTube. Por ejemplo:

El servicio de YouTube TV, que por una suscripción anual te permite ver los videos en directo de varias cadenas de televisión de pago a través del streaming (conexión directa de contenido de video y audio de un dispositivo a otro), o un servicio que a partir de poner un grupo o un cantante generase una lista de reproducción descargable en mp4 (archivos de audio y video) que contenga sus últimos videoclips oficiales.

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?

- ¿Cómo te sentiste durante la Lección?

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Debaten sobre la importancia de evitar el sesgo de muestreo y el papel que juegan las personas en el resultado del modelo de aprendizaje automático.

Lección 61

Infografía de la clase I

Propósito

Esta Lección busca que el estudiante ponga en práctica lo estudiado en las dos últimas actividades.

Secuencia para el aprendizaje

Conocimiento inicial (15 min)

Ampliación del conocimiento (15 min)

Transferencia del conocimiento (20 min)

Objetivos

Los estudiantes serán capaces de:

- Aproximarse al concepto de big data de forma práctica, comprobando la utilidad del almacenado, clasificación y análisis de datos.
- Realizar una recogida de datos cuantitativos.
- Calcular medias de acciones diarias.
- Combinar y comparar datos.
- Visualizar e interpretar datos mediante gráficos e infografías.

Preparación

- Imprima una [píldora informativa](#) para cada estudiante.
- Asegúrese de que cada estudiante tenga su [Diario de apuntes](#).

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- Píldora informativa – [Big data: cuando los datos se vuelven inteligentes](#)
- Documento para el docente – Datos por todas partes

Vocabulario

- **Big data:** conjunto grande de datos.

Estrategia de aprendizaje

Conocimiento inicial (15 min)

El concepto de big data

Individualmente, por parejas o por equipos, los estudiantes leen la píldora informativa [Big data: cuando los datos se vuelven inteligentes](#) para aproximarse al concepto de inteligencia de datos, con el que experimentarán en los pasos siguientes.

Ampliación del conocimiento (15 min)

Introducción de datos

Individualmente, los estudiantes deberán completar una pequeña base de datos simple que les ayudará a crear una infografía a partir de datos estadísticos simples.

Transferencia del conocimiento (20 min)

Visualización e interpretación de la infografía

Se visiona la gráfica resultante de los datos introducidos. Seguidamente, y a mano alzada, se pueden comparar y compartir los datos y comentar si ha habido algún dato que les haya sorprendido y por qué. La comparación permitirá ver en qué acciones cada estudiante está por encima o por debajo de la media de la clase.

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Recopilan datos utilizando herramientas computacionales y los transforman para hacerlos más útiles y confiables.

Lección 62

Dónde viven y viajan los datos

Propósito

Alguna vez se han preguntado dónde se alojan los datos, y cómo estos viajan desde un lugar determinado de la red hacia nuestras propias computadoras y viceversa.

En esta Lección, se espera que los estudiantes se adentren, en cuanto a conocimiento, en el mundo de los datos y sus particulares características.

Secuencia para el aprendizaje

Conocimiento inicial (15 min)

Ampliación del conocimiento (15 min)

Transferencia del conocimiento (20 min)

Objetivos

Los estudiantes serán capaces de:

- Conocer cómo viajan y se almacenan los datos.

Preparación

- (Opcional) vea el video [¿Dónde viven los datos?](#)
- Tener preparada y lista para presentar infografía interactiva [La red que soporta la red](#)
- Imprima una infografía [¿Cómo viajan los datos?](#) y guía de trabajo Antes pensaba, ahora pienso para cada estudiante.
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- Video – [¿Dónde viven los datos?](#)
- Infografía – [La red que soporta la red](#)
- Infografía interactiva – [¿Cómo viajan los datos?](#)
- Guía de trabajo – Antes pensaba, ahora pienso

Vocabulario

- **Datos:** secuencia de elementos que en informática se transforma en información.

Estrategia de aprendizaje

Conocimiento inicial (15 min)

Primera parte de “antes pensaba, ahora pienso”

De forma individual, los estudiantes deben contestar, a partir de sus conocimientos previos, a las preguntas de la guía de la destreza de pensamiento:

- ¿Dónde crees que se almacenan los datos que introdujiste en la sesión anterior?
- ¿Cómo crees que han viajado hasta allí?
- Escribe tres repercusiones medioambientales que crees que tiene el transporte y almacenaje de datos.

Ampliación del conocimiento (15 min)

Visualización del video e infografías

Los estudiantes deben visionar de forma grupal el video ¿Dónde viajan los datos? Y consultar individualmente la infografía La red que soporta la red y la infografía interactiva ¿Cómo viajan los datos?

Transferencia del conocimiento (20 min)

Segunda parte de “antes pensaba, ahora pienso”

Los estudiantes escribirán las nuevas ideas a las tres preguntas formuladas a partir de las informaciones vistas y realizarán una puesta en común para compartir y explicar los cambios de pensamiento.

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Responden preguntas alusivas al video, argumentan y reflexionan sobre sus repercusiones sobre el medio ambiente.

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Lección 63

Infografía de la clase II

Propósito

Esta Lección busca que el estudiante ponga en práctica lo estudiado en las tres principales actividades de esta unidad.

Secuencia para el aprendizaje

Conocimiento inicial (15 min)

Ampliación del conocimiento (15 min)

Transferencia del conocimiento (20 min)

Objetivos

Los estudiantes serán capaces de:

- Visualizar e interpretar datos mediante gráficos e infografías.
- Generar informaciones y supuestos a partir de la comparación cuantitativa de datos.
- Analizar el valor de los datos valorando las cinco V en el big data (volumen, velocidad, variedad, veracidad y visualización).

Preparación

- Imprima cada una de las píldoras informativas y la guía de trabajo para cada estudiante.
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- Gráficas – Datos por todas partes
- Píldora informativa – [Big data: cuando los datos se vuelven inteligentes](#)
- Píldora informativa – [La quinta V del big data](#)
- Guía de trabajo – Analizando datos

Vocabulario

- **Big data:** conjunto grande de datos.

Estrategia de aprendizaje

Conocimiento inicial (15 min)

Analizar y comparar datos

Se desarrollará grupalmente usando la Guía Analizando datos y consultando las gráficas – Datos por todas partes con los participantes en el programa.

Los estudiantes deberán comparar los datos de su clase con los de otras clases, regiones, sexos... buscando grupos de comparación que muestren los valores más alejados posibles entre ellos.

Ampliación del conocimiento (15 min)

Creando supuestos y justificaciones

Con los datos comparativos seleccionados de dos o varios grupos que hayan llamado la atención en el paso anterior por su disparidad, se generarán informaciones cuantitativas e idearán supuestos sorprendentes o curiosos que podrían justificar dicha referencia de datos.

Cada suposición debe ir acompañada de lo que haría falta para corroborarla y convertirla en información cualitativa.

Transferencia del conocimiento (20 min)

Puesta en común y las cinco V del big data

Puesta en común de los supuestos ideados en el paso anterior y análisis con ayuda de las cinco V del big data a partir de la consulta de la píldora informativa [Big data: cuando los datos se vuelven inteligentes](#) y de la píldora informativa [La quinta V del big data](#).

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Representan datos utilizando múltiples esquemas de codificación, considerando temas de su interés.
- Argumentan la toma de decisiones y sus resultados.

Contenido 3

Geometría en la naturaleza

Resumen

- **Lección 64:** Secuencia Fibonacci
- **Lección 65:** Exploración de Codeblocks de Tinkercad
- **Lección 66:** Análisis de patrones generados por código inspirados desde la naturaleza
- **Lección 67:** Encontrando inspiración en la naturaleza
- **Lección 68:** Investigación sobre lo que te inspira de la naturaleza
- **Lección 69:** Proyecto sobre cambio climático
- **Lección 70:** Introducción al pensamiento de diseño y definición del problema
- **Lección 71:** Idea como un ingeniero
- **Lección 72:** Haz tu prototipo
- **Lección 73:** Presenta tu trabajo!

Objetivos

- **OA 4.** Crear aplicaciones y realizar análisis mediante procesadores simbólicos, de geometría dinámica y de análisis estadístico.
- **OA g.** Elaborar representaciones, tanto en forma manual como digital, y justificar cómo una misma información puede ser utilizada según el tipo de representación.

Referencias

- <https://www.tinkercad.com/>
- Common Core (Math): www.corestandards.org/Math/Content/5/introduction/
- Next Generation Science Standards: www.nextgenscience.org/search-standards

*Varios de estos enlaces llevan a páginas en inglés. Sugerimos probar la traducción automática entregada por su navegador (de preferencia Google Chrome o Microsoft Edge).

Biomimetismo y naturaleza como compañera de diseño

Propósito

El biomimetismo es la imitación de modelos, sistemas y elementos de la naturaleza con el propósito de resolver problemas humanos complejos. En los campos de la arquitectura y la fabricación, el biomimetismo implica diseñar edificios y productos que simulen procesos que ocurren en la naturaleza. Después de que los estudiantes aprendan sobre el concepto de biomimetismo, aplicarán este conocimiento buscando patrones en la naturaleza, investigando problemas relacionados con el cambio climático y aplicando una mentalidad de ingeniería para diseñar un nuevo producto o estructura inspirada en la naturaleza. El plan incluye recursos complementarios para ayudarlo a guiar a los estudiantes a lo largo del proceso de ingeniería y diseño, y también involucrarlos en la comunicación creativa de sus ideas.

Podrá encontrar la duración estimada de las unidades al comienzo de cada una, por lo que tendrá la flexibilidad de ajustarlas según sea necesario.

Esta Lección no requiere experiencia previa en Tinkercad.

Objetivos

Los estudiantes serán capaces de:

- Identificar números Fibonacci y sus orígenes, y generar los siguientes números de la secuencia.
- Poner en práctica el pensamiento computacional al analizar un patrón generado por código, inspirado por la naturaleza.
- Identificar las maneras en que analizar e imitar a la naturaleza puede ayudar a resolver problemas.
- Investigar sobre el cambio climático y aplicar esos conocimientos a un auténtico desafío de diseño.
- Definir un problema a través del pensamiento de diseño.
- Aplicar una mentalidad de ingeniería al definir los criterios y limitaciones de un problema de diseño.
- Crear un modelo 3D en Tinkercad.

Recursos

Esta es una lista de Recursos que cada estudiante necesitará para poder completar esta Lección:

- Un computador o Tablet con acceso a internet y a Tinkercad
- Hojas para bocetear
- Regla
- Papel, plumones, lápices de colores o lápices pasta
- Un software de diapositivas, como Google Slides, PowerPoint o Keynote (opcional)

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Rúbrica de evaluación](#) [PDF]
- [La secuencia Fibonacci](#) [PPTX]
- [Análisis de un patrón generado por código inspirado por la naturaleza](#) [Video]

- Usar un prototipo digital para comunicar creativamente sus ideas.

Habilidades

- Diseño 3D
- Diseño Computacional
- Creatividad en la resolución de problemas
- Pensamiento de diseño
- Diseño de ingeniería
- Matemática computacional
- Investigación
- Comunicación visual

Contenido 3

- [Girasoles Fibonacci](#) [PPTX]
- [Inspiración en la naturaleza](#) [PPTX]
- [Explorar el cambio climático](#) [PPTX]
- [Introducción al pensamiento de diseño](#) [Video]
- [Definir el problema](#) [PPTX]
- [Criterios y limitantes](#) [PDF]
- [Prototipo paso 1](#) [Video]
- [Prototipo paso 2](#) [Video]
- [Prototipo paso 3](#) [Video]
- [Prototipo paso 4](#) [Video]
- [Prototipo paso 5](#) [Video]
- [Presenta tu trabajo](#) [PPTX]

Preparación de la clase

Esta Lección puede realizarse tanto de forma individual como en pequeños grupos. Si los estudiantes trabajan de forma remota, es recomendable utilizar una herramienta en línea aprobada por la institución para que puedan interactuar entre ellos, por ejemplo, Zoom. A lo largo del proyecto, los estudiantes deben detenerse y anotar sus ideas en respuesta a las indicaciones.

Revise la lista de recursos y pida a los estudiantes que se aseguren de tener los materiales correspondientes a mano cuando sean necesarios.

Varios de los enlaces presentes en la Lección llevan a páginas en inglés. Sugerimos probar la traducción automática entregada por su navegador (de preferencia Google Chrome o Microsoft Edge).

Lección 64

Secuencia Fibonacci

Propósito

En esta Lección, se busca que los estudiantes identifiquen la secuencia de Fibonacci. De esta forma, los estudiantes podrán profundizar su conocimiento y pensamiento matemático ante futuros problemas que demanden el uso de patrones numéricos.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (15 min)

Transferencia del conocimiento (10 min)

Objetivos

Los estudiantes serán capaces de:

- Identificar los números Fibonacci y su origen y generar el siguiente número en la secuencia Fibonacci.
- Articular las formas en que las personas practican la resolución inventiva de problemas en su vida cotidiana.

Preparación

- Tener preparada y lista para presentar la secuencia Fibonacci [PPTX]
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- [La secuencia Fibonacci](#) [PPTX] (Asegúrese de descargar la presentación).

Vocabulario

- **Secuencia Fibonacci:** En matemáticas, es una sucesión infinita de números naturales.

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Guíe una lluvia de ideas para determinar el conocimiento inicial y de fondo de los estudiantes.

Ampliación del conocimiento (15 min)

Guíe a los estudiantes a través de las diapositivas (revise el enlace en la sección de Recursos), pausando para que participen en las sugerencias. Esto podría culminar con una discusión en la cual los estudiantes podrían expresar su pensamiento matemático. Dependiendo del contexto y del nivel de los estudiantes, podrían discutir sobre:

- El concepto de razón, el uso del lenguaje relativo al tema para describir la relación de razón entre dos cantidades
- Relaciones proporcionales entre dos cantidades
- Revisar raíces cuadradas
- En un contexto artístico, podría extenderse a una discusión de la secuencia Fibonacci en las artes visuales y cómo los artistas y diseñadores usan un pensamiento matemático en sus composiciones
- Otras formas para extender la discusión: el áureo, Fibonacci en símbolos universales y naturaleza

Transferencia del conocimiento (10 min)

Para resumir esta Lección y reforzar los conceptos clave del tema y la habilidad de los estudiantes, encontrará sugerencias de actividades de redacción en las notas de la presentación. Puede extender esta Lección invitando a los estudiantes a utilizar Tinkercad para realizar un diseño inspirado en la secuencia Fibonacci, la razón de oro o el ángulo áureo, por ejemplo, manipulando los tamaños de las formas para reflejar los números.

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Instrucciones para el estudiante

Revisa las diapositivas (revisa el enlace en la sección de Recursos):

- La secuencia Fibonacci
- “el ángulo aureo” y “la razón de oro”
- Cómo reconocer patrones universales en el mundo
- Las formas en que los seres humanos incorporan la naturaleza en nuestra arquitectura, diseño, y procesos de resolución de problemas

Si estás enseñando de forma remota o independiente, coméntales a tus estudiantes que revisen las notas en la presentación para encontrar enlaces con información adicional y preguntas a considerar. Si estás enseñando en una sala de clases presencial, podrías guiarlos a una discusión de curso.

Como extensión de esta Lección, podrías crear un diseño en Tinkercad inspirado por los números de la secuencia Fibonacci o de la razón de oro.

Sugerencias para evaluación

Se sugiere los siguientes indicadores para evaluar formativamente los aprendizajes:

- Documentan los programas para que sean más fáciles de seguir, probar y depurar
- Crean aplicaciones y realizan análisis mediante procesadores simbólicos, de geometría dinámica y de análisis estadístico

Lección 65

Exploración de Codeblocks

Propósito

Los estudiantes explorarán los conceptos de diseño computacional por medio de la herramienta Tinkercard.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (15 min)

Transferencia del conocimiento (10 min)

Objetivos

Los estudiantes serán capaces de:

- Crear un modelo 3D usando Tinkercad

Preparación

- Tener preparada y lista para presentar las [actividades de Codeblock](#).
- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- Actividades en [Codeblock](#)
- [Guía tutorial](#)

Estrategia de aprendizaje

Revise esta guía tutorial con los estudiantes, la cual les presentará los aspectos básicos del uso de Codeblocks (bloques de código), de Tinkercad. Pida a los estudiantes que dediquen algo de tiempo para explorar el diseño computacional, completando las actividades “Intro to Shapes”, “Moving Shapes” y “Rotating Shapes” de bloques de código

Conocimiento inicial (5 min)

Guíe una lluvia de ideas para determinar el conocimiento inicial y de fondo de los estudiantes.

Ampliación del conocimiento (15 min)

Si te interesa crear modelos 3D usando bloques de código arrastrables, revisa esta guía tutorial, la cual te ayudará a conocer los aspectos básicos del uso de Codeblock, de Tinkercad. Dedicar algo de tiempo para explorar el diseño computacional, completando las actividades “Intro to Shapes”, “Moving Shapes” y “Rotating Shapes” de bloques de código

Conocer los aspectos básicos te ayudará a descubrir formas creativas de crear y construir tu propio proyecto inspirado en el biomimetismo.

Transferencia del conocimiento (10 min)

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Sugerencias para evaluación

Se sugiere los siguientes indicadores para evaluar formativamente los aprendizajes:

- Descomponen problemas y subproblemas en partes para facilitar el diseño, implementación y revisión de programas
- Utilizan programas para verificar teoremas relacionados con la geometría.
- Crean aplicaciones y realizan análisis mediante procesadores simbólicos, de geometría dinámica y de análisis estadístico

Lección 66

Análisis de patrones generado por código inspirados desde la naturaleza

Propósito

En esta Lección, los estudiantes identificarán patrones generados por códigos por medio de la herramienta Tinkercad.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (15 min)

Transferencia del conocimiento (10 min)

Objetivos

Los estudiantes serán capaces de:

- Practicar el pensamiento computacional al analizar un patrón generado por código inspirado en la naturaleza.

Preparación

- Tener preparado y listo para presentar el video [Phyllotaxis Bowl](#)
- Asegúrese de que cada estudiante tenga su [Diario de apuntes](#).

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- Video [Phyllotaxis Bowl](#)

Vocabulario

- **Patrón:** modelo o punto de referencia.

Estrategia de aprendizaje

Pida a los estudiantes que vean este video. Aquí aprenderá a crear patrones usando el espacio de trabajo de Codeblock. Analice los siguientes aspectos con los estudiantes:

- ¿Qué notas en los Codeblocks?
- ¿Cómo es que el diseñador usa un pensamiento matemático?
- ¿Qué notas sobre sobre lo que hacen las formas?
- ¿Qué notas respecto al efecto en general?

Si los estudiantes están listos para continuar con esta Lección, siga este enlace para que intenten crear sus propios patrones generados por código.

Conocimiento inicial (5 min)

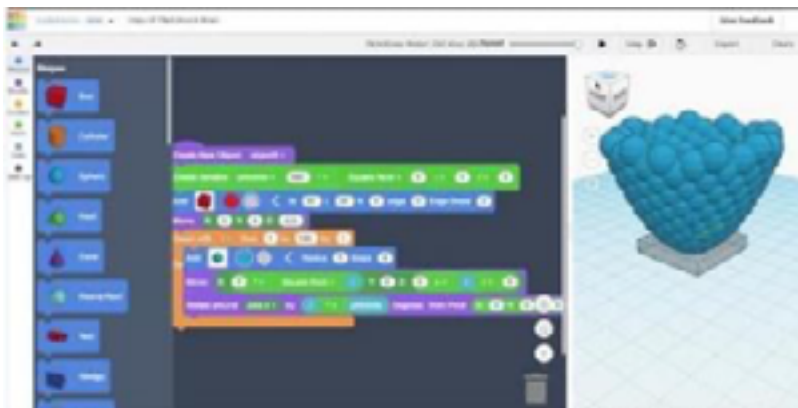
Guíe una lluvia de ideas para determinar el conocimiento inicial y de fondo de los estudiantes.

Ampliación del conocimiento (15 min)

Sigue este enlace para ver este video. Aquí aprenderás a crear patrones usando el espacio de trabajo de Codeblock.

- ¿Qué notan en los Codeblocks?
- ¿Cómo es que el diseñador usa un pensamiento matemático?
- ¿Qué notan sobre sobre lo que hacen las formas?
- ¿Qué notan respecto al efecto en general?

Si estás listo para continuar con esta Lección, sigue este enlace para que intentes crear tus propios patrones generados por código.



Transferencia del conocimiento (10 min)

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Prueban y refinan sistemáticamente programas utilizando una variedad de casos de prueba.
- Utilizan programas para verificar teoremas relacionados con la geometría.
- Crean aplicaciones y realizan análisis mediante procesadores simbólicos, de geometría dinámica y de análisis estadístico.

Lección 67

Encontrando inspiración en la naturaleza

Propósito

En esta Lección, los estudiantes conocerán diferentes tendencias que se han inspirado en la naturaleza, y cómo estas han contribuido a la evolución del arte y la arquitectura, por ejemplo.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (20 min)

Transferencia del conocimiento (15 min)

Objetivos

Los estudiantes serán capaces de:

- Identificar las formas en que analizar e imitar a la naturaleza puede ayudar a resolver problemas.

Preparación

- (Opcional) vea la PPTX [Inspiración en la naturaleza](#).
- Asegúrese de que cada estudiante tenga su [Diario de apuntes](#).

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- [Inspiración en la naturaleza](#) [PPTX] (asegúrese de descargar las diapositivas).

Vocabulario

- **Inspiración:** estímulo que favorece la creatividad.

Estrategia de aprendizaje

Guíe a los estudiantes a través de las diapositivas, parando ocasionalmente para conversar sobre las actividades sugeridas. Esta Lección debería conducir a una investigación autodidacta sobre cómo analizar e imitar la naturaleza puede ayudar a resolver problemas. Podría incluso asignar lecturas tales como:

<https://redshift.autodesk.es/>

<https://redshift.autodesk.es/biomimetica-arquitectura-sostenible/>

Conocimiento inicial (10 min)

Guíe una lluvia de ideas para determinar el conocimiento inicial y de fondo de los estudiantes.

Ampliación del conocimiento (20 min)

Revisa las diapositivas con la clase para explorar:

- La estructura o forma de un girasol
- Espirales de secuencia Fibonacci y resolución de problemas
- La relación entre naturaleza y arquitectura

Si estás enseñando de forma remota o independiente, coméntales a tus estudiantes que revisen las notas en la presentación para encontrar enlaces con información adicional y preguntas a considerar. Si estás enseñando en una sala de clases presencial, podrías guiarlos a una discusión de curso.

Transferencia del conocimiento (15 min)

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Lección 68

Investigación sobre lo que te inspira de la naturaleza

Propósito

En esta Lección, los estudiantes identificarán diferentes fuentes de la naturaleza que inspiran actualmente el diseño.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (15 min)

Transferencia del conocimiento (10 min)

Objetivos

Los estudiantes serán capaces de:

- Navegar la red para investigar fuentes de la naturaleza que inspiran el diseño.

Preparación

- (Opcional) vea la PPTX [Inspiración en la naturaleza](#).
- Asegúrese de que cada estudiante tenga su [Diario de apuntes](#).

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- [Inspiración en la naturaleza](#) [PPTX] (asegúrese de descargar las diapositivas).

Vocabulario

- **Inspiración:** estímulo que favorece la creatividad.

Estrategia de aprendizaje

Los estudiantes pueden usar la fuente que ellos deseen para buscar ejemplos de la naturaleza que los inspire. Entregue instrucciones de cómo deben compartir su investigación con usted.

Conocimiento inicial (5 min)

Guíe una lluvia de ideas para determinar el conocimiento inicial y de fondo de los estudiantes.

Ampliación del conocimiento (15 min)

Dedica al menos 15 minutos para investigar e identificar otros aspectos de la naturaleza que te inspiren. No olvides citar las fuentes de tu investigación. El profesor te dará indicaciones sobre cómo compartir tu descubrimiento.

Transferencia del conocimiento (10 min)

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Lección 69

Proyecto sobre cambio climático

Lección en línea

Propósito

En esta Lección, se espera que los estudiantes puedan identificar las principales características del cambio climático, y como este fenómeno influencia y desafía la forma de diseñar.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (25 min)

Transferencia del conocimiento (10 min)

Objetivos

Los estudiantes serán capaces de:

- Investigar sobre el cambio climático y aplicar esos conocimientos en un auténtico desafío de diseño.

Preparación

- (Opcional) vea la PPTX [Explorar el cambio climático](#).
- Asegúrese de que cada estudiante tenga su [Diario de apuntes](#).

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- [Explorar el cambio climático](#) [PPTX] (asegúrese de descargar las diapositivas).

Vocabulario

- **Cambio climático:** cambio de clima debido a la Lección humana.

Estrategia de aprendizaje

Revise las [diapositivas](#) para explorar los aspectos del cambio climático. Presente el desafío de diseño y explique a los estudiantes las habilidades que pondrán en práctica para abordarlo. Para comenzar, los estudiantes deben escribir una lista de oportunidades de innovación para combatir el cambio climático. Pueden trabajar de forma individual o en grupos. Se deben usar fuentes confiables al realizar la investigación. La infografía de “El futuro de la producción” es un buen punto de inicio para tener una vista general del asunto:

<https://adsknews.autodesk.com/views/future-of-making-infographic> (puede encontrar la infografía en español en las diapositivas)

Para conocer un punto de vista sobre cómo se puede combatir el cambio climático con innovación, pueden leer:

[Ni negación ni miedo: el cambio climático precisa políticas, innovación e infraestructuras \(autodesk.es\)](#)

Los estudiantes deben documentar su investigación, lo cual pueden hacer con artículos, videos o posts de blogs. Una vez que ya hayan realizado una investigación, de forma opcional, podrían entrevistar a una persona o grupo de personas que son (o que serán) afectados por el problema que estén investigando. Pueden realizar la entrevista en persona, vía e-mail o con una encuesta o formulario en línea. Cuando deban trabajar en equipos, los estudiantes aportarán ideas sobre posibles soluciones para combatir los problemas en los que se estén concentrando y luego diseñarán y crearán un prototipo en Tinkercad de la solución que hayan escogido.

Conocimiento inicial (10 min)

Guíe una lluvia de ideas para determinar el conocimiento inicial y de fondo de los estudiantes.

Ampliación del conocimiento (25 min)

Revisa las [diapositivas](#) para explorar los aspectos del cambio climático y para conocer el desafío de diseño en que trabajarás. En él, aplicarás biomimetismo para abordar un problema relacionado al cambio climático y después vas a comunicar tu idea construyendo un modelo en Tinkercad.

A través de este proceso, aplicarás tanto los pasos del pensamiento de diseño (design thinking) como la mentalidad de ingeniería, que podemos relacionar con el pensamiento digital.

En la última Lección, descubriste algunos ejemplos en la naturaleza que te inspiraron como diseñador. Ahora, vas a investigar sobre aspectos del cambio climático. Tu objetivo es idear una forma de aplicar un aspecto de la naturaleza para minimizar el impacto humano en el medio ambiente. No olvides que debes revisar varias fuentes, las cuales deben ser confiables.

Aquí hay algunos buenos puntos para comenzar:

- <https://adsknews.autodesk.com/views/future-of-making-infographic> (puede encontrar la infografía en español en las diapositivas).

- Para conocer un punto de vista sobre cómo se puede combatir el cambio climático con innovación, pueden leer: Ni negación ni miedo: el cambio climático precisa políticas, innovación e infraestructuras (autodesk.es)

Transferencia del conocimiento (10 min)

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Resuelven problemas con estrategias como dividirlo en problemas menores.

Lección 70

Introducción al pensamiento de diseño y definición del problema

Propósito

En esta Lección, los estudiantes reconocen el pensamiento de diseño como un enfoque que permite la definición y resolución de un problema.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (30 min)

Transferencia del conocimiento (10 min)

Objetivos

Los estudiantes serán capaces de:

- Definir un problema usando el pensamiento de diseño.

Preparación

- (Opcional) Vea el video [Design Thinking](#).
- Vea la PPTX [Definir el problema](#).
- Asegúrese de que cada estudiante tenga su [Diario de apuntes](#).

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- Video [Design Thinking](#)
- [Definir el problema](#) [PPTX] (asegúrese de descargar las diapositivas).
- (Opcional) Notas adhesivas, plumones, papel cuadriculado, pizarras o un espacio en blanco en una pared.

Vocabulario

- **Pensamiento de diseño (design thinking):** Enfoque de resolución de problemas específico del diseño.

Estrategia de aprendizaje

Haga clic en el [enlace](#) para reproducir el vídeo y aprender qué es el pensamiento de diseño y cómo sus estudiantes pueden utilizarlo para resolver una amplia variedad de problemas, incluidos los relacionados con el cambio climático. Después de que los estudiantes vean el vídeo, pídeles que compartan ejemplos de formas en las que el pensamiento de diseño podría ayudar a resolver problemas relacionados con el cambio climático.

Conocimiento inicial (5 min)

Parte 1: introducción al pensamiento de diseño

Haz clic en el [enlace](#) para reproducir el vídeo y aprender qué es el pensamiento de diseño y cómo puedes utilizarlo para resolver una amplia variedad de problemas, incluidos los relacionados con el cambio climático.



Muestre a los estudiantes las diapositivas (revise el enlace en la sección de recursos) para explorar las diferentes formas en que pueden utilizar el pensamiento de diseño para entender con claridad el problema y los puntos de vista de los afectados en la comunidad. Los estudiantes deberán realizar al menos una de las actividades. Deberán expresar lo que hayan aprendido en forma de "declaración de necesidades". Instruya a los estudiantes sobre la forma en que deben compartir su trabajo con usted.

Ampliación del conocimiento (30 min)

Instrucciones para el estudiante

Parte 2: definición del problema

Si has terminado la primera ronda de investigación, estás listo para la fase de "Definición". Revisa la

presentación de diapositivas (puedes encontrar el enlace en la sección de recursos) para obtener una visión general de cómo definir un problema. Una vez que hayas completado tu investigación, dedica algo de tiempo a analizar los datos que has recopilado. Esto puede hacerse de forma individual o en grupos. Debes realizar al menos una de las cuatro actividades siguientes:

- Clasificar y buscar patrones en tu investigación.
- Intentar profundizar en la raíz del problema.
- Trazar un mapa de las partes interesadas y de otros contextos relevantes para el problema.
- Crear un mapa de empatía sobre los más afectados por el problema, respondiendo a estas indicaciones.

Resume tus ideas redactando una declaración de necesidades para el usuario final de tu idea de diseño. Tu profesor te dará instrucciones sobre cómo compartir tu trabajo.

Transferencia del conocimiento (10 min)

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Prueban y refinan sistemáticamente programas utilizando una variedad de casos de prueba.

Lección 71

Idea como un ingeniero

Propósito

En esta Lección, se espera que los estudiantes identifiquen los distintos criterios y limitantes que el proceso de un diseño podría presentar.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (15 min)

Transferencia del conocimiento (10 min)

Objetivos

Los estudiantes serán capaces de:

- Aplicar una mentalidad de ingeniería al definir los criterios y limitantes de un problema de diseño.

Preparación

- (Opcional) vea el PDF [Criterios y limitantes](#).
- Asegúrese de que cada estudiante tenga su [Diario de apuntes](#).

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- [Criterios y limitantes](#) [PDF]
- (Opcional) Notas adhesivas, plumones, papel cuadriculado, pizarras o un espacio en blanco en una pared.

Vocabulario

- **Criterio:** juicio u opinión sobre algo.
- **Limitante:** que presenta límites.

Estrategia de aprendizaje

Revise la guía de trabajo. Presente a los estudiantes la fase de "Idear" del pensamiento de diseño y asegúrese de que dispongan de todos los recursos necesarios para elaborar una extensa lista de posibles soluciones al problema. Recuérdeles que deben evitar eliminar las ideas antes de anotarlas, es decir, solicíteles que anoten todas las ideas, incluso si no las van a utilizar o si son disparatadas. Anímelos a referirse a los resultados de su investigación.

Al final de la etapa de lluvia de ideas, deben elegir una solución potencial en la que se hayan centrado basándose en la evaluación de los criterios y las limitaciones. Deberán incluir un resumen de tres criterios que esperan demostrar a través del diseño y tres limitaciones que condicionarán el diseño. Instrúyalos sobre cómo compartir su trabajo.

Conocimiento inicial (5 min)

Guíe una lluvia de ideas para determinar el conocimiento inicial y de fondo de los estudiantes.

Ampliación del conocimiento (15 min)

Revisa la guía de trabajo (encontrarás el enlace en la sección de recursos). Cuando llegues a la fase de "idear", el objetivo es que se te ocurran tantas soluciones como sea posible. Todas las ideas están bien, incluso las más descabelladas, porque al final de esta fase aplicarás una mentalidad de ingeniería para reducir las opciones y encontrar la mejor solución, cuyo prototipo podrás crear en Tinkercad. Cuando los ingenieros idean, a veces también examinan las soluciones existentes para inspirarse y compararlas.

Solo o con tu grupo, pensarás en todas las ideas que puedas para abordar el problema que definiste en tu declaración de necesidades. Documenta todo lo que surja en tu grupo o lo que se te ocurra. Utiliza herramientas como los mapas mentales y las notas adhesivas para hacer un seguimiento de tus ideas.

Una forma de reducir la lista de posibles soluciones y decidir el diseño, es considerar los criterios y las limitaciones. Los "criterios" son las características positivas de un diseño que se pueden medir, mientras que las "limitaciones" son los factores que pueden condicionar el diseño. Los criterios y limitaciones deben abarcar al menos tres de las siguientes categorías: técnico, producción, mercado, negocios y sustentabilidad. Antes de pasar a la creación de prototipos en Tinkercad, debes esbozar tres criterios que esperas demostrar a través de tu diseño y tres limitaciones que condicionarán tu diseño, utilizando tu guía de trabajo. Tu profesor te orientará sobre cómo compartir tu trabajo.

Transferencia del conocimiento (15 min)

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Documentan los programas para que sean más fáciles de seguir, probar y depurar.

Lección 72

Haz tu prototipo

Propósito

En esta Lección, se busca que los estudiantes diseñen el prototipo de un modelo 3D con Tinkercad.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (15 min)

Transferencia del conocimiento (10 min)

Objetivos

Los estudiantes serán capaces de:

- Crear un modelo 3d en Tinkercad.
- Utilice un prototipo digital para comunicar creativamente sus ideas.

Preparación

- Asegúrese de que cada estudiante tenga su [Bitácora de trabajo](#).

Lección en línea

Vocabulario

- **Prototipo:** primer diseño de algo que sirve de modelo.

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Guíe una lluvia de ideas para determinar el conocimiento inicial y de fondo de los estudiantes.

Ampliación del conocimiento (20 min)

Tener un conocimiento básico de Tinkercad ayudará a los estudiantes con la creación de prototipos en esta Lección. Si los estudiantes aún no están familiarizados con Tinkercad, guíelos o pídeles que sigan de forma individual los vídeos de presentación. Si aún no lo han hecho, también puede animarlos a completar estas actividades rápidas de iniciación.

Anime a los estudiantes a utilizar el software para explorar las habilidades de diseño en 3D y desarrollar su alfabetización digital en 3D. Invítelos a compartir sus creaciones con sus compañeros para que les den su opinión. También pueden compartir sus prototipos con la persona o personas a las que entrevistaron al investigar el problema. Recuérdeles que tomen notas minuciosamente para registrar los comentarios. Puede darles más tiempo para que perfeccionen sus diseños en función de los comentarios.

Instrucciones para el estudiante

Utilizarás Tinkercad para aprender las habilidades básicas de diseño en 3D, como la navegación, los controles de la cámara, la colocación y manipulación de formas, el uso de un generador de formas en 2D, la duplicación, la selección, la alineación, la agrupación y mucho más. Conocer estos conceptos básicos te ayudará a crear tu propio prototipo. ¡Que disfrutes de tu trabajo!

Las siguientes secciones te guiarán a través de cinco pasos, utilizando la Avalon's House (visto en las actividades anteriores) como modelo para practicar cómo crear un diseño a partir de una forma biomórfica. Puedes seguir los pasos exactos del vídeo para desarrollar tus habilidades técnicas de diseño en 3D, o puedes seguir los pasos e improvisar con tus propias ideas. Aunque no es necesario, es recomendable utilizar un mouse cuando se diseña en 3D.

Después de revisar los vídeos que aparecen en las próximas secciones, continua con las actividades y crea un prototipo de tu idea utilizando Tinkercad.

A continuación, puedes "probar" tu diseño compartiéndolo con tus compañeros para que te den su opinión o incluso con los posibles usuarios de tu diseño. Anota lo que ha funcionado y lo que no. Esto te será útil para el siguiente paso.

Después de completar tu prototipo en Tinkercad, captura imágenes de tu diseño o comparte el enlace con tu profesor según las instrucciones.

Prototipo paso 1 [5 min]

[Lección en línea](#)

Haga clic en este [enlace](#) para ver el vídeo, en el cual podrá ver el primer paso del proceso de

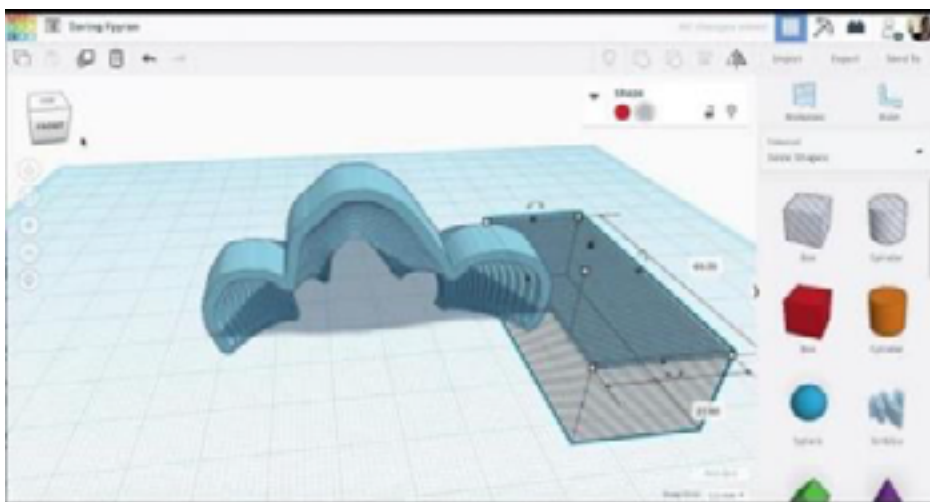
creación de prototipos.

Los estudiantes pueden empezar a crear su prototipo en Tinkercad mientras revisan los videos de los cinco pasos, o esperar hasta ver todas las demostraciones antes de comenzar. Instruya a los estudiantes para compartir su prototipo con usted una vez que lo hayan completado.

Instrucciones para el estudiante

Haz clic en este [enlace](#) para ver el vídeo, en el cual podrás ver el primer paso del proceso de creación de prototipos.

Paso 1: Crear una forma biomorfa con la herramienta Trazo (Scribble). Este vídeo también muestra cómo utilizar Smart Duplicate. Para duplicar un objeto en Tinkercad, usa Ctrl + D. Para explicarlo mejor, "Smart Duplicate" es como una memoria a corto plazo que recuerda lo que acabas de hacer y lo repetirá para crear patrones y repeticiones mientras copias objetos. Con estos pasos, esta función te permitirá crear complejos patrones de repetición en segundos.



Prototipo paso 2 [5 min]

Lección en línea

Haga clic en este [enlace](#) para ver el vídeo, en el cual podrá ver el segundo paso del proceso de creación de prototipos.

Los estudiantes pueden empezar a crear su prototipo en Tinkercad mientras revisan los videos de los cinco pasos, o esperar hasta ver todas las demostraciones antes de comenzar. Instruya a los estudiantes para compartir su prototipo con usted una vez que lo hayan completado.

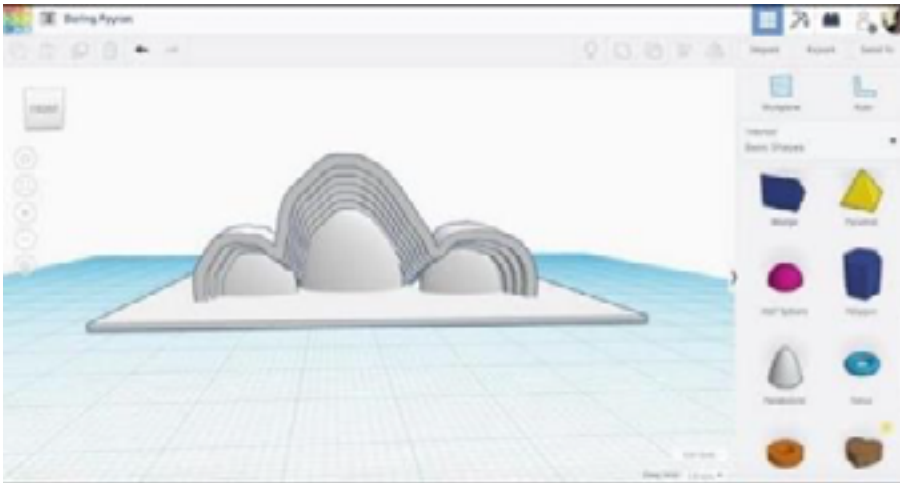
Instrucciones para el estudiante

Haz clic en este [enlace](#) para ver el vídeo, en el cual podrás ver el segundo paso del proceso de creación de prototipos.

Paso 2: Añadir dos o más formas básicas. Una forma primitiva es un punto de partida o bloque de

construcción para el diseño 3D.

Estas formas pueden añadirse, quitarse y combinarse entre sí para construir casi cualquier cosa. Entre ellas se encuentran: cubo (caja), cilindro, tubo, esfera, toro y cono.



Prototipo paso 3 [5 min]

Lección en línea

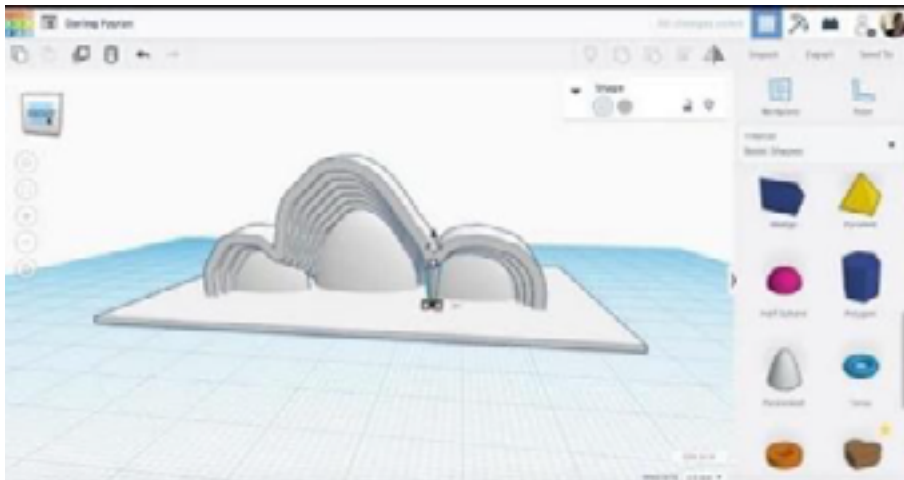
Haga clic en este [enlace](#) para ver el vídeo, en el cual podrá ver el tercer paso del proceso de creación de prototipos.

Los estudiantes pueden empezar a crear su prototipo en Tinkercad mientras revisan los videos de los cinco pasos, o esperar hasta ver todas las demostraciones antes de comenzar. Instruya a los estudiantes para compartir su prototipo con usted una vez que lo hayan completado.

Instrucciones para el estudiante

Haz clic en este [enlace](#) para ver el vídeo, en el cual podrás ver el tercer paso del proceso de creación de prototipos.

Paso 3: Añadir una forma que sirva de soporte. Este vídeo también muestra cómo utilizar la vista ortográfica. Esta es una vista 2D de un objeto tridimensional. Las vistas ortográficas representan la forma exacta de un objeto visto desde un sólo lado y, al mismo tiempo, desde una vista perpendicular. La profundidad no se muestra. En Tinkercad, puedes cambiar a la vista ortográfica haciendo clic en el icono que parece un bloque en la parte inferior de los controles de navegación. Esto es útil cuando estás centrando, apilando o alejando el zoom para analizar tu diseño.



Prototipo paso 4 [5 min]

Lección en línea

Haga clic en este [enlace](#) para ver el vídeo, en el cual podrá ver el cuarto paso del proceso de creación de prototipos.

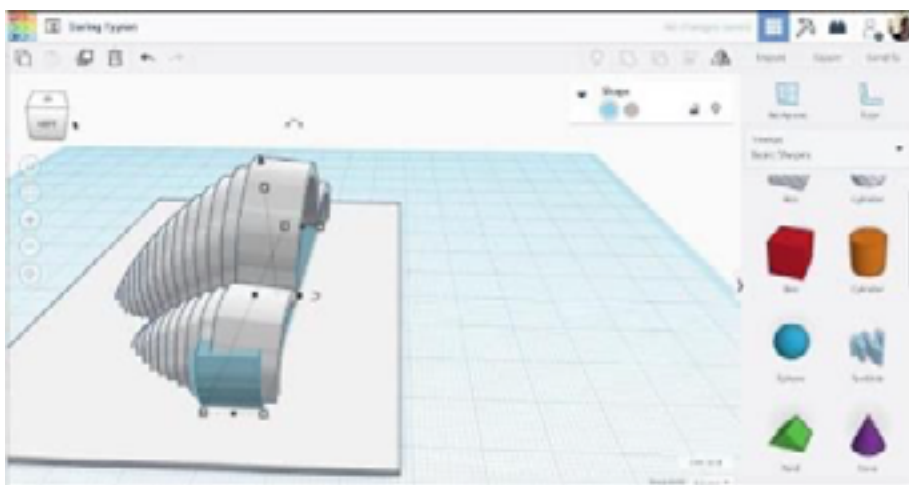
Los estudiantes pueden empezar a crear su prototipo en Tinkercad mientras revisan los videos de los cinco pasos, o esperar hasta ver todas las demostraciones antes de comenzar. Instruya a los estudiantes para compartir su prototipo con usted una vez que lo hayan completado.

Instrucciones para el estudiante

Haz clic en este [enlace](#) para ver el vídeo, en el cual podrás ver el cuarto paso del proceso de creación de prototipos.

Paso 4: Añadir una ventana o puerta. ¡Hazla transparente! Además de mostrar cómo hacer otra forma con la herramienta Trazo (Scribble), también muestra cómo rotar una forma.

¡Esto puede ser complicado! Quizás necesites intentarlo más de un par de veces.



Prototipo paso 5 [5 min]Lección en línea

Haga clic en este [enlace](#) para ver el vídeo, en el cual podrá ver el quinto paso del proceso de creación de prototipos.

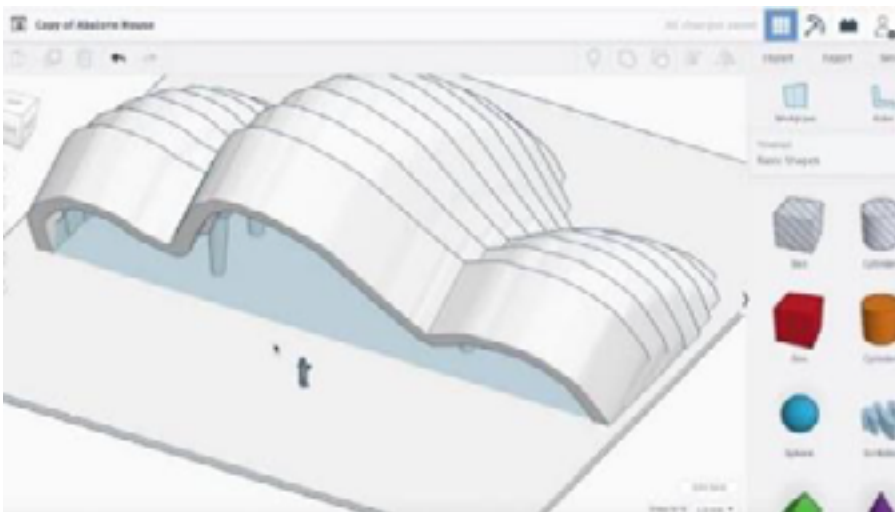
Los estudiantes pueden empezar a crear su prototipo en Tinkercad mientras revisan los videos de los cinco pasos, o esperar hasta ver todas las demostraciones antes de comenzar. Instruya a los estudiantes para compartir su prototipo con usted una vez que lo hayan completado.

Instrucciones para el estudiante

Haz clic en este [enlace](#) para ver el vídeo, en el cual podrás ver el quinto paso del proceso de creación de prototipos.

Paso 5: Añadir una persona u objeto a escala. Escalar significa cambiar el tamaño de un objeto para que sus dimensiones sean proporcionales al tamaño original. Puedes hacerlo manteniendo pulsada la tecla Shift mientras empujas y tiras de los puntos de control para cambiar el tamaño.

Si te sientes preparado, ¡utiliza estas habilidades que acabas de aprender para crear tu prototipo en Tinkercad!

**Transferencia del conocimiento (5 min)**Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?

- ¿Cómo te sentiste durante la Lección?

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Crean variables con nombres claros que representen diferentes tipos de datos y realice operaciones con sus valores
- Descomponen problemas y subproblemas en partes para facilitar el diseño, implementación y revisión de programas.

Lección 73

Presenta tu trabajo [60 min]

Propósito

En esta Lección, los estudiantes compartirán el diseño realizado como producto final. Para ello, lo harán en una presentación de diapositivas que contengan imágenes y textos que expliquen el proceso en sí.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (35 min)

Transferencia del conocimiento (15 min)

Objetivos

Los estudiantes serán capaces de:

- Crea una presentación visual para presentar el diseño como producto final.

Preparación

- (Opcional) vea la PPTX [Presenta tu trabajo](#).
- Asegúrese de que cada estudiante tenga su [Diario de apuntes](#).

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

- [Presenta tu trabajo](#) [PPTX] (asegúrese de descargar las diapositivas).

Estrategia de aprendizaje

Ofrezca un espacio para que los estudiantes puedan mostrar los resultados de su trabajo, incluyendo el prototipo final en 3D que crearon con Tinkercad (consulte el siguiente enlace para guiarlos en la creación de su propia presentación). Como parte de su presentación, deben describir su proceso de diseño y el aprendizaje relacionado. Los estudiantes pueden utilizar un software de presentación, como PowerPoint, Google Slides o Keynote.

De ser posible, podría ser una buena idea invitar a otras partes interesadas, como socios de la industria y de la comunidad, para que escuchen las ideas de los estudiantes y les den su opinión. Se puede reforzar el concepto de criterios y limitantes invitando a los estudiantes a dar su opinión sobre las soluciones de sus compañeros.

Conocimiento inicial (10 min)

Llegó el momento de compartir tu diseño y tu proceso de diseño con un público más amplio. Para ello, puedes crear una presentación de diapositivas combinando imágenes y texto (consulta el siguiente [enlace](#) como guía para realizar tu propia presentación). Asegúrate de repasar los consejos de esta presentación antes de empezar con la tuya.

Ampliación del conocimiento (35 min)

Comienza describiendo el aspecto de la naturaleza que te ha inspirado. A continuación, describe cómo has utilizado los métodos de pensamiento de diseño. Asegúrate de relatar también tu experiencia con el proceso de diseño. ¿Qué has aprendido? ¿Por qué debería importarle a los demás? ¿Por qué es importante tu diseño? ¿Cómo podría mejorar el mundo? Por último, explica cómo has aplicado la mentalidad de la ingeniería. Incluye al menos tres criterios que hayan guiado tu proceso de diseño.

Los criterios deben derivarse de al menos tres de las categorías: técnico, producción, mercado, negocios y sustentabilidad. No olvides incluir al menos tres de las limitaciones que han condicionado tu diseño. Las limitaciones deben derivarse también de al menos tres de las categorías. La imagen o imágenes que compartas de tu diseño deben ilustrar claramente los criterios que lo hacen fuerte. Puedes añadir subtítulos, símbolos o formas para resaltar las características más importantes.

Transferencia del conocimiento (15 min)

Escribir en el diario y charla rápida

El acto de escribir en sus diarios sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

Sugerencias para el diario:

- ¿Sobre qué se trataba la Lección de hoy?
- ¿Cómo te sentiste durante la Lección?

Lección 59

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Destreza de pensamiento: guía de trabajo Mi titular

Mi titular – Parte 1

Esta parte de la guía de trabajo se desarrolla individualmente. Cada estudiante consulta la infografía [Las increíbles cifras de internet](#), la infografía interactiva [Los datos también pesan](#) y, junto con el contenido del video que acaba de ver, debe resumir en un titular la esencia de toda esta información:

¿Qué titular te parece que recoge la idea principal de toda la información que has visto?

Mi titular – Parte 2

La parte 2 de la guía de trabajo se desarrolla en grupos de trabajo. Cada grupo compara los titulares realizados por cada uno de sus miembros y se consensua uno en conjunto. Finalmente, con toda la clase junta se hace una puesta en común de los titulares propuestos y su justificación.

Mi titular

Como acabas de ver en el video Datos por todas partes, el aumento del uso de internet ha generado grandes cantidades de datos que pueden ser analizados y transformados en información útil para resolver situaciones de nuestra vida.

Consulta la infografía Las increíbles cifras de internet y la infografía interactiva Los datos también pesan, y elabora un titular que recoja y resuma la idea principal de toda la información que has visto.

Mi titular – Individual

Júntense en un grupo de trabajo y comenten los titulares que han escogido individualmente y consensuen uno en conjunto.

El titular – Grupo de trabajo

Justificación - ¿Por qué han escogido este titular?

Lección 60

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Mi video en YouTube

Seguro que todos usan con asiduidad el servicio de YouTube.

Imaginen que acaban de subir un video a esta plataforma donde hacen una parodia de una coreografía o de una canción.

Consulten las píldoras de información de la Lección y conteste a las siguientes preguntas:

¿Qué datos cuantitativos podría obtener una empresa del video que acaban de subir?

¿Qué datos cualitativos podría obtener?

¿Qué metadatos del video necesitaría si una empresa quisiera crear listas de reproducción de videos de YouTube a partir de las preferencias marcadas por el usuario?

Debatan, toda la clase junta, nuevos servicios que se podrían crear a partir de los datos que se pueden extraer de un video en YouTube. Por ejemplo, el servicio de YouTube TV, que por una suscripción anual te permite ver los videos en directo de varias cadenas de televisión de pago a través del streaming, o un servicio que generase una lista de reproducción descargable en mp4 que contenga los últimos videoclips oficiales de tus cantantes o grupos favoritos.



Lección 61

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Datos por todas partes (documento para el docente)

Para el desarrollo de este bloque del programa educativo, será necesario que los estudiantes construyan una infografía a partir de los datos levantados.

La Lección

El objetivo de la Lección es obtener una infografía que represente un día normal de los estudiantes de la clase a partir de siete variables dadas. Con el conjunto de datos individuales se generará la infografía de la clase.

Las variables contempladas son:

- WhatsApps enviados y recibidos
- “Me gusta” y “comentarios” en Instagram
- Horas delante de pantallas (TV, PC, Tablet, smarthpone)
- Kilómetros recorridos en auto/moto, bicicleta, transporte público y a pie
- Horas dormidas
- Horas bailadas
- Horas de deporte

Para ello, cada estudiante irá completando en una hoja de cálculo con los valores levantados para cada una de las variables y con cálculos estadísticos simples deberá presentar una infografía o un video corto con las principales conclusiones.

Guíe una reflexión con el resto de la clase.

Lección 62

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Antes pensaba, ahora pienso

Todos hemos oído hablar de internet y de la nube, pero...

Antes pensaba

¿Dónde crees que se almacenan los datos que introdujiste en la sesión anterior?

¿Cómo crees que han viajado hasta allí?

Escribe tres repercusiones medioambientales que crees que tiene el transporte y almacenaje de datos.

Ahora pienso

Tras la visualización del video ¿Dónde viajan los datos?, consulta la infografía La red que soporta la red y la infografía interactiva ¿Cómo viajan los datos? y vuelve a contestar a las preguntas.

¿Dónde crees que se almacenan los datos que introdujiste en la sesión anterior?

¿Cómo crees que han viajado hasta allí?

Escribe tres repercusiones medioambientales que crees que tiene el transporte y almacenaje de datos.

Lección 63

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Analizando datos

Las infografías nos permiten visualizar y segmentar grandes cantidades de datos para analizarlos y obtener nuevas informaciones. Vamos a practicar usando la infografía y los datos de la Lección datos por todas partes.

Paso 1

Utilicen los datos de la Lección anterior. Pueden hacer comparaciones de clase, género u otros de las respuestas de todos los participantes. Experimenten haciendo varias comparaciones hasta que encuentren una que les llame la atención porque los datos que muestren sean muy diferentes unos de otros y anótenlo aquí.

Fuente de los datos 1	Fuente de los datos 2

Datos comparados que nos llaman la atención

Paso 2

Escriban la información que les muestra la comparación de datos escogida (por ejemplo, las chicas de una localidad envían la mitad de Whatsapps que las chicas de otra...).

Información

Paso 3

Imaginen ahora una posible explicación que nos diga el porqué de la información que han seleccionado.

Explicación

Recuerden que esto es solo una suposición y que debería verificarse con otros datos para convertirse en veraz.

Por último, lean las píldoras informativas [Big data: cuando los datos se vuelven inteligentes](#) y [La quinta V del big data](#) y anoten aquí las ideas para mejorar y presentar su supuesto, analizando las cinco V del big data.

Primera V – Volumen

¿Consideran suficiente el volumen de datos analizados para su supuesto? ¿Cómo lo ampliarían para obtener más valor en su análisis?

Segunda V – Velocidad

¿Con qué sistema podrían garantizar mayor velocidad en la recogida y análisis de datos?

Tercera V – Variedad

¿Qué otros datos completarían el análisis de su supuesto?

Cuarta V – Veracidad

¿Qué problemas de veracidad no están controlados en su supuesto? ¿Se podría resolver?

Quinta V – Visualización

¿De qué manera harían visibles los datos de su supuesto? Esbocen un esquema de cómo los presentarían para que la información sea clara y comprensible.

Unidad 04

¿Cómo se hace una app móvil?

Contenido 1:

Resolución de problemas

Contenido 2:

El proceso de diseño

Contenido 1

Resolución de problemas

Resumen

- **Lección 74:** Solución de problemas y datos
- **Lección 75:** Solución de problemas con big data
- **Lección 76:** Estructuración de datos
- **Lección 77:** Toma de decisiones con datos
- **Lección 78:** Interpretación de datos
- **Lección 79:** Automatizar decisiones de datos
- **Lección 80 - 81:** Proyecto- APP hacer una recomendación I y II

Objetivos

- **OA 5.** Desarrollar aplicaciones para dispositivos móviles y para dispositivos provistos de sensores y mecanismos de control.
- **OA j.** Desarrollar un trabajo colaborativo en línea para discusión y resolución de tareas matemáticas, usando herramientas electrónicas de productividad, entornos virtuales y redes sociales.

Referencias

- <https://studio.code.org/>
- <https://cuantrix.mx/> Fundación Televisa
- <https://educaixa.org/es/programa-bigdata>¹

Lección 74

Solución de problemas y datos

Propósito

En esta lección, los estudiantes usan el proceso de resolución de problemas. Después de revisar el proceso, se le presenta a la clase una decisión: si una ciudad debe construir una biblioteca, un refugio para mascotas o un departamento de bomberos. Los estudiantes trabajan en equipos para recopilar información en Internet para ayudarlos a decidir qué se debe construir, y luego usan esta información para construir un argumento que convenza a la municipalidad de su elección. A continuación, asignan lo que han hecho al proceso de resolución de problemas que han estado utilizando a lo largo del curso, comparando el proceso general de resolución de problemas con su aplicación específica a los problemas de datos.

Los estudiantes exploraron cómo las computadoras representan diferentes tipos de información o datos. En esta lección, aprenden cómo se pueden usar los datos para resolver problemas verbales reales, revisando el proceso de resolución de problemas a través del lente del análisis de datos.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Resolución de problemas con datos (70 min)

Proceso de resolución de problemas de datos

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Usar el proceso de resolución de problemas para responder una pregunta usando datos.
- Identificar y recopilar datos relevantes para ayudar a resolver un problema.
- Usar datos para sacar conclusiones.

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- Solución de problemas con datos- Ejemplar

Para los estudiantes:

- Solución de problemas con datos- Guía de actividades

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Observaciones: En la primera mitad de esta unidad, hablamos sobre diferentes tipos de datos, o información, que están almacenados en la computadora. Ahora vamos a ver cómo usamos los datos para resolver diferentes problemas.

Indicación: ¿Cuándo fue la última vez que buscó información en Internet? ¿Por qué lo necesitaba?

Observaciones: Todas estas cosas son datos, y todos nos ayudaron a resolver un problema en nuestras vidas. Hoy, vamos a ver nuestro proceso de resolución de problemas y cómo se puede usar junto con los datos.

Resolución de problemas con datos (70 min)

Grupo: Organice a los estudiantes en grupos de 3-4

Presente a los estudiantes el escenario como se explica en la guía de actividades. Puede elegir la mejor ubicación para tu clase o utilizar la ubicación de ejemplo de Columbia Mall en Columbia, MO.

Observaciones: Para esta lección, vamos a ser expertos contratados por el Concejo Municipal de Columbia, MO para dar consejos sobre lo que deben construir en una nueva ubicación. La ciudad tiene el dinero para construir algo cerca del Columbia Mall, pero el concejo no puede ponerse de acuerdo sobre qué es lo que más necesita la ciudad. Algunos miembros quieren una estación de bomberos, otros quieren un refugio de animales y otros quieren una biblioteca. Nuestro trabajo es investigar datos y utilizar el proceso de resolución de problemas para ayudarnos a hacer una recomendación convincente al consejo.

Revisión: Revise el proceso de resolución de problemas: definir, preparar, tratar, reflexionar.

Entregue la [Solución de problemas con datos- Guía de actividades](#) y deje que los estudiantes sepan que estarán trabajando en ella como clase.

Definir

Pida a los estudiantes que tomen unos momentos para escribir cuál es el problema y luego permita que lo discutan con un compañero. Permita que los estudiantes compartan sus

El objetivo de esta discusión es que los estudiantes se den cuenta de que usan Internet como recurso de información para ayudarlos a resolver problemas en sus propias vidas. Mientras los estudiantes intercambian ideas sobre por qué están en línea, anímelos a enmarcar su búsqueda de información como un paso en la resolución de problemas.

Los estudiantes deben entender que no solo se trata de tener una opinión sobre lo que se debe construir, sino de utilizar datos de datos para ayudar a decidir qué es lo que más necesita la ciudad.

respuestas y hablen como clase.

Preparar

Observaciones: Ahora que entendemos el problema, debemos decidir qué datos nos serán útiles. Todos escriban tres tipos de datos que pueden ayudarlos a decidir qué es lo que más necesita la ciudad.

Permita que los estudiantes compartan los diferentes tipos de datos y anótelos en el pizarrón. A medida que se sugieren más datos, pregunte a los estudiantes dónde creen que pueden encontrar esta información, centrándose en la información que está disponible en línea.

Esta es una gran cantidad de datos, y es posible que no podamos encontrarlo todo. Vamos a dividirnos en grupos para buscar esta información.

Asigne a cada grupo un tipo particular de datos para investigar en línea, asegurándose de que cada uno sepa dónde buscarlos.

Recorrer la sala: Mientras los grupos de apoyo intentan ubicar los datos en línea y registrar lo que han encontrado en la Guía de actividades.

Después de que cada grupo haya terminado, reúna a la clase para compartir los datos, y ponga los resultados de la investigación en el pizarrón.

Tratar

Una vez que todos los grupos hayan compartido sus datos, permita que cada grupo decida qué creen que debería estar en la ubicación y completa el cuadro con sus razones y datos. Asegúrese de que los estudiantes entiendan que pueden usar los datos de toda la clase, y no sólo los de su grupo en particular. Deben usar al menos dos tipos de datos para respaldar su decisión.

Recorrer la sala: Mientras los grupos de apoyo deciden qué se debe construir en la ubicación. Si los grupos no pueden llegar a un acuerdo, asegúreles que no existe una respuesta “correcta” para este problema, pero deben intentar averiguar qué pueden respaldar con dos tipos de datos.

Haga que los grupos compartan sus decisiones, así como los motivos y los datos que los respaldan. Asegúreles que está bien tener diferencia entre y dentro de los grupos.

Observaciones: Incluso las personas con los mismos datos a veces pueden llegar a conclusiones diferentes. Cuando tomas una decisión con datos, debes saber qué significan los datos y

Si los estudiantes tienen problemas para encontrar datos relevantes, puede intentar llevarlos a algunos de los siguientes sitios:

- ¿Qué bibliotecas, estaciones de bomberos o refugios para animales hay en el área? (Sitios de mapas, como Google Maps)
- ¿La gente está contenta con los servicios existentes? (revise los sitios, como Yelp)
- ¿Cuántas personas viven en la ciudad? ¿Qué tan grande es la ciudad? (Sitio web de la ciudad, o Wikipedia)

Los estudiantes deben comprender y sentirse cómodos con el hecho de que las decisiones que tomamos a partir de los datos implican la interpretación y la priorización. Deberían ser capaces de respaldar sus decisiones utilizando los datos que encontró la clase.

qué es importante para ti.

Reflexionar

Pida a los estudiantes que llenen la última porción de la hoja de trabajo, recordándoles que más datos pueden ayudarlos a tomar una mejor decisión.

Proceso de resolución de problemas de datos

Observaciones: Pudimos utilizar el proceso de resolución de problemas para ayudarnos a tomar una decisión con los datos, pero había algunas partes que se podían haber sentido nuevas.

Cuaderno: En su cuaderno, escriba una cosa que sintieran lo mismo sobre el uso del proceso de resolución de problemas, y dos cosas que se sintieran diferentes.

Después de que los estudiantes hayan terminado de escribir, dé unos minutos para compartir con un compañero y luego haz volver para una discusión grupal. Permita que los estudiantes compartan sus respuestas, y anótelas en el pizarrón, luego muestra el [Proceso de resolución de problemas para datos](#).

Observaciones: Aunque estamos usando el mismo proceso general de resolución de problemas, hacemos algunas cosas específicas cuando trabajamos con datos. Los pasos para el proceso de resolución de problemas de datos son: Definir, Recopilar, Interpretar y Reflexionar. Usaremos este proceso durante el resto del capítulo para ayudarnos a resolver problemas.

Los estudiantes deben tener en cuenta que el paso “Definir” del proceso fue muy similar, pero que el paso “Preparar” pedía a los estudiantes que **recogieran** datos, y el paso “Probar” les pidió que **interpretaran** los datos.

Transferencia del conocimiento (5 min)

Reflexión

Cuaderno: Pide a los estudiantes que piensen en un problema en sus vidas o en su comunidad que puedan resolver con datos. Deben responder las siguientes tres preguntas:

- ¿Cuál es el problema?
- ¿Qué datos podrían ayudarte a resolverlo?
- ¿Dónde podrías encontrar esa información?

Sugerencias de evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Organizan los datos recolectados y los presentan explicando el proceso

Lección 75

Solución de problemas con Big Data

Propósito

En esta lección, los estudiantes observan cómo las organizaciones recopilan y utilizan los datos para resolver problemas en el mundo real. La lección comienza con una revisión rápida del proceso de resolución de problemas de datos que ya han explorado. A continuación, se presentarán tres escenarios que podrían resolverse utilizando datos y una lluvia de ideas sobre los tipos de datos que desearían resolver y cómo podrían recopilar los datos. Cada problema está diseñado para reflejar un servicio del mundo real que existe. Después de una tormenta de ideas, los estudiantes ven un video sobre un servicio en el mundo real y registran notas sobre qué datos recopila el servicio del mundo real y cómo se usan. Al final de la lección, los estudiantes registran si los datos fueron provistos activamente por un usuario, si se registraron de forma pasiva o si fueron recopilados por sensores.

En esta lección, los estudiantes verán tres ejemplos de cómo se usa el proceso de resolución de problemas de datos para resolver problemas del mundo real. Esta lección amplía los tipos de problemas que los estudiantes consideran problemas de datos y les ayuda a relacionar lo que saben sobre los datos con sus experiencias del mundo real con servicios de Internet comunes. Los ejemplos también brindan la oportunidad de reflexionar sobre el hecho de que en sus propias vidas están produciendo intencionalmente y sin intención datos que las compañías recopilan y usan.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (10 min)

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Datos en el mundo real- Guía de actividades](#)
- [Video Code studio \(Netflix\)](#)

Objetivos

Los estudiantes serán capaces de:

- Brindar ejemplos de cómo se recopilan los datos de los sensores y el seguimiento del comportamiento del usuario.
- Determinar los datos que serían útiles para resolver un problema y cómo podrían recopilarse esos datos.
- Distinguir entre los datos que los usuarios producen intencionalmente y sin intención.

Preparación

- [Datos en el mundo real- Guía de actividades](#)
- Prepare el proyector para mostrar videos en la clase.

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Pregunte: En la lección previa, miramos el proceso de resolución de problemas de datos usando datos que encontramos en Internet. No nos preocupamos por cómo se pusieron en línea esos datos. Aunque hay muchas maneras en que las aplicaciones, las empresas o los gobiernos pueden recopilar datos. ¿Qué formas de recopilar datos ya conoce?

Discuta: Haga que los estudiantes piensen en sus ideas antes de compartirlas con la clase. Puede registrar sus ideas en el pizarrón para consultarlas más adelante en la clase.

Observaciones: Buen trabajo. Hoy vamos a ver algunos ejemplos del mundo real de cómo se recopilan los datos para resolver problemas. Esté atento a estas ideas y piense si está viendo otras nuevas.

Ampliación del conocimiento (40 min)

Grupo: Los estudiantes pueden completar esta lección individualmente o en parejas.

Distribuir: Dar a los estudiantes [Datos en el mundo real- Guía de actividades](#)

[Datos en el mundo real- Web Pix](#)

Presente a los estudiantes el problema “Web Pix” como se define en la primera página. Dé a los estudiantes unos minutos para escribir sus ideas y / o compartirlas con un compañero. Haga saber a los estudiantes que aún no podrán responder la pregunta sobre Netflix. Si los estudiantes terminan temprano, pida que piensen en otros tipos de datos y cómo se pueden recopilar. Permita que los estudiantes compartan rápidamente algunas ideas, luego presente el video de Netflix.

Observaciones: Muchas empresas, como YouTube, Facebook y Netflix, recomiendan videos y publicaciones a los usuarios en función de los datos que han recopilado sobre ellos. Vamos a ver un breve video sobre cómo Netflix hace esto. Preste atención a que Netflix recopila datos para ayudarlo a recomendar buenos videos.

[Video Code studio \(Netflix\)](#)

Objetivo: Esto es principalmente una lluvia de ideas. Algunos estudiantes pueden aportar mucho más conocimiento previo que otros a la clase y en este punto no se les ha enseñado explícitamente nada sobre la recopilación de datos fuera de las encuestas. Apunte principalmente para obtener ideas y establecer el escenario para la lección. Su objetivo es cambiar el enfoque de las encuestas que 10- 20 personas llevan a la gran cantidad de datos que podrían saber que son recolectados por las herramientas tecnológicas modernas.

Acceso a videos: se recomienda que la clase vea los videos de esta lección juntos en una pantalla grande o un proyector. Se pueden encontrar en sus propios niveles en Code Studio. Los estudiantes también tienen acceso a estos videos y, si lo desean, también pueden verlos por sí mismos.

Discute: Permita que los estudiantes compartan con un compañero, luego discutan con un grupo los tipos de datos que recopila Netflix para ayudarlo a hacer recomendaciones.

Routz

Pida a los estudiantes que pasen al problema de Routz. Una vez más, deberían tomarse unos minutos para trabajar individualmente o de a dos en los primeros dos problemas. Después de que los estudiantes hayan compartido con sus pares, presente el video de Waze.

[Video Code studio \(Waze\)](#)

Discute: Permita que los estudiantes compartan con un compañero, y luego dirija una breve discusión sobre los tipos de datos que Waze recopila para ayudarlo a encontrar la mejor ruta.

Nyle

Pida a los estudiantes que pasen al problema de Nyle. Una vez más, deberían tomarse unos minutos para trabajar individualmente o de a dos en los primeros dos problemas. Después de que los estudiantes hayan compartido sus pares, presenta el video de Amazon. Video Code studio (Amazon)

Discute: Permita que los estudiantes compartan con un compañero, y luego dirija una breve discusión sobre los tipos de datos que recopila Amazon para ayudarlo a encontrar los mejores productos para sugerir.

Reflexión: Revise las diferentes formas en que se pueden recopilar los datos.

Observaciones: Hemos encontrado muchas maneras diferentes en que la computadora puede recopilar datos. En general, los datos pueden recopilarse automáticamente de los sensores, ya que la aplicación Waze recopila datos de GPS; se puede recopilar de los usuarios que tienen la intención de producir datos, como cuando las personas evalúan los videos de Netflix; y puede recopilarse de usuarios que no están destinados a producir datos, como cuando Amazon registra qué productos mira.

Tipos de recopilación de datos: Pide a los estudiantes que llenen el cuadro al pie de la página que clasifica los diferentes tipos de datos. Luego, permite que los estudiantes comparen sus respuestas.

Objetivo: los estudiantes deben entender que Netflix basa sus sugerencias no solo en los datos de la encuesta que crean los usuarios, sino también en el comportamiento de los usuarios a medida que utilizan el sitio, como lo que miran y las clasificaciones que ofrecen en varios videos.

Objetivo: los estudiantes deben tener en cuenta que, aunque los usuarios agregan algunos datos de manera intencionada, Waze también recopila datos de GPS/ubicación de cada usuario sin la entrada activa del usuario. Los datos del sensor, como el tiempo, la temperatura y la ubicación, son recolectados automáticamente por las computadoras sin que el usuario los agregue activamente.

Objetivo: Hay algunos aspectos diferentes del video de Amazon que pueden ser de interés.

- El solo hecho de mirar algo en línea produce datos que pueden ser utilizados por anunciantes u otros.
- Amazon decide lo que puedes comprar al mirar a usuarios similares y usar su comportamiento para predecir el tuyo.
- Los diferentes tipos de datos, como “clics”, “me gusta” y “compras”, pueden

Transferencia del conocimiento (10 min)

Cuaderno: Hoy analizamos tres empresas que recopilan datos para resolver problemas. Haga una lluvia de ideas sobre otros sitios web, aplicaciones o empresas que usa o conoce. ¿Qué datos están recolectando? ¿Cómo lo están usando para resolver un problema?

Discuta: Pida a los estudiantes que compartan sus ideas con sus compañeros de clase.

Observaciones: Hoy vimos algunos ejemplos de diferentes fuentes de datos que las aplicaciones del mundo real y los sitios web usan para resolver problemas. Los sitios web a menudo le piden datos directamente, pero también pueden registrar su comportamiento en línea para recopilar datos. De hecho, a veces los sensores como una señal de GPS pueden recopilar datos sin que lo sepas. En la próxima clase vamos a ver cómo recopilaremos datos.

Sugerencias de evaluación

Se sugiere los siguientes indicadores para evaluar formativamente los aprendizajes:

- Presentan conclusiones de sus comparaciones describiendo y argumentando

– ponderarse de forma diferente.

- Las necesidades del usuario no son la prioridad. Las necesidades del anunciante son: Cualquiera de estos temas es relevante para la lección, pero lo más importante para que los estudiantes se den cuenta es que no siempre están al tanto de cuándo están produciendo datos.

Objetivo: Este mensaje está destinado a ayudar a los estudiantes a establecer conexiones entre sus experiencias personales con los servicios de recopilación de datos y lo que han aprendido en esta Lección. El objetivo aquí es conectar los ejemplos que los estudiantes vieron en la Lección de hoy con otras aplicaciones, sitios web o servicios de los que puedan estar enterados. Si necesita solicitarles a los estudiantes, puede sugerirles que piensen en sitios web de redes sociales, sitios web de medios, aplicaciones útiles que ellos o sus familias usan, etc.

Lección 76

Estructuración de datos

Propósito

En esta lección, los estudiantes profundizan en la recopilación e interpretación de datos, incluida la limpieza y visualización de datos. Los estudiantes primero observan cómo la presentación de datos de diferentes maneras puede ayudar a las personas a comprenderlo mejor, y luego crean visualizaciones de sus propios datos. Utilizando los resultados de una encuesta de pizza preferida, los estudiantes deben decidir qué hacer con los datos que no encajan fácilmente en el esquema de visualización que han elegido. Finalmente, los estudiantes observan qué partes de este proceso pueden ser automatizadas por una computadora y cuáles necesitan un ser humano para tomar decisiones.

Esta lección demuestra que los datos sin procesar deben interpretarse de alguna manera para ayudar a las personas a usarlos para tomar decisiones. Los estudiantes participan tanto en la visualización como en la limpieza de los datos, y ven cómo los datos pueden malinterpretarse si no se limpian adecuadamente. Los estudiantes también experimentan el trabajo con datos a mano y con herramientas computacionales, y ven cómo los datos deben estructurarse de maneras particulares para ser utilizados por una computadora.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Visualización de datos (70 min)

Transferencia del conocimiento (15 min)

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Datos de pizza \(descarga de csv\): recurso opcional](#)
- [Datos de pizza \(GSheets\) - Recurso opcional](#)
- [Estructurando Datos- Ejemplar](#)

Para los estudiantes:

- [Estructuración de datos - Guía de actividades](#)

Objetivos

Los estudiantes serán capaces de:

- Identificar y eliminar datos irrelevantes de un conjunto de datos.
- Crear un gráfico de barras basado en un conjunto de datos.
- Explicar por qué se debe limpiar un conjunto de datos antes de que una computadora pueda usarlos.

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Dirija a los estudiantes a los Niveles de Code Studio correspondientes a la lección.

Pida a los estudiantes que piensen por sí mismos por un momento, luego discutan sus respuestas con un compañero.

Discute: Haga que los estudiantes compartan sus respuestas para las preguntas en el pizarrón.

Observaciones: A veces, los datos “en bruto”, la forma en que se recopila la información por primera vez se debe poner en una forma diferente para que los humanos y las computadoras puedan entender más fácilmente lo que significa.

Visualización de datos (70 min)

Agrupe a los estudiantes en parejas y entregue a cada pareja Estructuración de datos- Guía de actividades.

Lea las instrucciones en conjunto, asegurándose que los estudiantes comprendan el problema que están tratando de resolver (eligiendo un relleno de pizza para la fiesta de pizza). Se les pide a los estudiantes que creen el gráfico de barras para el conjunto de datos brutos proporcionados. Algunas de las respuestas no se relacionarán fácilmente en las elecciones dadas. Anime a los estudiantes a usar su mejor juicio sobre las respuestas que son difíciles de poner en la tabla, y que estos desafíos son una parte normal del proceso de resolución de problemas de datos.

Discute: Después de que los estudiantes terminen de elaborar la tabla y llenen las preguntas de reflexión, haz que compartan sus respuestas con la clase

Observaciones: Podemos hacer el gráfico a mano, pero también es posible que la computadora lo haga por nosotros. Esto es especialmente útil cuando tienes muchos datos.

Envíe a los estudiantes al ejercicio 3, y pida que sigan las instrucciones en el nivel.

Pida a los estudiantes que discutan por parejas por qué la tabla se ve de la forma en que lo hace y luego compartan sus respuestas con la clase. ¿Por qué la computadora no pudo poner todo en la categoría correcta?

Cuando creamos nuestras tablas, sabíamos que necesitábamos

Los estudiantes deben comprender que las diferentes formas de datos facilitan que las personas tomen decisiones. También deberían ver que a la gente a menudo le va mejor con imágenes, como el gráfico de barras, mientras que la computadora funciona mejor con números, como la tabla.

Los estudiantes deben ver que hay varias formas en que las respuestas pueden ser difíciles de categorizar, ya sea que sean completamente irrelevantes, que no sean lo suficientemente específicas o que no sean una opción dada. Ignore la ortografía por ahora si los niños no la mencionan.

También puede completar esta lección usando Hojas de cálculo de Google o Excel. Los archivos de hoja de cálculo relevantes están vinculados en el Nivel 3 de la lección en línea o en el área de enlaces de recursos de este plan de lección.

dejar fuera algunas de las respuestas que no tenían sentido, y que algunas respuestas, como “pimientos” y “pimientos verdes”, realmente significaban lo mismo. También colocamos todo lo que había sido mal escrito en la categoría correcta. Las computadoras no saben cómo hacerlo, porque en realidad no entienden qué es un “pimiento”, o que una palabra mal escrita es igual a una palabra correctamente deletreada. Eso significa que tenemos que limpiar los datos antes de que la computadora pueda usarlos.

Diga a los estudiantes que crearán una nueva columna de datos “limpios” que será más fácil de interpretar para la computadora.

Envíe los estudiantes al ejercicio 4.

Haga clic en un encabezado en la lista “Datos limpios” y editarlo o eliminarlo. Demuestre que cuando eliminas / cambias las respuestas en la columna de limpieza de datos, el gráfico cambia automáticamente.

Pida a los estudiantes que, en parejas, limpien los datos hasta que solo se muestren las siete elecciones originales, y luego decidan qué topping (ingrediente) de pizza es la mejor opción.

Observaciones: Esto fue mucho trabajo, y solo fueron unos cincuenta votos.

¿Cuánto tiempo cree que tomaría limpiar los datos para una encuesta nacional? ¿Puede pensar en alguna forma de asegurarse de que tenemos datos claros desde el principio, para que ahorrar todo este trabajo?

Permita que los estudiantes lo discutan por parejas y luego compártalos con la clase.

Cuando trabajamos con grandes cantidades de datos, queremos automatizar tanto como podamos el proceso de mejora de problemas. Debido a que las computadoras no pueden hacer las mismas conexiones que las personas, eso significa que las personas tienen que ayudar a organizar los datos de manera que las computadoras puedan entenderlos. Eso significa limpiar los datos o recopilar datos de una manera que asegure que estén limpios cuando los obtengamos.

Transferencia del conocimiento (15 min)

Preguntar: Pida a los estudiantes que reflexionen sobre el desarrollo de las cinco prácticas de Descubrimientos CS (Resolución de problemas, Persistencia, Creatividad, Colaboración, Comunicación). Elija una de las siguientes indicaciones según lo considere apropiado.

Los estudiantes deben tener en cuenta que la computadora usó todas las respuestas en la tabla, incluso las que fueron irrelevantes. También deberían tener en cuenta que las ortografías diferentes de la misma elección no se agruparon juntas.

Al final, los estudiantes deben darse cuenta de que restringir las elecciones de un usuario mediante el uso de opciones múltiples en lugar de una respuesta de escritura hace que sea más fácil para una computadora usar los datos.

- Elija una de las cinco prácticas en las que crees que demostraste crecimiento en esta lección. Escriba algo que hizo que ejemplifica esta práctica.
- Elija una práctica en la que piense que puede seguir creciendo. ¿Qué le gustaría mejorar?

Elija una práctica que pensaste que era especialmente importante para la lección que completamos hoy. ¿Qué lo hizo tan importante?

Sugerencias para evaluar

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Presentan la información trabajada, formas de trabajarlo y conclusiones

Lección 77

Tomar decisiones con datos

Propósito

En esta lección, los estudiantes practican la toma de decisiones con datos basados en algunos problemas diseñados para que sean familiares para los estudiantes de secundaria. Los estudiantes trabajan en grupos discutiendo cómo usarían los datos presentados para tomar una decisión antes de que la clase discuta sus elecciones finales. No todas las preguntas tienen respuestas correctas y, en algunos casos, los estudiantes pueden y deben decidir que deben recopilar más datos. La lección concluye con una discusión sobre cómo diferentes personas podrían sacar conclusiones diferentes de los mismos datos, o cómo la recopilación de datos diferentes podría haber afectado las decisiones que tomaron.

Los estudiantes comienzan a explorar el proceso de resolución de problemas de datos en esta lección al final, tomando decisiones con datos que ya se han recopilado e interpretado. Los estudiantes probablemente no entren en este curso con mucha experiencia tomando decisiones basadas en datos de esta manera. Esta Lección intenta contextualizar este paso del proceso dentro de las decisiones que pueden ser familiares para su vida diaria.

Tomar decisiones con datos no es un proceso de formulación. Diferentes personas pueden sacar conclusiones diferentes con la misma información. Es importante que los estudiantes funden sus decisiones en los datos recopilados, pero generalmente no hay una “respuesta correcta”. Esta Lección es una oportunidad para introducir los matices de la toma de decisiones en un contexto controlado y compartido antes de que los estudiantes comiencen a recopilar más información propia.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (10 min)

Lección sin conexión
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Tomar Decisiones con Datos- Ejemplar](#)

Para los estudiantes:

- [Toma de decisiones con datos- Guía de actividades](#)

Objetivos

Los estudiantes serán capaces de:

- Usar tablas y visualizaciones que resuman datos para respaldar una decisión
- Interpretaciones actuales y críticas de tablas
- Identificar datos adicionales que podrían recopilarse para mejorar una decisión.

Preparación

- [Toma de decisiones con datos- Guía de actividades.](#)

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Preguntar: ¿Qué decisiones en su vida están recabando para ayudar a hacer algo? ¿Hay alguna decisión que piensen que los datos podrían ayudarlos a mejorar?

Discutir: Los estudiantes deben intercambiar ideas en silencio antes de compartirlas en grupos pequeños o directamente con la clase. No es necesario explorar esta cuestión con gran detalle, ya que podrá volver a ella al final de la clase.

Observaciones: Tomamos decisiones todos los días, pero no siempre tomamos decisiones usando datos. Hoy vamos a comenzar a explorar el proceso de resolución de problemas de datos que aprendimos en la última clase al saltar al último paso, tomar una decisión.

Ampliación del conocimiento (40 min)

Tomar decisiones con datos

Grupo: Los estudiantes pueden completar esta lección en grupos de 3 o 4 para que tengan compañeros con quienes hablar. Sin embargo, no es necesario que los grupos tomen decisiones compartidas.

Distribuir: [Toma de decisiones con datos- Guía de actividades](#)

Problema 1- Cuándo publicar:

Lea las secciones definir y preparar del problema como clase. Luego brinde a los grupos la oportunidad de ver los datos que se les presentan y reflexionar sobre las preguntas al final de la guía de actividades. Los estudiantes deben reflexionar individualmente sobre sus respuestas a cada pregunta, luego discutir la decisión que tomarán con el grupo. Una vez que los grupos han tenido la oportunidad de debatir, deben registrar sus decisiones, incluso si son diferentes entre sí.

Compartir: Haga que un miembro de cada grupo comparta los resultados de su discusión.

Problema 2- Hacer comidas:

Haga que los estudiantes repitan el proceso para este segundo problema. Deben revisar el problema, tomar una decisión

Objetivo: Esta lección debería ser una conversación de apertura muy rápida y está muy bien si los estudiantes no toman ninguna decisión con los datos. Puede señalar un par de ejemplos (por ejemplo, mirar el clima para decidir qué ponerse o revisar los datos sobre un producto en línea), pero está bien simplemente usar esto para abrir la clase y volver a la pregunta más adelante.

Asignar roles: haga que los grupos asignen a alguien responsable de compartir los resultados de su discusión antes de analizar cada decisión. Papeles alternos entre problemas.

Enfóquese en los datos: desde el principio, refuerce la importancia de que los estudiantes se refieran a los datos como fuente de su decisión. Si los estudiantes piensan que hay razones no representadas por los datos que les gustaría tomar su decisión, pregúntales si hay otros datos que quieran recopilar. Por ejemplo:

- Datos del fin de semana
- Datos de más de una semana
- Datos sobre lo que la gente está publicando (aquí no hay forma de saberlo)

personal sin escribirla, discutir con un grupo, registrar su decisión final y luego estar preparados para compartir en grupo.

Compartir: Nuevamente los grupos comparten cómo abordarían esta decisión.

Problema 3- Selección del casillero:

Repita el proceso por última vez con este problema.

Compartir: Haga que los estudiantes compartan los resultados de su conversación con la clase.

Transferencia del conocimiento (10 min)

Preguntar: ¿Dos personas necesitan tomar la misma decisión a partir de los mismos datos?

Discute: Haga que los estudiantes compartan sus respuestas.

Pregunta: Para los problemas que hemos analizado hoy, ¿hay datos diferentes que hubieran querido recopilar para tomar su decisión?

Discute: Haga que los estudiantes compartan sus respuestas

Observaciones: Sólo miramos formas en que los datos pueden ayudarnos a tomar una decisión. Vimos que las decisiones diferentes pueden venir de los mismos datos. También sabemos que estamos en el último paso del proceso de resolución de problemas de datos aquí. Dependiendo de qué datos hayamos recopilado, podríamos haber abordado estas decisiones de manera muy diferente. De ahora en adelante, vamos a explorar esos pasos anteriores con más detalle y veremos cómo afectan nuestras decisiones.

Sugerencias para evaluar

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Presentan la información trabajada, formas de trabajarlo y conclusiones

Interpretar el cuadro: El segundo y el tercer problema tienen tablas de información más complejas que pueden ser difíciles de entender para los estudiantes sin una guía. Es posible que desee modelar cómo leer los cuadros en estas dos actividades antes de pedir a los estudiantes que los utilicen para la toma de decisiones. Por ejemplo, pregunte a los estudiantes qué significa una celda particular en una mesa.

¿Dónde están mis amigos?:

Los estudiantes a menudo reaccionan al problema 3 diciendo que les importan otros datos, como dónde están los casilleros de sus amigos. No tenga miedo a decir que, por el momento, deben reflexionar sobre cómo los datos influyen en la forma en que tomamos decisiones.

Objetivo: Esta es una pregunta bastante matizada que se hace pasar por un simple sí/no. Pídales a los estudiantes que reflexionen sobre los momentos de las actividades de hoy en los que creen que los datos estaban más abiertos a la interpretación (por ejemplo, la decisión de los casilleros) y cuándo podrían ser más simples (por ejemplo, cuándo publicarlos). Si bien el objetivo de esta discusión es resaltar el hecho de que dos personas no deberían llegar a la misma conclusión a partir de los mismos datos, se necesitan seguimientos para hacer que esta conversación sea más efectiva.

Objetivo: Esta conversación puede recordar momentos en que los estudiantes querían más o diferentes datos para ayudarlos a tomar decisiones. Por ejemplo, es posible que desee saber dónde están los casilleros de su amigo antes de decidir cuál quiere. Esta discusión conduce a los comentarios finales de la Lección.

Lección 78

Interpretación de datos

Propósito

Los estudiantes comienzan la lección mirando una encuesta que les permite a los encuestados especificar la preferencia de un pastel como la de una cobertura (glaseado). Discuten cómo saber la relación entre el pastel y la preferencia de sabor les ayuda a decidir mejor qué combinación recomendar. Luego se les presenta la tabulación cruzada, lo que les permite graficar relaciones con diferentes preferencias. Usan esta técnica para encontrar relaciones en una encuesta de preferencia, luego intercambian ideas sobre los diferentes tipos de problemas que este proceso podría ayudar a resolver.

En las dos lecciones anteriores, los estudiantes utilizaron la visualización de datos para ayudarlos a tomar decisiones sobre una sola variable (qué casillero elegir). En esta lección, aprenden cómo encontrar relaciones entre variables usando la tabulación cruzada en las respuestas a diferentes preguntas de la encuesta. Determinar cómo se relacionan las opciones de respuesta entre sí les permitirá hacer predicciones sobre los usuarios en función de las respuestas anteriores. En el proyecto final, usarán este mismo tipo de análisis para ayudarlos a diseñar un algoritmo para su generador de recomendaciones.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Organizar los datos visualmente para resaltar las relaciones y respaldar un reclamo.
- Usar la tabulación cruzada para encontrar patrones y relaciones en los datos.

Lección sin conexión

[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Interpretación de datos - Guía de actividades](#)
- [Interpretación de datos- ejemplar](#)
- [Interpretación de datos- recurso](#)

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Pantalla: Muestre a los estudiantes la tabla en el [Nivel 2 de Code Studio](#).

Observaciones: Aquí hay algunos resultados más de la encuesta, pero esta vez, en lugar de mirar los ingredientes de la pizza, estamos viendo sabores de pasteles y cobertura.

Indicación: Si pudieras elegir un pastel con cobertura, ¿cuál sería?

Preguntar: ¿Qué pasaría si pudiera elegir dos pasteles con cobertura?

Pregunta: ¿De qué forma conocer ambas opciones nos ayuda a entender mejor qué tipo de pastel deberíamos pedir?

A veces no es suficiente mirar solo un tipo de datos. Necesitas ver cómo se relacionan los diferentes tipos de datos. Hoy veremos de una manera que podemos encontrar relaciones en los datos para ayudarnos a resolver problemas.

Ampliación del conocimiento (40 min)

Grupo: Organice a los estudiantes en grupos de 2-3.

Distribuir: Entregue a cada grupo [Interpretación de datos - ejemplar](#) e [Interpretación de datos- recurso](#).

Observaciones: Para nuestro ejemplo de pastel y cobertura, sólo hubo ocho resultados, por lo que pudimos ver las respuestas y obtener una buena idea de las relaciones entre ellos. En esta encuesta, tenemos muchos más resultados, por lo que vamos a utilizar una tabla para contarlos.

Lean las instrucciones como clase y luego pida a los estudiantes que miren la primera tabla de la guía de actividades.

Modelo: Muestre la primera tabla en el pizarrón y modela cómo completar la tabla con la clase. Para cada fila de resultados de la encuesta, agregue un recuento a la tabla.

Preguntar: Si a alguien le gustan los gatos, ¿qué lección es probablemente su favorita?

Permita que los estudiantes escriban su respuesta, luego verifica con tu grupo antes de compartir.

Preguntar: ¿Cómo cambiaría su respuesta si se le dijera que a la

No hay una sola recomendación que sea correcta, pero asegúrese de que los estudiantes entiendan que, aunque el chocolate fue el sabor de pastel más popular y el queso crema fue el sabor más popular, sólo una persona eligió un pastel de chocolate con cobertura de queso crema.

No es suficiente mirar las dos respuestas en forma aislada. Por ejemplo, si se eligen dos pasteles, el pastel de chocolate con cobertura de chocolate y el pastel de zanahoria con cobertura de queso crema es mucho mejor que el chocolate con queso crema y la zanahoria con chocolate. Mirar las relaciones entre las respuestas ayuda a ver qué opciones van bien juntas.

Aunque cada una de estas hojas tiene sólo una página, es mejor imprimirlas por separado, para que los estudiantes puedan ver los resultados de la encuesta mientras completan la tabla. Las clases que completen estas actividades en línea pueden usar dos computadoras por grupo, una para mostrar los resultados de las encuestas y otra para completar la tabla.

persona le gustan los perros?

¿Cuál es una relación más interesante entre la mascota y la elección favoritas?

Indique a los estudiantes que completen la hoja de trabajo en sus grupos.

El siguiente cuadro relaciona mascotas y deportes. Los estudiantes completan el cuadro y luego encuentran dos relaciones interesantes entre las preferencias de las mascotas y las preferencias deportivas. Los estudiantes repiten la lección para las preferencias de lección y deporte.

Antes de pasar a la pregunta de reflexión, brinde a los estudiantes la oportunidad de compartir todo lo interesante que hayan aprendido sobre las relaciones entre las diferentes preferencias.

Preguntar: ¿Cómo podrían las relaciones de conocimiento entre estos tipos de preferencias ayudarlo a abordar un problema del mundo real?

Transferencia del conocimiento (5 min)

Rápido:

1. ¿En qué otro problema de datos se podría utilizar este método?
2. ¿Qué preguntas harías?
3. ¿Qué relaciones buscarías?

Sugerencias para evaluar

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Presentan una consolidación y conclusión de la información trabajada.

Los estudiantes deben usar la tabla para encontrar relaciones entre las preferencias para que puedan diferenciar entre subgrupos. Pueden notar que, aunque las personas que eligen los gatos tienden a elegir el arte, a las personas que eligen los perros les gusta la música. Asegúrese de que la clase produzca ejemplos de las predicciones que funcionan en ambas direcciones (mascota a elección y elección a mascota). Por ejemplo, las personas que les gustan los videojuegos prefieren los perros.

Aunque no hay una respuesta "correcta" a esta pregunta, los estudiantes deben darse cuenta de que encontrar relaciones entre las preferencias puede ayudarlos a predecir una preferencia de conocer a otra. Esto podría ser para un motor de recomendación, para colocar anuncios o para promocionar determinadas publicaciones en las redes sociales.

Lección 79

Automatizar decisiones de datos

Propósito

En esta lección, los estudiantes observan un ejemplo simple de cómo se podría usar una computadora para completar el proceso de toma de decisiones del proceso de resolución de problemas de datos. Los estudiantes tienen la tarea de crear un algoritmo que podría sugerir un lugar de vacaciones. Los estudiantes luego crean reglas, o un algoritmo, que una computadora podría usar para tomar esta decisión automáticamente. Los estudiantes comparten sus reglas y qué opciones tomarían sus reglas con los datos de la clase. A continuación, utilizan sus reglas sobre los datos de sus compañeros de clase para probar si sus reglas tomasen la misma decisión que una persona. La lección concluye con una discusión sobre los beneficios y desventajas de usar computadoras para automatizar el proceso de resolución de problemas de datos.

Esta lección demuestra a los estudiantes que el último paso en el proceso de resolución de problemas de datos, tomar una decisión, es algo que una computadora puede hacer automáticamente si recibe un algoritmo. Se basa en la lección anterior que demuestra la importancia de diseñar la manera en que se recopilan los datos para que se puedan usar para la interpretación. Esto es especialmente cierto para las computadoras que son mucho más adecuadas para los datos de estilo de opción múltiple que verán en esta Lección.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (5 min)

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Automatizar decisiones de datos- Guía de actividades](#)
- [Automatizar decisiones de datos- Recursos](#)

Objetivos

Los estudiantes serán capaces de:

- Diseñar un algoritmo para tomar decisiones usando datos como entradas
- Explicar los beneficios y desventajas de usar computadoras para tomar decisiones automatizadas
- Interpretar los datos recopilados para identificar patrones

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Indicación: Imagine que va a programar una computadora para seleccionar automáticamente la ropa de tu armario por la mañana. ¿Qué tipo de datos necesitaría esta computadora? ¿Qué tipo de reglas te gustaría que usara?

Discute: Los estudiantes deben desarrollar silenciosamente respuestas, luego compartir en pequeños grupos, luego discutir con toda la clase.

Observaciones: Para que una computadora tome una decisión, necesita datos como entrada y un algoritmo para procesarla. Este es solo el modelo input-output-store-process del que hemos aprendido anteriormente. En la última clase aprendimos a interpretar datos para entender más sobre el mundo. Hoy vamos a ver cómo diseñar algoritmos para que una computadora pueda usar lo que entendemos para tomar decisiones automáticamente.

Ampliación del conocimiento (40 min)

Creando el algoritmo

Distribuir:

- [Automatizar decisiones de datos- Guía de actividades](#)
- [Automatizar decisiones de datos- Recurso](#)

Observaciones: Imagine que quiere utilizar una computadora para analizar las respuestas de alguien y hacer una recomendación automáticamente. Una computadora no sabe lo que es “la playa” o “la gran ciudad” y no tiene una opinión propia. Solo podrá ver qué respuesta eligió alguien, no el significado de esa respuesta. Se necesita una persona que le diga qué hacer para convertir las opciones de respuesta en una recomendación.

Introducir lección: Lea las instrucciones como una clase, asegurándote de que los estudiantes entiendan cómo funciona el algoritmo.

Preguntar: Según lo que ven en las tablas de tabulación cruzada, ¿por qué creen que alguien creó la primera regla del algoritmo? ¿Puedes pensar en una regla mejor?

Haga reglas: Haga que los estudiantes decidan individualmente sobre las reglas para sus algoritmos. Para cada opción de respuesta

Objetivo: Esta discusión puede ser difícil de mantener pequeña, pero el punto es solo hacer que los estudiantes piensen sobre el hecho de que las computadoras necesitan datos (entradas) y reglas (es decir, un algoritmo) para tomar una decisión (salida). Siempre que los estudiantes hayan tenido la oportunidad de ver un ejemplo de esos tres componentes del modelo de entrada-salida-tienda- proceso de la computadora, está bien seguir adelante. Dicho esto, he aquí algunas ideas que los estudiantes podrían mencionar

- Entradas: la temperatura, el clima, qué tipo de eventos tiene hoy, etc.
- Reglas: si la temperatura es inferior a 60 años, traiga una chaqueta, si el evento de lujo saca la ropa de lujo, si lleva gafas de sol, etc.

Asegúrese de que los estudiantes comprendan que la regla se basa en la primera fila de la tabla “Vacaciones y alimentos”. No hay una regla “correcta” basada en los datos, pero es muy probable que alguien que prefiera el helado prefiera la playa y que prefiera un parque de diversiones.

posible, los estudiantes deben agregar puntos a al menos una de las cuatro opciones de playa, parque de atracciones, parque nacional o gran ciudad.

Probando el Algoritmo

Grupo: Asignar estudiantes a grupos de 3-4

Una vez que los estudiantes hayan completado el algoritmo, deben sondear a otros dos estudiantes y registrar sus respuestas en la segunda página de la hoja de trabajo. Luego usan sus algoritmos para recomendar un lugar de vacaciones para esa persona.

Compartir decisiones: Haga que los grupos compartan los lugares de vacaciones elegidos en función de las diferentes opciones de respuesta, y si creen que el algoritmo fue una buena recomendación. Si no estaban satisfechos con las recomendaciones, anímelos a sugerir cambios en el algoritmo.

Transferencia del conocimiento (5 min)

Revisión: Esta Lección está estrechamente relacionada con el proceso de resolución de problemas de datos. Repasa con los estudiantes que procesan rápidamente y pide que indiquen dónde ven los datos que se recopilan, interpretan y toman una decisión. Pide que señalen las diferencias en el proceso ahora que una computadora toma la decisión.

Indicación: Utilice la sección Reflexión de la guía de actividades para que los estudiantes reflexionen sobre las siguientes indicaciones:

- ¿Cuál es la desventaja de utilizar un algoritmo para tomar decisiones?
- ¿Qué tipo de decisiones no le gustaría que una computadora haga automáticamente?

Sugerencias para evaluar

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Utilizan esquemas y diagramas para organizar la información
- Presentan el proceso y las conclusiones del trabajo realizado

Objetivo: Esta discusión se puede utilizar para seguir revisando el proceso de resolución de problemas de datos y también para establecer conexiones entre esta Lección y los lugares donde los estudiantes pueden haber visto que las computadoras toman decisiones. Los estudiantes deben comprender que la automatización de decisiones es conveniente en muchas situaciones, pero pueden perder algunos elementos que los humanos considerarían en una decisión. Por ejemplo:

- Recopilar datos que ya están limpios limita lo que podrías recopilar
- Automatizar las decisiones significa que a veces se obtiene una decisión que no habría tomado por su cuenta.

Sin embargo, hay casos en los que podríamos querer automatizar una decisión. Por ejemplo:

• Donde un ser humano puede olvidarse/aburrirse (por ejemplo, reordenar automáticamente los alimentos cuando el refrigerador está bajo, encender/apagar el aire acondicionado cuando cambia la temperatura)

• Cuando hay una gran cantidad de datos a

considerar para una decisión simple (por ejemplo, mirar a través de muchos productos para encontrar el que tiene el precio más bajo)

Lección 80 - 81

APP hacer una recomendación

Propósito

Los estudiantes diseñan un proyecto de recomendación basado en datos que recopilan y analizan de sus compañeros de clase. Después de ver un ejemplo de una aplicación de recomendación, los estudiantes siguen una guía de proyecto para completar esta lección de varios días. En los primeros pasos, los estudiantes eligen qué opción quieren hacer para ayudar al usuario, qué datos necesitan para dar la recomendación, crear una encuesta y recopilar información sobre las opciones de sus compañeros de clase. Luego interpretan los datos y usan lo que aprendieron para crear el algoritmo de recomendación. Por último, usan sus algoritmos para hacer recomendaciones a unos pocos compañeros de clase. Los estudiantes realizan una revisión por pares y hacen las actualizaciones necesarias a sus proyectos antes de preparar una presentación para la clase.

Los estudiantes deben demostrar que pueden aplicar el proceso de resolución de problemas de datos a un problema o dominio en el que estén interesados. Los estudiantes completan muchas de las mismas actividades, pero se unen en un solo proyecto integral. Por ejemplo, los estudiantes han usado tabulaciones cruzadas para encontrar relaciones entre preferencias y han creado algoritmos de recomendación, pero en este proyecto, usarán las relaciones que encuentren en los datos para crear sus algoritmos. El último paso en la lección es intencionalmente vago sobre cómo los estudiantes presentarán su trabajo para que pueda elegir el método que tenga más sentido para su clase.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (150 min)

Transferencia del conocimiento (10 min)

Extensión de programación (120 min)

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Haga una recomendación - Ejemplares](#)

Para los estudiantes:

- [Hacer una recomendación - Guía de proyectos](#)
- [Hacer una recomendación - Revisión por pares](#)
- [Haga una recomendación- Rúbrica](#)

Objetivos

Los estudiantes serán capaces de:

- Aplicar el proceso de resolución de problemas de datos a un tema personalmente relevante
- Determinar las fuentes apropiadas de datos necesarios para resolver un problema

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Revisión: Repase rápidamente los diferentes problemas que los estudiantes han visto hasta el momento:

- ¿Qué pizza deberíamos pedir?
- ¿Qué helado deberíamos elegir?
- ¿Qué video debería recomendar WebPix?
- ¿Qué ruta debe tomar un conductor?
- ¿Qué producto debería recomendar Nyle?

Señale que cada decisión de datos hasta ahora implica una recomendación o predicción e introduce el proyecto, en el cual los estudiantes utilizarán los datos para hacer una recomendación sobre un tema de su elección.

Observaciones: Hemos analizado muchos tipos diferentes de problemas de datos, pero todos tienen que ver con hacer recomendaciones (como el video principal) o predicciones (como la ruta más corta). Hoy, tendrá la oportunidad de usar datos para hacer su propia recomendación, en función de algo que le interese.

Ampliación del conocimiento (150 min)

Resuelve un problema de datos

Grupo: Este proyecto puede completarse individualmente, pero se recomienda que los estudiantes trabajen en parejas o, como máximo, en grupos de 3.

Distribuir: Entregue a los estudiantes [Hacer una recomendación-Guía de proyectos](#) y repase las instrucciones en la parte superior de la primera página con los estudiantes.

La aplicación de muestra

Dirija a los estudiantes a mirar la aplicación de muestra. Los estudiantes pueden ver y mezclar el código de esta aplicación en la siguiente [enlace](#).

Puede optar por pedirles a los estudiantes que miren el código, o simplemente que realicen la prueba para ver cómo funciona

Una revisión rápida: esta revisión rápida sirve para recordarles a los estudiantes los tipos de problemas que pueden resolver con datos antes de pasar a definir su propio problema de datos. Los estudiantes deben pasar a trabajar en el proyecto lo antes posible.

Si los estudiantes miran el código, la mayoría debería ser comprensible. La mayoría del código funciona enviando a usuarios a pantallas nuevas, que los estudiantes deben recordar de la Unidad 4. La aplicación también agrega puntos a diferentes opciones de vacaciones cuando el usuario hace clic en diferentes botones. Este tipo de funcionalidad debe ser familiar para los estudiantes de la Unidad 3. El único código desconocido debe ser la función “recomendar ()”, que encuentra la opción con la mayor cantidad de puntos, luego envía al usuario a la pantalla de esa opción.

desde la perspectiva del usuario.

Preguntar: ¿Qué opción ayuda la aplicación al usuario a hacer? ¿Qué datos usa para hacer esa recomendación?

Indicación: ¿Cómo crees que los creadores de la aplicación decidieron sobre el algoritmo que utilizan para hacer la elección?

Observaciones: En este proyecto, vamos a utilizar el proceso de resolución de problemas de datos para ayudarnos a crear nuestro algoritmo. Encuestaremos a las personas para encontrar las relaciones entre sus preferencias, luego usaremos lo que hemos aprendido para crear un algoritmo que pueda recomendar algo al usuario. En primer lugar, usted y su pareja deben pensar en el tipo de recomendación que desea hacer y las tres preguntas que formulará para ayudarlo a hacer esa recomendación.

Definir

Paso 1: Define tu problema

Dé a los estudiantes unos minutos para decidir sobre su tema de recomendación con su compañero.

Preparar

Paso 2: Decide qué datos necesitas

Dé tiempo a los estudiantes para que piensen en los datos que necesitan para ayudarlos a hacer la recomendación al usuario.

Paso 3: Crea tu encuesta

Una vez que los estudiantes han decidido sobre los datos, deben ponerlos en forma de encuesta. En la encuesta inicial, también deberán incluir una pregunta que les pregunte sobre lo que planean recomendar (por ejemplo, un lugar de vacaciones). Utilizarán estos datos para establecer conexiones entre las preferencias establecidas de un usuario y lo que eventualmente recomendarán.

Paso 4: Recopila tus datos de encuesta

Los estudiantes deben dar la encuesta a por lo menos veinte personas, para asegurarse de que haya suficientes datos para interpretar en el siguiente paso.

Paso 5: Interpreta tus datos

Guíe a los estudiantes a usar la tabulación cruzada para encontrar

Las primeras dos preguntas son bastante sencillas. La aplicación ayuda al usuario a elegir un lugar de vacaciones y utiliza datos sobre los alimentos, las superpotencias y las preferencias de los animales del usuario.

La siguiente pregunta pide a los estudiantes que piensen cómo se eligen los algoritmos. En una lección anterior, los estudiantes diseñaron un algoritmo para esta aplicación mirando tablas de tabulación cruzada para personas que habían expresado preferencias diferentes. Recuérdeles a los estudiantes que los creadores pueden usar los datos que han recopilado para crear su algoritmo, en lugar de simplemente inventar sus reglas a partir de sus suposiciones sobre el mundo.

El plan de la lección asume una estructura muy rígida para el algoritmo, con cuatro posibles recomendaciones basadas en tres preguntas de opción múltiple, cada una con cuatro opciones de respuesta. Es posible que desee dar a los estudiantes más libertad para crear su encuesta y algoritmo.

Cuando hacer una pausa: después del paso 3 o 4 es un buen lugar para detenerse después del primer día. La recopilación de guías de proyectos en este punto le permitirá revisar los problemas que puedan causar problemas

relaciones entre las diferentes preferencias del usuario. Si los estudiantes tienen problemas con esta parte del proyecto, es posible que desee compartir el ejemplar con ellos o completar algunas tablas como clase.

Paso 6: Define tu algoritmo

Los estudiantes usan las relaciones para crear reglas para su algoritmo de recomendación.

Paso 7: Prueba tu algoritmo

Dé tiempo a los estudiantes para usar su algoritmo para hacer una recomendación a tres de sus compañeros de clase, y luego reflexionar si creen que el algoritmo es efectivo.

Revisión por pares

Distribuir: Hoja de trabajo de revisión por pares [Hacer una recomendación- Revisión por pares.](#)

Paso 8: Revisión por pares.

Los estudiantes deben completar esta revisión por pares para la guía del proyecto de otro estudiante.

Presentación

Paso 9:

Los estudiantes deben incorporar sus comentarios de los compañeros haciendo ediciones en sus guías de proyectos. Luego, los estudiantes deben diseñar algún tipo de presentación de su trabajo para compartir con sus compañeros de clase. Si bien puede elegir muchos formatos diferentes, se proporciona el contenido sugerido para la presentación.

Transferencia del conocimiento (10 min)

Recopilar: La rúbrica está diseñada para ayudar a calificar la guía de proyecto completa, la presentación y la rúbrica de la revisión por pares.

Cuaderno: Pida a los estudiantes que escriban sobre

- ¿Qué es lo que te gusta de la forma en que se utilizan los datos para resolver problemas en tu vida?
- ¿Qué es lo que no te gusta o te preocupa acerca de cómo se

– a los estudiantes en la próxima parte de la lección. Los estudiantes también pueden usar este tiempo para recopilar datos de encuestas de personas fuera del aula.

Este paso es ligeramente diferente de la asignación de algoritmo anteriormente en el capítulo. Asegúrese de que los estudiantes entiendan que deberían estar creando reglas basadas en los datos de la encuesta que han recopilado e interpretado, no lo que creen que es verdad sobre el mundo.

Otro descanso: justo antes del paso 7 es otro lugar ideal para un descanso. Reúna las guías de proyectos de los estudiantes y vuelva a hacer un check-in rápido para asegurarse de que los estudiantes estén en camino.

¿Qué tipo de presentación?

La presentación puede tomar cualquier forma que considere mejor para su clase: diapositivas, póster, papel, etc. No dude en actualizar la presentación para que se ajuste a sus necesidades.

utilizan los datos para resolver problemas en tu vida?

Dirija a los estudiantes a Code Studio para completar su reflexión sobre sus actitudes hacia la informática. Aunque sus respuestas son anónimas, los datos agregados estarán disponibles una vez que al menos cinco estudiantes hayan completado la encuesta.

Ampliación del conocimiento (120 min)

Los estudiantes también pueden “remezclar” la aplicación de muestra para crear su propia aplicación de recomendación basada en este proyecto.

Lo siguiente deberá ser alterado para que las aplicaciones de los estudiantes funcionen.

En modo de diseño:

1. Cambie la pantalla de “introducción” para reflejar el nuevo tema.
2. Cambie el texto en las tres pantallas de “preguntas” para reflejar las nuevas preguntas.
3. Cambie las opciones de respuesta en cada pantalla de “pregunta”.
 - Los estudiantes pueden elegir un botón de texto o una imagen.
 - Se deberá cambiar la identificación de cada elemento de respuesta, o se eliminará el elemento antiguo y se creará un elemento nuevo.
4. Cambie las pantallas de “elección” para reflejar las nuevas opciones de recomendación.
 - Cambia el texto y la imagen.

En modo de código:

1. Cambie los nombres de las variables de las opciones.
2. Cambie las reglas para usar las nuevas variables y reflejar el algoritmo del alumno.
3. En las reglas para la Pregunta 3, actualice la lista de opciones pasadas a la función “recomendar ()”.

Objetivo: Estas indicaciones son una forma de cerrar la unidad. Los estudiantes han explorado el proceso de resolución de problemas de datos y ahora deberían poder reflexionar más ampliamente sobre las implicaciones de su aplicación generalizada. Muchos servicios que los estudiantes usan todos los días y que hacen sus vidas más fáciles o interesantes implican la resolución de problemas de datos. Sin embargo, probablemente ya hayan visto que se está recopilando mucha información sobre ellos, incluida información que no sabían que estaban proporcionando. Este último diario es una oportunidad para reflexionar sobre estos y otros intercambios que han visto en este capítulo.

Contenido 2

El proceso de diseño

Resumen

- **Lección 82:** Análisis de diseño
- **Lección 83:** Entender a su usuario
- **Lección 84:** Micro Lección de diseño centrado en el usuario
- **Lección 85:** Interfaces del usuario
- **Lección 86:** Comentarios y pruebas
- **Lección 87:** Identificación de las necesidades de usuario
- **Lección 88:** Proyecto- Prototipo de papel
- **Lección 89:** Investigación de impacto social
- **Lección 90:** Diseñar aplicaciones para el bien
- **Lección 91:** Investigación de mercado
- **Lección 92:** Prototipos de papel
- **Lección 93:** Pruebas de prototipos
- **Lección 94:** Diseño digital
- **Lección 95:** Vinculación de pantallas
- **Lección 96:** Prueba de la aplicación parte 1
- **Lección 97:** Mejorando e iterando parte 2
- **Lección 98:** Proyecto: presentación de la aplicación

Objetivos

- **OA 5.** Desarrollar aplicaciones para dispositivos móviles y para dispositivos provistos de sensores y mecanismos de control.
- **OA j.** Desarrollar un trabajo colaborativo en línea para discusión y resolución de tareas matemáticas, usando herramientas electrónicas de productividad, entornos virtuales y redes sociales.

Referencias

- <https://studio.code.org/>
- <https://cuantrix.mx/> Fundación Televisa

Lección 82

Análisis de diseño

Propósito

Para iniciar la lección dedicada a resolver problemas grupales y desarrollar productos para otros usuarios, los estudiantes comienzan investigando el diseño de varias teteras. Los estudiantes analizan cada tetera, intentando identificar cómo las necesidades específicas de los usuarios pueden haber influido en su diseño. Al considerar estas opciones de diseño e intentar relacionar cada tetera con un usuario potencial, los estudiantes pueden comenzar a tomar un enfoque centrado en el usuario para diseñar productos (tanto físicos como digitales) puede hacer que esos productos sean más útiles y utilizables. Para el cierre de la lección, se les pide a los estudiantes que propongan algunos cambios en una de las teteras que la haría más útil o utilizable.

Los estudiantes ingresarán a esta lección con una comprensión del proceso de resolución de problemas. El proceso de resolución de problemas utilizado es: Definir, Preparar, Tratar, Reflexionar. En esta lección, los estudiantes observan objetos del mundo real para comprender cómo se puede aplicar el proceso de resolución de problemas para ayudar a los demás. Comenzamos mirando objetos no técnicos para alentar a los estudiantes a pensar más ampliamente sobre lo que significa considerar al usuario final de un producto antes de perfeccionarse en cómo se aplica específicamente al diseño de software.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (35 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Expresar opiniones respetuosamente.
- Evaluar críticamente un objeto para ver si su diseño cumple

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Determine el usuario - Guía de actividades](#) se podrá entregar una a cada estudiante o proyectarlo para que todos puedan visualizarla.
- [Proceso de resolución de problemas con empatía- Imagen](#)

Vocabulario

- **Crítica:** Evaluar una idea de una manera detallada y constructiva.
- **Empatía:** En el diseño, prestar atención a los sentimientos y necesidades de un usuario al diseñar un producto.
- **Usuario:** Alguien que requiere y dará uso a un objeto, incluido software y hardware.

con un conjunto determinado de necesidades.

- Identificar empatía para el usuario como un componente importante del proceso de diseño.

Preparación

- Consultar [Determine el usuario- Guía de actividades](#)

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Introducción

Pantalla: Se mostrará el [proceso de resolución de problemas con empatía- Imagen](#) para presentar a los estudiantes el proceso de diseño centrado en el usuario como una extensión del proceso de resolución de problemas, agregando empatía.

Pregunta: ¿Qué significa empatizar? ¿Cómo puede el concepto empatizar en un proceso de resolución de problemas ser diferente? ¿En qué otros lugares que has escuchado sobre la empatía?

Discutir: Los estudiantes deberán discutir sus ideas como una clase. Finalmente, se dirigirá la conversación sobre el hecho de que empatizar con otras personas significa considerar sus deseos, necesidades o inquietudes.

Observaciones: En esta lección comenzarán a pensar mucho más sobre el diseño para otras personas. Esto sigue siendo en gran parte un proceso de resolución de problemas, pero ahora deberán pensar mucho más acerca de las necesidades de los usuarios. En otras palabras, la empatía será un tema importante a medida que aprendan más sobre el diseño.

Objetivo: Esta discusión puede ser bastante fluida y abierta. Si bien finalmente se desea llegar a una comprensión compartida de lo que significa empatizar, los estudiantes trabajarán en su comprensión durante la unidad. No es necesario establecer una única definición, por lo que debe considerarse esta discusión como una introducción a un tema de la unidad.

Ampliación del conocimiento (35 min)

¿Para quién fue esto diseñado?

Distribuir:

- [Determine el usuario- Guía de actividades](#)

Determinar el usuario: La primera parte de esta lección pedirá a los estudiantes que combinen diferentes teteras con el usuario más probable. Se pedirá que trabajen en esta lección en parejas, fomentando la discusión sobre por qué exactamente eligen establecer cada conexión. Este es un gran lugar para discutir el diseño de teteras y las necesidades de los usuarios.

Si bien algunos de los usuarios tienen una conexión obvia a una tetera específica, se podrían hacer argumentos para cualquier cantidad de conexiones. No hay respuestas correctas o incorrectas aquí; la discusión y la capacidad de respaldar razonablemente una decisión son lo que importa.

El primer paso para crear un diseño que pueda ser utilizado con eficacia por muchas personas es comprender que no todos piensan igual o miran un objeto o diseño de la misma manera. Este es el primer ejercicio en el que los estudiantes piensan fuera de ellos mismos y en cómo otros pueden percibir los elementos del mundo.

Crítica de diseño

Observaciones: Ahora aprenderán cómo criticar un diseño. Una crítica es cuidadosa cuando se da opinión sobre las partes buenas y malas de algo, como una pieza de escritura o una obra de arte. Es importante ver las críticas como retroalimentación hacia la mejora, en lugar de un juicio final. En este caso, criticarán los objetos con el fin de encontrar la identidad para los que pueden haber sido diseñados.

Esta segunda lección les da a los estudiantes la oportunidad de criticar específicamente cuatro de las teteras. La estructura de esta página refleja el enfoque general que utilizaremos para la crítica, utilizando las tres afirmaciones:

- Me gusta ... (una fortaleza)
- Deseo ... (una debilidad)
- ¿Qué pasa si ...? (una sugerencia)

Es posible que desee pedir a los estudiantes que apliquen este proceso de crítica a más teteras u objetos del salón para ayudar a reforzar el proceso.

Discutiendo las opciones de diseño

Discute:

Los estudiantes deben hablar sobre sus respuestas en la hoja de trabajo, ya sea en grupos pequeños o en clase.

Las preguntas podrían incluir:

- ¿Qué teteras elegiste para “Alguien que necesita servir té en la cena”? ¿Por qué elegiste esas teteras particulares?
- ¿Qué teteras elegiste para “Alguien a quien le gustan los objetos metálicos”? ¿Por qué elegiste esas teteras particulares?
- ¿Para qué usuarios fue más fácil encontrar coincidencias?
- ¿Para qué usuarios fue más difícil encontrar coincidencias?
- ¿Qué tetera era tu favorita? ¿Por qué?

Transferencia del conocimiento (5 min)

Mejorando en Diseños

Cuaderno:

- Elegirán uno o dos de los objetos que analizaron hoy (en cualquier Lección).
- Posterior se preguntarán ¿Qué podrías cambiar para que este objeto sea más útil para ti? Siéntete libre de usar palabras, imágenes o una combinación de ambos.

Experiencias de aprendizaje de profundización

Use estos Contenidos para ampliar el aprendizaje de los estudiantes. Se pueden usar como contenidos extras fuera del aula.

Explorando las cosas cotidianas

Traiga una variedad de objetos raros o poco comunes (las herramientas antiguas, los utensilios de cocina oscuros y los artilugios antiguos funcionan bien para esta lección). Distribuya los objetos alrededor de la sala y organice a los estudiantes en grupos de 3-4. Permite que los grupos recorran el salón y examinen los objetos. Para cada objeto, los grupos discutirán qué creen que hace el objeto y qué tipo de persona podría usarlo.

Cuando la clase haya tenido la oportunidad de explorar todos los objetos, vuelvan a reunirse como un grupo y discutan:

- ¿Qué equipos pensaron que cada objeto era específico para una tarea?
- Lo que cada objeto en realidad hace.
- ¿Qué características de cada objeto dieron pistas sobre lo que hizo, o qué tipo de persona podría usarlo?

Productos mal diseñados

Se pedirá a los estudiantes que encuentren cosas que utilicen en la vida cotidiana y que sientan que podrían beneficiarse de una revisión del diseño. Como clase, discutirán qué es ineficaz sobre el diseño existente y cómo podría mejorarse.

Lección 83

Entender a su usuario

Propósito

Los diseñadores deben comprender las necesidades de sus usuarios para crear productos útiles. Esta lección alienta a los estudiantes a pensar en cómo diseñar para otra persona mediante el juego de roles, como alguien más que usa un perfil de usuario y reacciona como ese usuario a una serie de productos. A cada estudiante se le asigna un perfil de usuario que describe a una persona, que luego utiliza para elegir los productos adecuados, criticar el diseño del producto y sugerir mejoras para el diseño.

Esta lección pide a los estudiantes identificarse con usuarios potenciales. A lo largo de esta lección, los estudiantes trabajarán para aplicar el proceso de resolución de problemas para otras personas. La empatía es un elemento clave en este tipo de diseño centrado en el usuario y una habilidad difícil de dominar para los estudiantes de esta edad.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (30 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Distinguir entre sus propias necesidades y las de los usuarios.
- Criticar un diseño a través de la perspectiva de un perfil de usuario.
- Diseñar mejoras en un producto en función de un perfil de usuario.

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Entendiendo al usuario- Diapositivas](#)

Para los estudiantes:

- [Reaccionando como tu usuario- Guía de actividades](#)

- Perfil 1, 2, 3 y 4 (se encuentran al final de la Lección)

Vocabulario

- **Usabilidad:** Qué fácil, eficiente y satisfactorio es usar un objeto o dispositivo fabricado por el ser humano (incluido el software).

Preparación

- Consultar [Reaccionando como tu usuario- Guía de actividades](#)
- Prepara preguntas rápidas para el ejercicio de calentamiento.
- Etiqueta cuatro esquinas del salón como A, B, C y D.
- Organiza el salón en grupos de 4 y asigna un número del 1 al 4 a cada integrante.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Funcionalidad vs. Estética

Pregunta: ¿Los productos atractivos siempre son fáciles de usar? ¿Puedes pensar en algún ejemplo de un producto que se vea bien pero que sea difícil de usar?

¿Qué pasa con algo que es fácil de usar, pero no le gustó cómo se veía? Se pedirá a los estudiantes escribir sus respuestas y prepararse para compartirlas.

Discute: Los estudiantes publicarán sus pensamientos. Mientras comparten con sus compañeros lo que escribieron, se pedirá que discutan si les importa que un producto sea utilizable o se vea bien. Después de un par de minutos, una pareja explicará lo que discutieron.

Observaciones: Claramente, como usuarios de productos, tenemos muchas necesidades. Es importante que un diseñador sea capaz de distinguir entre estas necesidades. La usabilidad de los diseños afectará si un usuario puede usar el producto en primer lugar.

Sin embargo, si un producto no es atractivo, puede que nunca se use. A medida que pensamos más sobre el diseño para otras personas, queremos tener en mente muchos tipos diferentes de necesidades.

¿Quién soy?

Distribuir: Se repartirán los perfiles de usuario a los estudiantes a medida que ingresan a clase y se indicará que comiencen a leer sobre su perfil asignado.

Observaciones: Tómense unos minutos para leer su perfil de usuario asignado: hoy tendrán que “ponerse al corriente” con el usuario para la Lección. De hecho, durante el resto del día aprenderán a empatizar con sus usuarios y responderá a las situaciones como lo hagan sus usuarios.

Recorra la sala: se dará a los estudiantes unos minutos para leer sus perfiles, animándolos a “meterse en la cabeza” de su usuario asignado. Durante el resto de la lección, los estudiantes intentarán empatizar con sus usuarios y responder a las situaciones.

Pantalla:

- [Entendiendo al usuario- Diapositivas 5-7](#)

Objetivo: Resalte que hay muchos tipos de necesidades de los usuarios con las que debemos sentir empatía como diseñadores. En particular, solicite que la usabilidad de un producto sea una consideración importante e independiente de si es meramente estético.

El primer paso para crear un diseño que pueda ser utilizado con eficacia por muchas personas es comprender que no todos piensan igual o miran un objeto o diseño de la misma manera. Este es el primer ejercicio en el que los estudiantes piensan fuera de ellos mismos y en cómo otros pueden percibir los elementos del mundo.

Adaptación de la lección: Las diapositivas provistas ya incluyen varios productos, pero debe agregar algunos productos e imágenes adicionales que cree que resonarán con sus alumnos.

Para cada diapositiva, solicitará a los alumnos que elijan un rincón según la imagen que atraiga a su usuario. Luego, los estudiantes caminarán hacia la esquina (A, B, C o D) que han decidido analizar con otros estudiantes.

Compartir: Los grupos en cada esquina discutirán brevemente (1-2 minutos) por qué creen que su usuario se siente atraído por esta versión del producto. Uno o dos estudiantes o grupos compartirán lo que discutieron.

Ampliación del conocimiento (30 min)

Mirando a través de los ojos de un usuario

Grupo: Organice a los estudiantes en grupos de 3 a 5 en función de su perfil de usuario asignado; los estudiantes con el mismo perfil trabajarán juntos durante la próxima Lección.

Distribuye: [Reaccionando como tu usuario- Guía de actividades](#)

Pantalla:

- [Entendiendo al usuario- Diapositivas 9-11](#)

Reaccionando como tu usuario

Respondiendo a los productos

Para cada producto en la plataforma de diapositivas, los equipos analizarán cómo reaccionaría su usuario relleno una fila en la sección “reaccionando como tu usuario”. La guía les pide a los estudiantes que hagan una distinción entre la usabilidad de un artículo para su usuario y si a un usuario le gusta o no. Es posible que deseen examinar el primero como grupo para modelar que un producto puede ser estéticamente atractivo, pero no muy útil, ni utilizable, ni estéticamente atractivo.

Compartir: Se pedirá a algunos equipos compartir sus reacciones a cada producto.

Encuentra un asiento

Pantalla:

- [Entendiendo al usuario- Diapositivas 9-11](#)

Los estudiantes podrán trabajar individualmente en la segunda página, que les pide considerar cuál de las sillas que se muestran en la diapositiva 12 se adapta mejor a su usuario.

Después de reflexionar sobre las características que los llevan a elegir una silla específica, los estudiantes tendrán un espacio para diseñar una silla aún más apropiada para su usuario.

Esta es una lección creativa, y los estudiantes pueden elegir abordarla de la forma más expresiva para ellos, incluido dibujar su diseño.

Compartir: Se recorrerá la sala para compartir algunos de los diseños de la silla, lo que lleva a los

estudiantes a definir qué hace específicamente que su diseño sea una mejor opción para el usuario.

Transferencia del conocimiento (5 min)

Pensando en la Empatía

Cuaderno: Pegue o copie tu objeto diseñado en tu cuaderno. Tómese de 1 a 2 minutos para escribir sobre lo que fue fácil y sobre lo que te resultó difícil identificarte con tu usuario.

Ticket de salida:

- ¿Cuáles fueron 3 cosas sobre su usuario que fueron diferentes de usted personalmente?
- ¿Qué fue 2 veces que le resultó difícil empatizar con su usuario?
- ¿Qué cosa crees que a tu usuario realmente le gustaría de la silla que diseñaste?

Lección 84

Micro lección de diseño centrado en el usuario

Propósito

Esta lección guía a los estudiantes a través de una versión abreviada del proceso de diseño. Los estudiantes primero generan una lluvia de ideas para una lista de usuarios potenciales de ropa inteligente. Como clase, agrupan estas ideas en categorías amplias y cada grupo elegirá una categoría de usuario. Los grupos repiten este proceso para generar una lluvia de ideas sobre las necesidades o inquietudes de sus usuarios, eventualmente categorizan estas necesidades y eligen una en la cual enfocarse. Finalmente, los estudiantes diseñan una pieza de ropa inteligente, usando las necesidades y preocupaciones específicas que intercambiaron para guiar sus decisiones.

Al final de la clase, los estudiantes comparten rápidamente su proceso de toma de decisiones y reciben comentarios sobre qué tan bien sus productos abordan las necesidades del usuario que seleccionaron.

Esta micro lección es el primero de tres proyectos de diseño. Es una introducción rápida al proceso de diseño centrado en el usuario, destinado a brindarles a los estudiantes una experiencia en la que puedan construir en proyectos posteriores. En esta lección, el objetivo principal es establecer que las decisiones de diseño se tomarán teniendo en cuenta las necesidades del usuario. La práctica de categorizar montones de ideas dispares para ayudar a tomar decisiones reaparecerá más adelante.

Esta lección es una adaptación de Design Charrette de la Universidad de Washington.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (40 min)

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Diseño centrado en el usuario- Guía de actividades](#)

Transferencia del conocimiento (15 min)

Objetivos

Los estudiantes serán capaces de:

- Empatizar con las necesidades de un usuario para diseñar un objeto.
- Crear categorías significativas a partir de una colección de ideas, específicamente en el contexto de una lluvia de ideas.

Preparación

- Asegúrate de tener notas adhesivas, bolígrafos y papel grande para que los estudiantes trabajen.
- Establece equipos con preferiblemente 3 estudiantes.

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Diseñando para otros

Pregunta: En las últimas dos actividades, hemos visto que los productos están diseñados con un propósito y que diferentes diseños son más útiles o agradables para diferentes personas. Dado que las personas tienen diferentes necesidades, intereses, etc., ¿cuáles podrían ser algunos de los desafíos si está intentando diseñar un producto para otra persona?

Discute: Se dará a los estudiantes un minuto para pensar en silencio antes de hacer que compartan con sus equipos y luego la clase como un todo.

Observaciones: Diseñar para otras personas puede ser un desafío por muchas razones, pero una de las más importantes es que nos desafía a considerar lo que otra persona valora, le gusta o le preocupa. En otras palabras, necesitamos tener empatía por otra persona. Hoy vamos a hacer una mini lección de diseño para ayudarnos a practicar todo el proceso nosotros mismos.

Ampliación del conocimiento (40 min)

Presenta la lección

Grupo: Los estudiantes deben estar en equipos de 3 a 5.

Distribuye: [Diseño centrado en el usuario - Guía de actividades](#), así como marcadores, notas adhesivas y papel para cada grupo.

Descripción general: Como clase, leerán la sección “Descripción general” para asegurar que los grupos entiendan el objetivo de la Lección.

Definir

1. **Lluvia de ideas para los usuarios:** Los estudiantes incluirán en sus guías de actividades a todos los usuarios potenciales de ropa inteligente que puedan imaginar. Se darán un par de minutos para intercambiar ideas de forma independiente. Una vez que terminen, crearán un post-it para los dos o tres tipos de usuarios que les parezcan más interesantes.
2. **Categorizar usuarios:** discutirán con su equipo al menos

Objetivo: Esta debería ser una introducción muy rápida a la lección. Está buscando decir que el diseño para otras personas requiere que considere sus necesidades en lugar de las suyas, lo que a menudo puede ser un desafío.

¿Por qué ropa inteligente?:

Esta lección se puede ejecutar fácilmente con un producto objetivo diferente. Se eligió este tipo de producto porque son aplicaciones amplias para combinar la tecnología informática con la ropa que podría beneficiar a muchos tipos de usuarios. Un beneficio adicional es que el campo aún no está bien definido y, por lo tanto, brinda a los estudiantes más margen de maniobra para desarrollar sus propias ideas sobre cómo resolver los problemas de las personas en lugar de confiar en soluciones más establecidas. Independientemente del producto que use, el punto menos primordial es que el producto sea factible y mucho más importante que los estudiantes estén pensando creativamente sobre cómo satisfacer las necesidades de otras personas en lugar de las suyas.

una categoría más grande de usuarios que vean en el pizarrón. Eventualmente llenan sus ideas con una discusión completa en clase. Deberán tratar de crear amplias categorías para cada usuario en el pizarrón

3. **Elige Usuario Específico:** elegirán una de las categorías que han creado para diseñar. Se deberá hacer todo lo posible para garantizar una buena combinación de usuarios en la sala, pero no es un problema si algunos grupos eligen al mismo usuario.

4. **Lluvia de ideas sobre las necesidades:** Los estudiantes repetirán el proceso de intercambio de ideas para identificar una lista de posibles inquietudes, intereses y necesidades del usuario que eligieron. Se animará a los estudiantes a pensar cuidadosamente sobre lo que podría ser importante para esas personas.

5. **Clasifica las necesidades:** Los estudiantes deberán repetir el mismo proceso de creación en trozos de papel para cada necesidad, interés o preocupación de su usuario y luego agruparlos. Los estudiantes pueden usar los marcadores y el papel para hacer este paso si has proporcionado esos materiales.

6. **Elige una necesidad específica:** los equipos elegirán la necesidad específica del usuario al que desean dirigirse. Deberán tratar de elegir una necesidad que crean que podría tratarse con ropa inteligente, por lo que en algunos casos pueden necesitar ser un poco creativos al pensar en estas necesidades.

Preparar

Soluciones de lluvia de ideas: Se pedirá a los estudiantes que piensen en posibles formas en que la ropa inteligente podría usarse para abordar el problema que han decidido resolver.

Discute los pros y los contras: Una vez que los estudiantes hayan intercambiado ideas, se invitará a los equipos a analizar los pros y los contras de las soluciones propuestas. Se reforzará que deberán tener esta conversación desde el punto de vista de su usuario. O bien las necesidades específicas que eligieron o las necesidades más amplias que han generado una lluvia de ideas deberá guiar la forma en que valoran cada idea.

Tratar

Describe tu producto: Los estudiantes deberán escribir una descripción de lo qué es su producto y cómo abordan la necesidad de su usuario en su guía de actividades.

Dibuja tu producto: Los estudiantes deben dibujar y etiquetar una

Lluvia de ideas: En esta lección los estudiantes usarán este proceso dos veces, primero para identificar a un usuario, luego para identificar la necesidad de un usuario. La primera vez debe modelar este proceso más cuidadosamente. Durante la lluvia de ideas, enfatice el hecho de que no hay ideas correctas o incorrectas. Al crear categorías, enfatice que nuevamente no hay categorías “correctas”. Recuérdeles a los estudiantes que el objetivo aquí es definir el problema que tratarán de resolver hoy y este proceso es una forma útil de enfocarse en un problema específico.

imagen de su producto. Específicamente, cualquier característica “inteligente” debe etiquetarse con descripciones breves. Los estudiantes también pueden usar papel y marcadores para esta parte de la lección.

Transferencia del conocimiento (15 min)

Reflexión

Presenta tu producto: Se asignará a los equipos un par de minutos para compartir lo que crearon. Podrán estructurar presentaciones en torno a los siguientes pasos:

- Quién es su usuario y qué necesidad específica han identificado.
- Las características del producto diseñado.
- Cómo las características abordaron la necesidad que eligieron.
- Al menos una característica de su producto que tal vez no hayan incluido personalmente pero que satisfaga las necesidades de su usuario.

Cuaderno: Según la lección de hoy, ¿qué desafíos prevés en el diseño de software para terceros (usuarios)?

Conocer las necesidades del usuario:

La mejor manera de empatizar con otra persona es hablar con ellos. En esta lección, los alumnos intercambian ideas sobre las necesidades potenciales de un usuario, pero en proyectos posteriores querrán escuchar directamente de los usuarios para que no hagan suposiciones sobre sus necesidades. Para esta lección, el objetivo es desarrollar un conjunto de necesidades plausiblemente realistas y luego pensar en cómo diseñar un objeto para esas necesidades en lugar de las suyas.

Empatizar con las necesidades del usuario:

Reforzar la necesidad de empatizar con las categorías de usuarios que los estudiantes eligen. Al sopesar los pros y los contras, debería ser desde el punto de vista de las necesidades e inquietudes de los usuarios que identificaron.

Lección 85

Interfaces de usuario

Propósito

Después del proyecto de mini diseño, los estudiantes miran hacia la siguiente fase del diseño: creación de prototipos de un producto que intenta abordar las necesidades del usuario. En equipos, los estudiantes examinan un prototipo en papel para una aplicación de chat llamada “Txt Ur”. Mediante el uso de este prototipo en papel, los estudiantes tienen la oportunidad de ver cómo un simple prototipo en papel se puede utilizar para probar rápidamente ideas y suposiciones antes de que lleguemos a la computadora. Después de “usar” el prototipo proporcionado, los estudiantes comienzan a identificar maneras de mejorar la próxima iteración.

Esta lección presenta a los estudiantes el concepto de un prototipo en papel como una herramienta rápida para iniciar el desarrollo de una aplicación. Los prototipos en papel volverán en las siguientes lecciones. Este prototipo también servirá como el contexto en torno al cual los estudiantes comenzarán a analizar y organizar los comentarios de una gran variedad de usuarios. Si bien esta lección les pide a los estudiantes que trabajen con un prototipo existente, pronto desarrollarán prototipos de su propio diseño.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Usar un prototipo en papel para probar una aplicación antes de programarla.

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Pantallas de la interfaz de usuario- Guía de actividades](#)
- [Prueba del usuario \(Computadora\)- Guía de actividades](#)
- [Prueba de usuario \(Usuario\)- Guía de actividades](#)

Vocabulario

- **Prototipo:** Primero o primer modelo de un producto que le permite probar suposiciones antes de desarrollar una versión final.
- **Interfaz de usuario:** Los elementos visuales de un programa a través del cual un usuario controla o se comunica con la aplicación. A menudo abreviado UI.

Preparación

- Consultar [pantallas de la interfaz de usuario- Guía de actividades](#)
- Consultar [Prueba del usuario \(Computadora\)- Guía de actividades](#)
- Consultar [Prueba de usuario \(Usuario\)- Guía de actividades](#)
- Recortar las pantallas de UI o proporcionar tijeras para que los estudiantes lo hagan ellos mismos.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

¿Qué es una aplicación?

Preguntar: ¿Qué es una aplicación? ¿Qué aplicaciones usas tú u otras personas que conoces?

Discute: Se mantendrá una lista de respuestas en el pizarrón. Una vez que algunos hayan tenido la oportunidad de compartir, anotarán una propuesta de definición compartida de una aplicación como “una pieza de software diseñada para un propósito”.

Observaciones: Cuando piense en una aplicación, probablemente se imagine el producto terminado. Ya han tenido la oportunidad de experimentar la fase más temprana del desarrollo, cuando estábamos intercambiaron ideas y crearon posibles usuarios y necesidades. Sin embargo, el camino entre esa lluvia de ideas y una aplicación final es bastante largo.

Comenta: ¿Qué pasos crees que entran en el desarrollo de una aplicación?

Ampliación del conocimiento (40 min)

Lectura y uso de un prototipo de papel

Grupo: Se organizará a los estudiantes en parejas.

Distribuir: Se entregará a los equipos [Pantallas de la interfaz del usuario- Guía de actividades](#). Y se otorgarán unos minutos para que recorten las tarjetas.

Decir: Las pantallas se llaman prototipos de papel. Esta es una de las formas más tempranas de un prototipo, y le permite a un desarrollador probar su idea antes de invertir una gran cantidad de tiempo de programación. Los prototipos de papel son una forma rápida de compartir la interfaz de usuario de su aplicación con usuarios potenciales.

Debate: Basado únicamente en las pantallas, ¿para qué crees que está diseñada esta aplicación?

Transición: Por cada par de estudiantes, se asignará un rol de Computadora y Usuario, o se permitirá que los estudiantes decidan. Una vez que se han asignado los roles, los pares se

Objetivo: Esta discusión tiene como objetivo presentar la definición de una aplicación. La mayoría de las sugerencias que brindan los estudiantes deberían poder vincularse a esta definición. Las aplicaciones pueden proporcionar servicios, organizar o proporcionar información, o pueden ser solo un juego u otra forma de entretenimiento. En todos los casos, solo desea establecer una aplicación como un software creado para un propósito.

Meta: El propósito de esta discusión es hacer que los estudiantes consideren el ciclo de vida de desarrollo de un producto de software.

Reducción de materiales impresos

Estos materiales manipulables pueden reutilizarse si los estudiantes no escriben en ellos. Los necesitará para la próxima lección.

Meta: Anima a los estudiantes a identificar elementos específicos del prototipo que respalde sus argumentos y considere las necesidades del usuario.

sentarán uno frente al otro.

Distribuir: A cada uno se le entregará [Prueba del usuario \(Computadora\)- Guía de actividades](#) y [Prueba de usuario \(Usuario\)- Guía de actividades](#)

Prueba de usuario

Esta lección tiene parejas de estudiantes que prueban el prototipo de papel, con uno que actúa como el usuario y otro como la computadora. El proceso básico es el siguiente:

1. La Computadora coloca la pantalla de inicio (la que se llama “Txt Ur”) frente al Usuario.
2. El usuario selecciona una tarea de la tabla en su guía de actividades.
3. El usuario intenta completar la tarea haciendo clic en la pantalla del prototipo de papel que tiene delante.
4. Cada vez que el Usuario hace clic en la pantalla, la Computadora consulta el Diagrama de navegación en su guía de actividades.
 - Si el diagrama de navegación muestra una línea conectada al elemento al que se hizo clic, busca la pantalla en el otro extremo de la línea y colócala frente al usuario.
 - Si el Diagrama de navegación no muestra una línea conectada al elemento al que se hizo clic, no hagas nada.
5. Cuando el usuario completa una tarea (o decide que es imposible en el prototipo actual), completa las columnas “Lo que intenté” y “Mi reacción”

Hay varias filas adicionales en la tabla de tareas para que los usuarios creen sus propias tareas para intentar. La primera tarea se completa como un ejemplo.

Prototipo de informe

Compartir: Se pedirá a los usuarios que compartan su experiencia con toda la clase.

Debate: Como clase discutirán algunos de los problemas comunes con los que tropezaron los usuarios en esta lección, e identificarán mejoras potenciales al prototipo.

Transferencia del conocimiento (5 min)

Criticar el prototipo

Cuaderno: Al principio de la Lección formularon una hipótesis sobre las necesidades del usuario que este prototipo tenía que abordar. Ahora que han experimentado la aplicación de primera mano, ya sea como usuario o como computadora, regresarán a esa pregunta. En sus cuadernos, responderán lo siguiente:

- ¿Qué necesidades de usuario encajan bien con esta aplicación?
- ¿Qué necesidades de usuario podría esta aplicación mejorar?

- ¿Cuáles son algunas de las necesidades de los usuarios que no se abordan en absoluto, pero que tú harías una buena adición?

Compartir: Si hay tiempo, los estudiantes compartirán sus necesidades. Considerando escribir esto en el pizarrón o en papel, donde se puede consultar en la próxima lección.

Sugerencias para evaluar

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Explican los esquemas utilizados para resolver problemas
- Describen con algoritmos en forma de pseudocódigo.
- Argumentan sus decisiones.

Objetivo: La próxima lección profundizará en el análisis de los comentarios sobre esta aplicación y en la propuesta de mejoras, por lo que no invierta demasiado tiempo en los detalles de las mejoras específicas aquí. El objetivo de esta discusión es hacer que los estudiantes se muevan en la preparación de la siguiente lección.

Refuerce el vocabulario: Use esta conversación final para reforzar los términos de vocabulario presentados en esta lección, en particular “interfaz de usuario” y “prototipo”.

Lección 86

Comentarios y pruebas

Propósito

En esta lección, los alumnos utilizan los comentarios de los “usuarios” de la aplicación prototipo de papel de la lección anterior para desarrollar mejoras en la interfaz de usuario de ese prototipo de papel. La lección comienza con una reflexión sobre el hecho de que los diseñadores necesitan traducir las necesidades humanas con la tecnología. Los estudiantes reciben una colección de comentarios y solicitudes de los usuarios de la aplicación de la Lección anterior. En equipos, los estudiantes clasifican los comentarios e identifican las formas en que las necesidades expresadas en los comentarios se pueden cumplir mediante cambios en la interfaz de la aplicación. Luego, en equipos, los estudiantes implementarán algunos de estos cambios para cumplir con una de las necesidades que identificaron.

Esta lección presenta varias habilidades que los estudiantes necesitarán usar más adelante. Aprenderán a categorizar los comentarios para organizar y priorizar su respuesta. Aprenden a interpretar las necesidades humanas con una pieza de tecnología en términos de los cambios en la interfaz de usuario. Tienen la primera oportunidad de crear un prototipo de papel propio diseñando una nueva pantalla de una aplicación.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Traducir las necesidades del usuario en cambios y mejoras en la interfaz de usuario de una aplicación.
- Categorizar y priorizar los comentarios de los usuarios de una

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Comentarios de prototipos - Guía de actividades](#)
- [Mejorar una pantalla - Guía de actividades](#)
- [Pantallas de interfaz de usuario - Guía de actividades](#)

aplicación.

- Crear un prototipo de papel para las pantallas de una aplicación.

Preparación

- Consultar [Comentarios de prototipos- Guía de actividades](#)
- Consultar [Mejorar una pantalla para cada estudiante.](#)
- Lápices de colores o marcadores para los prototipos de papel.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Preguntar: Pensarán en una pieza de software (como una aplicación) que sea difícil de usar. Tal vez no pudieron encontrar algo que necesitaban, estaba mal organizado o simplemente era difícil de usar. ¿Qué específicamente no le gustó al respecto?

¿Por qué no cubrió sus necesidades? ¿Cómo podría haberse mejorado?

Discutir: Los estudiantes deberán escribir sus ideas en silencio. Luego pasarán a una discusión como clase sobre los tipos de problemas que los estudiantes han tenido con las aplicaciones en el pasado. Abordando la conversación específicamente sobre la interfaz de usuario o la experiencia del usuario y cómo podrían mejorarse esos aspectos de la aplicación.

Observaciones: La tecnología está hecha para satisfacer nuestras necesidades humanas y así es como generalmente pensamos en ellas. Escuché muchos buenos ejemplos de cosas que necesitabas para que la tecnología funcione mejor. Como creadores de una aplicación queremos satisfacer las necesidades de nuestros usuarios, pero tenemos que pensar en ellos en términos de lo que tenemos control en la aplicación. En otras palabras, debemos comenzar a pensar cómo traducir las necesidades humanas en las interfaces de usuario y las experiencias que creamos.

Ampliación del conocimiento (40 min)

Sentido de los comentarios de los usuarios

Grupo: Coloca a los estudiantes en equipos de 2 o 3

Distribuir: Se entregará a cada equipo [Comentarios de prototipos - Guía de actividades](#) . También distribuirá copias de las [pantallas de la interfaz de usuario- Guía de actividades](#) o podrá mostrarlas en un proyector.

Comentarios del prototipo

Clasifica la retroalimentación: Trabajando en equipos, los estudiantes deben crear categorías, las cuales deben corresponder con necesidades o problemas similares a los que se dirigen los comentarios. Por ejemplo, varios comentarios pueden ser sobre el hecho de que la fuente es demasiado pequeña para leer.

Objetivo: Esta conversación tiene como objetivo resaltar que las necesidades humanas con el software se pueden resolver cambiando la experiencia del usuario del software o la interfaz, términos que los estudiantes aprendieron en la clase anterior. Esto es principalmente una lluvia de ideas y si estos puntos no aparecen naturalmente en la conversación, conviértelos en la transición a la Ampliación del conocimiento.

Reducción de materiales impresos

Las pantallas se pueden reutilizar, siempre y cuando los alumnos no escriban en ellas. Las pantallas de interfaz de usuario ya deberían estar impresas de la lección anterior.

Selección de categorías:

Circule por la sala pidiendo a los equipos que expliquen cómo están formando sus categorías. Recuerde a los estudiantes que su usuario no siempre dice exactamente lo que necesita. Necesita interpretar o traducir lo que han dicho. No hay una agrupación correcta de comentarios aquí, pero los estudiantes deben estar preparados para justificar sus categorías.

Compartir: Una vez que todos los equipos hayan categorizado sus comentarios, solicitará algunas sugerencias sobre los tipos de categorías que crearon.

Respondiendo a Comentarios

Distribuir: [Mejorar una pantalla - Guía de actividades](#) para cada equipo.

Mejora una pantalla

Mejoras potenciales: Como equipo, los estudiantes tomarán los comentarios que categorizaron y propondrán algunas posibles mejoras a la aplicación. Cada mejora propuesta debe estar claramente conectada a una de las categorías de retroalimentación que se crearon en la última lección.

Rediseño de pantalla: Cada estudiante deberá dibujar una versión mejorada de una de las pantallas en la aplicación prototipo en papel. La guía de actividades contiene una página para cada pantalla de la aplicación para que el grupo divida.

Transferencia del conocimiento (5 min)

Compartir: Algunos equipos presentarán brevemente la pantalla que han actualizado. Se pedirá que describan específicamente:

- La categoría de necesidades que eligieron abordar.
- Diferentes formas en que consideraron abordar esas necesidades.
- Los cambios en la interfaz de usuario y la experiencia del usuario que diseñaron para satisfacer esas necesidades.

Sugerencias para evaluar

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Explican los esquemas utilizados para resolver problemas
- Describen con algoritmos en forma de pseudocódigo.
- Expresan y argumentan sus puntos de vista.

Reducción de materiales impresos

La Guía de actividades se puede completar en línea o como una lección de diario. Los estudiantes que completen la lección en línea pueden optar por utilizar una herramienta de dibujo en línea para bocetar la interfaz de usuario mejorado o enviar el boceto por separado en papel.

Elección de mejoras:

Los estudiantes pueden necesitar ayuda para mejorar sus aplicaciones. Por ejemplo, si varios usuarios se quejan de texto pequeño, entonces podrían intentar aumentar el tamaño de la fuente en la pantalla que están mejorando. También pueden elegir agregar una configuración de “tamaño de texto” en su página “Configuración”. Siempre hay muchas mejoras en una aplicación que pueden tener el efecto deseado. Para esta lección, es más importante que el cambio aborde razonablemente la necesidad que eligieron. Habrá más oportunidades para investigar qué es realista cambiar en la interfaz de usuario de la aplicación más adelante en la unidad.

Lección 87

Identificación de las necesidades del usuario

Propósito

Hasta este momento, los estudiantes se han centrado en diseñar para usuarios que, hasta cierto punto, están distanciados de ellos. Ya sea a través de lluvia de ideas, perfiles o comentarios de texto, la conexión con un usuario final nunca ha sido directa. Esta distancia está diseñada para ayudar a los estudiantes a salir de su cabeza cuando piensan en los usuarios, pero para obtener información más directamente de un usuario real, los estudiantes deben confiar en sus compañeros de clase. En esta lección, los estudiantes se emparejan para convertirse en usuarios (y diseñadores) el uno para el otro, lo que les permite a todos entrevistar directamente a su usuario final y hacer preguntas para informar mejor su diseño. Cada par de estudiantes se entrevista entre sí, tratando de identificar una necesidad específica que una aplicación podría abordar.

Hasta este punto, los estudiantes han explorado a los usuarios y la empatía a distancia, ya sea analizando un perfil de usuario, generando ideas sobre posibles usuarios y necesidades, o respondiendo a los comentarios escritos de los usuarios. En esta lección hacemos la transición a una interacción más directa con los usuarios, a través de entrevistas.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (10 min)

Objetivos

Los estudiantes serán capaces de:

- Entrevistar a un compañero para conocer sus necesidades.

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [la entrevista del usuario](#)
- [Guía de actividades](#)
- [Prototipo de papel](#)
- [Guía de proyectos](#)

- Analizar notas de la entrevista para desarrollar preguntas de seguimiento.
- Hacer una lluvia de ideas sobre posibles soluciones a un problema específico.

Preparación

- Consultar [la entrevista del usuario- Guía de actividades](#)
- Consultar [Prototipo de papel- Guía de proyectos](#)
- Materiales para intercambiar ideas (bolígrafos, notas adhesivas, etc.).

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Meta personal

Cuaderno: ¿Qué es lo que siempre has querido aprender (o aprender a hacer mejor)? Los estudiantes lo compartirán con un compañero, así que se asegurarán de que sea algo que se sienta cómodo compartir. ¿Qué se interpuso en tu camino al aprender esto?

Ampliación del conocimiento (40 min)

Escuchando a los usuarios

Grupo: Se agrupará a los estudiantes en parejas con el objetivo de que intenten resolver un problema que no les pertenece, por lo que, en la medida de lo posible, se agruparán con alguien de intereses y antecedentes diferentes.

Distribuir: Se entregará a cada pareja [Entrevista del usuario: Guía de actividades](#).

Observaciones: Hasta este momento, la información que hemos recibido sobre los usuarios siempre ha sido, en el mejor de los casos, de segunda mano y una direccional (del usuario a ustedes). A menudo, este es el único tipo de información que podemos obtener sobre los usuarios, pero siempre que sea posible, lo mejor es ir directamente a la fuente. Ahora cada uno de ustedes está emparejado con su nuevo usuario, y su trabajo es diseñar una aplicación para este usuario. Sus aplicaciones están diseñadas para eliminar una barrera para aprender algo nuevo: el mismo que cada uno identificó en la lección de calentamiento.

Entrevista del usuario

Esta lección de entrevista se divide en dos partes: dos preguntas iniciales que aparecen en la primera página y tres preguntas de seguimiento que los alumnos crearán en la segunda página.

1. **Preguntas iniciales:** Estas primeras preguntas se centran en aprender qué específicamente cada usuario desea aprender, y qué piensan que está en el camino de eso. Esto deberá dar a cada estudiante alguna información explícita para que funcione. Se dará a cada estudiante aproximadamente tres minutos para completar esta parte de la entrevista con su compañero, e insértalos para que presenten preguntas de seguimiento sobre el

Impulsar a los estudiantes a pensar más allá de la información de aprendizaje o las materias académicas, a enfocarse en más habilidades de aspiración. Tocar un instrumento, hacer una manualidad o hacer algo son buenos ejemplos para las siguientes actividades.

El objetivo aquí es guiar a los estudiantes hacia los tipos generales de barreras que les permitirán trabajar juntos para generar una lluvia de ideas alrededor de una amplia categoría de barreras, sin preocuparse por las características específicas de sus usuarios. Lo ideal sería que estas categorías se parezcan a las siguientes:

- Gestión del tiempo (encontrar el tiempo para aprender algo).
- Adquisición de recursos (obtención de recursos necesarios para aprender algo).
- Mentoría (encontrar apoyo humano para aprender algo).
- Requisitos previos (desarrollar las habilidades básicas para aprender algo).

Si sus alumnos tienen dificultades para identificar las barreras en este nivel, considere proporcionarles estas categorías y pídeles que revisen las notas de la entrevista para encontrar ejemplos de cada tipo de barrera.

terreno que mantendrán al usuario hablando durante los tres minutos completos.

2. **Desarrollo de preguntas de seguimiento:** Después de que todos hayan tenido la oportunidad de hacer las preguntas iniciales, se dará a los estudiantes unos minutos para leer sus notas. Los estudiantes deben buscar y destacar cualquier declaración sobre la que les gustaría saber más y cualquier declaración ambigua que deba aclararse. El objetivo de este momento es “leer entre líneas” y tratar de descubrir qué es lo que tu usuario aún no ha mencionado, o qué suposiciones puede tener que validar. Una vez que los estudiantes hayan revisado las respuestas a las preguntas iniciales, deberán escribir tres preguntas de seguimiento. Se dará a los estudiantes otras 2-3 preguntas cada uno para hacer sus preguntas de seguimiento.

3. Identificación de barreras: El objetivo final de esta entrevista es identificar un puñado de barreras para que el usuario aprenda lo que quiere.

Compartir: Se dará a cada estudiante la oportunidad de compartir algunas de las barreras que identificaron, haciendo un seguimiento de todas ellas en el pizarrón.

Discute: Como clase, hablarán acerca de cómo pueden categorizar todas estas barreras. En el pizarrón, crearán 4-5 categorías generales identificadas en la discusión (consulta el objetivo de discusión para obtener consejos) y clasificarán las barreras de las categorías anteriores.

No es esencial que cada barrera encaje perfectamente en una de las categorías, pero la clase debería proponer un conjunto lo suficientemente amplio de categorías para cubrir la mayoría de las barreras identificadas.

Soluciones de lluvia de ideas

Compartir: Cada alumno seleccionará una de las barreras identificadas como el foco de su aplicación. La barrera que eligieron debería estar dentro de una de las categorías creadas en la discusión previa. Luego, los estudiantes comparten cuál es su categoría (no la barrera específica).

Grupo: Según las categorías elegidas, se agrupará a los estudiantes en categorías comunes.

Lluvia de ideas: En equipos pensarán en posibles soluciones basadas en aplicaciones para la categoría de barrera. Si bien cada estudiante debe mantener su mente de barrera específica durante esta lección, el objetivo es considerar soluciones a la categoría más amplia de barreras: esto dará a los estudiantes el cambio para ver soluciones potenciales que quizás no hayan considerado para sus problemas específicos.

Transferencia del conocimiento (10 min)

Preparación del proyecto

Distribuir: Se entregará a cada estudiante [Prototipo de papel- Guía de proyectos](#).

Indicaciones: Repase esta guía de proyecto. Iniciaremos este proyecto basado en la entrevista y lluvia de ideas que completaron hoy. En la primera página, completa la sección Definir.

Lección 88

Proyecto – prototipo de papel

Propósito

Con base en la entrevista entre iguales de la lección anterior, a cada alumno se le ocurre una idea para una aplicación que abordará el problema de su usuario. Luego, los estudiantes pueden crear su propio prototipo en papel de sus ideas de aplicaciones dibujando “pantallas” en tarjetas de notas individuales. Una guía de proyecto dirige a los estudiantes a través del proceso, incluida la construcción de la aplicación y la prueba con su usuario para ver si sus suposiciones sobre las interfaces de usuario que crearon son precisas.

Esta lección es una oportunidad para mostrar las habilidades que los estudiantes han desarrollado. **Desarrollarán una interfaz de usuario basada en las necesidades de su compañero de clase que identificaron en la lección anterior.** Además, practican obtener comentarios de los usuarios al probar su prototipo en papel. En el próximo capítulo, los estudiantes continuarán utilizando muchas de estas habilidades, pero programarán sus prototipos en el App Lab.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (15 min)

Objetivos

Los estudiantes serán capaces de:

- Diseñar la funcionalidad de una aplicación para abordar las necesidades específicas de un usuario.
- Identificar mejoras en una aplicación basada en las pruebas del usuario.
- Diseñar la interfaz de usuario de una aplicación.

Lección sin conexión
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Prototipo de papel- Rúbrica](#)
- [Prototipo de papel - Guía de proyectos](#)

Preparación

- Consultar [Prototipo de papel- Guía de proyectos](#)
- Consultar [Prototipo de papel- Rúbrica](#).
- 6 tarjetas de notas para cada estudiante.
- Suministros de dibujo (lápices de colores, marcadores, reglas, etc.).
- Papel para cada estudiante.

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Lo que hace una buena interfaz de usuario

Preguntar: Hoy comenzarán a realizar las interfaces de sus prototipos de papel. Antes de comenzar, reflexionarán sobre la aplicación “Txt Ur” que han mejorado u otras aplicaciones que han usado en el pasado. ¿Qué características hacen una buena interfaz de usuario? ¿Cómo podemos asegurarnos de que las aplicaciones sean fáciles de usar y respondan a las necesidades de nuestros usuarios?

Discute: Los estudiantes compartirán sus ideas. Podrán anotarlos en el pizarrón como referencia durante toda la clase.

Ampliación del conocimiento (40 min)

Empezando

Grupo: Se colocará a los estudiantes con el compañero que entrevistaron en la lección anterior.

Distribuir: Si fueron recolectados después de la última clase, se entregará a cada alumno una copia del Prototipo de papel: Guía del proyecto y Prototipo de papel. Rúbrica

Proyecto prototipo de papel

Descripción general: Como clase, leerán la descripción general del proyecto, revisando lo que necesitarán producir y cómo se evaluará.

Definir: Los primeros pasos de este proyecto piden a los alumnos que anoten los resultados de su entrevista y la posterior discusión en clase de la lección anterior.

Prepárate: Los estudiantes deberán intercambiar ideas sobre el conjunto mínimo de características que su aplicación necesitaría para abordar las necesidades de sus usuarios. Esto se puede completar como una lista con viñetas en la guía de actividades.

Decida sus pantallas: A los estudiantes se les proporcionará espacio en su guía de actividades para esbozar sus ideas para las pantallas de su aplicación. Esta es una oportunidad para hacer un primer borrador rápido de su aplicación. Esto debería tomar alrededor de 10 minutos y centrarse en:

Meta: Antes de enviar a los estudiantes a diseñar sus prototipos en papel, les da la oportunidad de intercambiar ideas sobre atributos de diseños efectivos. Por ejemplo, pueden aparecer

- Interfaces simples
- Reutilizando elementos de diseño
- Colocando elementos comunes en el mismo lugar en cada pantalla
- Presta atención a lo que es importante

Proyectos de alcance: Anime a los estudiantes a que alcancen la funcionalidad de su aplicación con la cantidad mínima necesaria para satisfacer las necesidades de sus usuarios. A medida que funcionen, es probable que descubran nuevos componentes que necesitan agregar, por lo que es mejor comenzar con un conjunto pequeño desde el principio.

¿Por qué seis pantallas?: No hay nada mágico sobre el número seis. Es una estimación aproximada de la cantidad de pantallas en las que la mayoría de los estudiantes deberían trabajar para obtener sus ideas y es una forma concreta de ayudar a las aplicaciones de los estudiantes. También se adapta muy bien en una sola hoja de papel. Siéntete libre,

- ¿Cuántas pantallas necesitarán?
- ¿Qué información o funcionalidad aparece en cada pantalla?
- ¿Cómo se conectarán las pantallas entre sí?

Prototipos de papel

Distribuir: Se dará a cada estudiante hasta seis tarjetas de notas para crear sus pantallas y también para dibujar suministros.

Crear pantallas: Los estudiantes deberán crear las pantallas de su aplicación.

Crear diagrama de navegación: Los estudiantes pueden hacer su diagrama de navegación cuando lo deseen, pero en general tendrá sentido esperar hasta que se terminen sus pantallas.

Pruebas de prototipos

Pon a prueba su aplicación: Los estudiantes deben reagruparse con el socio para el que diseñaron su aplicación. Usando la misma estructura deberían probar la aplicación. Cada uno debería actuar como la “computadora” para la aplicación que diseñaron y permitir que su usuario interactúe con las diferentes pantallas de la aplicación. Más instrucciones existen en la guía del proyecto.

En la guía del proyecto, los estudiantes tienen espacio para registrar lo que observaron al probar su aplicación. Los estudiantes también pueden y deben preguntarles a sus usuarios si hay algo más que les gustaría ver en la aplicación.

Interpretación de la opinión del usuario: Los estudiantes tienen una tabla T en la guía del proyecto para interpretar los comentarios de los usuarios e identificar posibles cambios o mejoras. En el lado izquierdo de la gráfica T, los estudiantes enumeran todas las observaciones interesantes que hicieron durante las pruebas (como las suposiciones que tuvo el usuario, las formas en que trataron de interactuar con la aplicación o las declaraciones que hicieron al usar la aplicación). El lado derecho de la tabla es un área para interpretar lo que podría significar cada una de esas observaciones. Este proceso ayudará a los estudiantes a identificar qué cambios deberían impulsar su próxima iteración.

—
sin embargo, de imponer cualquier otro límite que desee.

Cuánto tiempo: Haz saber a los alumnos cuánto tiempo tendrán que trabajar en sus proyectos. Se recomienda que los estudiantes tengan al menos una hora para diseñar sus pantallas y es probable que necesiten más. Ayuda a los estudiantes a planear recordándoles cuánto tiempo tienen para trabajar en cada pantalla individual.

Transferencia del conocimiento (15 min)

Comparte tu trabajo

Crear un póster: Como una extensión opcional de este proyecto, los estudiantes pueden crear un póster de su proyecto para presentar. El cartel podría incluir:

- Las pantallas de su aplicación.
- Alguna forma de demostrar el diagrama de navegación, quizás esta vez usando las pantallas mismas.
- Una descripción de lo que el usuario está tratando de aprender y su barrera.
- Una descripción de cómo funciona la aplicación y cómo responde a las necesidades del usuario.

Reflexiona sobre el Proyecto

Revisa: Piense en todo el proceso de desarrollo de tu prototipo en papel, desde la entrevista inicial hasta el diseño del prototipo, las pruebas y la lluvia de ideas para nuevas mejoras.

- ¿Cuáles fueron los mayores desafíos en el diseño de una pieza de software para un usuario diferente?
- ¿Qué habilidades de este proyecto cree que serán útiles al programar prototipos de aplicaciones más adelante?

Discutir: Se pedirá a los alumnos que compartan sus reflexiones con la clase.

Objetivo: Resalte la dificultad para ver proyectos desde la perspectiva de otra persona, especialmente cuando usted es el que está creando. Ésta es una habilidad que los estudiantes necesitarán continuar usando en esta unidad. Habilidades como la interpretación de las necesidades del usuario a partir de entrevistas y pruebas en vivo, el alcance de proyectos grandes, el diseño de interfaces de usuario y la organización general son las que los estudiantes continuarán utilizando en la unidad.

Lección 89

Investigación de impacto social

Propósito

A partir de esta lección los estudiantes se centrarán en identificar una problemática de su comunidad que les permita investigar, plantear y desarrollar una solución que ayude y/o resuelva dicho problema. Las siguientes lecciones permitirá a los estudiantes aplicar los conocimientos adquiridos hasta el momento sobre prototipos centrados en el usuario y en este caso aplicado a una problemática cercana a ellos, permitiéndoles pasar de los prototipos de papel al desarrollo digital.

Secuencia para el aprendizaje

Ampliación del conocimiento (45 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Identificar una problemática de su comunidad.
- Investigar a profundidad las causas y consecuencias del problema.
- Plantear posibles soluciones enfocadas a prototipos centrados en el usuario.

Preparación

- Investigar previamente problemáticas de la comunidad, en caso de que esta parte se dificulte para los estudiantes.

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- Posibles problemáticas de la comunidad (en caso de ser necesarias).
- Cuadernos / hojas para cada equipo.
- Computadoras para realizar la investigación.

Estrategia de aprendizaje

Ampliación del conocimiento (45 min)

Se agrupará a los estudiantes en equipos, preferentemente de 5 integrantes o los que se consideren pertinentes según la capacidad de la sala y cantidad de estudiantes.

Equipos: Se pedirá a los estudiantes que se agrupen y tengan sus cuadernos, o en su caso se les entregarán hojas.

Identificar problemáticas

En equipos, se pedirá a los estudiantes que debatan y definan una problemática de su comunidad que sea de su interés y pueda ser abordada desde lo aprendido durante el curso.

Esta decisión posiblemente tome a los estudiantes tiempo y discusión entre ellos, por lo que será importante definir un tiempo y se recomendará hacer una tabla de pros y contras con el objetivo de que todos los equipos tomen una decisión sobre la problemática a abordar.

Para este punto será importante identificar que los equipos se enfoquen en diversas problemáticas y no en la misma todos.

Causas y consecuencias

Los equipos pasarán a los computadores para realizar una investigación a fondo sobre la problemática que eligieron, estos serán puntos clave para su investigación:

- ¿Cuál es el problema central?
- ¿Qué provoca el problema?
- ¿A quiénes afecta?
- ¿Se derivan otros problemas?

Posibles soluciones

Una vez que los estudiantes tengan claras las causas, consecuencias y afectados de la problemática elegida, pensarán juntos en posibles soluciones enfocadas en lo que han aprendido durante el curso (prototipos centrados en el usuario, prototipos de papel). Es importante que los estudiantes comprendan que este momento no es para iniciar el desarrollo de una solución, simplemente es la ideación de posibles soluciones.

- ¿Cómo lo resolverían?
- ¿Cómo imaginan que sería el prototipo?
- ¿Qué necesidades deberá tener para satisfacer al usuario?

- ¿Conocen las necesidades del usuario?

Transferencia del conocimiento (5 min)

Se pedirá a los equipos compartir con el resto de la clase sus problemáticas y posibles soluciones, esto permitirá que los equipos puedan retroalimentarse y abrir el panorama de su problema. Se recomendará a los estudiantes, realizar una encuesta a otras personas de la comunidad para complementar su idea sobre las necesidades de los usuarios.

Sugerencias de evaluación

Se sugiere los siguientes indicadores para evaluar formativamente los aprendizajes:

- Analizan y evalúan el impacto de las tecnologías digitales en contextos sociales, económicos y culturales.
- Trabajan colaborativamente en la generación, desarrollo y gestión de proyectos y la resolución de problemas, integrando las diferentes ideas y puntos de vista.

Lección 90

Diseñar aplicaciones para el bien

Propósito

Para iniciar el proyecto de diseño de la aplicación, los estudiantes primero exploran una serie de aplicaciones diseñadas para el impacto social que han sido creadas por otros estudiantes. Luego, la clase revisa los pasos Definir, Preparar, Probar y Reflexionar del proceso de resolución de problemas mientras desarrollan una idea para una aplicación propia con impacto social. Finalmente, los estudiantes formarán equipos de proyecto y diseñarán un contrato sobre cómo funcionará el equipo durante el desarrollo de su aplicación.

Esta lección pasa algún tiempo revisando y reforzando el proceso de diseño centrado en el usuario que usarán los estudiantes durante su proyecto, el objetivo principal es brindarles a los estudiantes la oportunidad de ver cómo otras personas de su edad han utilizado la tecnología para abordar problemas que les preocupan. Los recursos proporcionados son un excelente punto de partida, pero puede hacer que este paquete de lecciones tenga un mayor impacto si encuentra ejemplos (incluso los de tus clases previas) que demuestren ejemplos locales de estudiantes que diseñan aplicaciones para el bien social.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (45 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Identificar formas en que las aplicaciones pueden afectar el cambio social.

Lección sin conexión

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los profesores:

- [Aplicaciones de ejemplo para bien- Recurso del maestro](#)

Para los estudiantes:

- [Inicio de diseño de la aplicación- Guía de actividades](#)

Preparación

- Asignar equipos de 4 a 5, idealmente en función de un interés similar en un tema de la aplicación.
- Consultar [Inicio de diseño de la aplicación- Guía de actividades](#).
- Elige 1-3 de las aplicaciones de [Ejemplo de aplicaciones para bien- Recurso del maestro](#) para compartir con la clase.
- Organiza los asientos en el aula para equipos preseleccionados de 4-5 estudiantes.
- Pon notas adhesivas para cada equipo.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Aplicaciones creadas por estudiantes

Grupo: Los estudiantes deberán estar sentados en sus equipos pre-asignados de 4-5 estudiantes, pero no se les dirá que está es su asignación de equipo.

Pantalla: Se compartirá con los estudiantes las aplicaciones de ejemplo que se ha seleccionado, ya sea desde Aplicaciones de ejemplo para bien- Recurso del maestro) o en otro lugar.

Para cada aplicación:

- Discuta: ¿Cuál es el problema social que esta aplicación fue diseñada para abordar, y cómo fue diseñada para hacerlo?
- Preguntar: ¿Hay otras aplicaciones o piezas de tecnología de las que tengan conocimiento que el trabajo aborde un problema social?

Ampliación del conocimiento (45 min)

¿Qué hay en una aplicación?

Comenta: Preguntará a los estudiantes qué piensan que es una aplicación. Si la discusión se detiene, intentará hacer algunas de las siguientes indicaciones:

- ¿Cuál creen que es la diferencia entre una aplicación (como Microsoft Word) y una aplicación como las que vimos antes?
- ¿Cuáles son algunas de las cosas que notaron que hicieron para construir su aplicación?
- ¿Qué aplicaciones creían que eran las más creativas? ¿La más útil? ¿La más sorprendente?
- ¿Qué aplicaciones creen que serían difíciles de crear? ¿Por qué?

Revisión: Mostrará el gráfico del proceso de resolución de problema. Hará saber a los estudiantes que los micro y mini proyectos en los que trabajaron anteriormente se centraron en los pasos Definir y Preparar. Con este proyecto, también trabajarán en estos dos pasos, pero luego continuarán con los pasos Probar y Reflexionar. También volverán a recorrerlos al menos una vez para refinar su aplicación.

La razón para no revelar las asignaciones del equipo al comienzo de la lección es para que los estudiantes puedan comenzar a formar sus propias ideas acerca de las aplicaciones en las que deseen trabajar individualmente antes de presentarlas al equipo.

Las reglas en el contrato del equipo son lo suficientemente generales como para funcionar en la mayoría de las salas, pero para hacerlo más impactante, debe considerar personalizar el contrato para incluir las palabras de su clase o las reglas de la escuela.

En la siguiente lección usaremos la investigación de mercado (investigando soluciones existentes para el problema de cada grupo) como una forma de aprender cómo otros ya han abordado las necesidades de los usuarios. Si bien esa será una forma común en que todos los equipos pueden aprender sobre sus usuarios, use esta discusión para ayudar a los estudiantes a considerar formas de aprender sobre sus usuarios fuera de las lecciones proporcionadas.

Transición: Si los estudiantes aún no están sentados con sus equipos de aplicaciones, pedirá que lo hagan ahora.

Distribuir: Una copia por equipo de inicio de diseño de la aplicación- Guía de actividades.

Inicio de diseño de la aplicación

Haz saber a los estudiantes que actualmente están sentados con el equipo con el que trabajarán a lo largo de este proyecto de diseño de la aplicación.

Contrato de equipo

Para apoyar un ambiente de trabajo en equipo positivo y productivo, cada equipo comenzará desarrollando y firmando un contrato. Puede utilizar este contrato durante el transcurso del proyecto para abordar y mediar en los problemas entre los miembros del equipo. Repase cada sección del contrato con toda la clase.

- **Comunicación:** La sección de comunicación cubre las pautas básicas de comunicación respetuosa. Hay dos espacios abiertos para que los equipos agreguen reglas propias. Se pedirá a los equipos que consideren los modos de comunicación que les gustaría usar y cuáles deberían ser las expectativas. Por ejemplo, usaremos el correo electrónico para comunicarnos fuera de la sala y los miembros del equipo responderán dentro de las 24 horas.
- **Toma de decisiones:** Mientras que otras secciones son solo listas con viñetas, la sección de toma de decisiones está numerada para reforzar la prioridad de los pasos de toma de decisiones.
- **Participación:** La sección de participación está destinada a evitar que los estudiantes hagan girar sus ruedas en situaciones en las que no están seguros de cómo proceder. Considerará agregar reglas a esta sección como una clase completa.

Una vez que confirme que todos los estudiantes comprenden y aceptan los términos del contrato, haga que los miembros del equipo firmen sus nombres en la parte inferior.

Distribuir: Reparta notas adhesivas a cada grupo.

Transición: Si el espacio lo permite, dé a cada equipo un poco de espacio para trabajar en la siguiente lección donde puedan dispersarse.

Reunión creativa

Inicio de diseño de la aplicación- Guía de actividades ofrece espacio para que los equipos puedan intercambiar ideas sobre sus aplicaciones.

Revisión: Recuerde a los estudiantes sobre la lluvia de ideas que hicieron en la “Micro lección de Diseño Centrado en el Usuario”, donde pasaron unos minutos proponiendo tantos tipos de usuarios como les era posible. Haga saber que harán el mismo ejercicio nuevamente, pero esta vez se centrarán en anotar tantos usuarios posibles para el tema de su aplicación como puedan. En particular, recuerde:

- a) Un usuario por nota adhesiva

- b) No hay malas respuestas
- c) Desarrolla las sugerencias de los demás con “Sí, y ...”

Finalmente, recuérdelos que después de la lluvia de ideas con las notas adhesivas, dispondrán de unos minutos para clasificarlas y analizarlas, pero su primer objetivo es escribir tantas ideas como sea posible, sin importar cuán locas sean.

Nombre del equipo: Para el precalentamiento y practicar lluvia de ideas, dé a los equipos un minuto para que asignen el nombre del equipo y lo registren en la guía de actividades.

Tema: Dé a los equipos aproximadamente cinco minutos para elegir un tema general para su aplicación. Es importante recordarles que no están tratando de descubrir qué hará la aplicación ni detalles específicos en este punto, sino simplemente acordar un tema general que deberían abordar.

Equipos de usuarios: Brinde a los estudiantes un tiempo para generar ideas para los usuarios y luego seguir el proceso de agrupación de usuarios. Recuerde a los estudiantes que al agrupar usuarios no tienen que tomar una decisión sobre su usuario objetivo durante este paso, solo hacer las agrupaciones. Deben organizar las notas adhesivas en sus tablas en categorías, y si hay tiempo, proponga un nombre corto para la agrupación. Mencione que tendrán entre 1 y 2 minutos para llegar con los usuarios y entre 1 y 2 minutos adicionales para agruparlos.

Una vez que los equipos tienen un grupo de posibles usuarios para su tema, deben trabajar juntos para reducir su grupo potencial de usuarios objetivos a una o dos categorías generales. Diga a los estudiantes que tendrán tiempo durante las próximas dos actividades para seguir investigando y perfeccionando su grupo de usuarios objetivos, pero deben tener un punto de partida claro para orientar esa investigación.

Compartir: Pida a cada equipo que comparta brevemente su nombre, tema y grupo de usuarios.

Transferencia del conocimiento (5 min)

Planificación para la investigación

Debate: Dado que los equipos están abordando grandes problemas del mundo real con estas aplicaciones, es poco probable que tengan acceso directo a posibles usuarios para entrevistar. Discuta con la clase cómo creen que pueden aprender más sobre quiénes son sus usuarios y cuáles podrían ser sus necesidades.

Lección 91

Investigación de mercado

Propósito

En esta lección, los estudiantes investigan aplicaciones similares a la que pretenden crear para comprender mejor las necesidades de sus usuarios. Los estudiantes trabajan en sus equipos para buscar otras aplicaciones en Internet y luego evalúan las que les parecen interesantes. Al final de la lección, cada equipo tendrá una idea más clara sobre el tipo de aplicación que desea crear y reforzará aún más quiénes son sus usuarios objetivos. Cada equipo mantendrá una lista de todas las aplicaciones que examinaron para usar en su presentación final.

Esta lección es la principal oportunidad que tienen los estudiantes para obtener la opinión del usuario sobre la concepción original de su aplicación. **Al mirar aplicaciones similares a las que han propuesto, pueden inspirarse para cambiar la estructura de su aplicación o incluso identificar un aspecto del problema que habían ignorado previamente.** En lecciones posteriores, los estudiantes probarán prototipos de su aplicación para realizar mejoras adicionales.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Ubicar las aplicaciones que se dirigen a un grupo de usuarios específico o necesitan.
- Identificar las necesidades del usuario a las que se dirige una aplicación.

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Investigación de mercado - Guía de actividades](#)

Preparación

- Organizar asientos en el aula para equipos de 4-5 estudiantes.
- Sacar una copia por equipo de [Investigación de mercado- Guía de actividades](#).

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Lluvia de ideas de la aplicación

Grupo: Haga que los estudiantes se sienten en las mesas de su equipo. Ésta será la disposición de asientos predeterminada.

Preguntar: Hoy trataremos de obtener más información sobre las necesidades de los usuarios sobre su aplicación. Una forma de que podamos conocer las necesidades de nuestros usuarios es ver cómo otras aplicaciones intentan cumplirlas. Antes de que busquemos en línea, analicen con su grupo los tipos de aplicaciones que pueden posiblemente encontrar. Prepárense para compartir.

Discuta: Brinde a los equipos un minuto para generar una lluvia de ideas sobre los tipos de aplicaciones que esperan encontrar. Pida que compartan sus pensamientos. Aprovechen esta oportunidad para ampliar el alcance de las soluciones que buscan. Idealmente, escuchar los pensamientos de otros grupos proporcionará nuevas ideas.

Revisión: Recuerde a la clase que todavía están en la fase Definir de su proyecto. Su objetivo hoy es encontrar y analizar otras aplicaciones en su área temática. Deberán trabajar dentro de sus equipos para seguir pensando en sus usuarios objetivos y avanzar hacia el consenso en cuanto al tipo de aplicación que quieren hacer. No deberían planificar cómo se verán las pantallas individuales de su aplicación. Eso sucederá durante la siguiente fase.

Ampliación del conocimiento (40 min)

Distribuir: Entrega una copia de Investigación de mercado- Guía de actividades a cada equipo

Investigación de mercado

En esta lección, los equipos buscarán aplicaciones en el tema que elijan. Se espera que cada estudiante encuentre al menos una aplicación para que el equipo la analice.

A medida que el equipo encuentre una aplicación, deberán trabajar juntos como grupo para determinar si están inspirados en ella y desean hacer referencia a ella a medida que crean la

Enfóquese en el usuario:

Asegúrese de que los estudiantes se concentren en la parte de investigación de esta lección. Algunos equipos querrán comenzar a diseñar pantallas para sus propias aplicaciones. Anímelos a tomar notas sobre lo que encuentran inspirador en otras aplicaciones, pero recuérdelos que todavía no deben comenzar su proceso de diseño.

propia. Es importante que registren el nombre y la ubicación de la aplicación que investigan, para que luego puedan crear una cita en su presentación final.

Además de investigar otras aplicaciones, cada equipo deberá generar ideas para sus propias aplicaciones. En la parte posterior de la Guía de actividades hay espacio para que escriban estas ideas a medida que los miembros del equipo las propongan. Haga énfasis en que tendrán tiempo en la próxima lección para hacer una elección final y más tiempo en futuras actividades para diseñar la aplicación. Su objetivo de hoy es simplemente generar ideas sobre algunas otras mientras realizan su investigación.

Lugares sugeridos para que los estudiantes busquen:

- Galería pública de Code.org
- Clases de aplicaciones móviles o competencias como:
 - Aplicaciones para bien
 - Technovation
 - Desafío de la aplicación Verizon
- Tiendas de aplicaciones móviles como Apple App Store o Google Play
- Búsqueda web general con Google o Bing
 - Prueba frases como “concurso de aplicaciones” o “aplicaciones para el bien social”
 - Incluye tu nombre de estado para ideas locales o competencias.

Ideas de la aplicación de lluvia de ideas

Después de que los equipos hayan encontrado y revelado suficientes aplicaciones, déles aproximadamente 10 minutos para completar la sección de Lluvia de ideas de esta lección. Esta lluvia de ideas también se puede hacer con notas Post It. Los estudiantes pueden escribir sus ideas como lo han hecho en ejercicios de agrupación anteriores, luego, como equipo, seleccionar las mejores 3-6 ideas y registrarlas en tu guía de actividades.

Transferencia del conocimiento (5 min)

Compartir hallazgos

Pida a cada grupo que se prepare para compartir al menos una aplicación que encontraron inspiradora para su proyecto.

Lección 92

Prototipos de papel

Propósito

Antes de comenzar a diseñar aplicaciones, debemos ayudar a los estudiantes a que alcancen mejor sus expectativas. Debido a que los estudiantes eventualmente crearán prototipos de estas aplicaciones en App Lab, estarán en mejor forma si sus ideas se alinean con los tipos de aplicaciones que son fácilmente prototipadas en App Lab. Los equipos comienzan este análisis de varias aplicaciones ejemplo creadas en App Lab. Los equipos entonces pueden elegir una (o más) de las aplicaciones como base para las suyas. A partir de ahí, los equipos tienen tiempo para analizar la funcionalidad básica de su aplicación antes de usar tarjetas de 3x5 para desarrollar prototipos de papel.

Esta lección inicia la fase de diseño real de este proyecto. Con el fin de simplificar el proceso de diseño de la aplicación, los estudiantes presentan varias categorías de aplicaciones para elegir en el desarrollo de su propia idea. Los estudiantes crearon por última vez prototipos de papel en lecciones anteriores, pero en ese momento se colocaron muy pocas restricciones en la interfaz de usuario del prototipo. Debido a que estos prototipos eventualmente se convertirán en aplicaciones de App Lab, es importante conocer mejor el tipo de diseño de aplicaciones y los elementos de interfaz que utilizan.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (70-90 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Comunicar el diseño y el uso previsto del programa.
- Demostrar el flujo de usuarios a través del diseño de una

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Prototipo de papel del teléfono](#)- Plantillas de 3x5
- [Prototipo de lluvia de ideas](#)- Guía de actividades
- [Code Studio - ejemplos Apps](#)

Vocabulario

- **Prototipo:** Primer modelo de un producto que le permite probar suposiciones antes de desarrollar una versión final.

aplicación utilizando un prototipo de papel.

- Identificar las necesidades del usuario a las que se dirige una aplicación.

Preparación

- Reúne fichas de 3x5, notas adhesivas, lápices de colores y lápices, tijeras para cada equipo.
- Sacar una copia de [Prototipo de lluvia de ideas- Guía de actividades](#) para cada equipo.
- Consultar previamente [Code Studio- ejemplos Apps](#).

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Obtener inspiración

Discute: ¿Qué hemos establecido para nuestras aplicaciones hasta ahora? ¿Qué tenemos que decidir?

Transición: Naveguen en [Code Studio - ejemplos Apps](#) para explorar los diferentes tipos de aplicaciones. Para cada caso, discutan cómo podría aplicarse a su problema.

En esta lección, se les presentará a los estudiantes una pequeña cantidad de aplicaciones relativamente restringidas:

- Examinar
- Tomador de decisiones
- Administrador de listas
- Crowdsourcing (colaboración abierta distribuida)

Para los estudiantes más jóvenes o sin experiencia, es posible restringirlos solo a estos tipos de aplicaciones. Elija específicamente estos ejemplos, porque son relativamente fáciles de crear prototipos en App Lab.

Si se permite que los estudiantes se desvíen de estos, deberá tener en cuenta que algunos tipos de aplicaciones son más difíciles de crear en App Lab y pueden requerir más creatividad para desarrollar un prototipo. Por ejemplo, App Lab no tiene funcionalidad de mapeo, por lo que, si los estudiantes desearan crear una aplicación de mapeo, necesitarían encontrar o crear una imagen estática de un mapa para usar en su aplicación.

Examinar

Se puede utilizar una aplicación de cuestionario para enseñar o ayudar a reforzar la información importante para el tema elegido.

En su aspecto más básico, una aplicación de prueba se compone de una pantalla para cada pregunta con dos o más botones para respuestas de opción múltiple. Cada botón se vincula a una página que confirma que es correcto o le proporciona información sobre por qué estaba equivocado.

Las aplicaciones de prueba más avanzadas podrían incluir características tales como:

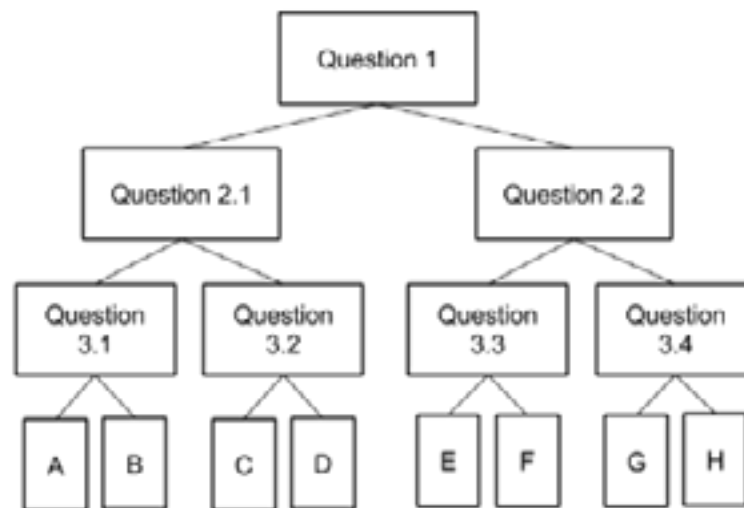
- Diferentes tipos de preguntas (emparejar, respuesta libre o incluso juegos interactivos)
- Hacer aleatorio el orden de las preguntas
- Cambiar las preguntas según cómo lo haremos.
- Proporcionar un puntaje final

- Incorporar material de estudio

Tomador de decisiones

Similar a la aplicación de prueba, una aplicación de árbol de decisión le hace al usuario una serie de preguntas. La principal diferencia entre los dos es que un árbol de decisiones no verifica las respuestas correctas o incorrectas, sino que intenta sugerir al usuario o ayudarlo a tomar una decisión en función de sus respuestas.

Este simple tomador de decisiones solo hace tres preguntas, pero las más avanzadas en realidad toman la forma de un árbol como se ve a continuación:



Dependiendo de la respuesta del usuario a cada pregunta, se les da una página diferente para la siguiente pregunta. Esto permite que la aplicación reduzca gradualmente la decisión una pregunta a la vez. Este tipo de aplicación puede ser realmente útil para ayudar a los usuarios a comprender procesos complejos; por ejemplo, puede presentar un programa de reciclaje utilizando una aplicación de árbol de decisión para ayudar a los usuarios a determinar en qué contenedor ordenar los elementos; primero preguntar de qué material está hecho, entonces si está limpio o no, y así sucesivamente

Administrador de listas

La aplicación de administrador de listas le permite al usuario hacer un seguimiento del estado de la información y puede modificarla para cumplir diferentes propósitos, como registrar / retirar elementos prestados, hacer un seguimiento de las tareas personales.

Si bien este ejemplo se centra en una sola lista, las aplicaciones de lista más sofisticadas podrían permitir al usuario crear múltiples listas, o incluso mover elementos entre las listas. Esta aplicación también solo realiza un seguimiento del nombre de un elemento y de si está o no "Hecho", pero podría extenderse para rastrear y organizar cualquier cantidad de información sobre los tiempos de la lista.

Crowdsourcing (colaboración abierta distribuida)

Las aplicaciones de Crowdsourcing (colaboración abierta distribuida) permiten que múltiples usuarios envíen contenido para que todos lo usen (la fuente del contenido es la multitud). Esta aplicación es muy similar a la aplicación de la lista, con la diferencia principal de que la aplicación de la lista asume un solo usuario, mientras que esta aplicación permite a todos los usuarios agregar a la misma lista y enviar propuestas “me gusta”.

Este ejemplo utiliza una lista de información de fuentes múltiples para simplificar, pero puede mostrar cualquier tipo de información en muchos formatos diferentes. Una aplicación de Crowdsourcing (colaboración distribuida) podría permitir a los usuarios enviar ubicaciones para mostrarlas en un mapa, imágenes para acceder a una galería o cualquier cantidad de otros tipos de información.

Lluvia de ideas: Elija uno de los tipos de aplicación y haga una descripción de un párrafo de la aplicación.

Ampliación del conocimiento (70-90 min)

Diseño de prototipos

Definir: Para establecer las expectativas, es importante reiterar continuamente a los estudiantes que están desarrollando prototipos, cada uno agregando más funcionalidad que la anterior.

Observaciones: Para mejorar la idea de su aplicación, cada equipo creará y probará un prototipo de papel. Al igual que con los prototipos que crearon anteriormente, esto permitirá una exploración rápida con una inversión mínima. Los diseñadores pueden iterar y cambiar bocetos rápidamente en función de los comentarios de los usuarios.

Discute: Involucre a la clase en una discusión sobre por qué este proceso podría ser preferible a solo saltar en la programación de las aplicaciones.

Indicación: Mientras hacen esto, también desearán crear un diagrama de navegación para describir el flujo entre las pantallas. Esto será útil para las pruebas de usuario para su presentación final.

Discutir: Aplicación Ejemplar

Pantalla: [¿Qué hay para el almuerzo? Prototipo](#). Aquí hay un ejemplo del prototipo de baja fidelidad para nuestra aplicación de ejemplo “¿Qué hay para el almuerzo?”

El proceso en el que participan los estudiantes se conoce como diseño participativo: una filosofía de diseño para sistemas informáticos que se originó en Escandinavia en los años ochenta. El objetivo es lograr que los usuarios finales del diseño participen en el proceso de manera temprana y frecuente.

Los estudiantes pueden proponer varios motivos válidos, pero querrá asegurarse de resaltar al menos lo siguiente:

- Es más fácil y rápido esbozar una idea
- Probar ideas en papel le permite encontrar problemas en sus suposiciones antes de dedicar tiempo a la programación
- Los usuarios son más propensos a dar su opinión sobre algo que parece menos pulido/terminado.

Comenta: ¿Qué notas sobre este prototipo?

Distribuir:

- [Prototipo de lluvia de ideas- Guía de actividades](#)

Preparación para prototipos de papel

Prototipo de lluvia de ideas

Pantallas sugeridas: La primera página contiene una lista de pantallas sugeridas para los tipos de aplicaciones con los que los estudiantes experimentaron durante el precalentamiento. Los equipos ya deben haber seleccionado un tipo de aplicación para que puedan usar esto como punto de partida para determinar qué pantallas necesitarán.

Elementos de UI: La segunda página contiene una tabla de elementos de diseño disponibles en App Lab. Para cada elemento, los equipos discutirán qué creen que es y cómo se puede usar en su aplicación. Los equipos no necesitan (y probablemente no lo hagan) incorporar todos los elementos, pero deben considerar cómo se pueden usar en el contexto de su aplicación.

Compartir: Después de dar tiempo a los equipos para completar la tabla, revise la lista y pida a un grupo que comparta cómo podrían usar cada uno de los elementos. Tiene la oportunidad de asegurarse de que la clase comprenda claramente qué es y qué hace cada elemento.

Las pantallas de nuestra aplicación: La página final proporciona una tabla para que los equipos enumeren todas las pantallas que necesitarán. Haga saber a los estudiantes que es probable que necesiten más pantallas de las que se enumeran en las “pantallas sugeridas”, y es posible que no incluyan todas las pantallas que finalmente creen, pero debería abarcar todas las funciones que actualmente planean incorporar.

Prototipos de papel

Distribuye: Fichas de 3x5, notas adhesivas, lápices de colores y lápices, tijeras para cada equipo.

Indicaciones: Utilizarán esta guía de actividades, dibujarán cada una de las pantallas de sus aplicaciones en una tarjeta separada.

Recorrer la sala: Durante este período prolongado de trabajo, los equipos necesitarán dividirse y al mismo tiempo asegurarse de que están trabajando hacia un objetivo común. Asegúrese de que

Incentive las conexiones a la discusión previa. ¿Cómo este prototipo logra (o no logra) los objetivos de un prototipo de baja fidelidad?

- ¿Están suficientemente detallados para saber qué hará cada pantalla?

- ¿Estos se ven muy pulidos? ¿O esbozados?

- ¿Están relativamente limpios?

¿O al menos lo suficientemente ordenado para que el usuario de la prueba pueda leer lo que necesita?

- ¿Qué características de la interfaz de usuario hay en cada pantalla? (buscando cosas como botones, imágenes, texto, etc)

- ¿Piense que un niño podría hacer clic en estos elementos con sus dedos pequeños? ¿Cree que los dedos de un adulto podrían caber en las características importantes?

- ¿Está claro el flujo a través de las pantallas?

- Si fuera un usuario que prueba esta aplicación, ¿qué comentarios podría hacerles a los desarrolladores?

Si fuera el desarrollador de la aplicación, ¿qué le gustaría saber de los usuarios de prueba?

Presentamos los elementos de App Lab en este punto para que los estudiantes comprendan qué herramientas tendrán disponibles. Los estudiantes pueden desviarse de los elementos proporcionados, pero cuando llegue el momento de desarrollar su prototipo digital necesitarán

los equipos continúen comunicándose a medida que desarrollan sus pantallas y anime a los equipos a ver con frecuencia cómo funcionan las pantallas juntas.

Transferencia del conocimiento (5 min)

Recoger prototipos

Recolectar: Reúna los prototipos de papel para que puedan probarse en la próxima lección. El acceso a estos prototipos será esencial para las próximas actividades, así que deberá considerar establecer una ubicación específica en la sala para que los equipos los almacenen.

→
cambiar a uno de los
elementos admitidos o usar
una imagen como suplente.

Lección 93

Pruebas de prototipos

Propósito

El objetivo principal del desarrollo de prototipos de papel es que permiten realizar pruebas e iteraciones rápidas antes de escribir cualquier código. Esta lección se centra en dar a los equipos la oportunidad de probar sus prototipos antes de pasar a App Lab. Los equipos desarrollan un plan para probar con los usuarios antes de ejecutar pruebas prototipo con otros estudiantes en la clase (y potencialmente fuera de la clase). Para probar el prototipo con los usuarios, los estudiantes tendrán que asignar roles en la prueba (el “narrador”, la “computadora” y los “observadores”), así como algunas preguntas preparadas para que el usuario responda después de la prueba.

El objetivo de esta lección es brindar a los estudiantes un formato claro para probar e iterar sus aplicaciones. Esta será la primera de múltiples oportunidades que los equipos tendrán para probar varias etapas de sus prototipos, y cada etapa tendrá un propósito diferente. En este punto, el propósito principal es controlar las suposiciones sobre cómo se debe diseñar y navegar la aplicación: este no es el momento para que los estudiantes se preocupen demasiado por los detalles.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento 1 – Prueba (45 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Probar un prototipo con un usuario, registrando los resultados.
- Analizar una prueba de usuario para identificar posibles problemas o mejoras.

Lección sin conexión
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Prueba de usuario prototipo de papel - Guía de actividades](#)

Preparación

- Tener a otras personas alineadas para probar los prototipos de papel de cada equipo, o programar tiempo suficiente para que los equipos prueben los prototipos de los demás.
- [Prueba de usuario de Prototipo de papel- Guía de actividades](#)

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Preparándose

Distribuir: Asegurarse de que cada equipo tenga sus prototipos a la mano.

Preguntar: Antes de considerar probar con otros usuarios, tomen un momento como equipo para trabajar en sus pantallas.

Ampliación del conocimiento 1 – Prueba (45 min)

Preparar

Cuando realizan un experimento en la clase de ciencias, hacen su mejor esfuerzo para probar una hipótesis a la vez. Por ejemplo, si desean probar una teoría de que la comida se echará a perder más rápidamente fuera del refrigerador que dentro, no querrá probar con una nevera caliente. Tampoco querrían probar con temperaturas oscilando violentamente, eso podría afectar los resultados de cuales se echan a perder primero. Por lo tanto, les conviene controlar la temperatura “variable” de la comida tanto dentro como fuera del refrigerador y ver qué sucede.

Es lo mismo cuando pruebas software. Aunque las personas son muy variables, deben eliminar tantas “variables” como sea posible.

Una forma de hacerlo es asegurarse de hacer las mismas preguntas cada vez que prueban un software. Así que vamos a trabajar en una lista de preguntas para aplicar a nuestros usuarios cuando hayan terminado de probar nuestros prototipos de baja fidelidad, para que podamos comparar las reacciones de los diferentes usuarios con sus aplicaciones y sus respuestas a sus mismas preguntas.

Pantalla: Muestre [Para qué es el almuerzo de prueba- Video](#) . Nota: activar los subtítulos en español.

Discute: ¿Qué notaron sobre cómo se realizó esta prueba? Específicamente investiguen los siguientes roles que se jugaron:

- El “usuario” es la persona que está probando la aplicación en forma de prototipo de baja fidelidad. El usuario debe pretender ejecutar la “aplicación” presionando el prototipo con los dedos de la manera que tenga más sentido. La parte más importante es que el usuario debe hablar en voz alta lo que está pensando mientras realiza las acciones y hace muchas preguntas si hay cosas que no entiende. También pueden ofrecer sugerencias útiles con oraciones que comienzan con “Me gusta ...”, “Deseo...” y “Me pregunto ...”
- El “computador” es la persona que manipula el prototipo de fidelidad en función de lo que hace el usuario. Por ejemplo, si el usuario presiona un botón que debería hacer que la aplicación vaya a otra pantalla, el “computador” quitará la simulación de la pantalla anterior y la reemplazará con la simulación de la siguiente pantalla. El “computador” comienza la prueba presentando al usuario la primera pantalla de la aplicación.
- El “narrador” es la persona que está ejecutando la prueba. Esta persona presentará a los

miembros del equipo, la aplicación y su propósito. Esta persona también le recordará al usuario que debe hablar en voz alta mientras manipula la aplicación y le recordará a la “computadora” y a los “observadores” que eviten tratar de guiar al usuario en lo que creen que es la forma correcta de usar la aplicación, a menos que el usuario pida ayuda.

- Los “observadores” son los otros estudiantes del equipo. Observarán la interacción y anotarán en sus notas lo que ven que hace el usuario en respuesta a la computadora.

Distribuir: Una copia de [Prueba de usuario prototipo de papel-Guía de actividades](#) para cada equipo

Prueba de usuario prototipo de papel

Descripción general: Como clase, revisen los objetivos para la prueba del usuario. En particular, respondan a cualquier pregunta sobre los diferentes roles.

Asignar funciones para la prueba: Pida a los equipos que asignen roles para sus pruebas. Si lo desean, los roles pueden cambiarse entre las pruebas.

Identificar usuarios: Los equipos deben estar vinculados con otro equipo para probar su aplicación.

Prueba de usuario

Configuración: Decida cómo se juntarán los equipos para las pruebas y colocar el arreglo donde los estudiantes puedan ver.

Indicaciones: Al utilizar [Prueba de usuario prototipo de papel-Guía de actividades](#), probarán su aplicación con un usuario.

Recorrer la sala: Los estudiantes comenzarán sus pruebas durante aproximadamente 5-7 minutos cada uno. Anime a los estudiantes a que sigan con la tarea y a los observadores a escribir todo lo que puedan. Después de que los estudiantes terminen, pida que regresen a su equipo original.

Resumir los hallazgos: Haga que los equipos discutan lo que observaron y registren sus hallazgos en la primera página de la guía de actividades. En particular, pregunte qué significan sus observaciones en términos de cambios que deberán realizar para la interfaz de usuario de su prototipo.

Los estudiantes pueden proponer varios motivos válidos, pero es importante resaltar al menos lo siguiente:

- Es más fácil y rápido esbozar una idea
- Probar ideas en papel le permite encontrar problemas en sus suposiciones antes de dedicar tiempo a la programación
- Los usuarios son más propensos a dar su opinión sobre algo que parece menos pulido/terminado.

Transferencia del conocimiento (5 min)

Reflexión

Escribe en tu cuaderno las respuestas a estas preguntas:

- ¿Hubo alguna diferencia entre las pruebas con un usuario que estuvo involucrado en el desarrollo de su prototipo de baja fidelidad (lo que hicimos ayer) y las pruebas con un usuario que nunca había visto esta aplicación?
- ¿Cuáles fueron algunas de las similitudes entre los dos tipos de usuarios?
- ¿Qué diferencia viste entre los dos tipos de usuarios?

Sugerencias de evaluación

Se sugiere los siguientes indicadores para evaluar formativamente los aprendizajes:

- Documenta las pruebas realizadas, obteniendo conclusiones y mejoras
- Reconoce los aportes de los colaboradores describiendo los beneficios en el trabajo realizado

Lección 94

Diseño digital

Propósito

Después de haber desarrollado en colaboración un prototipo en papel para sus aplicaciones, los equipos ahora se dividen y organizan para comenzar a trabajar en una versión digital interactiva basada en el prototipo de papel. Usando el modo de diseño de arrastrar y soltar de App Lab, los estudiantes trabajan individualmente a través de una progresión de niveles de desarrollo de habilidades para aprender a construir versiones digitales de un prototipo de papel. Desde allí, cada miembro del grupo construye al menos una página de su aplicación en App Lab, para luego combinarla en una sola aplicación.

Esta lección presenta a los estudiantes la interfaz del modo de diseño de App Lab y el proceso de traducción de un prototipo en papel a digital. Utilizarán esta herramienta para iterar en sus prototipos de papel, construyendo primero una pantalla de aplicación ejemplar y, finalmente, un prototipo de aplicación interactiva al final de la Lección.

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (40 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Traducir un prototipo de papel a un formato digital.
- Seleccionar el elemento de entrada apropiado para un tipo de información dado.

Lección en línea
[Ver en Code Studio](#)

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Propiedad de la pantalla - Guía de actividades](#)
- [Diseño de pantalla - Guía de actividades](#)
- [Diseño de pantalla del modelo- Guía de actividades](#)

Preparación

- Propiedad de la pantalla- Guía de actividades
- Diseño de pantalla- Guía de actividades
- Diseño de pantalla del modelo- Guía de actividades.

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Divide y conquistarás

Observaciones: Hoy tomaremos los prototipos de baja fidelidad en los que han estado trabajando y comenzaremos a desarrollar las versiones digitales que conducirán a su prototipo final. Con sus equipos, repartan las pantallas entre los miembros de su grupo.

Distribuir: Entregue a cada equipo una [propiedad de pantalla: guía de Lección](#)

Propiedad de la pantalla: Haga que los equipos revisen sus prototipos en papel para identificar todas las pantallas únicas que necesitarán crear. A partir de ahí, pueden dividir las pantallas de manera equitativa entre los miembros del grupo y documentarlo en el gráfico.

Si hay más pantallas que estudiantes, los equipos tienen en cuenta los siguientes factores al dividir el contenido:

- ¿Es cada página realmente única, o algunas representan la misma página con contenido ligeramente modificado?
- ¿Hay páginas similares que puedan ser creadas fácilmente por la misma persona?
- ¿Hay varias páginas simples que se pueden agrupar?

Ampliación del conocimiento (40-60 min)

Modo de diseño en App Lab

Pantalla: Ponga el [diseño de la pantalla del modelo - Guía de actividades](#) donde los estudiantes puedan verlo, o distribuya copias impresas. Este prototipo también estará disponible para ellos durante la progresión del nivel.

Observaciones: Antes de comenzar a diseñar sus propias aplicaciones en App Lab, todos vamos a trabajar primero en la conversión de este prototipo de papel. Revise el prototipo para asegurarse de saber para qué están construyéndolo. Tome en cuenta que la mayoría de los elementos en la pantalla están etiquetados: les conviene asegurarse de usar las mismas etiquetas exactas a medida que construyen esta pantalla en App Lab.

Distribuir: Entregue [Diseño de pantalla - Guía de actividades](#) cuando los estudiantes alcancen el nivel final de esta lección. Los

Reglas para los nombres de pantalla: Para cada pantalla, los equipos deben asignar un nombre corto y claro, no debe contener espacios ni caracteres especiales. Este nombre se usará en App Lab para identificar su pantalla.

interactuar con su aplicación. Esta pantalla prototipo necesitará tres botones. Ya hemos agregado el botón Buscar, por lo que solo debe agregar Contacto y Acerca de.

7. **Desarrollar buenos hábitos de programación:** Seguir las convenciones de nomenclatura claras y consistentes es un hábito importante para desarrollar, pero los beneficios raramente son inmediatamente obvios para los estudiantes. Realmente vale la pena a largo plazo ser un riguroso para las identificaciones bien nombradas, particularmente porque salvará tu propia cordura cuando llegue el momento de leer los programas de tus estudiantes, pero también simplificará el proceso de combinar la pantalla de cada alumno en una aplicación de equipo completo

8. **Espaciado de nombres:** Asegúrate de que todos los estudiantes creen identificaciones únicas para todos los elementos y el espacio de nombre adecuado, les ayude a evitar problemas en la próxima Lección, por lo que vale la pena ser un riguroso aquí. Puedes proporcionarles a los estudiantes la Guía de actividades de diseño de pantalla opcional para realizar un seguimiento de sus ID de elemento.

9. Trabajando juntos, promueve la discusión activa entre los equipos mientras diseñan pantallas para asegurarte de que todos estén trabajando hacia un objetivo común.

Compartir diseños

Discute: ¿Cómo cambiaron sus diseños iniciales de papel cuando los volvió a crear en App Lab?

¿Qué tipo de ajustes tienen que hacer?

Compartir: Pida a algunos estudiantes que compartan sus pantallas con la clase, centrándose en mostrar una variedad de diseños

Transferencia del conocimiento (5 min)

Preguntas grupales

Discute: Dé a los equipos unos minutos para analizar su experiencia al convertir sus diseños de papel en digitales. En particular, se concentrará en los cambios que se deben realizar en la conversión para que todo el equipo conozca los cambios que se tuvieron que realizar, y que es necesario tenerse en cuenta.

Sugerencias de evaluación

Se sugiere los siguientes indicadores para evaluar formativamente los aprendizajes:

- Identifican mejoras a los problemas
- Planifican y priorizan las mejoras
- Gestionan roles de trabajo y comunican
- Documentan y explican el proceso de trabajo
- Reconocen los beneficios de la colaboración

Lección 95

Vinculación de pantallas

Propósito

En esta lección, los equipos combinan las pantallas que diseñaron en la lección anterior en una sola aplicación, que luego pueden vincular utilizando código. Los estudiantes aprenden programación basada en eventos básicos mediante la construcción de la aplicación modelo que comenzaron en la lección anterior. Además de la pantalla que los estudiantes diseñaron, aprenderán a crear pantallas adicionales e incluso a importar pantallas hechas por otros.

Esta lección y la siguiente se centran en conceptos de programación, y la medida en que lo hacen es mínima. El objetivo de esta Lección no es enseñar programación en App Lab, sino más bien enseñar la cantidad mínima de habilidades de programación para permitir que los estudiantes hagan sus aplicaciones interactivas.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (45-75 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Escribir programas que respondan a la entrada del usuario.
- Integrar pantallas diseñadas por otros en una aplicación propia.
- Colaborar con otros para desarrollar un prototipo interactivo.

Lección en línea
[Ver en Code Studio](#)

Código

- `onEvent(id, type, function(event)){ ... }`
- `setScreen(screenId)`
- `console.log`

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Reaccionando a eventos

Inducción: Recuerde cuando estaba programando en Game Lab, cuando querían responder a la interacción del usuario, ¿cómo lo hicieron?

Observaciones: En Game Lab usamos condicionales dentro del ciclo draw para verificar si un usuario había interactuado con el juego treinta veces por segundo. Esta técnica funciona muy bien en juegos y animaciones donde hay muchas otras cosas que también le está diciendo al computador que haga. La mayoría de las aplicaciones de teléfono, sin embargo, no están haciendo cosas constantemente. De hecho, muchas aplicaciones no hacen absolutamente nada más que esperar a que el usuario haga clic en algo.

Para aplicaciones como ésta, es mejor usar un concepto de programación llamado “Evento”, que le permite definir qué código debe ejecutarse cuando ocurren diferentes eventos. En App Lab, los “manejadores de eventos” se pueden unir a los elementos que se crearon en el modo de diseño, donde pueden observar la ocurrencia de varios eventos (como hacer clic en un botón).

Ampliación del conocimiento (45-75 min)

Vinculación de pantallas

Consejo de enseñanza: Durante esta lección, y para el resto de esta unidad, los estudiantes deberán compartir enlaces de proyectos entre sí para importar su trabajo compartido. Considera proporcionar a cada grupo un documento compartido para rastrear los enlaces de sus proyectos a fin de facilitar este proceso.

Esta lección comienza con la aplicación del modelo que los estudiantes comenzaron en la lección anterior. Agregarán e importarán pantallas adicionales, y luego escribirán código que vincula sus pantallas a botones.

Niveles de Code Studio

Multi-Screen Apps: Puede imaginarse que una aplicación típica tiene una cantidad de “pantallas” o vistas que atraviesa un usuario como parte del uso de tu aplicación. A continuación, se muestra un ejemplo típico con una pantalla de bienvenida, una pantalla que reproduce un juego en la pantalla.



¿Qué es una pantalla?

Una pantalla es solo un elemento de la interfaz de usuario como cualquier otra cosa en el modo de diseño. Pero tiene algunos elementos especiales.

Instrucciones para el alumno

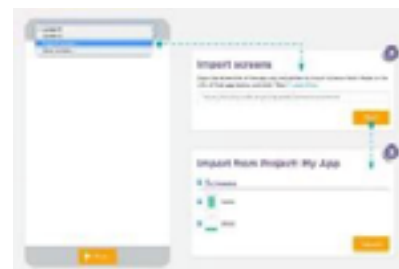
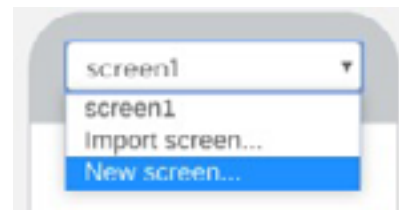
1. **Agregar una nueva pantalla:** Para completar tu aplicación, vas a necesitar más que solo una pantalla de inicio. Cada tarjeta que creaste en tu prototipo de papel está representada por una “pantalla” en App Lab. Para crear más pantallas, simplemente haz clic en el menú desplegable en la parte superior de la pantalla de la aplicación y selecciona “Nueva pantalla ...”
2. **Diseña la pantalla Acerca de:** Para agregar elementos a su nueva pantalla Acerca de, deberá seleccionarla primero en el menú desplegable de pantallas.
3. **Importación de pantallas:** Has visto cómo crear una aplicación con varias pantallas, pero ¿qué hay de trabajar con otras personas para colaborar en una aplicación? Afortunadamente, App Lab te permite importar pantallas desde otra aplicación. Desde el mismo menú desplegable donde puede agregar una pantalla, simplemente selecciona “Importar pantalla” y sigue las instrucciones (necesitarás el código para compartir desde la aplicación que estás importando las pantallas). Además de agregar nuevas pantallas en blanco, puedes importar pantallas desde “Importar pantalla ...”. Para importar una pantalla de otra persona, necesitarás que te proporcionen la URL para compartir.

Nota: ¡Las pantallas que importes no deben compartir ningún ID con elementos que ya estén en tu aplicación!

Respondiendo a la entrada del usuario

El modo de diseño hace que sea realmente fácil diseñar la interfaz de usuario para su aplicación, pero necesitaremos sumergirnos en el modo de código para que realmente responda a los usuarios. En Game Lab, el bucle Draw ejecuta código constantemente, usando condicionales para verificar las interacciones del usuario. App Lab, por otro lado, espera responder a ciertos eventos que ocurren en un programa. Es la manera en que App Lab dice “¡Oye, algo acaba de pasar!”. Esto significa que no es necesario que haya código en ejecución cuando no está interactuando directamente con una aplicación, y su código puede responder a cada uno de estos eventos a medida que ocurren.

Tarjetas a pantallas: Para simplificar el proceso de creación de prototipos, los equipos pueden pensar en cada tarjeta que crearon en Paper Prototyping como una pantalla separada en la iteración del App Lab de su aplicación. En realidad, las pantallas que son muy similares a menudo se pueden desarrollar como una sola pantalla con contenido que cambia según la interacción del usuario. En la próxima lección, los estudiantes verán algunos ejemplos de cómo podría funcionar, pero no se espera que incorporen ese tipo de funcionalidad en este proyecto.



En App Lab, todo el manejo de eventos comienza agregando un bloque especial de código llamado onEvent. Aquí está cómo hacerlo:



Eventos de botones

Con nuestras pantallas en su lugar, es hora de comenzar a programar realmente la aplicación para que responda a los clics del botón. Para cada botón en cada pantalla, desearás agregar un onEvent bloque que que “al hacer clic en ese botón, responda de manera adecuada”. Para comenzar, simplemente miraremos el botón “home_search_btn” e imprimiremos algo en la consola cuando se haga clic

Hacer esto

Ahora está en el modo de código (puedes usar los botones situados encima de tu aplicación para cambiar entre el modo de código y el de diseño). Aquí es donde puedes escribir el código que responderá a los usuarios que interactúan con tu aplicación.

- Arrastra un onEvent bloque desde la caja de herramientas del código.
- Selecciona “home_search_btn” del menú desplegable “id”.
- Arrastra un console.log bloque desde el cajón de variables.
- Ejecuta tu aplicación.
- Haz clic en el botón Buscar y busca mensajes en la consola.

```
onEvent(▼ "home_search_btn", ▼ "click", function(event) {
  console.log("Search clicked!");
})
```

Poniéndose al día: Si los estudiantes se perdieron la lección anterior o no pudieron completar la pantalla de inicio, puede hacer que importen esta versión completa de la pantalla de inicio.

Vea el [mapa de importación de pantalla](#) para más detalles.

Promoviendo el trabajo en equipo:

Los siguientes niveles prepararán a los equipos para fusionar todas sus pantallas por separado en una sola aplicación. Hay dos posibles obstáculos que pueden encontrar durante el proceso de importación:

- URL mal comunicadas: Las URL compartidas son largas y complicadas, por lo que los estudiantes necesitarán una ubicación compartida donde puedan copiar/pegar sus URL
- ID de pantalla en conflicto: la importación de una pantalla con la misma ID que una existente le permitirá sobrescribir la pantalla existente.

Esto es particularmente problemático cuando los estudiantes dejan la identificación de pantalla predeterminada “screen1”

- ID de elementos en conflicto: si los estudiantes han asignado correctamente todos sus elementos, esto no debería ser un problema, pero cualquier elemento que tenga la ID predeterminada o que comparta una ID con un elemento existente evitará que se importe una pantalla.

Una vez que los equipos

Cambio de pantallas

Usando el comando setScreen para cambiar las pantallas con el código

El comando setScreen cambia la pantalla para mostrar la pantalla que desee. Usted especifica la pantalla por su ID.



Cambio de pantallas al hacer clic

El setScreen bloque cambiará tu aplicación a la pantalla que especifiques. Si colocas un setScreen bloque dentro de un onEvent bloque, puedes programar tus botones para cambiar a la pantalla correcta

Receso por el día: Dependiendo de la cantidad de tiempo que tengas en clase, este puede ser un buen momento para un receso.

Tiempo de equipo: Aunque los estudiantes están trabajando en sus copias individuales de la aplicación de su equipo en este punto, anime a los equipos a ser activos durante esta parte de la lección. Cualquier problema que los estudiantes encuentren al importar las pantallas de sus compañeros de equipo solo se puede resolver mediante el trabajo en equipo, y es probable que sea un problema para todos los miembros del equipo.

Comparando con Game Lab

En Game Lab podríamos usar el siguiente código para verificar si se hizo el clic en un sprite llamado “botón” y hacer algo al respecto:

```
función draw () {
  if ( mousePressedOver ( button ) ) {
    // hacer algo
  }
}
```

pasan a la fase de programación de este proyecto, se vuelve mucho más difícil para los estudiantes colaborar en la misma aplicación. La fase de diseño es la oportunidad principal para que todos los estudiantes colaboren en la aplicación, después cada alumno trabajará en su propia copia.

Solo los hechos: Aquí hay mucho que consumir y muchos de los detalles sobre cómo funcionarán los eventos en App Lab no son totalmente necesarios para comprender nuestras necesidades en esta unidad. Si los estudiantes están luchando con la forma en que funciona la programación en App Lab, concéntrese en estos elementos esenciales:

- No hay ningún draw clic que se ejecute para siempre
- El onEvent () bloque específica:
 - un elemento ID
 - un evento (como “clic” o “mouseover”)
 - código que debe ejecutarse cuando ocurre el evento (este código es técnicamente una función de devolución de llamada anónima, representada por el bloque verde)
 - Code Inside onEvent () blocks se ejecuta en cualquier momento que ocurra el evento especificado

En App Lab, dado un “botón” ID, podríamos escribir el siguiente código para lograr el mismo objetivo:

```
onEvent ( "botón" , "click" , función () {
  // hacer algo
})
```

En el ejemplo de AppLab, la función es simplemente otro argumento pasado a la función OnEvent, sucede que el argumento debe ser una función en si misma. También podríamos definir la función por separado y escribir este mismo programa de la siguiente manera:

```
function doSomething () {
  // hacer algo
}
onEvent ( "botón" , "click" , hacer Algo )
```

Combinar pantallas de proyectos

En el segundo ejemplo observa que cuando pasamos la función doSomething a la onEvent no incluimos el paréntesis al final. Esto es porque en realidad estamos pasando toda la función como un objeto.

Revisión: En la Lección anterior, cada estudiante desarrolló una pantalla para la aplicación de su equipo. Dará a los equipos unos minutos para revisar el estado de sus pantallas, identificando cualquier trabajo restante antes de combinarlos. En el último nivel de esta Lección pedirá a los estudiantes que importen y vinculen cada una de las pantallas en las aplicaciones de su equipo.

Transferencia del conocimiento (5 min)

Equipo

Discuta: Ofrecerá a los equipos la oportunidad de compartir sus prototipos entre ellos. Debido a que cada estudiante está trabajando en una versión separada de la aplicación del equipo, cada aplicación probablemente será ligeramente diferente. Alentará a los miembros del equipo a analizar sus diferentes soluciones para el mismo problema: ¿cuáles son los beneficios o los intercambios?

Programación limitada

requerida: El objetivo de esta unidad es hacer que los estudiantes piensen en el papel que juega el diseño en el desarrollo de un producto y ayudarlos a generar empatía con los usuarios finales. Esta no es una unidad centrada en la programación, y el grado en el que se espera que los estudiantes programen está representado en su totalidad aquí.

Experiencias de aprendizaje de profundización

Use estos Contenidos para ampliar el aprendizaje de los estudiantes. Se pueden usar como Contenidos extras fuera del aula.

Par de pantalla

Brindará a los estudiantes la oportunidad de practicar diseños rápidos y creación de prototipos emparejándolos para diseñar rápidamente y fusionar dos pantallas para una nueva aplicación.

Lluvia de ideas: Como clase, pensarán en una lista de ideas de aplicaciones para fomentar una buena variedad.

Agrupación: Colocará a los estudiantes en parejas, preferentemente con compañeros de un equipo diferente. Dará a cada par unos minutos para elegir una aplicación de la lista y acordará la pantalla que cada pareja diseñará, luego comenzarán un nuevo proyecto de App Lab. Después de un tiempo limitado dedicado al diseño de sus pantallas, los pares pueden intercambiar códigos compartidos e intentar importar las pantallas de los demás. El objetivo aquí no es crear una aplicación que funcione, sino practicar prototipos rápidos e iteración. A través de esta práctica, los estudiantes pueden comenzar a priorizar mejor lo que debe y lo que no debe entrar en una primera iteración.

Sugerencias de evaluación

Se sugiere los siguientes indicadores para evaluar formativamente los aprendizajes:

- Identifican mejoras a los problemas
- Planifican y priorizan las mejoras
- Gestionan roles de trabajo y comunican
- Documentan y explican el proceso de trabajo
- Reconocen los beneficios de la colaboración

Lección 96

Prueba de la aplicación parte 1

Propósito

Cada equipo debe tener un prototipo mínimo viable de su aplicación. El objetivo principal de esta Lección es hacer que el equipo realmente pruebe la aplicación con otras personas, preferiblemente del público objetivo para la aplicación, o de diferentes equipos en la clase, mientras que los observadores del equipo registrarán los resultados en las hojas de trabajo que utilizaron en la fase de planificación. Al igual que con la prueba de los prototipos en papel, los equipos comenzarán planificando los escenarios específicos antes de ejecutar y observar las pruebas.

Por ahora, los estudiantes deberían ser bastante experimentados en las pruebas con los usuarios. ¡Esta vez, sin embargo, la computadora realmente hace el papel de la computadora! Esto significa que el equipo que realiza la prueba debe centrar su energía en observar cuidadosamente lo que hace el usuario y alentar al usuario a decir en voz alta lo que está experimentando. Si bien puede parecer al principio que las pruebas con un prototipo interactivo son más fáciles (ya que nadie tiene que cambiar manualmente las pantallas), en realidad puede ser mucho más difícil observar la experiencia del usuario.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento (45 min)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Escribir un plan detallado de cómo probarán su prototipo de baja fidelidad con otras personas.
- Ejecutar una prueba de usuario en una aplicación y registre lo que dicen los usuarios sobre su producto viable mínimo.

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Prueba de usuario de la aplicación- Guía de actividades](#)

Preparación

- [Prueba de usuario de la aplicación- Guía de actividades](#)
- Si es posible, tenga a los usuarios externos disponibles para probar las aplicaciones.

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Revisar el proceso de prueba

Discutir: Pedirá a los estudiantes que recuerden el proceso utilizado en la prueba del prototipo en papel. Las buenas preguntas para discutir con la clase incluyen:

- ¿Cuáles fueron los pasos que usamos para probar el prototipo de baja fidelidad? (Respuesta esperada: planificación, prueba, análisis)
- ¿Cuáles fueron algunos de los roles que jugaron nuestros compañeros de equipo en nuestras pruebas?
- ¿Crees que necesitamos el papel de la “computadora” en este escenario de prueba?
- ¿Qué es una tarea y cómo se relaciona con un caso de prueba?
- ¿Cuáles fueron algunas de sus tareas que revelaron información valiosa en las pruebas?
- ¿Cuáles fueron algunas de las buenas preguntas que hizo al final de la sesión de prueba?
- ¿Qué crees que será diferente sobre las pruebas con la aplicación vs las pruebas con el prototipo?

En su mayoría, esta revisión consiste en cambiar la perspectiva de la implementación y volver a las pruebas. También es para mostrarles lo similar que será esta sesión de prueba con lo que hizo la clase con los prototipos de papel.

Ampliación del conocimiento (45 min)

Preparación para la prueba

Distribuir:

- [Prueba de usuario de la aplicación- Guía de actividades](#)

Prueba de usuario de la aplicación

Esta guía de actividades es muy similar a la utilizada por los estudiantes para probar sus prototipos en papel. La principal diferencia es que no existe una función de “Computadora”, ya que la computadora real se encargará de eso. Esto también permite que el grupo ejecute más pruebas a la vez, así que considera dividir y conquistar a los equipos en las pruebas de esta iteración.

Esta prueba debe ser más larga que la prueba con el prototipo de baja fidelidad, ya que desea que el usuario sea lo más detallado posible. Si los estudiantes terminan temprano, pídeles que retrocedan y hagan una segunda ronda con la misma persona, tal vez para ver si esa persona se perdió algo o está dispuesta a “decir en voz alta” más información acerca de sus pensamientos mientras usan la aplicación.

Transición: Cuando los equipos hayan tenido tiempo suficiente para preparar sus casos de prueba, realizará la transición a la fase de prueba.

Ejecutando pruebas de usuario

Configuración: Decidirá cómo se juntarán los equipos para las pruebas y colocará el acuerdo donde los estudiantes puedan ver.

Indicaciones: Al usar la [Prueba del usuario de la aplicación- Guía de actividades](#) , prueba tu aplicación con un usuario.

Recorrer el salón: Los estudiantes comenzarán sus pruebas, que ejecutarán durante aproximadamente 10 minutos cada uno. Animará a los estudiantes a que sigan con la tarea y a los observadores a escribir todo lo que puedan. Después de que los estudiantes terminen, pedirá que regresen a su equipo original.

Transferencia del conocimiento (5 min)

Equipo

Discutir: Los equipos revisan los hallazgos de las pruebas de hoy y planifican las oportunidades de evaluación fuera del aula.

Lección 97

Mejorando e iterando parte 2

Propósito

Los equipos han desarrollado en este punto un prototipo de aplicación que ha pasado por múltiples iteraciones y rondas de pruebas de usuario. Con la información y la orientación obtenidas de la última ronda de pruebas de usuario, cada alumno tiene la oportunidad de planificar e implementar mejoras en la aplicación del equipo. Según el tiempo que tengas disponible y el interés del alumno, puedes ejecutar el ciclo de prueba e iteración tantas veces como te parezca.

Si bien es tentador hacer de esta Lección una inmersión más profunda en la programación para todos los estudiantes, anima a cada alumno a encontrar los elementos del diseño de la aplicación que te interesen y concéntrense en eso para sus mejoras personales. Si bien la mejora puede significar agregar funcionalidad programática, también puede significar un diseño más llamativo, una copia de texto más informativa, una mejor uniformidad de la iconografía o cualquier cantidad de otras características no relacionadas con la programación.

En este punto del proceso de desarrollo, pueden optar por dedicar todo el tiempo que desees a las iteraciones y las pruebas. No hay más tiempo presupuestado para la prueba después de esta iteración, pero pueden utilizar las dos lecciones anteriores como una guía para el ciclo a través de iteraciones adicionales.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Lección 1 (30 min)

Lección 2 (30-90 min)

Transferencia del conocimiento (15 min)

Objetivos

Los estudiantes serán capaces de:

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Análisis de prueba del usuario de la aplicación - Guía de actividades](#)

Vocabulario

- **Error:** Parte de un programa que no funciona correctamente.
- **Característica:** Un elemento funcional individual de software, como la capacidad de hacer algo nuevo, trabajar en múltiples plataformas o realizar de manera más eficiente.

- Analizar los comentarios de los usuarios de la Lección anterior y determina una lista de errores (fallas) que deben corregirse y características que podrían agregarse a la aplicación.
- Priorizar los errores y características de acuerdo con el impacto y la facilidad de implementación.

Preparación

- [Análisis de prueba del usuario de la aplicación: guía de actividades](#)
- Colocar notas adhesivas y papel para cada equipo.

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Errores y características

Indicación: En función de las pruebas de usuario, ¿cuáles son algunos de los errores que descubriste en tus aplicaciones? Manten un registro de estos en el tablero.

Discute: Es probable que encuentren muchos errores, pero ¿qué otros comentarios recibiste de los usuarios? Ayudará a la clase a identificar los comentarios que implican la necesidad de nuevas funcionalidades. Vamos a llamar a estas solicitudes funciones. Hoy los equipos trabajarán para identificar los errores y las características descubiertas a través de las pruebas, priorizarlas y planificar mejoras.

Los errores a menudo son más fáciles de identificar para los estudiantes que los comentarios que las características. Aliente a los estudiantes a considerar realmente cómo los usuarios podrían haber supuesto que la aplicación debería funcionar, o los lugares donde necesitaron ayuda o dirección para navegar con éxito.

Lección 1 (30 min)

Distribuye: Papel, notas adhesivas y una copia del Análisis de pruebas del usuario de la aplicación: Guía de actividades para cada equipo.

Análisis de prueba del usuario de la aplicación

1. **Interpretar los comentarios del usuario:** Los equipos comienzan completando una tabla en T que ayudará a conectar las observaciones de prueba de usuarios específicos con los posibles errores o características que faltan que revelan. Esta información será la base de una sesión de lluvia de ideas para crear una lista de errores encontrados en la aplicación existente y características que el equipo quisiera agregar a nuestra próxima iteración de la aplicación en la siguiente Lección.

2. **Sesión de lluvia de ideas:** Una vez que los equipos han organizado todos sus comentarios en la tabla T, pueden pasar a la fase de lluvia de ideas. Completa una nota adhesiva por cada error o característica identificada. La parte superior de la nota adhesiva debe decir ERROR o FUNCIÓN. La mitad de la nota adhesiva debe ser una descripción de cuál es el error o las características, incluidos los pasos para reproducir el problema si es necesario. La parte inferior de la aplicación debe tener una estimación rápida del tiempo (en minutos) que tardará en solucionar este error o implementar esta característica. Deja que los estudiantes sepan que tendrán la oportunidad de refinar este cálculo más adelante

Es tentador centrarse únicamente en los errores que son la causa de, o se puede resolver con, el código. Recuerde a los estudiantes que hay muchos roles y conjuntos de habilidades en los equipos de desarrollo de software, y que no todos los errores y características son tratados por los programadores. El color inconsistente, el texto confuso y los diseños contraintuitivos son todos errores potenciales que es importante tratar. Esta es una oportunidad para que los estudiantes con otros conjuntos de habilidades brillen y hagan una fuerte contribución a sus aplicaciones, así que haga un esfuerzo extra para ayudar a resaltar a esos estudiantes.

en la Lección; todo lo que es necesario ahora es un cálculo aproximado.

3. **Análisis de fallas y funciones:** En una hoja de papel, dibuja los dos cuadros en la página dos de esta guía de actividades. Para cada una de las notas adhesivas generadas en la fase anterior, discute si es urgente o no, y si parece ser fácil o difícil de implementar. Con base en esa discusión, coloca el adhesivo en el cuadrante apropiado.

Discute: Ahora que tienen sus errores y características categorizados, cuál de las cuatro categorías debería ser la primera que aborda. ¿Cuál debería ser el último?

Lección 2 (30-90 min)

Reparar errores y agregar funciones

Niveles de Code Studio

Implementación de funciones y reparaciones de forma individual:

Debido a que App Lab es difícil para que equipos grandes de estudiantes trabajen de forma simultánea, los estudiantes trabajan en la implementación de funciones y soluciones individualmente o en pares. El beneficio de este enfoque es que permite que el equipo en su conjunto explore más características potenciales que si estuvieran trabajando en la misma aplicación, pero también significa que la aplicación de cada miembro del equipo será diferente de las otras a medida que se realicen más cambios.

Como Lección de extensión opcional, podría proporcionar un día de trabajo para que los estudiantes compartan sus cambios entre ellos e intenten integrar todas las funciones agregadas en una sola aplicación.

Mantener registro de los cambios:

Cuantas más funciones agreguen los estudiantes a su aplicación, más probabilidades tendrán de presentar nuevos errores imprevistos. Para garantizar que los estudiantes siempre guarden la copia de trabajo más reciente de su aplicación, deben hacer un remezcla antes de agregar nuevas funciones. Asegúrese de que los estudiantes dejen el remix solo y vuelvan a este nivel para hacer su programación; esto garantizará que cuando vayan a enviar su aplicación para su revisión, sea la versión más reciente

Esta Lección puede tomar tanto o tan poco tiempo como desee, siempre y cuando a cada alumno se le permita el tiempo para implementar al menos una corrección o característica. Incluso puede considerar volver a esta Lección más adelante en el año cuando tenga algo de tiempo libre; la distancia desde un problema puede agregar una perspectiva útil.

Transferencia del conocimiento (15 min)

Preguntar: Pedirá a los estudiantes que reflexionen sobre el desarrollo de las cinco prácticas de Descubrimientos CS (Resolución de problemas, Persistencia, Creatividad, Colaboración, Comunicación). Elegirá una de las siguientes indicaciones según lo considere apropiado:

- Elige una de las cinco prácticas en las que crees que demostraste crecimiento en esta Lección. Escribe algo que hiciste que ejemplifica esta práctica.
- Elige una práctica en la que pienses que puede seguir creciendo. ¿Qué te gustaría mejorar?
- Elige una práctica que pensaste que era especialmente importante para la Lección que completamos hoy. ¿Qué lo hizo tan importante?

Experiencias de aprendizaje de profundización

Use estos Contenidos para ampliar el aprendizaje de los estudiantes. Se pueden usar como Contenidos extras fuera del aula.

Fusionar nuevas características

En esta fase de iteración, cada estudiante ha estado trabajando en su propia copia de la aplicación del equipo. Si tienes tiempo disponible, considera la posibilidad de que los equipos vuelvan a reunirse para fusionar sus actualizaciones en una sola aplicación maestra. Esto puede ser bastante complicado, especialmente cuando los estudiantes pueden haber agregado códigos conflictivos o elementos de diseño, pero es un desafío excelente para desarrollar habilidades para resolver problemas y fortalecer la dinámica del equipo.

Seguimiento del progreso

Utilizando la tabla de tareas, comience a mover notas adhesivas desde el cuadrante Urgente / Fácil a la columna de tareas pendientes. Elija al menos dos tareas por miembro del grupo y escriba en la carpeta adhesiva a quién está asignada la tarea. A medida que trabajes en la mejora de la aplicación, moverás las notas adhesivas a través de las etapas de esta tabla.

Sugerencias de evaluación

Se sugiere los siguientes indicadores para evaluar formativamente los aprendizajes:

- Presentan un análisis de los resultados del trabajo
- Describe errores y mejoras
- Planifican la resolución de errores

Lección 98

Proyecto – Presentación de la aplicación

Propósito

En este punto, los equipos han investigado un tema de importancia personal y social, han desarrollado y probado tanto un prototipo en papel como un prototipo digital, y han repetido la aplicación inicial para incorporar nuevas funciones y correcciones de errores. Ahora es el momento para que revisen lo que han hecho y junten una presentación coherente para demostrar su proceso de creación. Utilizando la plantilla de presentación proporcionada, los equipos se preparan para presentar sobre su proceso de desarrollo de aplicaciones, incluido el problema que se proponen resolver, las formas en que han incorporado los comentarios de las pruebas y sus planes para el futuro.

Esta Lección es la culminación de las últimas semanas de trabajo del proyecto. Al prepararse para presentar su trabajo, es importante que los estudiantes vean esto como algo más que el elemento de programación de su aplicación: el objetivo de la presentación es reconocer la investigación, el diseño, las pruebas y la iteración que han experimentado los equipos.

Secuencia para el aprendizaje

Conocimiento inicial (5 min)

Ampliación del conocimiento 1 (40-50 min)

Ampliación del conocimiento 2 (10 min por equipo)

Transferencia del conocimiento (5 min)

Objetivos

Los estudiantes serán capaces de:

- Presentar la información técnica claramente a usuarios no técnicos.

Lección en línea

Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- [Plantilla de presentación de la aplicación](#)
- [Presentación de la aplicación: Ejemplar](#)
- [Presentación de la aplicación- Rúbrica](#)

- Reflexionar sobre el desarrollo de un proyecto en curso.

Preparación

- Determina cuánto tiempo tiene para presentar cada grupo. Normalmente, esto es de 8 a 10 minutos por equipo, pero recuerda dejar tiempo para la transición entre equipos.
- Crea una copia del archivo de diapositivas de la plantilla de presentación para cada equipo.
- [Presentación de la aplicación- Rúbrica para cada equipo.](#)

Estrategia de aprendizaje

Conocimiento inicial (5 min)

Inspeccionando el ejemplar

Discute: Discutirá con la clase todo lo que han completado hasta ahora. Este ha sido un proyecto grande, y los estudiantes pueden no darse cuenta completamente de lo que han logrado hasta ahora. A medida que la clase se acerque a los logros, preguntará cómo podrían presentar las cosas bastante técnicas que han hecho a una audiencia no técnica (como usuarios potenciales de sus aplicaciones).

Pantalla: Recorrerá la [presentación de la aplicación: ejemplo \(en inglés\)](#), explicando que esta es la estructura general de la presentación que realizará cada equipo.

Ampliación del conocimiento 1 (40-50 min)

Preparación de presentación

Distribuir: Distribuirá la [Presentación de la aplicación - Rúbrica](#) y hará que cada equipo haga una copia de la [Plantilla de presentación de la aplicación](#) para construir su presentación.

Plantilla de presentación

La mayor parte del contenido de estas diapositivas ya debería haberse completado en partes anteriores de este proyecto. Las últimas diapositivas (en particular, Trabajo futuro y Reflexión) requerirán más reflexión grupal o trabajo en equipo, por lo que los estudiantes deben abordarlas primero.

- **Diapositiva 1:** Nombre de la aplicación y lista del equipo.
- **Diapositiva 2:** Una breve introducción del problema que la aplicación pretende abordar y cómo lo hace, así como el impacto social de esta aplicación.
- **Diapositiva 3:** Describe la investigación de mercado que se realizó, incluidas las aplicaciones específicas que se usaron como inspiración.
- **Diapositiva 4:** Muestra imágenes del prototipo de papel y describe el estado de la aplicación durante la creación de prototipos de papel.
- **Diapositiva 5:** Muestra imágenes del prototipo digital y

La comunicación es una habilidad crucial a menudo pasada por alto en informática, tanto internamente (con otras personas en el equipo) o externamente (a otros en la empresa o incluso fuera de la empresa). Los estudiantes están practicando estas habilidades en esta presentación, y estas habilidades serán transferibles a muchas de otras clases.

Si los estudiantes están “atascados” sobre lo que salió bien o mal en el proyecto, pídeles que piensen en la comunicación entre los miembros del equipo, o la falta de tiempo para completar la tarea. Por lo general, estas son las dos cosas principales que pueden ser problemáticas en un proyecto.

describe los cambios introducidos en esta iteración.

- **Diapositiva 6:** Demostrar la aplicación, intentando abordar todas las funciones principales. Los equipos pueden elegir enfocarse en la versión más funcional de su aplicación, o pueden demostrar la funcionalidad en múltiples versiones de la aplicación.
- **Diapositiva 7:** Enumera los comentarios clave encontrados en cada ronda de prueba del usuario. Alienta a los estudiantes a enfocarse en los comentarios que más directamente afectaron las siguientes iteraciones.
- **Diapositiva 8:** Describe cuáles serían los próximos 3-5 cambios si el equipo siguiera una iteración adicional. Estos cambios deben ser impulsados por los comentarios de los usuarios que el equipo no tuvo tiempo de abordar.
- **Diapositiva 9:** Reflexiona sobre el proceso como un todo. ¿Qué salió bien? ¿Qué no? ¿Qué aprendiste y qué harías de manera diferente?
- **Diapositiva 10:** Documenta los recursos utilizados durante este proyecto, incluidas las aplicaciones encontradas durante la investigación de mercado, el origen de las imágenes utilizadas y cualquier otro recurso o sitio web utilizado.

La [Presentación de la aplicación- Rúbrica](#) se centra en el contenido específico de estas diapositivas, pero alienta a los equipos a que realmente hagan suya esta presentación: esta es su oportunidad de “lanzar” su aplicación de manera efectiva.

Recorrer el salón: A medida que los equipos trabajen en el desarrollo de sus presentaciones, los motivará a escribir de una manera que sea accesible para una audiencia no técnica.

Compartir: Si es tiempo, juntará a los equipos para practicar sus presentaciones. recuerda a los equipos que su objetivo es mostrar su presentación no más de 10 minutos.

Ampliación del conocimiento 2 (10 min por grupo)

Presentaciones

Transición: Recordará a cada equipo cuántos minutos tienen para su presentación. Demuestra cómo mantendrás el tiempo.

Indicación: Llamará a cada equipo individualmente y permite presentar su trabajo. Cada equipo debe destinar unos minutos para permitir las preguntas de los otros estudiantes.

Es muy importante que mantenga la clase a tiempo con las presentaciones, no sólo para que todas las presentaciones puedan hacerse, sino también para que los estudiantes puedan ver el valor de la preparación de la presentación.

Transferencia del conocimiento (5 min)

Cuaderno: Presentará a los estudiantes las siguientes indicaciones para el cuaderno:

- ¿Qué presentación le pareció más interesante?
- ¿Qué presentación pensó que sería la más efectiva para resolver el problema que se propusieron resolver?
- Para las siguientes preguntas, escriba la frase “Todavía no”, “Casi” o “Entendido” dependiendo de dónde se encuentre con respecto a estos temas.
- Me siento cómodo investigando los deseos y las necesidades de los usuarios. (Todavía no, casi, lo tengo)
- Me siento cómodo con la fase de investigación de este proyecto. (Todavía no, casi, lo tengo)
- Me siento cómodo con el diseño y la fase de prototipo de baja fidelidad de un proyecto (incluidas las pruebas y la iteración) (Todavía no, casi, lo tengo)
- Me siento cómodo programando en el laboratorio de aplicaciones (Todavía no, casi, lo tengo)
- Me siento cómodo probando nuestra aplicación con los usuarios y haciendo cambios basados en esas pruebas. (Todavía no, casi, lo tengo)
- Me siento cómodo preparando una presentación técnica. (Todavía no, casi, lo tengo)
- Me siento cómodo dando una presentación técnica sobre mi trabajo. (Todavía no, casi, lo tengo)

Experiencias de aprendizaje de profundización

Use estos Contenidos para ampliar el aprendizaje de los estudiantes. Se pueden usar como Contenidos extras fuera del aula.

App Celebración

Invitará a los padres, a otros miembros de la escuela y a cualquier persona que haya sido entrevistada fuera de la clase a que vengan a una celebración. Los equipos pueden instalar cabinas donde puedan presentar sus aplicaciones y hablar sobre la experiencia.

Sugerencias de evaluación

Se sugiere los siguientes indicadores para evaluar formativamente los aprendizajes:

- Presentan un proceso completo del trabajo
- Describen con claridad el proceso realizado
- Proponen aspectos a mejorar y oportunidades futuras

Lección 74

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

¿Cómo resolver un problema de datos?

La ciudad está planeando un nuevo edificio para sus residentes, pero el ayuntamiento no está seguro de lo que debería construir. Algunos miembros quieren una estación de bomberos, otros quieren un refugio de animales y otros quieren una biblioteca. Su trabajo es usar datos para hacer una recomendación convincente al ayuntamiento.

Definir

¿Cuál es el problema que estás tratando de resolver?

Preparar

¿Qué datos necesitas para resolver tu problema?

Con su grupo, encuentre sus datos en Internet y grábelos aquí. Necesitará combinar sus datos con los datos de otros grupos para tomar la mejor decisión posible, por lo tanto, no haga una recomendación hasta que escuche de todos los otros grupos también.

Tratar

¿Qué debería construirse en la nueva ubicación?



Enumere dos razones para su elección y los datos que respaldan esa razón.

<p>Motivo 1</p>	<p>Datos para la Razón 1</p>
<p>Motivo 2</p>	<p>Datos para la Razón 1</p>

Reflexión

¿Qué otros datos podrían ayudarlo a tomar una mejor decisión?

¿Dónde podrían encontrar esta información?



Lección 75

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Datos en el mundo real

Web Pix es un sitio web donde los usuarios pueden ver televisión y películas a pedido. Tiene miles de películas y espectáculos y siempre agrega nuevos.

Web Pix recomienda un Video Top a cada uno de sus usuarios todos los días. Su trabajo es recopilar información sobre sus usuarios y sus videos para hacer la mejor recomendación que pueda para el video principal de cada usuario.

1. ¿Qué tipo de información debe recopilar sobre usuarios o videos para mejorar sus recomendaciones?
2. ¿Cómo podría obtener esta información?
3. Vea el video sobre Netflix y explique cómo solucionar este problema.



Routz es una compañía que ayuda a las personas a navegar por el tráfico. Tiene mapas de ubicaciones en todo el mundo y conoce información en tiempo real sobre las condiciones del tráfico. Cuando ingresa un destino al que desea ir, le brindará la mejor ruta que pueda encontrar.

Routz quiere mejorar sus datos para proporcionar direcciones aún mejores. Su trabajo es ayudar a la empresa a decidir qué datos podría recopilar para mejorar esas instrucciones.

1. ¿Qué datos debe recopilar Routz para mejorar sus recomendaciones?
2. ¿Dónde podría obtener esa información?
3. Mire el video sobre Waze y explique cómo solucionar este problema.

Nyle es una compañía que vende todo tipo de productos en línea. Los usuarios en el sitio pueden buscar y pedir productos que sean interesantes para ellos.

A los anunciantes les gustaría publicitar sus productos en Nyle para aumentar las ventas. Idealmente, solo mostrarían anuncios a las personas que probablemente comprarían sus productos.

1. ¿Qué tipo de información debe recopilar? ¿Qué es para sus clientes?
2. ¿Cómo podría recolectar esta información?
3. Vea el video sobre Amazon y explique cómo solucionar este problema



Reflexión

Hay muchas formas en que se pueden recopilar datos para resolver problemas. Complete tantos ejemplos de sus discusiones de hoy que pertenecen a cada una de las categorías a continuación.

<p>Usuario activamente provisto</p> <p>El usuario proporciona activamente datos y entiende que sus acciones están proporcionando datos a la empresa</p>	<p>Usuario pasivamente provisto</p> <p>Es posible que el usuario no sepa que su comportamiento en el sitio web genera datos para resolver un problema</p>	<p>Datos del sensor</p> <p>Los sensores recopilan automáticamente datos sobre el mundo. Por lo general, esto no mide nada sobre las personas</p>

A veces las personas no se dan cuenta de que están proporcionando datos cuando usan la tecnología.

¿Cuál es un ejemplo de datos de arriba que un usuario puede no querer compartir?

¿Por qué un usuario podría querer mantener esta información privada?

Lección 76

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Estructuración de datos



Problema de fiesta de pizza

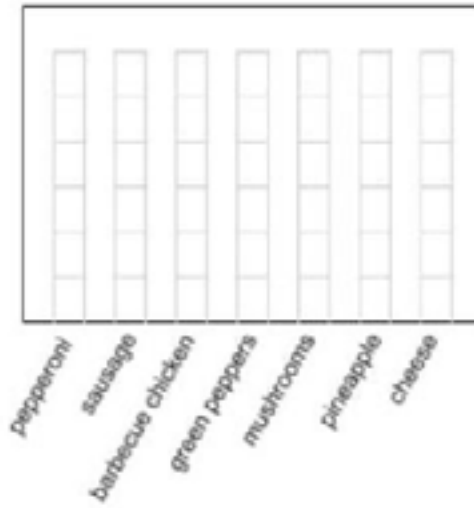
La pizzería local ha donado una pizza gigante de un ingrediente a una clase de octavo grado para su fiesta anual de pizza, y tienen que decidir qué tipo de cobertura pedirán para la pizza.

Sus opciones son: pepperoni, salchichas, pollo asado, pimientos verdes, champiñones, piña y queso extra.

Grafica los datos

Usa los datos para hacer un gráfico de barras que muestre cuántos votos recibió cada topping.

- | | |
|------------------------|-------------------------|
| Pepperoni | Piña |
| Queso | Pimientos |
| Pollo a la barbacoa | Pimientos verdes |
| Salchicha | Pepperoni |
| Hongos | Hongos |
| Voy a estar ausente :(| No me gusta la pizza |
| Pollo a la barbacoa | Pepperoni |
| Piña | Queso |
| Aceitunas | Pollo a la barbacoa |
| Todo está bien | !!!!Fiesta de pizza!!!! |
| Solo queso | Pepperoni |
| Queso | Las cuatro |
| Salchicha | Sin carne, por favor! |
| Pollo a la barbacoa | Pepperoni |



Más difícil de categorizar

A veces puede ser difícil averiguar qué hacer con algunos de sus datos. Elija dos votos que fueron difíciles de poner en la tabla. Para cada voto, explique qué lo hizo difícil y lo que finalmente hizo con la respuesta.

Votar	Fué difícil por que...	Yo decidí...
--------------	-------------------------------	---------------------

Lección 77

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Decisión 1 - cuándo publicar

Definir

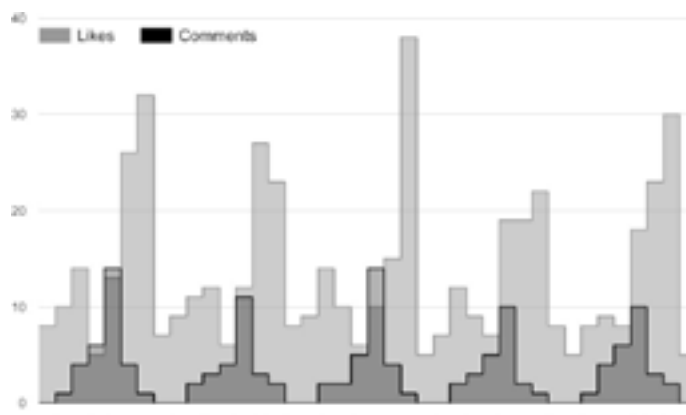
Desea saber cuándo publicar una imagen en línea para obtener la mejor respuesta.

Preparar

Para responder a esta pregunta, usted y sus amigos publicaron imágenes en sus aplicaciones favoritas durante una semana. Mantuvo un registro de me gusta y comentarios para ver si había patrones diferentes.

Tratar

Obtuviste información sobre cuándo hiciste una publicación y cuántos likes y comentarios recibiste. A continuación, se muestran los me gusta promedio y los comentarios para cada publicación.



Reflexión

Haz una recomendación sobre a qué hora publicar tomando en cuenta las siguientes afirmaciones:

- ¿Si quieres obtener más me gusta?
- ¿Si quieres obtener mayor cantidad de comentarios?
- ¿Qué otra información le gustaría recopilar para tomar una decisión aún mejor?

Decisión 2 - hacer comidas

Definir

A la cafetería le gustaría decidir la mejor manera de reducir el desperdicio de alimentos después de notar que están tirando la comida todos los días.

Preparar

Para responder a su pregunta, la cafetería decidió recopilar información sobre la cantidad de comidas preparadas, la asistencia diaria, el tipo de comida y la cantidad de comidas consumidas y no desperdiciadas.

Tratar

La cafetería recopiló estos datos y creó el cuadro a continuación.

	Lunes	Martes	Miércoles	Jueves	Viernes
Comida	Pizza	Hamburguesa	Dedos de pollo	ensalada	Queso asado
Comidas preparadas	100	100	100	100	100
Asistencia	85	90	90	95	85
Comidas comidas	80	82	88	75	80
Comidas perdidas	20	18	12	25	20

Reflexión

Ahora es tu turno de ayudar. ¿Puede hacer una recomendación sobre cómo la cafetería podría reducir el desperdicio de alimentos?

- ¿Cuántas comidas deben preparar cada día?
- ¿Deberían variar la cantidad de comidas que hacen en función de la comida servida o la asistencia?
- ¿Hay más información que desee recopilar para tomar su decisión?

Lección 78

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Interpretación de datos

Encontrar relaciones

Para encontrar relaciones entre las preferencias, harás una tabulación cruzada de los datos. Eso significa que no solo cuenta cuántas personas les gustó cada cosa (por ejemplo, “perro”), sino a cuántas personas les gustaron dos cosas juntas (“perros” y “acampar”). Esto te ayudará a ver cómo las diferentes respuestas se relacionan entre sí.

Mascotas y actividades

	Perro	Gato	Pájaro	Tiburón
Acampar				
Arte				
Video juegos				
Música				

- Si a alguien le gustan los gatos, ¿qué Lección es probablemente su favorita?
- ¿Cuál es una relación más interesante entre la mascota y la Lección favoritas?

Mascotas y deportes

	Perro	Gato	Pájaro	Tiburón
Natación				
Fútbol				
Basquetbol				
Volleyball				

- ¿Cuáles son dos relaciones interesantes entre la mascota y el deporte favoritos?

Actividades y deportes

	Acompañar	Arte	Video Juegos	Música
Natación				
Fútbol				
Basketbol				
Volleyball				

- ¿Cuáles son dos relaciones interesantes entre la Lección y el deporte favoritos?

Reflexión

¿Cómo pueden las relaciones entre este tipo de datos ayudarte a abordar un problema del mundo real?

Lección 79

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Recurso - automatización de decisiones de datos

Hacer reglas a partir de datos

Alguien ha dado una encuesta y ha puesto los resultados en las siguientes tablas. Usa la información para decidir qué reglas quieres para tu algoritmo.

Vacaciones y comida

	Playa	Parque de atracciones	Ciudad	Parque nacional
Helado	5	2	1	-
Pizza	1	2	2	1
Ensalada	-	-	1	5
Sandwiches	2	-	-	3

Vacaciones y super poder

	Playa	Parque de atracciones	Ciudad	Parque nacional
Volar	3	1	-	4
Invisibilidad	-	-	3	3
Super elasticidad	-	1	-	2
Teletransportación	5	2	1	-

Vacaciones y animal

	Beach	Amusement Park	Big City	National Park
Lobo	2	1	-	3
Bellona	2	-	-	2
Ratón	-	3	2	-
Perro	4	-	2	4

Automatización de decisiones de datos

Elegir unas vacaciones

Vas a crear un algoritmo que ayudará a tus compañeros de clase a elegir un buen lugar de vacaciones en función de los datos que recopiles sobre sus preferencias. Puede sugerir la playa, un parque de diversiones, un parque nacional o una gran ciudad. Después de crear su algoritmo, lo probará en tres de sus compañeros de clase.

Su algoritmo

Su algoritmo utilizará las respuestas que da un usuario para asignar puntos a diferentes tipos de vacaciones. Para cada respuesta posible a la pregunta, decida cuántos puntos obtendrá cada ubicación. La ubicación con más puntos después de hacer las tres preguntas será donde su usuario debería irse de vacaciones.

	Respuesta	Instrucciones
¿Cuál es tu comida favorita?		
	Helado	Agrega 2 puntos a la playa. Agrega 1 punto al parque de atracciones.
	Pizza	
	Ensalada	
	Sandwiches	
¿Cuál es el mejor superpoder?		
	Volar	
	Invisibilidad	
	Super elasticidad	
	Teletransportación	
¿Cuál es tu animal favorito?		
	Lobo	
	Ballena	
	Ratón	
	Perro	

El primero está hecho para ti, pero puedes cambiarlo si no estás de acuerdo.

Probando su algoritmo

Ahora, pruebe su algoritmo preguntando a dos de sus compañeros de clase que respondan las preguntas y que sumen los puntos para encontrar el lugar de vacaciones sugerido.

Compañero de clases 1

Respuesta 1: _____ Respuesta 2: _____ Respuesta 3:

Playa Puntos	Parque de atracciones Puntos	Parque nacional Puntos	Ciudad Puntos

- ¿Cuál es el lugar recomendado?

Compañero de clases 2

Respuesta 1: _____ Respuesta 2: _____ Respuesta 3:

Playa Puntos	Parque de atracciones Puntos	Parque nacional Puntos	Ciudad Puntos

¿Cuál es el lugar recomendado?

Reflexión

- ¿Estuviste de acuerdo con las sugerencias que hizo tu algoritmo?
- ¿Por qué no?
- ¿Cómo podría mejorar este algoritmo para que pueda hacer mejores recomendaciones?
- ¿Cuál es la desventaja de que una computadora tome una decisión automáticamente?
- ¿Cuál es un tipo de decisión que no quiere que una computadora haga automáticamente?
- ¿Por qué?

Lección 80 - 81

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Resuelva un problema de datos

Resumen

En esta unidad, ha visto cómo se pueden usar los datos para resolver todo tipo de problemas. Ahora es tu turno de usar datos para ayudar a alguien. En este proyecto, usarás los datos para hacer una recomendación a un compañero de clase.

Vas a...

- Trabajar con un socio
- Definir un problema que podría abordarse mediante una recomendación
- Identifique los datos que necesita y cree una encuesta para recopilarlos
- Interprete los datos para encontrar relaciones entre las respuestas de la encuesta
- Crea un algoritmo para hacer una predicción basada en datos
- Pon a prueba tu algoritmo
- Presente su trabajo a sus compañeros de clase

Usted enviará ...

- Guía completa del proyecto (este documento)
- Revisión por pares completada
- Cualquier material utilizado para presentar su trabajo

Comenzando

Antes de comenzar su proyecto, mire la aplicación de recomendación de ejemplo en línea.

- ¿Qué opción ayuda la aplicación al usuario?
- ¿Qué datos usa para hacer esa recomendación?

- ¿Cómo crees que los creadores de la aplicación decidieron sobre el algoritmo que utilizaron para hacer la elección?

Pasos del proyecto

Paso 1- Defina su problema

En la aplicación de muestra, el recomendador ayudó al usuario a decidir dónde ir de vacaciones. Con su pareja, decida qué tipo de problema resolverá su recomendador.

- ¿Qué pregunta responderá su recomendador para el usuario?

¿De qué posibles recomendaciones debería elegir?

- 1.
- 2.
- 3.
- 4.

Paso 2: decide qué datos necesitas

En la aplicación de muestra, el recomendador utilizó datos sobre la alimentación de un usuario, superpotencia y preferencias de los animales para decidir qué recomendar.

¿Qué datos pueden ayudarte a hacer una recomendación?

<p>Tipo de datos Describe el tipo de información que deseas recolectar</p>	<p>Posibles preguntas y respuestas Describe las preguntas y las opciones de respuesta que podrías incluir en una encuesta</p>

Paso 3: crea tu encuesta

Mire los tipos de datos que ha decidido recopilar. Úselos para crear preguntas de encuesta

- Pregunta 1
- Opciones de respuesta

1.	3.
2.	4.

- Pregunta 2
- Opciones de respuesta

1.	3.
2.	4.

- Pregunta 3
- Opciones de respuesta

Para averiguar cómo se relacionan estas respuestas con la recomendación que desea hacer, también deberá preguntarle a todos su preferencia por lo que desea recomendar.

1.	3.
2.	4.

- Pregunta de preferencia
- Opciones de respuesta

Paso 4: recopila tus datos de encuesta

Ahora que tiene una encuesta, está listo para recopilar sus datos. Entregue su encuesta a al menos a veinte personas diferentes y registre sus respuestas aquí. Utilizará estos datos para descubrir cómo las respuestas a las primeras tres preguntas pueden predecir las preferencias de una persona.

#	Respuesta 1	Respuesta 2	Respuesta 3	Preferencia
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				

Paso 5- Interpreta tus datos

Use la tabulación cruzada para descubrir cómo las respuestas a cada una de las tres preguntas se relacionan con la preferencia que desea recomendar. Rotule cada tabla con la pregunta que hizo y luego complete con las respuestas con las respuestas de la encuesta. Una vez que haya puesto sus datos en la tabla, intente encontrar algunas relaciones que le ayuden a hacer su algoritmo de recomendación.

Respuesta 1

- ¿Qué relaciones podrían ayudarte a hacer una regla?

Respuesta 2

- ¿Qué relaciones podrían ayudarte a hacer una regla?

Respuesta 3

- ¿Qué relaciones podrían ayudarte a hacer una regla?

Paso 6: defina su algoritmo

Ahora que ha encontrado algunas relaciones interesantes entre las respuestas a las preguntas y las preferencias del usuario, puede usarlas para hacer su algoritmo. Para cada pregunta, enumere las posibles opciones de respuesta y explique las instrucciones para agregar puntos a las opciones de recomendación pertinentes.

	Respuestas	Instrucciones
Pregunta 1		
Pregunta 2		
Pregunta 3		

Paso 7: prueba tu algoritmo

Pon a prueba tu algoritmo tratando de hacer una recomendación para tres compañeros de clase que no tomaron tu encuesta original. Para cada compañero de clase, enumere las cuatro recomendaciones posibles en la primera fila de la tabla, luego use la segunda fila para contar los puntos que cada recomendación obtiene de las respuestas de la encuesta de ese compañero de clase.

Compañero de clase 1

Respuesta 1: _____ Respuesta 2: _____ Respuesta 3: _____

Recomendación				
Puntos				

¿Cuál es la recomendación

Compañero de clase 2

Respuesta 1: _____ Respuesta 2: _____ Respuesta 3:

Recomendación				
Puntos				

- ¿Cuál es la recomendación?
- ¿Sus usuarios estuvieron de acuerdo con las recomendaciones que hizo? Explique.
- ¿Hay algún cambio que creas que deberías hacer a tu algoritmo?

Paso 8: Revisión por pares

Su maestro le proporcionará una hoja de Revisión por pares. Intercambie proyectos con otro grupo y complete la revisión por pares. Como parte de este proceso, debe desarrollar nuevas ideas sobre cómo puede mejorar su recomendación.

Paso 9: finalizar y presentar

En función de los resultados de los comentarios de sus pares, haga las adiciones o cambios que necesite realizar a la forma en que definió su problema, los datos que desea recopilar o la forma en que se analizarán. Luego, prepare una presentación de su solución para compartirla con sus compañeros. Debe incluir:

¿Qué elección estás ayudando al usuario a hacer?

Los tipos de datos que recopila para ayudar al usuario a tomar esa decisión Las relaciones que encuentre al interpretar los datos de tu encuesta

La forma en que usaste esta información para crear tu algoritmo de recomendación Los resultados de probar el algoritmo en usuarios

Debería poder encontrar toda esta información dentro de su guía de proyecto.

Revisión por pares: hacer una recomendación

Previo a la revisión

Nombre del creador:

Una cosa sobre la que quiero recibir comentarios es ...

Sección del revisor

Nombre del revisor:

Preguntas	Clasificación	Comentarios
El problema está bien definido. Puede decir fácilmente cuál es el tema, qué problema específico está tratando de resolver y qué tipo de recomendación / predicción se usará para resolverlo.	✓ ✗	
Los datos recopilados ayudarán a resolver el problema. Todos los datos necesarios deben hacer que la recomendación o predicción se incluya en la encuesta.	✓ ✗	
Los datos se han utilizado para ayudar a resolver el problema. Se han identificado relaciones importantes en los datos de la encuesta.	✓ ✗	
La forma en que se usaron los datos para crear el algoritmo fue razonable. El algoritmo de toma de decisiones se conecta claramente a las relaciones encontradas en los datos.	✓ ✗	

Retroalimentación de respuesta gratuita

- Me gusta...
- Deseo...
- Y si...

Reflexión del Creador

1. ¿Qué comentario te fue más útil? ¿Por qué?

2. ¿Qué comentario te sorprendió más? ¿Por qué?
3. En función de los comentarios, ¿qué cambios harías en tu propuesta de aplicación?

Reflexión de las prácticas

Práctica	Cosas para celebrar	Cosas para trabajar
Resolución de problemas		
Persistencia		
Creatividad		
Colaboración		
Comunicación		

Lección 82

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Determina el usuario





Junto a cada descripción de un usuario, escribe la letra de cada tetera que este pueda usar. Prepárate para justificar tus respuestas.

1. Alguien que quiere una taza individual de té antes de acostarse.
2. Alguien que a menudo deja caer cosas.
3. Alguien a quien le gustan los objetos muy ornamentados.
4. Alguien que necesita servir té en la fiesta.
5. Alguien a quien le gusta ver cómo se prepara su té.
6. Alguien que a menudo tiene prisa.
7. Alguien que tiene una cocina con muchos electrodomésticos negros.
8. Alguien a quien le gustan los objetos simples.
9. Alguien que prefiere el té helado.
10. Alguien que puede usar una estufa para calentar el agua.
11. Alguien que posee un titular de marihuana
12. Alguien que quiere servir té a un amigo que vino a conversar.
13. Alguien que tiene una cocina moderna.
14. Alguien a quien le gustan los objetos altos y delgados.
15. Alguien que colecciona antigüedades.
16. Alguien que tiene una cocina muy pequeña.

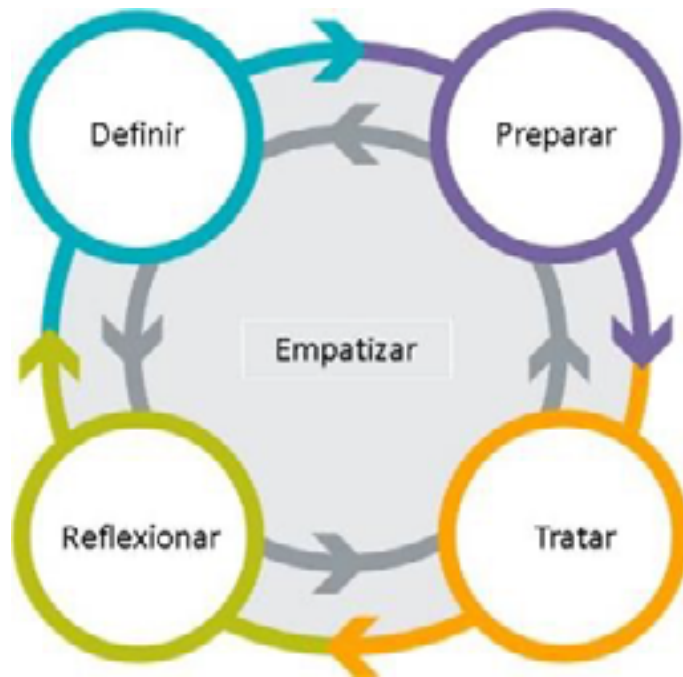


Diseño crítico

Para cada una de estas teteras, termina las afirmaciones de la derecha que critican el diseño del objeto:

	<p>Me Gusta _____</p> <p>_____</p> <p>Deseo _____</p> <p>_____</p> <p>Y si _____</p> <p>_____</p>
	<p>Me Gusta _____</p> <p>_____</p> <p>Deseo _____</p> <p>_____</p> <p>Y si _____</p> <p>_____</p>
	<p>Me Gusta _____</p> <p>_____</p> <p>Deseo _____</p> <p>_____</p> <p>Y si _____</p> <p>_____</p>
	<p>Me Gusta _____</p> <p>_____</p> <p>Deseo _____</p> <p>_____</p> <p>Y si _____</p> <p>_____</p>

Proceso de resolución de problemas con empatía



Lección 83

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha
















Reaccionando como tu usuario

Evaluación del producto

Tu profesor te presentará una serie de productos diferentes para que puedas reaccionar. Para cada producto que se muestra, analice y califique cada uno de los elementos que se le muestran, desde el punto de vista del usuario descrito en el perfil del usuario.

Alerta Falsa

Es fácil pensar en esto en términos de lo que te gusta o no, pero deberías responder en función de lo que a tu usuario le gustaría o no.

Producto	Análisis de funciones
<p>Nombre del producto: _____</p> <p>      </p> <p>Me gusta No esta mal No me gusta</p>	<p>A mi usuario le gusta / no le gusta</p> <p>Porque</p>
<p>Nombre del producto: _____</p> <p>      </p> <p>Me gusta No esta mal No me gusta</p>	<p>A mi usuario le gusta / no le gusta</p> <p>Porque</p>
<p>Nombre del producto: _____</p> <p>      </p> <p>Me gusta No esta mal No me gusta</p>	<p>A mi usuario le gusta / no le gusta</p> <p>Porque</p>

Rediseño del producto

Revisa las 8 sillas como su usuario asignado y responde las siguientes preguntas:

1. ¿Qué silla preferiría tu usuario?
2. Nombra 2 características de tu silla que le gusten a tu usuario. ¿Cómo sabes que a tu usuario le gustaría?
3. Teniendo en cuenta lo que has aprendido sobre tu usuario, diseña una mejor silla a continuación.

Discusión

1. Nombra dos características del nuevo diseño de tu silla y describe por qué atraen a tu usuario.
2. Nombra dos características de otras sillas que elijas no incluir y explica por qué.
3. Compara las sillas dentro de tu grupo. ¿Qué tienen en común? ¿Qué es diferente?

Perfil de usuario

Randi: Profesor de actividades al aire libre

Perfil de usuario

Randi es una mujer de 35 años que vive en Wolf Point, Montana. Ella es miembro de la Tribu Sioux y enseña Native American Studies en Fort Peck Community College. Ella sigue de cerca al equipo de básquetbol femenino de la universidad, los Buffalo Chasers, y usa los colores rojo, amarillo y negro del equipo cada vez que puede. Randi disfruta de los deportes de invierno como el esquí y la pesca en el hielo, y con frecuencia trae a su perro Scout con ella en sus aventuras al aire libre. Ella siempre ha querido visitar el océano en un lugar cálido como Florida, pero a ella generalmente no le gusta el clima caluroso y húmedo. Cuando no está al aire libre, le encanta acurrucarse en el sofá de su casa con un buen libro y una tajada de pastel de chocolate, aunque prefiere tomar té con él, ya que los productos lácteos le provocan molestias estomacales. No le gusta mirar televisión, zapatos de tacón alto y música de punk rock. Por las noches, ella a veces trabaja en línea comunicándose con sus estudiantes y otros profesores y haciendo investigación. Ella no tiene un espacio de oficina, así que lo hace principalmente desde su sofá. Le encantaría tener una mejor manera de separar su espacio de trabajo de su espacio personal cuando esté en casa.

Declaración de punto de vista

Randi necesita una forma de trabajar de manera más eficiente desde su casa porque no tiene una oficina / espacio de trabajo organizado.

Perfil de usuario

Gabriela: Madre Dedicada

Perfil de usuario

Gabriela es una mujer de 34 años que vive en Cicero, Illinois. Ella nació en Indiana y se mudó a Illinois después de conocer a su esposo Michael en el ejército. Gabriela ahora trabaja para una agencia de bienes raíces y pasa la mayor parte del tiempo cuidando a su hija que cursa el 3er grado en la escuela primaria local. Ella y Michael tienen mañanas tempranas para ir a la escuela y luego a trabajar, y los fines de semana están ocupados haciendo recados. Gabriela disfruta asistir a los juegos de fútbol de su hija, participar en grupos comunitarios en su vecindario y asistir a clases de baile. Mientras estaba en el ejército ella trabajaría más a menudo, pero en los últimos años ha tenido cada vez menos tiempo. También desea tener más tiempo para ver a su familia en Indiana y teme que su hija no vea lo suficiente a sus abuelos.

Declaración de punto de vista

Gabriela tiene una vida abarrotada en Illinois. Ella desea tener un poco más de tiempo para ella y quiere que su hija se relacione con su familia en Indiana.

Perfil de usuario

Tony: Ajustador de seguros

Perfil de usuario

Tony (nombre completo Antonio) es un ajustador de seguros de más de 20 años que vive fuera de Los Ángeles. Él trabaja para una gran compañía de seguros en el área, y tiene que conducir mucho por el trabajo, lo que significa estar atrapado en el tráfico escuchando la radio la mayor parte del día. Tony realmente desea tener más tiempo libre y poder darse el lujo de vivir más cerca de la ciudad. Preferiría no seguir trabajando en seguros y preferiría encontrar un trabajo en atención médica. Completó la escuela secundaria pero nunca asistió a la universidad y, aunque le gustaría volver a la escuela para cambiar su carrera, Tony tiene dificultades para ahorrar dinero.

Los fines de semana, Tony toca el bajo en una banda de reggae con sus compañeros de cuarto, pero rara vez llegan a tocar o practicar. Tony también adora ir al cine, y se enorgullece de conocer los nombres y la historia de todos los actores. Él descubrió que es muy difícil hacer amigos fuera de sus compañeros de habitación y desea tener alguna forma de compartir su conocimiento e interés en las películas con más personas para que pueda encontrar nuevos amigos con ideas afines.

Declaración de punto de vista

Tony quiere una forma de compartir su interés en las películas con una gama más amplia de personas que podrían convertirse en nuevos amigos.

Perfil de usuario

Amelia: cajera jubilada

Perfil de usuario

Amelia es una cajera de la tienda de abarrotes jubilada que, después de moverse por el país con su esposo David, se estableció en Atlanta. Le encanta cocinar, y particularmente disfruta probar nuevas recetas de diferentes regiones del mundo. Amelia y su esposo han descubierto que realmente disfrutan de la excelente comida y museos en Atlanta, pero los veranos han demostrado ser demasiado calurosos y húmedos para su gusto y han comenzado a ahorrar para viajar un poco.

Amelia tiene 4 hijos adultos y 7 nietos. Le resulta muy difícil vivir tan lejos de todos sus nietos, por lo que se esfuerza mucho por encontrar actividades divertidas y emocionantes para hacer con ellos cuando vienen de visita. Como es nueva en el área y tiene nietos de entre 4 y 15 años, Amelia a veces tiene dificultades para encontrar actividades que todo el grupo pueda disfrutar, pero que todavía son asequibles según su presupuesto.

Declaración de punto de vista

Amelia quiere encontrar actividades interesantes en su área que sean apropiadas para una gran variedad de edades, y le gustaría saber antes de salir de la casa cuánto costará la Lección.

Lección 84

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Diseño centrado en el usuario

Resumen

Diseñar un producto requiere que pienses en las necesidades de sus usuarios, incluso si esas necesidades son muy diferentes a las tuyas. En esta Lección diseñarás un producto desde cero para las necesidades específicas de un usuario.

El producto que diseñarás es una prenda inteligente. Para esta Lección considera la ropa inteligente como cualquier prenda que también integra la tecnología informática. Puede tener sensores que graban datos, cambian colores o muestran información, o incluso se conectan a Internet. Empuja los límites de lo que es posible. Siempre que tu diseño cumpla con las necesidades de tu usuario, es apropiado para la Lección de hoy.

Definición

1. Lluvia de ideas de usuarios

En un espacio, haz una lluvia de ideas sobre tantos tipos de personas como sea posible que puedan beneficiarse de la ropa inteligente.

2. Categorizar usuarios

Crea trozos de papel para cada uno de tus usuarios. Como grupo o como clase, crea categorías de usuarios agrupando tipos similares de usuarios.

3. Elige un usuario específico

4. Elige una categoría específica de usuario para diseñar

5. Lluvia de ideas necesidades

En un espacio, piensa en las necesidades, intereses y preocupaciones de su usuario.

6. Clasificar necesidades

Crea trozos de papel para cada una de las necesidades, intereses e inquietudes de tus usuarios. Como grupo o como clase, crea categorías de necesidades agrupando necesidades similares.

7. Elige una necesidad específica

Escribe la necesidad específica que seleccionó tu grupo:

Preparar

8. Lluvia de ideas soluciones

En un espacio, haz una lluvia de ideas sobre cómo se podría usar ropa inteligente para ayudar a abordar la necesidad que tu grupo eligió.

9. Discutir pros y contras

Con tu grupo discute los pros y los contras de las ideas que intercambiaste. En cada caso, toma en cuenta la necesidad específica de tu usuario.

Tratar

10. Describe tu producto

En el espacio a continuación, escribe una breve descripción de tu producto y cómo satisface las necesidades de tu usuario.

11. Dibuja tu proyecto

En un espacio dibuja una imagen de la prenda inteligente que deseas crear. Etiqueta los diferentes componentes que agregarás a la ropa.

Cierre

12. Presenta tu proyecto y realiza una reflexión con la clase

Prepárate para compartir tu idea de usuario, necesidad e indumentaria inteligente con tus compañeros de clase

Lección 85

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

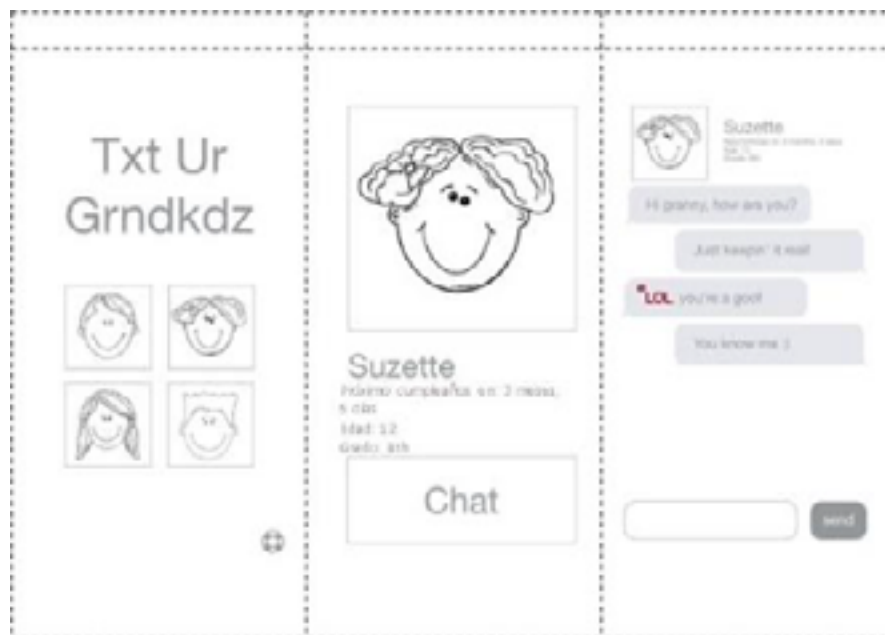
Periodo

Fecha

Pantallas de interfaz de usuario

Interfaz de usuario (UI)

Estas imágenes representan los elementos visuales de esta aplicación. Recorta cada tarjeta de pantalla.



Prueba de usuario (computadora)

Diagrama de navegación

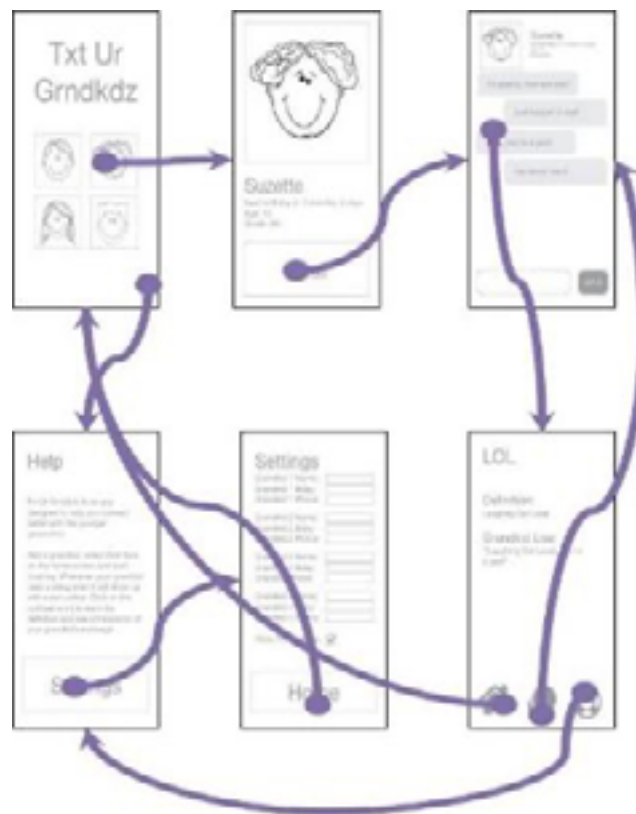
El diagrama de navegación a la derecha describe cómo cada pantalla de este prototipo está conectada a otras pantallas. Cada flecha conecta un elemento (como un botón, imagen o icono) a otra pantalla de la aplicación. Combinado con las pantallas de la interfaz de usuario, este diagrama le permite “usar” el prototipo de papel como si fuera una aplicación real.

Prueba de prototipo

Vas a probar este prototipo con un compañero. Su función es ser la computadora, tomando la información del usuario y respondiendo en consecuencia.

Reglas para la computadora

- Coloca la primera pantalla en frente del usuario
- Coloca el resto de las pantallas a un lado
- Cuando el usuario hace clic en una carta, consulta el diagrama:
 - Si el elemento cliqueado tiene una línea, cambia a la pantalla a la que la línea se vincula
 - Si el elemento cliqueado no tiene línea, no hagas nada

**Prueba de usuario (usuario)**Casos de prueba

Un caso de prueba es una interacción específica del usuario que el desarrollador de una aplicación desea probar. Elegir buenos casos de prueba puede garantizar que cuando pruebas una aplicación con un usuario, accedas a todas las funciones importantes. En la tabla a continuación, cada fila es un caso de prueba compuesto por una tarea específica que está tratando de realizar, un espacio para que documentes lo que intentaste y un espacio para registrar tu reacción.

Pruebas de prototipos

Vas a probar este prototipo con un compañero. Su función es ser el Usuario, interactuando con las

pantallas de la Interfaz de usuario como si fuera una aplicación real. Tu compañero (la Computadora) se encargará de asegurarse de que el prototipo responda tal como fue diseñado. Usa la tabla a continuación para probar diferentes casos de prueba y registrar los resultados. Las últimas filas se dejan en blanco para que pueda explorar otras áreas de la aplicación que te interesan. La primera fila se completa como un ejemplo.

Tarea	Lo que intente	Mi reacción
Ver la información de Suzette	Hice clic en cada una de las caras hasta que encontré una que cambió a la pantalla de Suzette.	No estaba claro quién era quién en la pantalla de inicio. Sería bueno si mostrara el nombre de cada niño debajo de su foto
Obtenga ayuda sobre cómo usar la aplicación		
Descubre cuándo es el cumpleaños de Suzette		
Enviar un mensaje a Suzette		
Obtener la definición de un acrónimo desconocido		

Tarea	Lo que intente	Mi reacción
Cambiar la edad de Suzette		
Cambiar la imagen de Suzette		
Permitir notificaciones		

Lección 86

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Mejora una pantalla

<p>1. Me está resultando difícil descubrir cómo agregar a mis nietos a esta aplicación. Parece que será muy útil, pero he estado haciendo clic y parece que no puedo encontrarlo.</p>	<p>11. Me pregunto si esta aplicación debe ser solo para personas que quieren hablar con sus nietos. Soy voluntario en mi escuela primaria local y me encantaría usar esta aplicación para estar en contacto con mis alumnos. ¿Hay alguna forma de que se conecte menos en los nietos?</p>
<p>2. ¿Hay alguna forma de ejercer esta aplicación?</p>	<p>12. ¿Por qué tengo que pasar por la pantalla de ayuda para ir a la pantalla de configuración? Me tomó mucho tiempo resolver esto y no creo que tenga mucho sentido.</p>
<p>3. ¿Cómo agrego nietos a esta aplicación?</p>	<p>13. Me acabo de dar cuenta de que necesito llegar a la definición de una palabra para volver a la pantalla de inicio. Siento que siempre deberían poder acceder a la pantalla de inicio.</p>
<p>4. Acabo de descargar Text U-Grnditz. Sé que muchas aplicaciones tienen nombres que deletrean palabras incorrectamente, pero siento que es difícil encontrar esta aplicación en línea porque está mal escrita.</p>	<p>14. ¿Cómo puedo agregar una foto de mis nietos? Encontré una manera de agregar su información, pero no hay forma de cambiarla de la imagen predefinida.</p>
<p>5. ¿Hay alguna manera de hacerán para que cuando haga clic en un grandid abra el chat de inmediato? Me gustan los recordatorios de cumpleaños, pero por lo general solo uso esta aplicación para hablar con mis nietos y no necesito toda esa información adicional.</p>	<p>15. ¿Hay alguna manera de cambiar la foto de mis nietos?</p>
<p>6. Me está resultando difícil que el texto sea tan pequeño en esta aplicación. ¿Hay alguna manera de hacerlo más grande?</p>	<p>16. La fuente en muchas de estas páginas es realmente pequeña y difícil de leer. ¿Podemos hacerlo más grande?</p>
<p>7. ¿Qué es un Grnditz? No entiendo por qué la aplicación se llama así.</p>	<p>17. Siento que a menudo quiero volver a la página de inicio y no hay forma de hacerlo. Se siente como que a veces se puede y a veces no se puede y no hay un patrón.</p>
<p>8. Quiero cambiar la información de mis nietos, pero parece que no puedo encontrar ningún lugar para hacerlo.</p>	<p>18. No hay forma de agregar la calificación de mi nieto. ¿Simplemente se calcula usando su edad?</p>
<p>9. Cada vez que voy a la página de chat de un nieto, parece que no puedo regresar a la pantalla de inicio. Ahora mismo tengo que cerrar toda la aplicación.</p>	<p>19. Esto dice que mi hijo está en séptimo grado, pero se saltó un grado y está en octavo. ¿Cómo se muestra este grado?</p>
<p>10. Me gustaría una forma de hablar con más de 4 nietos. Tengo 11 y esperaba que esta aplicación fuera una buena forma de mantenerse en contacto.</p>	<p>20. ¿Hay un lugar para ingresar a la calificación de un estudiante?</p>

Mejoras potenciales

Usando los comentarios que categorizaste en la última Lección, ¿qué mejoras podría hacer tu grupo en la aplicación? Para cada mejora, identifique a qué categoría de comentarios se dirige.

Mejora de propuesta	Comentarios dirigidos

Rediseño de pantalla

Cada miembro del grupo seleccionará una de las pantallas y una de las mejoras. Usando la página para la pantalla elegida, esboce cómo se verá la interfaz de usuario mejorada. Describa los cambios en la experiencia del usuario escribiendo descripciones o dibujando flechas a los cambios que ha realizado en el espacio a continuación. ¿Cómo funcionarán los botones, entradas, etc. de forma diferente en esta pantalla actualizada?

Pantallas de interfaz de usuario

Interfaz de usuario (UI):

Estas imágenes representan los elementos visuales de esta aplicación. Recorta cada tarjeta de pantalla.



Lección 87

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Prototipo de papel

Resumen

Para este proyecto, crearás un prototipo en papel de una aplicación para ayudar a un compañero a aprender algo nuevo. Ya deberías haber entrevistado a tu compañero de clase e identificado una barrera para su aprendizaje. Tendrás que crear una aplicación que los ayude a superar esta barrera. Siéntete libre de probar nuevos enfoques creativos para el problema que has identificado, pero sea lo que sea que decidas diseñar, asegúrate de empatizar con las necesidades de tu usuario.

Enviarás:

- Esta guía de proyecto.
- Las pantallas de tu aplicación dibujadas en tarjetas de notas. Como máximo, puedes tener 6 pantallas.
- Un diagrama de navegación de su aplicación, que muestra cómo tu usuario se mueve entre las pantallas.
- Un póster que muestra tu prototipo y describe cómo funciona.

Definir

1. Ya deberías haber entrevistado a tu compañero de clase para identificar lo que quiere aprender y una barrera para aprender esta habilidad. Registra los resultados de esa entrevista a continuación.
 - ¿Qué quiere aprender tu compañero de clase?
 - ¿Qué barrera le impide aprenderlo?
 - ¿Qué tipo de aplicación ayudará a abordar las necesidades de tu usuario?

Preparar

2. Como clase, deberías haber analizado los tipos de aplicaciones que pueden satisfacer las necesidades de tu usuario. Ahora es el momento de empezar a pensar qué hará específicamente esta aplicación. En el espacio a continuación, enumera las funciones mínimas que tu aplicación necesitará para satisfacer las necesidades de tu usuario.
3. Dibuja tus pantallas: En una hoja de papel en blanco dibuja el contorno de tus pantallas y cómo

un usuario navegará entre ellas. Para mantener tu aplicación simple, debes limitarte a 6 pantallas como máximo. Sin embargo, en muchos casos, tu aplicación puede no necesitar tantas.

Algunas cosas para tener en mente:

- Centrarse en la estructura. ¿Cuál es el propósito de cada pantalla? Qué características necesitan para incluir.
- Comienza con las características principales. Solo agrega “extras” una vez que estés seguro de que son necesarios.
- Decide qué características de diseño consistentes usarás (por ejemplo, un menú que siempre esté en la parte superior).
- No te preocupes por dibujar las cosas a la perfección. Este es un boceto para ayudarte a diseñar tu borrador final.

Intentar

4. Crea tus pantallas: Se te entregarán tarjetas de notas para dibujar la versión final de tus pantallas. Con lápices, marcadores, lápices de colores o cualquier otra cosa que desees, crea la interfaz de usuario de tu aplicación.
5. Crea tu diagrama de navegación: Encontrarás un diagrama de navegación que puedes usar para dibujar cómo se moverá su usuario dentro de su aplicación. Cualquier información sobre cómo el usuario interactúa con la aplicación debe incluirse en este diagrama.

Reflexión

6. Prueba de usuario: Probarás tu aplicación con tu usuario. Para organizar este proceso, completa la Guía de prueba del usuario a continuación.
7. Interpretar los comentarios del usuario: Usando tus notas de las pruebas de los usuarios, enumera todas tus observaciones de cosas interesantes que el usuario dijo o hizo, e intenta identificar lo que eso podría significar acerca de cómo funciona tu aplicación, o debería funcionar.

El usuario dijo/hizo	Que significa
Seguí haciendo clic en la cara de Suzette para tratar de cambiar su configuración.	Las imágenes de Grandkid deben enlazar a la página de configuración.

8. Crear un cartel: Combina las diferentes piezas de trabajo que completaste para este proyecto para crear un póster único. Además de tu bosquejo de lluvia de ideas, pantallas de usuario final y diagrama de navegación, asegúrate de incluir la siguiente información.

- ¿Qué está tratando de aprender tu usuario?
- ¿Qué es una barrera para su aprendizaje?
- ¿Cuál es tu aplicación? ¿Cómo funciona?
- ¿Cómo responde su aplicación a las necesidades de sus usuarios?
- Mejoras que harías según tus pruebas

9. Diagrama de navegación: Dibuja bocetos simples de tus pantallas en el espacio a continuación. Luego, dibuje flechas que muestren cómo un usuario navegaría entre sus pantallas. Cualquier información sobre cómo el usuario interactúa con la aplicación debe incluirse en esta página.

Guía de prueba del usuario

Casos de prueba

Crear tareas de usuario: Completa la columna izquierda de la tabla con diferentes tareas comunes que su usuario querrá hacer con su aplicación.

Pon a prueba tu prototipo

Dale esta guía de prueba a tu usuario. Prueba la aplicación comenzando en la pantalla de inicio y permitiéndoles navegar a través de ella haciendo clic. Intenta completar cada tarea enumerada en la tabla.

Aquí hay algunas pautas:

- Como la “computadora” solo utiliza piezas de interacción incluidas en el diagrama de navegación para moverse por la aplicación.
- No expliques cómo funciona la aplicación para tu usuario. Observa cómo lo usarían sin guía.

- El usuario puede y debe pensar en voz alta. Esta es una forma de ayudarte a comprender su experiencia.
- Mantén una hoja de papel para marcar cualquier otra cosa que notes durante la prueba.

Resultados de registro

En las columnas “Lo que intenté” y “Mi reacción”, incluye lo que hizo tu usuario para completar cada tarea. En la siguiente columna, incluye sus reacciones sobre cuán fácil o accesible era la aplicación. Objetivo de la aplicación.

Entrevista del usuario

Preguntas iniciales

Usando las preguntas y el espacio a continuación, entrevista a tu compañero para aprender más acerca de sus necesidades y obstáculos con respecto a aprender algo nuevo. Si bien sólo se proporcionan dos preguntas, debes hacer preguntas de seguimiento basadas en las respuestas de tu entrevistado para obtener más información que te ayude en las fases posteriores. Asegúrate de anotar todo lo que dice o hace tu entrevistado en respuesta a cada pregunta.

- Entrevistado:
- Pregunta 1: ¿Qué cosa nueva quisiste aprender y por qué?
- Pregunta 2: ¿Qué puede (o podría) obstaculizar tu aprendizaje?

Desarrollo de preguntas de seguimiento

Lee tus notas de las dos primeras preguntas de la entrevista y resalta cualquier afirmación o respuesta sobre la que desees obtener más información. Tu objetivo en esta entrevista es aprender tanto como sea posible sobre lo que tu usuario desea aprender y las barreras que podrían encontrar en su camino. Después de revisar las respuestas a las dos primeras preguntas, presenta tres preguntas de seguimiento que podrían ayudarte a aprender más o a aclarar las ambigüedades.

- Pregunta de seguimiento:
- Pregunta de seguimiento 2:
- Pregunta de seguimiento 3:

Identificación de barreras

Usando las notas que has tomado de todas las preguntas de la entrevista, elabora una lista a continuación de todas las barreras para que tu usuario aprenda lo que quería aprender. Intenta encontrar las dos barreras que declaró explícitamente tu usuario, así como algunas que encuentre al leer entre líneas.

- Barreras:

Lección 88

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Rúbrica - prototipo de papel

Rúbrica específica del proyecto

Criterio	Si / No	Comentarios
Todas las secciones de la guía del proyecto están completas		
El prototipo de papel incluye pantallas múltiples con elementos de pantalla claramente definidos		
El prototipo de papel se completa ordenadamente con atención a los detalles visuales		
El diagrama de navegación indica claramente cómo un usuario navegaría la aplicación		
El póster describe lo que el usuario quiere aprender y la barrera que la aplicación debe superar		
El póster explica cómo funciona la aplicación y cómo está diseñada para satisfacer las necesidades del usuario		
El póster describe los resultados de las pruebas de usuario		
El póster describe las mejoras propuestas basadas en los resultados de la prueba del usuario.		

Práctica de reflexión

Práctica	Cosas para celebrar	Cosas para trabajar
Resolución de problemas		
Persistencia		
Creatividad		
Colaboración		
Comunicación		

Prototipo en papel

Resumen

Para este proyecto, crearás un prototipo en papel de una aplicación para ayudar a un compañero a aprender algo nuevo. Ya deberías haber entrevistado a tu compañero de clase e identificado una barrera para su aprendizaje. Tendrás que crear una aplicación que los ayude a superar esta barrera. Siéntete libre de probar nuevos enfoques creativos para el problema que has identificado, pero sea lo que sea que decidas diseñar, asegúrate de empatizar con las necesidades de tu usuario.

Enviarás:

- Esta guía de proyecto
- Las pantallas de tu aplicación se dibujarán en notecards, como máximo, puedes tener 6 pantallas
- Un diagrama de navegación de su aplicación, que muestra cómo tu usuario se mueve entre las pantallas

- Un póster que muestra tu prototipo y describe cómo funciona

Definir

Ya deberías haber entrevistado a tu compañero de clase para identificar lo que quiere aprender y una barrera para aprender esta habilidad. Registra los resultados de esa entrevista a continuación.

- ¿Qué quiere aprender tu compañero de clase?
- ¿Qué barrera les impide aprenderlo?
- ¿Qué tipo de aplicación ayudará a abordar las necesidades de su usuario?

Preparar

Como clase, deberías haber analizado los tipos de aplicaciones que pueden satisfacer las necesidades de tus usuarios. Ahora es el momento de empezar a pensar qué hará específicamente esta aplicación.

En el espacio a continuación, enumera las funciones mínimas que tu aplicación necesitará para satisfacer las necesidades de tu usuario:

Dibuja tus pantallas

En una hoja de papel en blanco dibuja el contorno de tus pantallas y cómo un usuario navegará entre ellas. Para mantener tu aplicación simple, debes limitarte a 6 pantallas como máximo. Sin embargo, en muchos casos, tu aplicación puede no necesitar tantos. Algunas cosas para tener en mente:

- Centrarse en la estructura. ¿Cuál es el propósito de cada pantalla? ¿Qué características necesitan para incluir?
- Comience con las características principales. Solo agregue “extras” una vez que esté seguro de que son necesarios.
- Decida qué características de diseño consistentes usará (por ejemplo, un menú que siempre esté en la parte superior).
- No te preocupes por dibujar las cosas a la perfección. Este es un boceto para ayudarlo a diseñar su borrador final.

Intentar

Crea tus pantallas: Se te entregarán tarjetas de notas para dibujar la versión final de tus pantallas. Con lápices, marcadores, lápices de colores o cualquier otra cosa que desee, crea la interfaz de usuario de tu aplicación.

Crea tu diagrama de navegación: Encontrará un diagrama de navegación que puede usar para dibujar cómo se moverá su usuario dentro de su aplicación. Cualquier información sobre cómo el usuario interactúa con la aplicación debe incluirse en este diagrama.

Reflexión

Prueba de usuario: Probarás tu aplicación con tu usuario. Para organizar este proceso, complete la Guía de prueba del usuario a continuación.

Interpretar los comentarios del usuario: Usando sus notas de las pruebas de los usuarios, enumere todas sus observaciones de cosas interesantes que el usuario dijo o hizo, e intente identificar lo que eso podría significar acerca de cómo funciona su aplicación, o debería funcionar.

El usuario dijo/hizo	Qué significa
<i>Seguí haciendo clic en la casa de Suzette para tratar de cambiar su configuración.</i>	<i>Las imágenes de granídad deben enlazar a la página de configuración</i>

Crear un cartel: Combina las diferentes piezas de trabajo que completaste para este proyecto para crear un póster único. Además de tu bosquejo de lluvia de ideas, pantallas de usuario final y diagrama de navegación, asegúrate de incluir la siguiente información.

- ¿Qué está tratando de aprender tu usuario?
- ¿Qué es una barrera para su aprendizaje?
- ¿Cuál es tu aplicación? ¿Cómo funciona?
- ¿Cómo responde su aplicación a las necesidades de sus usuarios?
- Mejoras que harías según tus pruebas

Diagrama de navegación

Dibuja bocetos simples de tus pantallas en el espacio a continuación. Luego, dibuja flechas que muestren cómo un usuario navegaría entre sus pantallas.

Cualquier información sobre cómo el usuario interactúa con la aplicación debe incluirse en esta página.

Guía de prueba del usuario: Casos de prueba

Crear tareas de usuario: Completa la columna izquierda de la tabla con diferentes tareas comunes que tu usuario querrá hacer con tu aplicación.

Pon a prueba tu prototipo: Dale esta guía de prueba a tu usuario. Prueba la aplicación comenzando en la pantalla de inicio y permitiéndoles navegar a través de ella haciendo clic. Intenta completar cada tarea enumerada en la tabla. Aquí hay algunas pautas:

- Como la “computadora” solo utiliza piezas de interacción incluidas en el Diagrama de navegación para moverse por la aplicación.
- No expliques cómo funciona la aplicación para su usuario. Deseas observar cómo lo usarían sin guía.
- El usuario puede y debe pensar en voz alta. Esta es una forma de ayudarlo a comprender su experiencia.
- Mantén una hoja de papel para marcar cualquier otra cosa que notes en la prueba.

Resultados de registro: En las columnas “Lo que intenté” y “Mi reacción”, incluye lo que hizo tu usuario para completar cada tarea. En la siguiente columna, incluya sus reacciones sobre cuán fácil

o accesible era la aplicación

Tarea	Lo que intenté	Mi reacción

Lección 90

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Lista de recursos de la aplicación

Esta es una lista de recursos de las aplicaciones a las que se hace referencia en esta Lección:

1. El ganador del Verizon App Challenge creó “Hello Navi”. Ayuda a los estudiantes ciegos a desplazarse a la escuela y a otros lugares.
 - (2:57 min) Video de la presentación
2. Trisha Prabhu creó “ReThink”. Ayuda a prevenir el acoso cibernético al hacer que los usuarios piensen antes de enviar un texto malo.
 - ReThink sitio web
3. Technovation 2016 MS equipo ganador de California Coders: “Loc8Don8”. Encuentra sitios de donaciones y rastrea varios tipos de donaciones de caridad.
 - (3:51 min) Loc8Don8 pitch video
 - (1:27 min) Loc8Don8 demo video
 - (6:30 min) Loc8Don8 PP Presentation
4. La tripulación Tech de Winchester, MA creó “SpeechReach”. Ayuda a los usuarios a mejorar sus habilidades para hablar en público.
 - (3:23 min) SpeechReach pitch video
 - (1:53 min) SpeechReach demo video
5. El equipo ganador de Verizon App Challenge 2014 de Kennewick, WA creó “Safe and Sound”. Ayuda a los adolescentes a lidiar con la depresión.
 - Youtube video
 - Artículo publicado en Huffington

Inicio de diseño de aplicaciones

1. Contrato de equipo

Su primera decisión como equipo es acordar cómo trabajarán juntos. Seguirás las normas acordadas

sobre aula para la comunicación, la toma de decisiones y la participación. También hay espacio para acuerdos adicionales.

Lean este contrato, discútanlo en equipo y luego firmen en el cuadro siguiente para indicar que están de acuerdo.

2. Comunicación

Cuando trabajamos juntos en el aula, acordamos que nos trataremos unos a otros con respeto.

Eso significa:

- Nuestra comunicación escrita y verbal será considerada (sin cambiarnos los nombres)
- Nos turnaremos cuando compartamos ideas
- Todos escucharemos cortésmente cuando otros hablen

3. Decisión Making

Al tomar decisiones en equipo, primero hablaremos juntos como un grupo. Bien entonces:

- Tratar de llegar a un consenso donde todos estén de acuerdo con la misma decisión.
- Si eso no es posible después de varios intentos, entonces tomaremos un voto. La mayoría siempre gana.
- Si es un empate, le pediremos al maestro que decida.

Una vez que se toma una decisión, todos aceptamos trabajar en ella con nuestro mejor esfuerzo, incluso si no fue nuestra elección personal.

4. Participación

Estamos de acuerdo en que este proyecto solo tendrá éxito si todos participamos y trabajamos en él.

Nos comprometemos a:

- Hacer nuestro mejor esfuerzo en cada Lección.
- Pedir ayuda si la necesitamos.
- Entregar nuestro trabajo a tiempo.
- Verificar con otros miembros del equipo o el maestro que el proyecto pueda mantenerse en el buen camino

Nombre y apellido	Fecha de hoy
1.	
2.	
3.	
4.	
5.	

1. **Firmado**
2. **Lluvia de ideas**
3. **Nombre del equipo**

Elijan un nombre de equipo y escríbanlo a continuación:

4. **Tema**

¿En qué tema general se centrará su aplicación? No se preocupen por lo que específicamente hará la aplicación en este momento, simplemente consideren qué tipo de problema les interesa. Por ejemplo, puede considerar problemas en su comunidad escolar, fomentar el servicio cívico, promover las artes, mejorar el medioambiente, o mejorar la salud y el bienestar de los demás.

5. **Equipos de usuarios**

¿Para qué tipo de personas podrías desarrollar esta aplicación? Considera todos los diferentes equipos de usuarios afectados por su tema elegido.

6. **Otras notas:**

Lección 91

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Investigación de mercado

Alerta falsa

El objetivo de esta investigación es ayudarte a decidir el tema, el grupo de usuarios objetivos y el tipo de aplicación que desea crear. Aún no estás tomando decisiones sobre cómo se verán las pantallas individuales de tu aplicación.

Encontrar aplicaciones similares

Completa la información a continuación sobre las aplicaciones que encuentres durante tu investigación. Para cada aplicación que encuentres, deberás identificar quiénes son los usuarios objetivos, qué te gustó de la aplicación (y, por lo tanto, podrías usar como fuente de inspiración) y lo que no te gustó.

Nombre de la aplicación	¿Quién es el usuario objetivo? ¿Cómo se satisfacen sus necesidades?	Me gusta...	No me gustó...

Ideas de la aplicación de lluvia de ideas

Haz una lluvia de ideas con tu equipo con ideas potenciales para tu aplicación. Cuando hayas encontrado una idea en la que el equipo pueda ponerse de acuerdo, documéntela aquí. Concéntrense en el conjunto mínimo de características que les permitirán prototipar y probar su idea central.

Idea de la aplicación de equipo:

Aplicaciones utilizadas para inspiración:

Nombre de la APP y URL	Características

Lección 92

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Prototipo lluvia de ideas

Ejemplo de estructura:














Pantallas sugeridas:

Recordatorio: Debe haber al menos una pantalla por alumno en el equipo.

Tipo de aplicación	Ejemplo	Pantallas sugeridas
Toma de decisiones	¿Puedo reciclar esto?	<ul style="list-style-type: none"> • Inicio / Sobre nosotros • Preguntas y respuestas • Página de respuesta
Gestión de listas	Recordatorio de la tarea	<ul style="list-style-type: none"> • Inicio / Sobre nosotros • Lista de artículos • Añadir artículo • Remover el artículo • Ver todos los artículos que se han hecho
Pregunta	Ayudante español	<ul style="list-style-type: none"> • Inicio / Sobre nosotros • Preguntas / respuestas (secuenciales) • Resultados de seguimiento • Compare los resultados con otros
Crowdsourcing (colaboración abierta distribuida)	¿Qué contenedores de reciclaje están llenos?	<ul style="list-style-type: none"> • Lista de ubicaciones • Para cada ubicación, puede hacer algo: informar, verificar • Página de resumen personal • Página de resumen global hizo todo el mundo

Elementos de la interfaz de usuario:

La siguiente es una tabla de los elementos de la interfaz de usuario que estarán disponibles en App Lab. Para cada elemento, discute con un equipo que cree que es y cómo puede usarlo en su aplicación. Anótalo en la columna de Uso previsto.

Elementos de la interfaz de usuario	Uso previsto
 Button	
 Text Input	
 Label	
 Dropdown	
 Radio Button	
 Checkbox	
 Image	
 Canvas	
 Text Area	
 Chart	
 Slider	

Lección 93

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Prototipo lluvia de ideas

Ejemplo de estructura:














Pantallas sugeridas:

Recordatorio: Debe haber al menos una pantalla por alumno en el equipo.

Tipo de aplicación	Ejemplo	Pantallas sugeridas
Toma de decisiones	¿Puedo reciclar esto?	<ul style="list-style-type: none"> • Inicio / Sobre nosotros • Preguntas y respuestas • Página de respuesta
Gestión de listas	Recordatorio de la tarea	<ul style="list-style-type: none"> • Inicio / Sobre nosotros • Lista de artículos • Añadir artículo • Remover el artículo • Ver todos los artículos que se han hecho
Pregunta	Ayudante español	<ul style="list-style-type: none"> • Inicio / Sobre nosotros • Preguntas / respuestas (secuenciales) • Resultados de seguimiento • Compare los resultados con otros
Crowdsourcing (colaboración abierta distribuida)	¿Qué contenedores de reciclaje están llenos?	<ul style="list-style-type: none"> • Lista de ubicaciones • Para cada ubicación, puede hacer algo: informar, verificar • Página de resumen personal • Página de resumen global hizo todo el mundo

Elementos de la interfaz de usuario:

La siguiente es una tabla de los elementos de la interfaz de usuario que estarán disponibles en App Lab. Para cada elemento, discute con un equipo que cree que es y cómo puede usarlo en su aplicación. Anótalo en la columna de Uso previsto.

Elementos de la interfaz de usuario	Uso previsto
 <p>Button</p>	
 <p>Text Input</p>	
 <p>Label</p>	
 <p>Dropdown</p>	
 <p>Radio Button</p>	
 <p>Checkbox</p>	
 <p>Image</p>	
 <p>Canvas</p>	
 <p>Text Area</p>	
 <p>Chart</p>	
 <p>Slider</p>	

Diseño de pantalla

Elige un nombre de pantalla

Cada pantalla debe tener un nombre único (dentro de la aplicación) y descriptivo. El nombre de pantalla solo puede contener letras y números, sin espacios, signos de puntuación u otros caracteres especiales.

Configurando el espacio de nombres

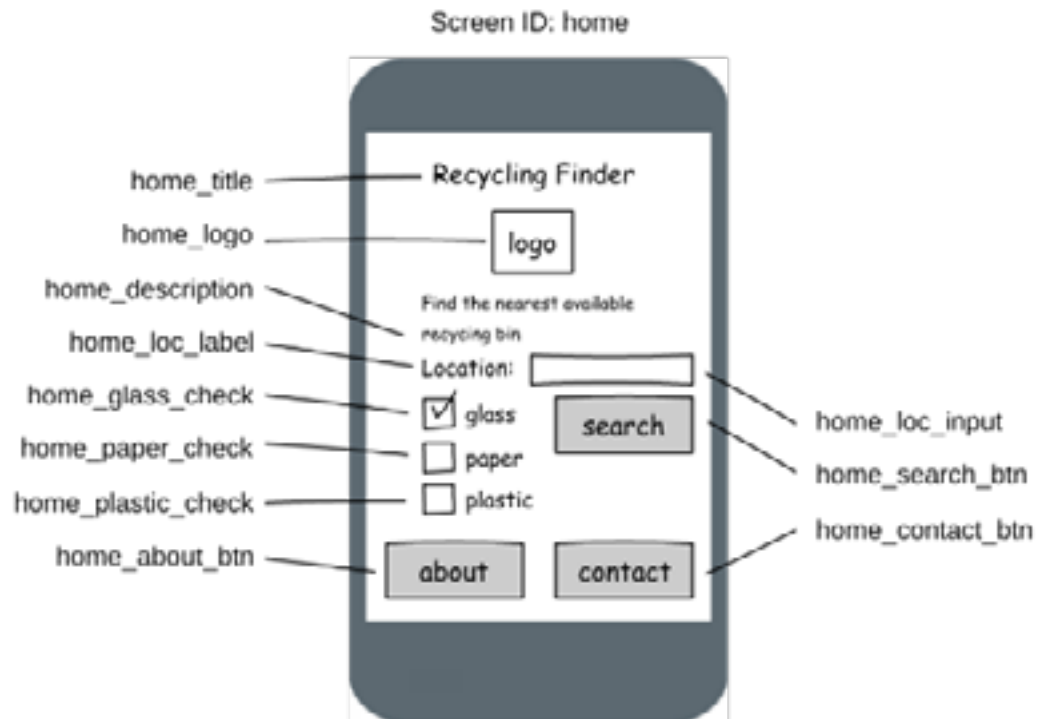
Un espacio de nombres es una cadena de texto única que agregará a todos sus ID de elementos. De esta forma, puedes asegurarte de que los elementos que creas no entren en conflicto con los creados por tus compañeros de equipo. Siempre que todos los integrantes de tu equipo utilicen un espacio de nombres diferente, cuando llegue el momento de combinar todas tus pantallas en una sola aplicación, no tendrás que preocuparte por las identificaciones conflictivas.

Una buena forma de crear un espacio de nombres es combinar el nombre de la pantalla con un guión bajo (_). Por ejemplo, si tu nombre de pantalla es una cuenta y tiene un cuadro de entrada para el nombre del usuario, tu ID para ese cuadro de entrada sería `account_firstName`

Diseño de pantalla del modelo

Traduciendo de papel a digital

Usando Design Mode en App Lab, crea una versión digital de este prototipo de papel. Asegúrate de que los ID que uses coincidan con los ID de la derecha. Si un elemento de la pantalla no está etiquetado con un id, crea



Lección 96

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Prueba de usuario de la aplicación

Resumen

El grupo probará el prototipo de su aplicación con otros miembros de su clase y potencialmente usuarios fuera de la clase. Para obtener la mejor retroalimentación posible, se asignarán diferentes roles en el proceso, de modo que mientras algunos miembros del equipo ejecuten la simulación, los demás solo se concentren en escribir comentarios.

Asignar roles para pruebas

Decide quién en tu equipo tendrá cada uno de los siguientes roles para la prueba. Si lo deseas, puede cambiar roles entre pruebas.

- **Narrador:** La persona que ejecuta la prueba. Explican lo que le está sucediendo al usuario, responde cualquier pregunta (aunque no ayuda al usuario) y asigna nuevas tareas a los usuarios.
- **Observadores:** Observa la interacción y escribe en sus notas lo que ve que hace el usuario en respuesta a la computadora.

Identificar usuarios

Decide quién en tu clase será tu usuario de tu aplicación. Si lo deseas, también puedes realizar esta prueba con personas ajenas a la clase que podrían ser parte del público objetivo de la aplicación.

Crear y ejecutar casos de prueba

En la siguiente hoja, encontrarás la hoja de casos de prueba que usarás para ejecutar y registrar tu prueba.

Resumir los hallazgos

Una vez que hayas ejecutado tus pruebas, registra los hallazgos o cambios más importantes que hayas encontrado en sus pruebas.

El usuario hizo / dijo	Que significa
<i>Seguir haciendo clic en la cara de Suzette para tratar de cambiar su configuración.</i>	<i>Las imágenes de GrandId deben enlazar a la página de configuración.</i>

Casos de prueba

Crear tareas de usuario: Completa la columna izquierda de la tabla con diferentes tareas comunes que su usuario querrá hacer con su aplicación.

Pon a prueba tu prototipo: Da esta guía de prueba a tu usuario. Prueba la aplicación comenzando en la pantalla de inicio y permite navegar a través de ella haciendo clic. Intenta completar cada tarea enumerada en la tabla.

Aquí hay algunas pautas:

- No expliques cómo funciona la aplicación para su usuario. Deseas observar cómo lo usarían sin guía.
- El usuario puede y debe pensar en voz alta. Esta es una forma de ayudarlo a comprender su experiencia.
- Mantén una hoja de papel para marcar cualquier otra cosa que notes durante la prueba.

Resultados de registro: En las columnas “Lo que el usuario trató” y “Mi reacción”, incluye lo que hizo tu usuario para completar cada tarea. En la siguiente tabla, incluye sus reacciones sobre cuán fácil o accesible era la aplicación.

Tarea	Lo que el usuario trató	Mi reacción

Lección 97

(reproducir e imprimir copia para cada estudiante)

Nombre(s)

Periodo

Fecha

Análisis de prueba del usuario

Interpretar los comentarios del usuario

Revisa sus notas de las pruebas de los usuarios y anote las observaciones de su equipo de todos los comentarios de los usuarios.

El usuario dijo / hizo	Que significa

Sesión de lluvia de ideas

Usando notas adhesivas y el análisis anterior, piense en los errores que el usuario identificó en la aplicación y las características que sugirió para la aplicación. Registre cada error o característica en una nota adhesiva por separado. Asegúrese de escribir si se trata de un ERROR o una FUNCIÓN, y calcule la cantidad de tiempo que el equipo cree que tomaría implementar la corrección o la nueva característica. ¡Recuerde que los errores y las características no solo están relacionados con el código! Texto confuso, elección de color inconsistente y diseños confusos son todos errores potenciales también.

Análisis de fallas y funciones

Duplique las tablas a continuación en una hoja de papel. Discuta entre su equipo dónde cree que irán los errores y las solicitudes de características en la tabla, colocando el adhesivo en el cuadrante apropiado. Si no está seguro acerca de la dificultad de implementar una característica, haga lo mejor posible.

	Fácil de implementar	Difícil de implementar
D		
D		

Qué hacer	Obra	Hecho
-----------	------	-------

Lección 98

(reproducir e imprimir copia para cada estudiante)

Nombre(s) Periodo Fecha

Rúbrica - presentación app

Rúbrica específica del proyecto

Criterio	Sí / No	Comentario
Organizado de manera clara y coherente		
El problema está bien definido		
La investigación de mercado incluye aplicaciones específicas y sus citas		
El prototipo de papel incluye pantallas múltiples con elementos de pantalla claramente definidos		
El prototipo digital demuestra la respuesta a los comentarios del prototipo en papel		
Demostración cubre la funcionalidad básica y muestra claramente cómo debería funcionar la aplicación		
La reflexión muestra receptividad a los comentarios de los usuarios		
Los recursos son citados apropiadamente		

Práctica de reflexión

Práctica	Cosas para celebrar	Cosas para trabajar
Resolución de problemas		
Persistencia		
Creatividad		
Colaboración		
Comunicación		

Referencias

Todas las fuentes tienen licencias de uso abierto y están diseñadas para la escuela. Hemos procurado buscar fuentes en español o con posibilidad de ser traducidas sin dificultad.

- <https://studio.code.org/>
- <https://cuantrix.mx/>
- <https://www.tinkercad.com/>
- <https://educaixa.org/es/programa-bigdata-en-el-aula>
- <https://www.iste.org/>
- <https://www.ibm.org/activities/machine-learning-for-kids>
- <https://www.csteachers.org/>

Glosario⁴

ABSTRACCIÓN

Sacar diferencias específicas para que una solución funcione para varios problemas.

ALGORITMO

Una lista de pasos para finalizar una tarea. Un conjunto de instrucciones que se pueden realizar con o sin un computador.

AUTENTICACIÓN

Verificación de la identidad de una persona o proceso.

BINARIO

Una forma de representar información usando solo dos opciones. Por ejemplo: verdadero o falso, sí o no, 0 o 1, encendido o apagado.

BOOLEANO

En honor al matemático británico George Boole (1815-1864) es un tipo de dato o expresión que representa información binaria pudiendo ser verdadero o falso.

BUG/ERROR

Algo que evita que un programa funcione como se esperaba.

BÚSQUEDA DE PATRONES

Encontrar similitudes entre las cosas.

CABLE DE FIBRA ÓPTICA

Una conexión que usa la luz para transmitir información.

⁴ <https://studio.code.org/>; <https://cuantrix.mx/>; <https://www.tinkercad.com/>; <https://educaixa.org/es/programa-bigdata-en-el-aula>; <https://www.csteachers.org/>; <https://www.iste.org/>.

CAJA DE HERRAMIENTAS

Es la barra gris en la sección central del sistema de aprendizaje en línea de Code.org, donde están todos los comandos que puede utilizar para escribir un programa.

CÓDIGO

Uno o más comandos o algoritmos diseñados para ser ejecutados por un computador.

COMANDO

Una instrucción para el computador. Muchos comandos juntos componen algoritmos y programas de computadores.

CONDICIONAL

Tipo de instrucción de un lenguaje de programación que permite realizar diferentes cálculos o acciones dependiendo si una condición booleana que aparece en un programa es verdadero o falsa.

CONDICIONALES

Declaraciones que solo se ejecutan bajo ciertas condiciones o situaciones.

CIBER SEGURIDAD

Protección contra el acceso no autorizado o su alteración, mediante el uso de tecnología, procesos o capacitación.

CIENCIAS DE LA COMPUTACIÓN

Disciplina de estudio que desarrolla conocimientos relacionados con los computadores y algoritmos; abarcando desde sus principios fundamentales, el hardware, software, sus aplicaciones y su impacto en la sociedad.

CIUDADANO DIGITAL

Alguien que actúa con seguridad, responsabilidad y de forma respetuosa en instancias online ante cualquier interacción con la tecnología.

COLABORACIÓN ABIERTA

Obtener ayuda de un gran grupo de gente para terminar algo más rápido.

CONTROLADOR DE EVENTOS

Un monitor para un evento o una acción específica en un computador. Cuando escribes un código para un controlador de eventos, se ejecutará cada vez que ocurre un evento o una acción. Muchos controladores de eventos responden a acciones humanas como los clics del mouse.

DATOS

Cantidades, caracteres o símbolos que son las entradas y salidas de programas de computador.

DEPURACIÓN/ DEBUGGING

Buscar, encontrar y corregir errores en programas.

DESCOMPONER

Dividir un problema en piezas más pequeñas.

DNS (SERVICIO DE NOMBRES DE DOMINIO)

El servicio que asocia información con nombres de dominio.

DSL/CABLE

Un método para enviar información, utilizando cables de teléfono o televisión.

DIGITAL

Característica de la tecnología electrónica que usa valores discretos, generalmente 0 y 1, para generar, almacenar y procesar datos.

DIRECCIÓN IP

Un número asignado a cualquier artículo que está conectado a Internet.

EJECUTE EL PROGRAMA

Hacer que el computador realice los comandos que has escrito en el programa.

ENTRADA

Datos, señales u órdenes que recibe una computadora.

ESPACIO DE TRABAJO

El área blanca en el lado derecho del sistema de aprendizaje en línea de Code.org, donde arrastra y suelta comandos para construir su programa.

EVENTO

Una acción que causa que suceda algo.

FUNCIÓN

Un fragmento de código que se puede insertar fácilmente una y otra vez. Las funciones a veces se denominan "procedimientos". Para utilizar una función primero debe ser definida, para lo cual se escribe el segmento de código que incluye los pasos que se busca reutilizar. Luego, podrá utilizarse llamándola en el segmento de código, típicamente dentro de la lógica principal del programa, que invoca la función.

HUELLA DIGITAL

La información que queda grabada sobre las actuaciones de alguien en Internet.

INTERNET

Un grupo de computadores y servidores que están conectados entre sí.

ITERACIÓN

Una acción o comando repetitivo normalmente creado con loops/ciclos de programación.

LOOP/CICLO

La acción de repetir un comando múltiples veces

NOMBRE DE USUARIO

Un nombre que inventas para poder ver o hacer cosas en un sitio web.

OFFLINE

Fuera de línea. Indica que un dispositivo, software o usuario está desconectado a internet.

ONLINE

En línea. Indica que un dispositivo, software o usuario está conectado a internet.

PAQUETES

Pequeños trozos de información que han sido cuidadosamente formados a partir de grandes cantidades de información.

PENSAMIENTO COMPUTACIONAL

Procesos y estrategias mentales que incluyen el cuerpo de conocimientos de las Ciencias de la Computación y el desarrollo de las habilidades necesarias para desenvolverse en el contexto del siglo XXI.

PERSISTENCIA

Intentar algo una y otra vez, incluso cuando es muy difícil.

PROGRAMA

Un algoritmo que ha sido codificado en algo que pueda ser ejecutado por una máquina.

PROGRAMACIÓN EN BLOQUES

El lenguaje de programación visual utilizado en el aprendizaje en línea de Code.org sistema para estudiantes K-5.

SERVIDORES

Computadores que existen sólo para proporcionar cosas a los demás.

STREAMING

Conexión directa de contenido de video y audio de un dispositivo a otro.

URL (LOCALIZADOR DE RECURSOS UNIVERSAL)

Una dirección fácil de recordar para llegar a una página web (como www.code.org).

VARIABLE

Un marcador de posición para un dato que puede cambiar.

WI-FI

Un método inalámbrico para enviar información mediant

Contenido, diseño y
diagramación de estas guías
están bajo una licencia
Creative Commons.



Ideo Digital

