



Paquete educativo de 16
Libro guía para educadores



Paquete educativo de 16
Libro guía para educadores

Material con copyright

Mini guía para educadores, paquete de 16, de Sphero

Copyright © 2023 por Sphero, Inc.

Reservados todos los derechos. Impreso en los Estados Unidos de América.

Ninguna parte de esta publicación puede reproducirse, almacenarse en un sistema de recuperación ni transmitirse, de ninguna forma ni por ningún medio (electrónico, mecánico, fotocopia, grabación o de otro tipo) sin el permiso previo por escrito del editor, excepto la inclusión de citas breves en una reseña.

Para obtener información sobre este título o solicitar otros libros y/o medios electrónicos, comuníquese con el editor: Sphero, Inc education@sphero.com www.sphero.com

Tabla de contenido

Bienvenido	6
Conoce Sphero mini	6
Conoce tu mini paquete de 16	7
 ¿Por qué mini?	 9
Aprendizaje alineado con los estándares para grados 3 y superiores	9
Pensamiento computacional	10
Ciencias de la Computación para TODOS los estudiantes	11
 Empezando	 13
Operación	14
Cargando	16
Cuidado	17
La aplicación Sphero Edu	18
Conectando como educador	19
Conectando como estudiante	20
 Actividades de clase	 22
Lección 1: Dibujar Croquet De Lona	24
Lección 2: El tejido de la mochila	29
Lección 3: Bola de datos	34
Lección 4: La parte superior que cambia de color	40
Lección 5: Laberinto Caos	46
Lección 6: Evita al Minotauro	51
Lección 7: Que personaje	55
Lección 8: Funciones divertidas y divertidas	60
 Ir más lejos	 65
Ofertas curriculares adicionales	66
 Recursos	 67
Objetivos	68
Glosario	70
Biblioteca de bloques	71

Bienvenido

Conoce Sphero mini

Lo suficientemente pequeño como para aprender sobre una mesa pero lo suficientemente resistente como para rodar por el suelo, Sphero mini fue diseñado para estudiantes de todas las edades. mini fomenta el amor por la robótica, la codificación y los principios STEM, todo a través del aprendizaje basado en el juego. Con los mini estudiantes pueden:

- programa con lenguajes de dibujo, bloques y texto en cualquier tipo de dispositivo informático.
- recopile datos de los sensores integrados, incluidos un giroscopio y un acelerómetro.
- enseñar los conceptos básicos de codificación e informática



Conoce tu mini paquete educativo de 16

El mini Education 16 Pack es su ventanilla única para lanzar y mantener una iniciativa de programación y STEM exitosa. Además de 16 mini robots Sphero, el paquete educativo incluye:

- 16 cubiertas de bolas robóticas (como ponerse los zapatos)
- 16 juegos de 6 pines y 3 conos
- 16 juegos de construcción
- 16 cables USB divisores
- 16 juegos de 15 tarjetas de actividades



Esta Guía para educadores está diseñada para ayudarle a comenzar y maximizar el valor de sus nuevos minis en el aula. Vas a:

- Explore los muchos beneficios de mini para la participación y el aprendizaje de los estudiantes.
- Conozca las características, funcionalidad y uso de sus mini robots.
- acceda a ocho lecciones listas para implementar que puede usar con sus estudiantes para iniciar el aprendizaje
- Sumérgete en la aplicación Sphero Edu y aprende cómo usarla para configurar y administrar tu miniaula.
- Descubra formas de hacer que todo funcione en cualquier entorno de aula.

MISIÓN ESFERA

Sphero está transformando la educación desde preescolar hasta 12.º grado con herramientas accesibles que fomentan la exploración, la imaginación y la perseverancia a través de STEM y la informática. Con la ayuda de educadores de todo el mundo, estamos capacitando a estudiantes de todos los orígenes y habilidades para que descubran sus intereses y pasiones, mientras los equipamos con las habilidades que necesitan para ser los futuros agentes de cambio del mundo.

¿Por qué mini?

Aprendizaje alineado con estándares para edades de 7 años en adelante

Sphero mini se puede utilizar para enseñar conceptos de informática y complementar materias en cualquier área de contenido. Sphero ofrece amplias actividades de aprendizaje que están alineadas con los estándares estadounidenses y otros estándares nacionales e internacionales para una fácil integración en los planes de estudio.

Usted, como profesor, no necesita ser un experto en programación para integrar mini en su instrucción en el aula. La aplicación Sphero Edu ofrece tres “lienros” de codificación diferentes (Dibujar, Bloquear y Texto) que van desde habilidades de codificación principiantes hasta habilidades de codificación más avanzadas. Los tres lienros de codificación le facilitan el progreso como programador junto con sus alumnos.

		
Dibujar lienzo Programa mini con un interfaz de dibujo.	Bloque de lienzo Programe mini arrastrando y soltando bloques que representan líneas de código.	Lienzo de texto Programa mini con el lenguaje de programación JavaScript.

Sphero mini es una gran herramienta para presentar conceptos de codificación y programación a los estudiantes. Para los estudiantes jóvenes, mini hace que comenzar a programar sea atractivo y divertido. Este enfoque práctico hace que aprender a codificar sea más atractivo y tangible.

El Sphero mini proporciona una forma práctica de enseñar conceptos de ciencia, tecnología, ingeniería y matemáticas (STEM). Los estudiantes pueden explorar la física, la geometría y varios principios de ingeniería a través de los movimientos e interacciones del robot.

Pensamiento computacional

Sus estudiantes desarrollarán importantes habilidades de pensamiento computacional (CT) cuando aprendan con mini a través de lecciones decididas y alineadas con los estándares, así como un juego abierto y no estructurado.

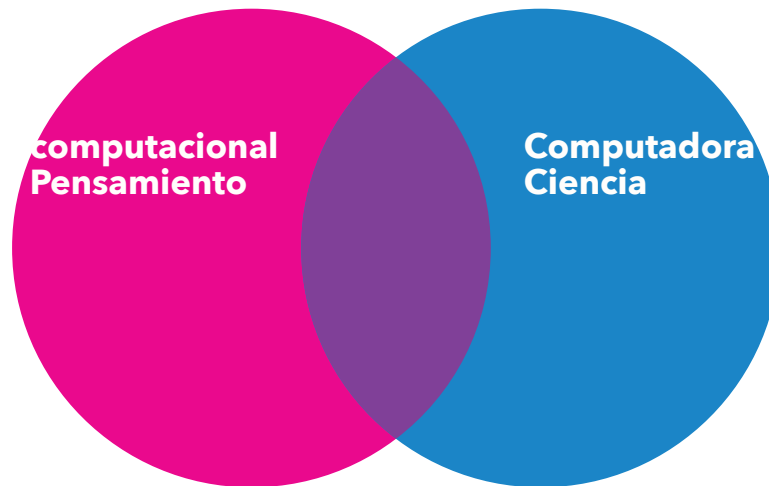
El pensamiento computacional es un conjunto sistemático de procesos para abordar problemas complejos. Estos procesos a menudo se describen como “cuatro pilares”. La siguiente tabla define cada pilar y proporciona descripciones de cómo el aprendizaje con mini desarrolla la habilidad de pensar.

Pilar CT	DEFINICIÓN	Ejemplo con mini
 Descomposición	La división de algo, como un problema complejo, en partes más pequeñas y manejables.	Describir un desafío de programación, como un laberinto, en partes, y luego programar mini para realizar cada parte discreta en secuencia.
 Patrón Reconocimiento	El análisis de objetos o ideas similares para ampliar o crear patrones para comprender mejor un problema.	Analizar cómo tres entradas (rumbo, velocidad y duración) gobiernan la ejecución de los bloques de balanceo del mini para ganar precisión sobre el movimiento del robot.
 Abstracción	El proceso de eliminar la información importante e ignorar los detalles irrelevantes.	Usar diagramas de flujo para planificar, ejecutar y comunicar un mini programa de forma clara y eficiente
 Algoritmico Diseño	El desarrollo de pasos utilizados para resolver un problema, a menudo un conjunto secuencial de reglas que se siguen.	Programar un control if...else para ejecutar diferentes grupos de código cuando un mini sensor recopila datos dentro de diferentes rangos

Considere las siguientes formas en que la CT beneficia a los estudiantes de informática, otras áreas temáticas y su vida cotidiana.

- **CT ayuda a los estudiantes a resolver problemas relacionados con las computadoras.**
Por ejemplo, un estudiante puede descomponer un objetivo de programación difícil en partes manejables más pequeñas utilizando un lenguaje común llamado pseudocódigo.
- **CT ayuda a los estudiantes a aplicar las computadoras a los problemas del mundo que los rodea.**
Por ejemplo, un estudiante puede crear un algoritmo para programar mini para navegar por un laberinto o seguir un camino específico, lo que le obliga a dividir los problemas en partes más pequeñas y manejables.
- **CT ayuda a los estudiantes a abordar problemas no relacionados con la informática.**
Por ejemplo, un estudiante puede utilizar la abstracción para centrarse en detalles importantes de un libro para identificar los temas.

La TC se suele considerar indistintamente con la informática, pero, si bien están relacionadas, tienen algunas diferencias clave. Las habilidades CT que sus estudiantes desarrollan mientras resuelven problemas de programación con mini los beneficiarán en sus futuros esfuerzos de programación, así como en otros aspectos de sus vidas.



Ciencias de la Computación para TODOS los estudiantes

mini fue diseñado para hacer que el aprendizaje de los fundamentos de la programación y la informática sea accesible y divertido para todos los estudiantes, independientemente de sus antecedentes o habilidades. A medida que continuamos avanzando hacia un futuro cada vez más digitalizado tanto en nuestro trabajo como en nuestra vida cotidiana, necesitaremos más trabajadores en ocupaciones relacionadas con la informática. La Oficina de Estadísticas Laborales de EE. UU. predice que la cantidad de empleos en la industria aumentará un 13% entre 2020 y 2030, más rápido que en otras ocupaciones.¹

Sin embargo, el campo de la informática todavía lucha por atraer y retener una fuerza laboral que represente con precisión la diversidad de género, racial, étnica y socioeconómica de nuestras

¹ Oficina de Estadísticas Laborales de EE. UU. (<https://www.bls.gov/ooh/computer-and-information-technology/home.htm>)

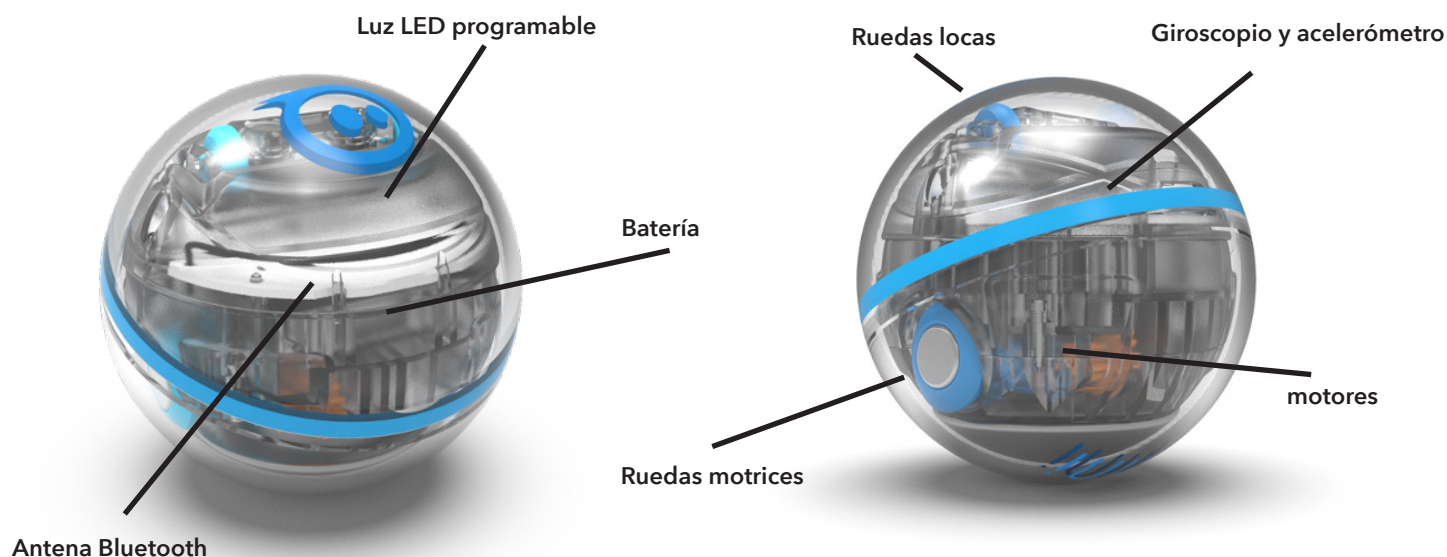
comunidades. Una fuerza laboral más diversa, junto con las ricas perspectivas y antecedentes que la acompañan, es esencial para resolver los problemas informáticos del mañana.

El trabajo de organizaciones como code.org y Girls Who Code ha demostrado que introducir la informática a los estudiantes en las primeras etapas de su educación desde preescolar hasta 12.º grado es clave para atraer a una gama más amplia de estudiantes a profesiones relacionadas con la informática. Sphero mini puede hacer que las experiencias iniciales de programación de los estudiantes sean prácticas y visuales. Arrastre un bloque rodante al lienzo, ejecute el programa e inmediatamente observe el efecto que su programa tiene en el robot. Considere cómo los siguientes usos activos de mini hacen que el aprendizaje de la informática sea accesible y tangible:

- Usando prueba y error para programar el camino de Mini a través de un laberinto.
- comunicarse con un compañero de aprendizaje para ajustar los valores en un comparador
- Crear un juego con mini para mostrar la comprensión del sensor de orientación de cabeceo y balanceo.
- compartiendo una historia única a través de un mini programa con luces, sonidos y movimientos

En comparación con métodos de instrucción más pasivos, como ver un video tutorial o escuchar a un maestro, aprender con mini coloca a los estudiantes directamente en el centro del proceso de aprendizaje.

Empezando



mini es compacto (aproximadamente del tamaño de una pelota de ping pong), robusto, programable y divertido. Sus características ayudarán a los estudiantes de todas las habilidades y edades a desarrollar sus habilidades de programación y pensamiento.

minis **batería** Durará 1 hora completa, pero solo tardará 45 minutos en cargarse.

Mini está equipado con **dos motores de accionamiento**, uno a cada lado del robot, para permitir movimientos como rodar y girar. Los codificadores de motor informan datos sobre velocidad y distancia.

mini tiene un **luz LED programable** que puede mostrar cualquier color

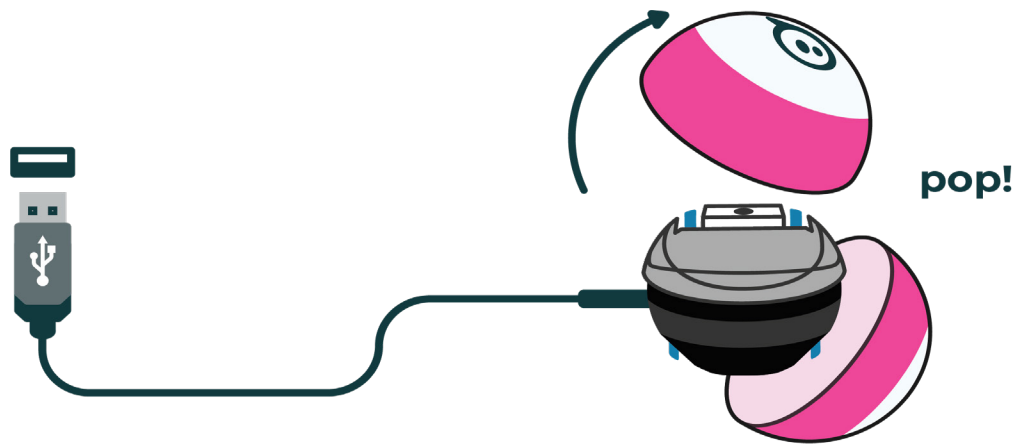
mini a bordo **sensores** le permite recopilar datos durante la ejecución del programa que pueden usarse para programar lógica interesante en la aplicación Sphero Edu. Aquí hay una descripción general:

- **Acelerómetro** : El acelerómetro mide la aceleración lineal y se puede utilizar para detectar cambios de velocidad y colisiones.
- **Giroscopio** : El giroscopio mide el movimiento de torsión o rotación alrededor del cabeceo (eje x), balanceo (eje y) y guiñada (eje z).
- **Codificadores de motor** : Los codificadores del motor proporcionan información sobre la posición, la velocidad o la dirección de rotación del motor.

Operación

Para encender mini:

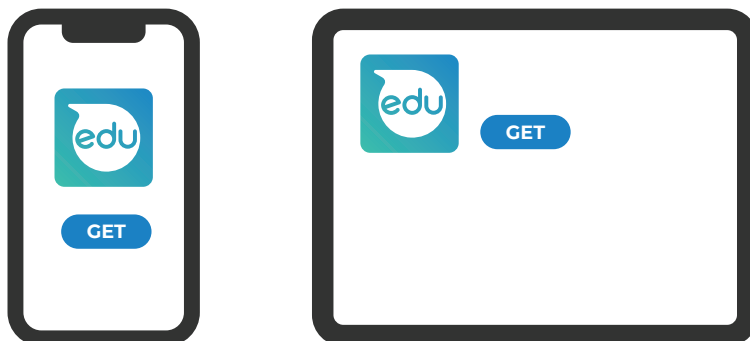
1. Sphero Mini se carga mediante un cable de carga Micro USB. Se proporciona un cable en el paquete, pero necesita un enchufe de pared de CA exclusivo. Para cargar, simplemente retire la carcasa del Sphero Mini, ubique el pequeño puerto de carga micro USB y conecte el Sphero Mini a la fuente de alimentación.



Para conectarse a la aplicación Sphero Edu:

1. Descargue la aplicación Sphero Edu en iOS, Android, Kindle, Mac, Windows o Chrome:

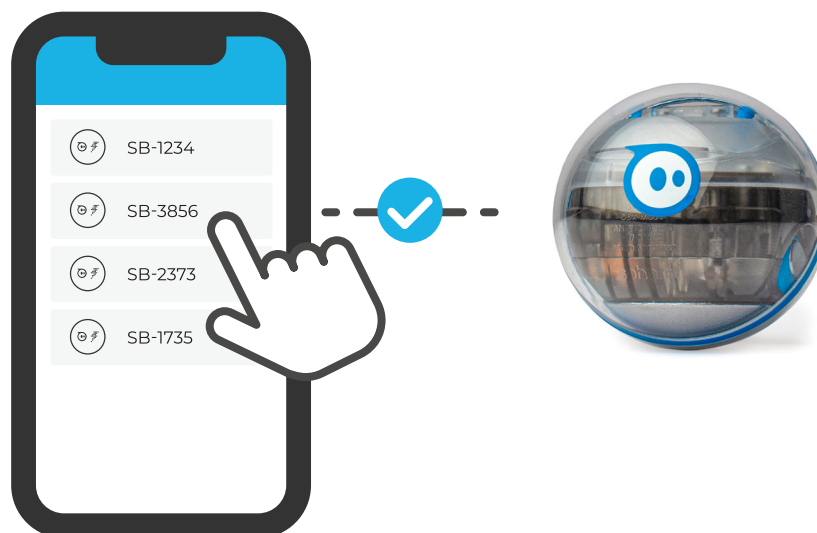
<https://sphero.cc/edu-d>



2. Abra la aplicación Sphero Edu y busque el botón 'Conectar robot'.



3. Sostenga su dispositivo cerca de su mini.
4. Seleccione mini de la lista de robots y luego seleccione su robot de la lista para conectarse.



Cargando

Para cargar los 16 mini al mismo tiempo necesitarás un concentrador de carga USB con hasta 16 puertos o 2 concentradores con 8 puertos.

- Una carga de 1 hora te dará 45 minutos de tiempo de reproducción.
- Sphero Mini no puede conducir cuando está enchufado a una fuente de alimentación porque los motores están desactivados.

Cuidado de la batería

La mejor manera de prolongar la vida útil de la batería es guardar su Sphero Mini desconectado. La batería se agotará con el tiempo y el robot se apagará. Deberá cargarlo la próxima vez antes de usarlo. Esto evita que la batería repita ciclos de carga y descarga.

Sphero Mini tiene dos modos de energía diferentes, Apagado y En Espera. Dependiendo de en cuál de estos modos de energía se encuentre tu Mini, tendrás que encenderlo de manera diferente...

- **Apoyar** -- si simplemente cerró la aplicación y dejó que Sphero Mini permanezca inactivo hasta que se apaguen las luces, estará en modo de espera. Para despertar, simplemente inicie la aplicación Sphero Play junto a su robot.
- **Apagado** -- si navegó a la página de configuración de la aplicación Sphero Play y seleccionó "Apagar", su Mini se apaga. Para reactivarlo después del apagado, conecte su Mini, espere a que se encienda la luz LED y luego inicie la aplicación.

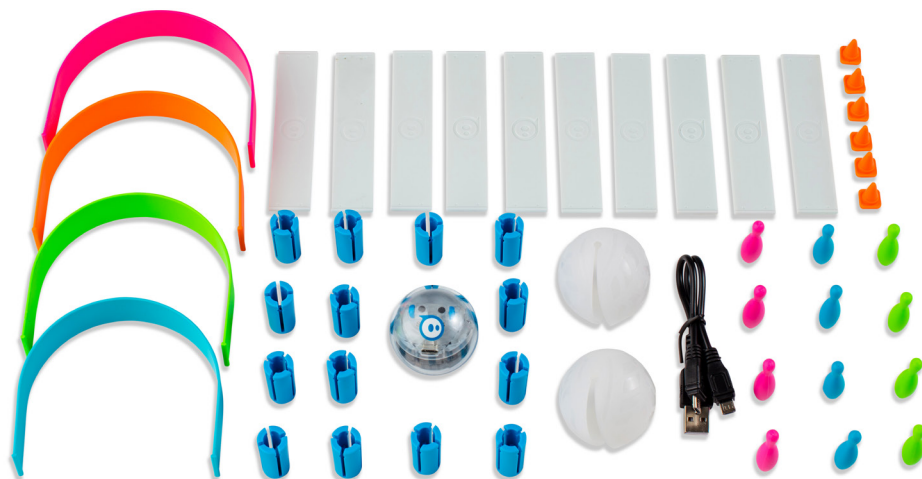
Cuidado

El hardware minis está encerrado en una carcasa que puede abrir para acceder al puerto de carga. Si bien el mini es bastante duradero, debes evitar dejarlo caer desde cualquier altura. mini no es resistente al agua, así que evite llevar el robot a nadar o sumergirlo en pintura.

Accesorios mini EDU, paquete de 16

Además de 16 mini robots, su mini EDU Pack viene con los siguientes accesorios para impulsar el aprendizaje y la diversión de los estudiantes:

- 16 cubiertas de bolas robóticas
- 16 juegos de 6 pines y 3 conos
- 16 juegos de construcción
- 16 cables USB divisores
- 16 juegos de 15 tarjetas de actividades



La aplicación Sphero Edu

La aplicación Sphero Edu está diseñada para permitirle a usted y a sus alumnos aprovechar las funciones de mini y crecer con su robot a medida que desarrolla su comprensión de la programación.

La aplicación Sphero Edu incluye muchas funciones, entre ellas:

- actividades de aprendizaje alineadas con los estándares para una variedad de niveles de habilidades y áreas de contenido
- Programas precreados para mini. Ejecute el programa tal cual o úselo como punto de partida para codificar el suyo propio.
- Lienzos de dibujo, bloques y texto para crear programas
- información adicional sobre los robots de Sphero

Dispositivos compatibles

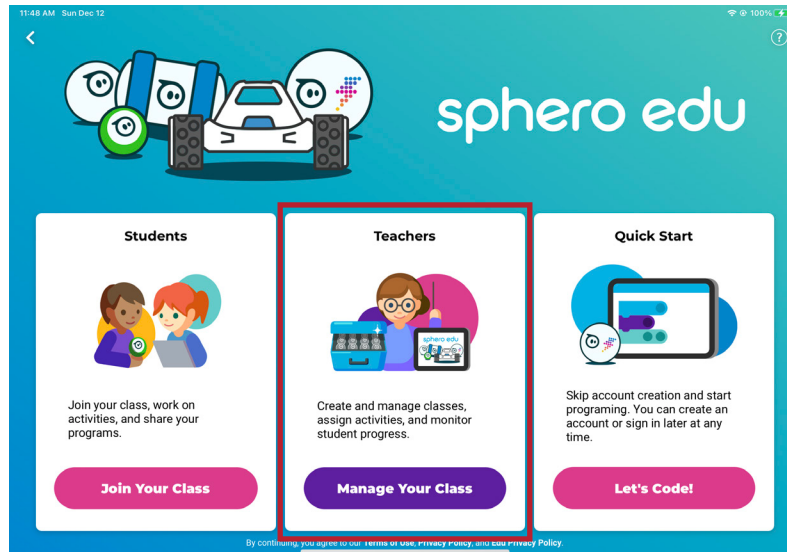
Para comenzar, será necesario descargar la aplicación en los dispositivos de aprendizaje de los estudiantes. La aplicación está disponible de forma gratuita en las tiendas de aplicaciones de iOS, Google Play, MacOS, Microsoft y Amazon para su uso en cualquier tipo de dispositivo del aula. Para usar en Chromebooks, deberá instalar la aplicación de Android Sphero Edu desde la tienda Google Play.

Descarga la aplicación para tu dispositivo:



Conectando como educador

Para comenzar como profesor con la aplicación Sphero Edu, seleccione "Profesores" la primera vez que abra la aplicación.



Se le guiará a través de la creación de la cuenta. Una vez que haya creado su cuenta, tendrá acceso a las mismas herramientas de programación que sus estudiantes. También podrá iniciar sesión en su cuenta en edu.sphero.com para configurar clases y asignar actividades a sus estudiantes.

CONSEJO PARA EL AULA: Es una buena idea configurar su clase e invitar a los estudiantes antes de que utilicen la aplicación. Si los estudiantes crean cuentas antes de que se realice una clase, deberán ingresar un correo electrónico del tutor para la aprobación de la cuenta.

Obtenga más información con el siguiente enlace sobre cómo configurar y administrar su aula Sphero:

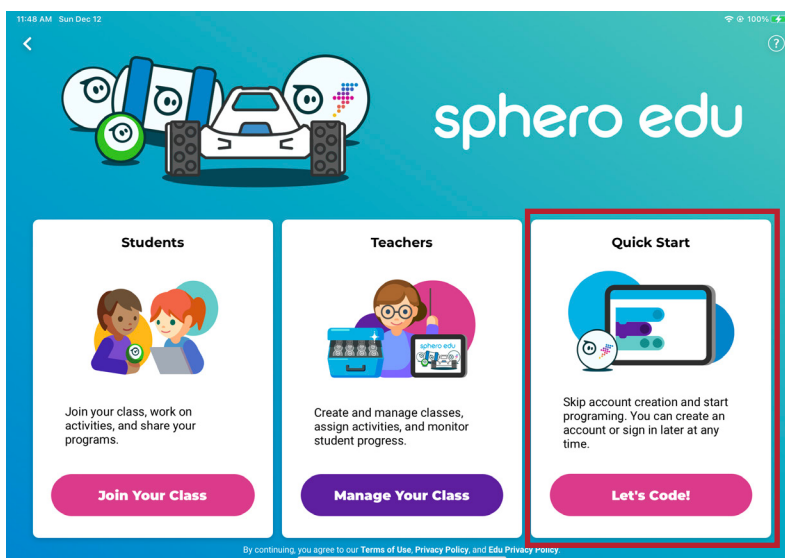
sphero.cc/gettingstarted



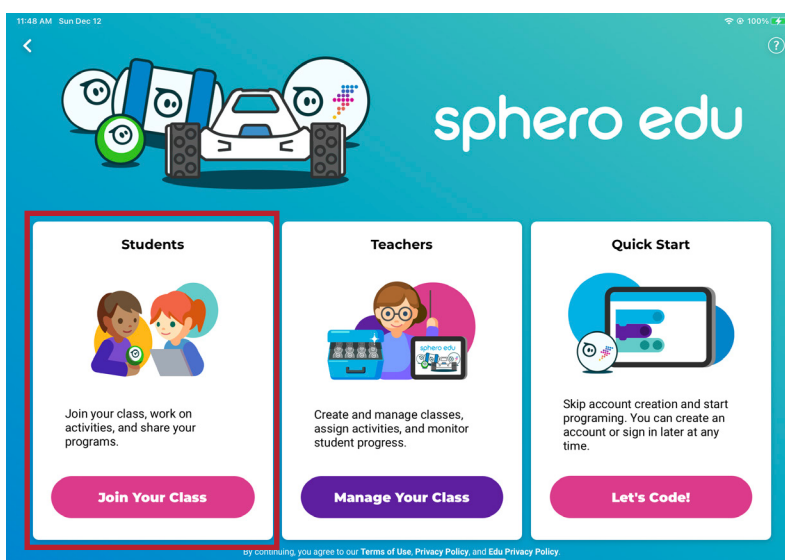
Conectando como estudiante

Una vez instalada la aplicación, los estudiantes tienen dos opciones para conectar sus dispositivos con BOLT y comenzar con la programación:

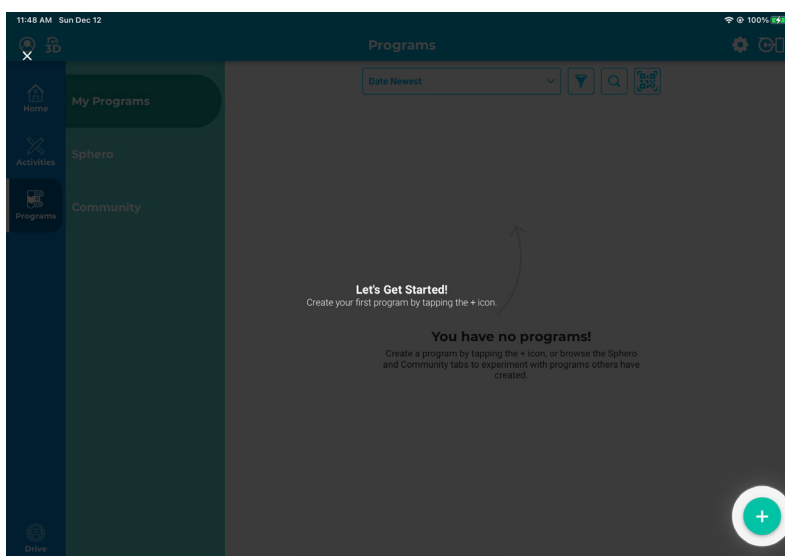
1. Seleccione "Codificar" en el cuadro Inicio rápido para comenzar sin crear una cuenta. Los estudiantes podrán crear y ejecutar programas. Sin embargo, ninguno de sus trabajos se guardará en la nube para estar disponible desde otro dispositivo.



2. Seleccione Unirse a su clase en el cuadro Estudiantes para abrir la aplicación con un código de clase (si ya creó una clase) o inicie sesión con una cuenta de Google, Clever o Sphero. Este método permitirá a los estudiantes guardar su trabajo en la nube.



3. Una vez que se abra la aplicación, elija "Programas", luego seleccione el símbolo más en la esquina inferior derecha para iniciar su primer programa.



Actividades de clase

Su paquete mini EDU 16 es su aula STEM nada más sacarlo de la caja. Pero ahora que lo tienes, ¿cómo puedes poner en marcha el aprendizaje STEM? Las ocho actividades listas para usar, desempacar y listo incluidas en esta Guía para educadores lo ayudarán a usted y a sus estudiantes a comenzar a usar mini y los lienzos de dibujo, bloques y texto rápidamente, sin necesidad de experiencia previa. Cada lección está diseñada para estudiantes de todas las edades y habilidades y se puede completar en un período de clase de 45 a 60 minutos. Complételos en orden o salte según el nivel de comodidad de sus estudiantes con mini y programación. Si bien los estudiantes deben tener la aplicación Sphero Edu instalada en sus dispositivos, no necesitarán crear cuentas ni iniciar sesión para trabajar en los programas.



DIBUJAR

ENFOQUE DE CONTENIDO ENFOQUE DE PROGRAMACIÓN

Lección 1: Dibujar croquet de lona

Ciencias de la Computación

secuencias



BLOQUE INICIO

Lección 2: El tejido de la mochila

Ciencias de la Computación

bloques de movimiento

Lección 3: Bola de datos

Ciencias de la Computación

Bloques de movimiento, luces y sonido.



BLOQUE INTERMEDIO

Lección 4: La camiseta que cambia de color

Ciencias de la Computación

declaraciones condicionales

Lección 5: Caos en el laberinto

matemáticas

eventos



BLOQUE AVANZADO

Lección 6: Evita al Minotauro

Ciencias Sociales

variables, funciones

Lección 7: ¡Qué personaje!

Artes del lenguaje

Bloques de movimiento, luces y sonido.



TEXTO INICIAL

Lección 8: Funciones divertidas y divertidas

Ciencias de la Computación

Sintaxis de JavaScript

Antes de enseñar:

- Lea las instrucciones del educador para la actividad, pruebe el desafío de programación y anticipe los obstáculos que encontrarán sus alumnos.
- Copie y/o prepárese para proyectar las instrucciones de la lección dirigidas a los estudiantes.
- Asegúrese de que todos los robots y dispositivos de programación estén completamente cargados.
- Planifique dos estudiantes por mini robot.

Durante la enseñanza:

- Utilice los pasos de Exploración y Desarrollo de habilidades para iniciar la actividad e introducir habilidades fundamentales.
- Brinde a los estudiantes la oportunidad de participar en el Desafío sin guía. Circule por el espacio de aprendizaje para celebrar el trabajo en equipo, los errores y la resolución de problemas.
- Anime a los estudiantes a ampliar su aprendizaje con las opciones del Desafío Extendido.

Después de enseñar:

- Reflexionar sobre el aprendizaje de los estudiantes. ¿Qué fue fácil para los estudiantes? ¿Qué fue más difícil? ¿Hubo algún artefacto del aprendizaje de los estudiantes que pudiera compartir con toda la clase para abordar un desafío común o mostrar un concepto de programación o una estrategia de resolución de problemas?
- Planifica tu próxima lección. Utilice otra lección de esta Guía para educadores o pruebe otra actividad de Sphero en la aplicación Sphero Edu.



Lección 1: Dibujar Croquet De Lona

Descripción general

Bienvenido a tu primera Mini actividad en la aplicación Sphero Edu. Aprenda cómo conectar su dispositivo de programación a Mini y luego juegue un juego de croquet en Draw Canvas.

Nivel de programación

Sorteo principiante

Objetivos de aprendizaje

1. Puedo programar mini para que ruede y cambie de color en Draw Canvas.
2. Puedo revisar y depurar mi código para solucionar problemas.

Vocabulario

- **Dibujar lienzo:** El editor de la aplicación Sphero Edu para crear programas dibujando líneas
- **Ejecutar:** Para ejecutar o completar un bloque o programa

Estándares

- Descomponer (descomponer) los problemas en subproblemas más pequeños y manejables para facilitar el proceso de desarrollo del programa.
- Cree programas que incluyan secuencias, eventos, bucles y condicionales.

Conexiones de contenido

- Ciencias de la Computación

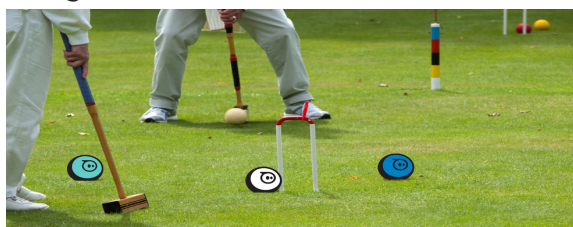
Materiales

- Paquete mini EDU de 16
- Programar dispositivos con la aplicación Sphero Edu instalada
- cinta de laberinto u otra cinta adhesiva

Preparación

- Monta una cancha de croquet (imagen 2).
- Utilice trozos de 15 cm de cinta azul para hacer los portillos por los que pasará Mini.
- ¡No te preocupes por las longitudes exactas! El croquet siempre es divertido incluso si la cancha no es perfecta. Obtenga una vista previa y prepárese para compartir las Instrucciones para el estudiante (página 28).

Configuración



Plan de estudios [Enlace](#) :

Consejo para el educador:

Si está enseñando esta actividad con un grupo de estudiantes, considere ahorrar tiempo instalando de 3 a 5 canchas de croquet con anticipación para prepararse para el aprendizaje. Esto permitirá que sus alumnos se concentren más en la programación de sus Sphero Minis y menos en configurar el juego. Al croquet se puede jugar con hasta seis jugadores y pelotas por cancha. Ajusta el tamaño de la cancha hacia arriba o hacia abajo según el espacio que tengas disponible.

Desarrollo de habilidades 1

Ahora que tienes la cancha de croquet configurada, es hora de conectar tu dispositivo de programación a tu Mini e intentar conducir.



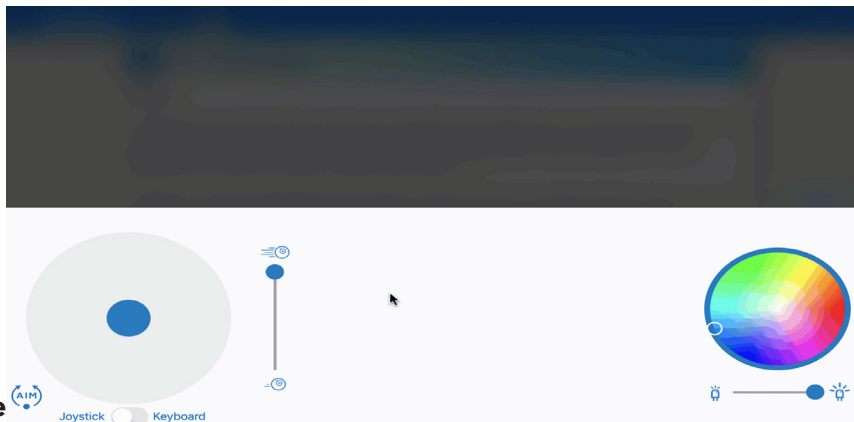
HACER:

1. Seleccione el ícono Drive en la esquina superior derecha de la aplicación.
2. Cuando se le solicite, seleccione Mini y luego seleccione su robot.
3. Una vez que su robot esté emparejado, seleccione AIM.
4. Mueva el punto azul para que la luz trasera del Mini mire hacia ti. Esta es la parte trasera del Mini.
5. Explora cómo cambiar la velocidad y el color de Mini.
6. Practica conducir Mini por la cancha de croquet.

CONVERSAR:

¿Cómo puedes diferenciar tu Mini de los demás?

¿Qué velocidad te da el mejor control mientras conduces?



Consejo para e

La actividad Introducción a Sphero Edu presenta a los estudiantes la aplicación Sphero Edu, los lienzos de programación y la conexión y conducción de su robot. Si los estudiantes ya completaron esa actividad, puede pedirles que se salten este paso. De lo contrario, es posible que los estudiantes necesiten un poco más de tiempo para completar y explorar este paso.

Quando se gestionan varios estudiantes y varios robots al mismo tiempo, emparejar dispositivos informáticos con Minis puede ser un desafío. Recomendamos utilizar el método de Harry Potter para asignar robots a los estudiantes. El estudiante no puede elegir al robot, sino que el robot elige al estudiante.

- Haga que los estudiantes se acerquen al paquete de Mini del aula uno a la vez.
- Los estudiantes seleccionan el Mini que esté en la parte superior de la lista en su dispositivo.
- El Mini que se enciende y se conecta a su dispositivo es su robot del día.
- Los estudiantes llevan su Mini a sus asientos.

Con los robots redondos, realizar un seguimiento de hacia dónde se avanza puede ser un desafío para los estudiantes. Anime a los estudiantes a alinear sus cuerpos, sus dispositivos de programación y la dirección de puntería de sus mini robots. De esta manera, un encabezado de 0 siempre estará directamente lejos de los estudiantes.



Desarrollo de habilidades: dibuja tu curso

LEER:

La aplicación Sphero Edu tiene tres lienzo de programación:

- Draw Canvas te permite programar Mini dibujando líneas.
- Block Canvas te permite arrastrar y soltar bloques para decirle a Mini qué hacer.
- Text Canvas le permite programar Mini con JavaScript, un lenguaje de programación popular.

En esta actividad, programarás tu Sphero Mini para que se mueva por la cancha de croquet con Draw Canvas.

HACER:

1. Coloque Mini en inicio y apunte el robot directamente al primer portillo.
2. Seleccione el botón Nuevo programa en la barra lateral derecha.
3. Dibuja una línea vertical en el Draw Canvas.
4. Seleccione Iniciar para ejecutar o ejecutar el programa. ¿Qué pasó? ¿Mini pasó por encima del primer portillo? ¿Los dos primeros portillos?
5. Revise el programa hasta que su robot pase sobre los dos primeros portillos.

CONVERSAR:

- ¿Cómo afecta el grosor de la línea a la ejecución del programa por parte de Minis?
- ¿Importa si trazas la línea de arriba hacia abajo o de abajo hacia arriba?

Consejo para el educador:

Tómese un tiempo para reforzar algunos aprendizajes críticos sobre Draw Canvas.

Las líneas más gruesas significan velocidades más rápidas. Las líneas más delgadas significan velocidades más lentas.

El color de la línea muestra el color del Mini LED mientras rueda.

La direccionalidad de la línea importa. Una línea trazada de derecha a izquierda le indicará al Mini que conduzca con un rumbo de 90 (hacia la derecha). Una línea trazada de izquierda a derecha le indicará al Mini que conduzca con un rumbo de 270 (hacia la izquierda).



Desafío

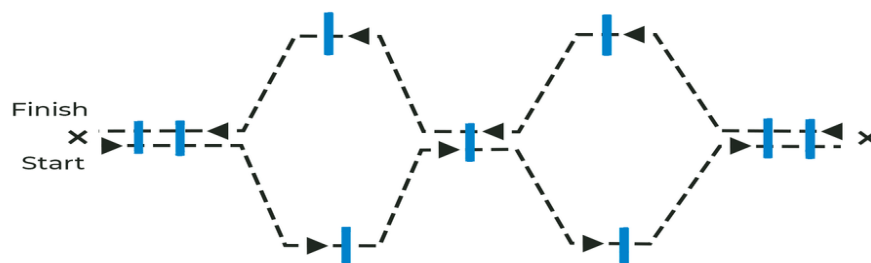
Es hora de jugar al croquet. Aquí hay algunas reglas simplificadas:

6. Los jugadores se turnan para ejecutar programas Draw Canvas para mover sus Minis sobre los portillos de la cancha de croquet.
7. Los minis deben pasar sobre los portillos en la dirección correcta.
8. Los jugadores pueden dibujar una, dos o tres líneas por programa.
9. Los jugadores obtienen un turno adicional si pasan por encima de un portillo O si chocan con otro Mini.
10. El primer jugador que logre completar el recorrido gana.

Luego, el equipo ganador se convierte en un explorador y puede continuar turnándose para ayudar o interrumpir a los jugadores restantes.

HACER:

11. Reúna un grupo de dos a cinco jugadores.
12. Lanza una moneda para ver quién va primero.
13. Elija diferentes colores para sus Mini robots para que pueda distinguirlos.
14. ¡Jugar!





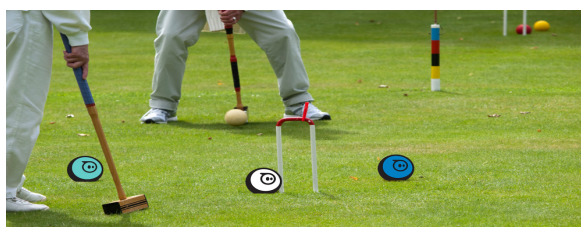
Lección 1 : Dibujar Croquet De Lona

Bienvenido a tu primera Mini actividad en la aplicación Sphero Edu. Aprenda cómo conectar su dispositivo de programación a Mini y luego juegue un juego de croquet en Draw Canvas.

Objetivos de aprendizaje

1. Puedo programar mini para que ruede y cambie de color en Draw Canvas.
2. Puedo revisar y depurar mi código para solucionar problemas.

Configuración



Código QR del programa



Exploración:

Configure un curso de croquet, abra la aplicación Sphero EDU. Empareje su mini con su dispositivo de programación y practique conducir el recorrido usando el ícono de manejo.



Desarrollo de habilidades:

En esta actividad, programarás tu Sphero Mini para que se mueva por la cancha de croquet con Draw Canvas. Seleccione el botón Nuevo programa en la barra lateral derecha para hacer esto.



Desafío:

Los jugadores se turnan para ejecutar programas Draw Canvas para mover sus Minis sobre los portillos de la cancha de croquet. El primer jugador que logre completar el recorrido gana.



Lección 2: El tejido de la mochila

Descripción general

Bienvenido a la Mini lección 2, su introducción a la programación de su Sphero Mini en Blocks Canvas. ¡Prepárate para aprender todo lo que hay que saber sobre el bloque rodante!

Nivel de programación

Bloque inicial

Objetivos de aprendizaje

1. Puedo controlar el movimiento de Mini con las tres entradas de un bloque de balanceo: velocidad, rumbo y duración.
2. Puedo programar a Mini para que ruede entre obstáculos y los rodee.

Vocabulario

- **Bloque de lienzo** : El editor en la aplicación Sphero Edu para crear programas de bloques
- **Velocidad** : Entrada para programar qué tan rápido (o lento) se moverá un robot Sphero
- **Título** : Entrada para programar la dirección en la que apunta un robot Sphero durante o antes de un giro, entre 0° y 360°
- **Duración** : Entrada para programar la cantidad de tiempo que mini ejecutará un movimiento como un giro o un giro en segundos

Estándares

- Compare y refine múltiples algoritmos para la misma tarea y determine cuál es el más apropiado.
- Cree programas que incluyan secuencias, eventos, bucles y condicionales.

Conexiones de contenido

- Ciencias de la Computación
- matemáticas

Materiales

- Mini robots Sphero completamente cargados
- Programar dispositivos con la aplicación Sphero Edu instalada
- mochilas u otros objetos del aula para usar como obstáculos

Preparación

- Despeje un espacio de 20' por 20' para configurar la actividad. Si el espacio de su salón de clases es limitado, considere realizar esta lección en un gimnasio, cafetería u otro salón de usos múltiples disponible.
- Obtenga una vista previa y prepárese para compartir las instrucciones de los estudiantes (página 33).

Código QR del programa





Exploración

- ¡El Canvas de Bloques en la aplicación Sphero Edu hace que aprender a programar sea fácil al unir bloques!
- Cada bloque es una instrucción de comando de programación para su robot.
- Arrastra un bloque de la biblioteca de bloques.
- Colóquelo en el bloque del programa de inicio.
- Seleccione Iniciar.

Esta actividad comienza con un programa que ya se ha iniciado para usted.

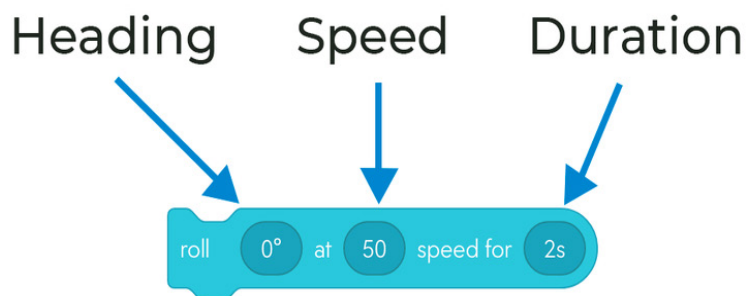
HACER:

- Abra el programa asociado a la actividad.
- Asegúrese de que su Mini esté emparejado con su dispositivo de programación.
- Seleccione Iniciar.

CONVERSAR:

Mire de cerca el bloque de desplazamiento debajo del programa de inicio.

- ¿Qué significan los tres números en el bloque de rollo?
- ¿Por qué Mini giró en la dirección que lo hizo?



Consejo para el educador:

Esta actividad supone que sus estudiantes ya tienen experiencia abriendo la aplicación Sphero Edu y conectando o emparejando su Mini robot con su dispositivo de programación. Si los estudiantes son totalmente nuevos en Sphero Edu, pídeles que completen la Introducción a Sphero Edu y/o Mini 1: Dibujar croquet en lienzo antes de comenzar esta lección.

La entrada de rumbo 0 debería hacer que Mini avance recto. Sin embargo, la dirección de mira de Mini define qué dirección es realmente la directa. Para ayudar a los estudiantes con los títulos, recuérdelos que alineen sus cuerpos, sus dispositivos de programación y la dirección de mira de los robots.



Desarrollo de habilidades: el bloque de rollo

El bloque roll es el bloque más utilizado en la biblioteca Sphero Block. Aprender cómo funciona te permitirá programar tu Mini para que ruede donde quieras. El bloque de rodadura tiene tres áreas, llamadas entradas, para los números que controlan cómo rueda Mini:

- **Velocidad** : qué tan rápido rodará tu Mini, donde 0 es el más lento y 255 es el más rápido
- **Título** : la dirección en la que rodará tu Mini, donde 0 es hacia adelante, 90 es hacia la derecha, 180 es hacia atrás y 270 es hacia la izquierda
- **Duración** : la cantidad de tiempo que tu Mini rodará, en segundos

HACER:

1. Experimente cambiando las entradas en el bloque de balanceo.
2. Seleccione Iniciar para ejecutar el programa y ver qué sucede.
3. Intentar hacer **mini** Haz lo siguiente:
 - rueda lo más rápido posible directamente hacia la derecha
 - avance en línea recta y deténgase después de 3 segundos
 - rodar hacia atrás a la menor velocidad posible

Consejo para el educador:

Es posible que sus alumnos sean nuevos en el concepto de entradas. Las entradas son datos o información que le dicen a una computadora u otros sistemas tecnológicos cómo ejecutar un comando. Los estudiantes pueden cambiar las entradas con los controles deslizantes integrados en el bloque de desplazamiento. También pueden tocar/hacer clic/seleccionar el número directamente e ingresar el número deseado con el teclado.

Familiarizarse con la entrada de encabezados suele ser la entrada más desafiante para los estudiantes. Utilice un transportador, una rosa de los vientos y/o un reloj para reforzar la direccionalidad de diferentes rumbos en grados de 0 a 359. Muchos estudiantes están familiarizados con los giros de 180 o 360. Úselos como puntos de referencia para entender otros títulos.

Explique que antes de que los estudiantes puedan programar su robot para obtener dos números, deben poder usar un **bloque rodante** para programar mini para rodar a un número.

4. Discuta las tres entradas en el **bloque rodante** .
 - **Título** es la dirección mini apunta entre 0° y 360°. 0° es la dirección de mira, mirando directamente en dirección opuesta a la luz trasera azul.
 - **Velocidad** es un número entre 0 y 255 que controla qué tan rápido rueda BOLT.
 - **Duración** es el tiempo en segundos para una tirada.

Desarrollo de habilidades: bloques de retardo

1. Coloca una mochila delante de tu Mini.
2. Arrastra un segundo bloque rodante al programa.
3. Ingrese entradas en los dos bloques de rollo para hacer un mini tejido alrededor de la mochila.



CONVERSAR:

- ¿Qué notas sobre el giro que dio tu Mini?
- ¿Se detuvo por completo antes de comenzar el segundo lanzamiento? ¿O dio una vuelta redondeada?

A veces querrás que tu robot se detenga por completo entre bloques rodantes. Para que esto suceda, puede colocar un bloque de retraso (de la categoría de control de la biblioteca de bloques) entre sus bloques de desplazamiento y configurar la entrada de retraso en un segundo.

HACER:

- Coloque un bloque de retraso de la categoría de controles en la biblioteca de bloques entre los dos bloques de rollo.
- Establezca la entrada en el bloque de retardo en 1 segundo.
- Seleccione Iniciar para ejecutar el programa.

CONVERSAR:

¿Qué efecto tuvo el bloque de retraso en la ejecución de su programa por parte de Minis?



Desafía el tejido de la mochila

1. Prepara tres mochilas.
2. Programe Mini para recorrer las mochilas con bloques de avance y retraso.
3. Pruebe y depure su programa hasta que su mini se mueva con precisión a través de la carrera de obstáculos.
4. Para divertirte más, busca otro amigo y corre a través de la carrera de obstáculos. ¿Quién puede terminar la carrera de obstáculos más rápido?

**Consejo para el educador:**

Los estudiantes pueden usar bloques rodantes adicionales para programar un camino único a través de los obstáculos.

Anime a los estudiantes que resuelvan rápidamente el desafío a programar el camino de Minis a través de la carrera de obstáculos.



Lección 2: El tejido de la mochila

Bienvenido a la Mini Lección 2, su introducción a la programación de su Sphero Mini en el Canvas de Bloques. ¡Prepárate para aprender todo lo que hay que saber sobre el bloque rodante!

Objetivos de aprendizaje

1. Puedo controlar el movimiento de Mini con las tres entradas de un bloque de balanceo: velocidad, rumbo y duración.
2. Puedo programar a Mini para que ruede entre obstáculos y los rodee.

Configuración



Código QR del programa



Exploración:

Abra un programa de bloques. Cada bloque es una instrucción de comando de programación para su robot. Arrastre un bloque de la biblioteca de bloques y ajústelo al bloque del programa de inicio. Ahora seleccione Iniciar y vea qué sucede.



Desarrollo de habilidades:

El bloque roll es el bloque más utilizado en la biblioteca Sphero Block. Aprender cómo funciona te permitirá programar tu Mini para que ruede donde quieras. El bloque de rodadura tiene tres áreas, llamadas entradas, para números que controlan cómo rueda Mini.



Desafío:

Programe Mini para recorrer las mochilas con bloques de avance y retraso.



Desafío extendido:

Para divertirse más, busca otro amigo y corre a través de la carrera de obstáculos. ¿Quién puede terminar la carrera de obstáculos más rápido?



Lección 3: Bola de datos

Descripción general

Su Mini puede hacer mucho más que rodar, encender luces y reproducir sonidos a través de su dispositivo de programación. La Unidad de Medición Inercial (IMU) de tu Mini es un sensor que constantemente recopila datos sobre su entorno y puedes usar estos datos en tus programas de bloques.

Nivel de programación

Bloque inicial

Objetivos de aprendizaje

1. Puedo leer datos del sensor IMU de Mini.
2. Puedo usar eventos de programación para hacer que un Mini robot responda a los datos del sensor.

Vocabulario

- **Asincrónico:** Programación que ejecuta múltiples bloques o comandos al mismo tiempo.
- **Sincrónico:** Programación que ejecuta un bloque o comando a la vez.

Estándares

- Descomponer (descomponer) los problemas en subproblemas más pequeños y manejables para facilitar el proceso de desarrollo del programa.
- Utilice datos para resaltar o proponer relaciones de causa y efecto, predecir resultados o comunicar una idea.

Conexiones de contenido

- matemáticas
- Ciencias de la Computación

Materiales

- Mini robots Sphero completamente cargados
- Dispositivos de programación con la aplicación Sphero Edu instalada

Preparación

- Planifique áreas de trabajo para los estudiantes alrededor del espacio de aprendizaje. Cada grupo de estudiantes necesitará un área de aproximadamente 5' por 5' de espacio.

Código QR de actividad





Exploración - Sensores de Mini

¿Alguna vez te has preguntado cómo sabe un teléfono o una tableta en qué dirección está arriba?
¿O qué tal cómo un avión permanece en el cielo?

La respuesta es uno de los sensores más comunes e importantes de toda la robótica: la Unidad de Medición de Inercia o IMU.

¡Tu Sphero Mini tiene una IMU que te permite hacer más con tu robot! En esta actividad, aprenda a leer los valores de los sensores Minis y luego úselos para mejorar su programación.

LEER:

La IMU dentro de su Mini en realidad se compone de dos sensores separados:

- El acelerómetro detecta fuerzas, incluida la gravedad, y la aceleración de su robot.
- El giroscopio mide la posición exacta de su robot y la tasa de cambio de esas posiciones.

La IMU recopila datos a lo largo de tres ejes: x, y y z. También puedes pensar en estos ejes como cabeceo, balanceo y guiñada.

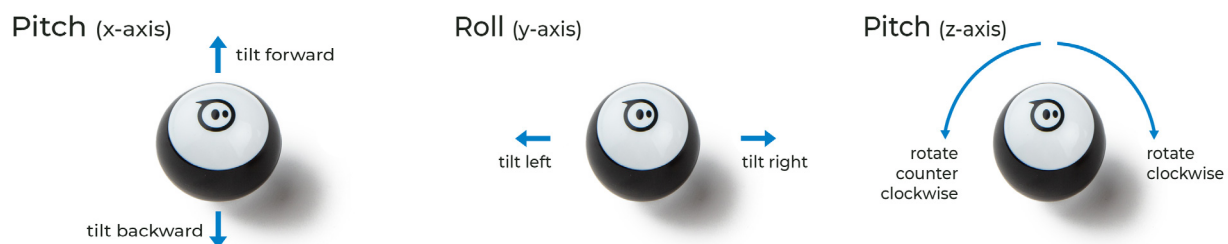
- El tono mide la inclinación hacia adelante y hacia atrás alrededor del eje x.
- Roll mide la inclinación hacia la izquierda y hacia la derecha alrededor del eje y.
- La guiñada mide la rotación en sentido horario y antihorario alrededor del eje z.

CONVERSAR:

¿Comparar y contrastar los diferentes ejes en relación con las dimensiones del movimiento de un avión?

- ¿Qué significa si el avión se inclina hacia adelante?
- ¿Qué pasa si el avión gira hacia la izquierda?
- ¿Cuándo cambiaría la posición de guiñada de un avión?

Comprender la IMU puede resultar confuso, ¡pero Mini puede ayudar! ¡Continúe con el siguiente paso para obtener más información!

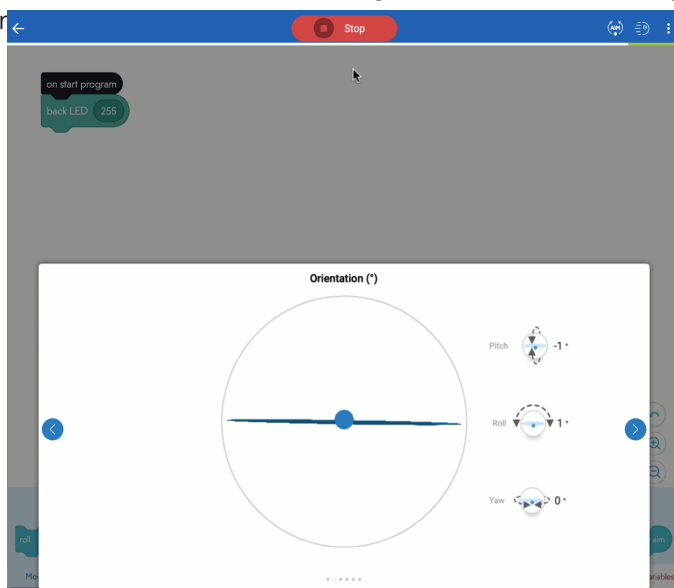




Desarrollo de habilidades: datos de sensores

1. **Conecte su Mini a su dispositivo de programación** y abra el programa asociado a esta actividad.
2. **Apunta tu mini** de modo que la luz trasera apunte hacia usted.
3. Seleccionar **Comenzar**.
4. Selecciona el **Pestaña Datos del sensor en vivo** en la parte inferior de la aplicación.
5. Desplácese por el **orientación, giroscopio**, y **acelerómetro** Vea y juegue con Mini para ver el efecto en los datos del sensor.

Revise la información en la imagen a continuación mientras explora. Preste mucha atención a los datos totales del acelerómetro. Un G es la fuerza por unidad de masa que experimentan los objetos en la superficie terrestre. En reposo, el acelerómetro total del Minis marca 1 G. Aumenta más de 1 cuando se lanza al aire. En caída libre, cuando acelera con la gravedad, indica 0. Luego, el acelerómetro total vuelve a dispararse al aterrizar.



Consejo para el educador:

Otra forma de obtener los valores del Mini acelerómetro y giroscopio es hacer que el programa los lea durante la ejecución del programa. Este [imagen](#) muestra cómo leer datos del sensor en voz alta con un bloque de cadena de construcción.

- Arrastra un bloque hablado para leer el valor.
- Coloque un bloque de cadena de construcción dentro del bloque de habla.
- Edite el bloque de cadena de compilación para incluir una entrada numérica.
- Arrastre el sensor que desee dentro del bloque de cadena de construcción.

Coloque los bloques dentro de un bloque de bucle permanente para seguir leyendo el valor del sensor una y otra vez.

Después de que se haya ejecutado un programa, aún puede ver los datos del sensor.

- Selecciona los tres puntos en la esquina superior derecha de la aplicación.
- Seleccione Datos del sensor.
- Vea los gráficos en la aplicación o descargue los datos como un archivo CSV.



Desarrollo de Habilidades - Programación de Eventos

Puede programar Mini para que responda a los valores de cada uno de estos sensores con eventos de programación. Los eventos, que se encuentran en la categoría Eventos de la biblioteca de bloques, son activadores basados en valores de sensores que hacen que otros bloques se ejecuten en el programa.

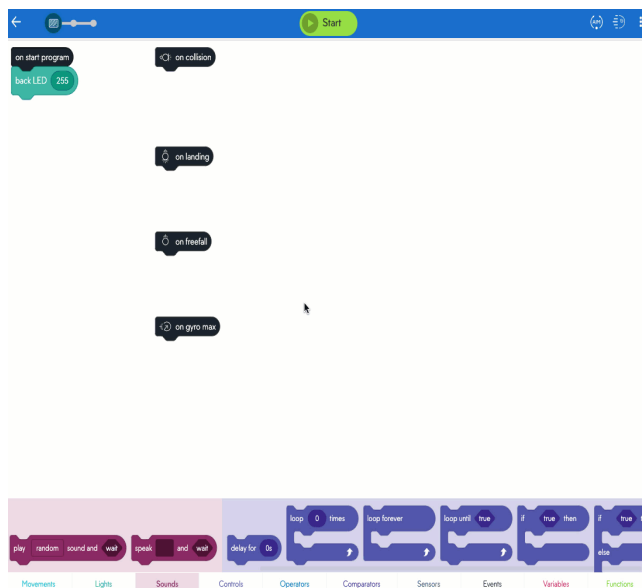
Mini tiene cuatro bloques de eventos:

- **en colisión** Utiliza el sensor de acelerómetro total para detectar cuando tu Mini ha chocado contra un objeto.
- **al aterrizar** Detecta cuando tu Mini ha chocado con un objeto después de estar en caída libre.
- **en caída libre** detecta cuando tu Mini está en caída libre, lo que significa que el acelerómetro total indica 0 Gs.
- **en giroscopio** max detecta cuando tu mini gira en el sentido de las agujas del reloj o en el sentido contrario a las agujas del reloj a un ritmo rápido.

HACER:

1. Arrastra un **reproducir bloque de sonido** y mantenlo configurado para reproducir un **sonido aleatorio**.
2. **Adjuntar** el bloque al **en caso de colisión**.
3. **Prueba el programa**. ¿Puedes hacer que Mini reproduzca el sonido?
4. **Verifique los datos del sensor**. ¿Puedes encontrar el punto de datos que desencadenó el evento?

Repita los pasos 2-4 para el **al aterrizar**, **en caída libre**, y **en eventos gyro max**. NOTA: No deje caer el Mini más de 18 pulgadas. Si se deja caer desde una altura mayor, la carcasa podría desprenderse y el Mini podría romperse.



Consejo para el educador:

Los estudiantes pueden notar que solo pueden arrastrar uno de cada bloque de eventos al lienzo de programación. Esto se debe a que Mini sólo puede ejecutar un conjunto de bloques a la vez. En otras palabras, sólo puede responder a cada evento de una única manera.

Los estudiantes pueden notar que un evento como el aterrizaje no se activa si se reproduce el efecto de sonido de la caída libre porque cuando se inició el evento de aterrizaje, el dispositivo de programación todavía estaba reproduciendo un efecto de sonido. Si los estudiantes tienen dificultades para notar cada evento, intente tener solo uno de estos al mismo tiempo.



Desafío

Ahora que sabes algunas cosas sobre la programación de eventos, es hora de ser creativo y divertirse.

En este paso, adjuntarás bloques de luz y sonido a cada uno de los eventos de programación para mostrar cómo cada evento hace sentir a tu Mini. Qué luces y sonidos muestran cómo se siente tu Mini cuando:

- choca con otro objeto?
- durante una caída libre?
- cuando aterriza en el suelo?
- ¿O cuando se hace girar lo más rápido posible?

HACER:

- Adjunte uno o más bloques a cada uno de los cuatro eventos.
- Pruebe y depure su programa.
- ¡Continúe trabajando hasta que su programa se ejecute como se esperaba!

Nota: Es posible que tenga problemas para ejecutar los bloques de aterrizaje. Si los bloques en caída libre tardan demasiado, seguirán ejecutándose cuando se produzca el aterrizaje. ¡Haz que tus bloques en caída libre sean bloques rápidos, como encender el LED, para que los bloques en aterrizaje puedan funcionar!

NOTA: No deje caer el Mini más de 18 pulgadas. Si se deja caer desde una altura mayor, la carcasa puede saltar y el Mini puede romperse. Ahora que BOLT conoce la situación del terreno, invite a los estudiantes a programar su camino a casa.



Compartir

HACER:

- **Encuentra un compañero de clase o profesor .**
- **Mostrarles** lo que creaste.
- **Pregúntales** qué les gusta y qué creen que mejoraría su programa.
- **Revisa tu programa y realizar mejoras .**

CONVERSAR:

- ¿Qué es un sensor IMU y qué datos recopila sobre tu Mini robot?
- ¿Cómo se pueden utilizar los eventos de programación para hacer más con tu Mini?
- ¿Qué te gustaría programar para que haga tu Mini a continuación?



Lección 3: Bola de datos

Su Mini puede hacer mucho más que rodar, encender luces y reproducir sonidos a través de su dispositivo de programación. La Unidad de Medición Inercial (IMU) de tu Mini es un sensor que constantemente recopila datos sobre su entorno y puedes usar estos datos en tus programas de bloques.

Objetivos de aprendizaje

1. Puedo leer datos del sensor IMU de Mini.
2. Puedo usar eventos de programación para hacer que un Mini robot responda a los datos del sensor.

Código QR del programa



Exploración:

¿Alguna vez te has preguntado cómo sabe un teléfono o una tableta en qué dirección está arriba?
¿O qué tal cómo un avión permanece en el cielo?

La respuesta es uno de los sensores más comunes e importantes de toda la robótica: la Unidad de Medición de Inercia o IMU.

¡Tu Sphero Mini tiene una IMU que te permite hacer más con tu robot! En esta actividad, aprenda a leer los valores de los sensores Minis y luego úselos para mejorar su programación.



Desarrollo de habilidades:

La IMU dentro de su Mini en realidad se compone de dos sensores separados: el acelerómetro y el giroscopio. La IMU recopila datos a lo largo de tres ejes: x, y y z. Piutche, roll y guiñada. ¿Comparar y contrastar los diferentes ejes en relación con las dimensiones del movimiento de un avión?

Utilice la pestaña de datos del sensor en vivo en su programa y desplácese **orientación, giroscopio**, y **acelerómetro**. Vea y juegue con Mini para ver el efecto en los datos del sensor. Ahora programe Mini para que responda a los valores de cada uno de estos sensores con eventos de programación.

Eventos, que se encuentra en la categoría Eventos en la biblioteca de bloques



Desafío:

Adjuntar **luz** y **bloques de sonido** a cada uno de los eventos de programación para mostrar cómo cada evento hace sentir a tu Mini? Qué luces y sonidos muestran cómo se siente tu Mini cuando:

- choca con otro objeto?
- durante una caída libre?
- cuando aterriza en el suelo?
- ¿O cuando se hace girar lo más rápido posible?



Lección 4: La parte superior que cambia de color

Descripción general

Su Sphero Mini puede hacer aún más cuando agrega diferentes bloques de programación. Usar condicionales es una forma de hacer que tu Mini se comporte de manera diferente en determinadas situaciones. ¡Coge tu Mini y comencemos a programar!

Nivel de programación

Bloque Intermedio

Objetivos de aprendizaje

1. Puedo controlar las luces de Mini con una declaración condicional.
2. Puedo programar Mini usando bloques if else

Vocabulario

- **Declaraciones condicionales :**
Declaraciones que se ejecutan sólo bajo ciertas condiciones. Por ejemplo, si la condición A es verdadera, realice esta acción específica.

-

Estándares

- Desarrollar programas con secuencias y bucles simples, para expresar ideas o abordar un problema.
- Descomponer (descomponer) los pasos necesarios para resolver un problema en una secuencia precisa de instrucciones.

Conexiones de contenido

- Ciencias de la Computación

Materiales

- Robots mini robots Sphero completamente cargados
- Dispositivos de programación con la aplicación Sphero Edu instalada

Preparación

- Planifique áreas de trabajo para los estudiantes alrededor del espacio de aprendizaje. Cada grupo de estudiantes necesitará un área de aproximadamente 5' por 5' de espacio.

Código QR del programa





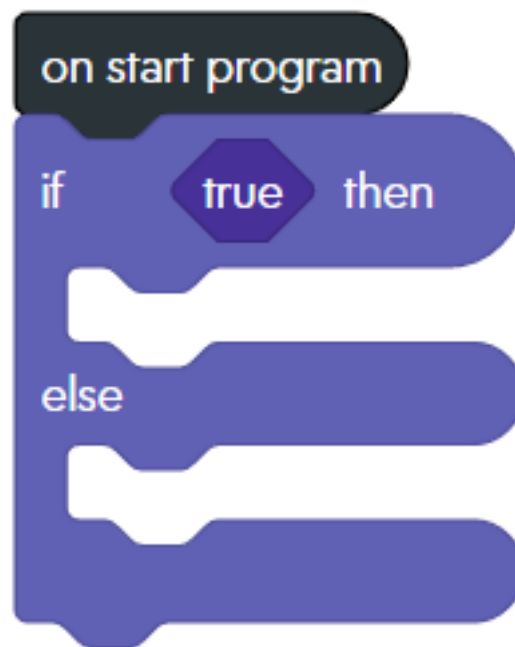
Exploración

Las declaraciones condicionales se encuentran en lenguaje, matemáticas y programación. Por lo general, están estructurados en la forma Si _____, entonces _____. Por ejemplo: si está lloviendo, entonces necesito un paraguas o si afuera está oscuro, entonces es hora de dormir.

El principio básico o declaraciones condicionales es que:

- **SI** la primera parte de la afirmación es verdadera,
- **ENTONCES** la segunda parte de la afirmación se vuelve verdadera.

Los condicionales son un elemento básico en cualquier lenguaje de programación. Permiten que su Mini tenga diferentes comportamientos dependiendo de si una afirmación que usted crea es verdadera o falsa. Mientras que en el ejemplo anterior involucraron el lenguaje If / Then, en programación se les conoce como **Si no** bloques o declaraciones.



Consejo para el educador:

Esta es una oportunidad para plantear causa y efecto. Las declaraciones if else son una extensión de este tema, con vocabulario diferente. Si (causa), entonces (efecto). Si no (causa), entonces (efecto).



Desarrollo de habilidades

En la actividad de la Lección 3, se le presentó el sensor Minis IMU y el concepto de cabeceo, balanceo y guiñada. En este paso, usaremos su conocimiento sobre esto para experimentar con declaraciones condicionales.

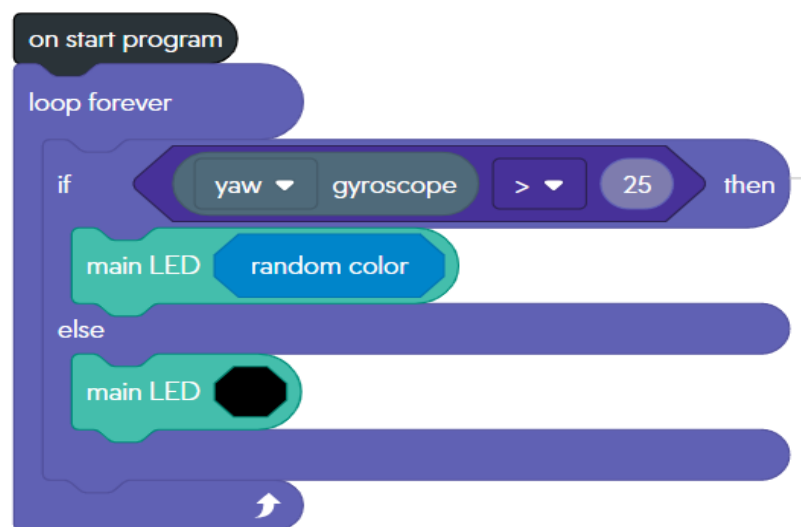
1. Abra el programa asociado con esta actividad.
2. Mira el código adjunto al **al iniciar el bloque de programa** . ¿Qué crees que hará el programa cuando lo ejecutes?
3. Inicie el programa y haga girar su Mini como si fuera un trompo en una dirección. ¿Qué hizo Mini?
4. Ahora gira tu Mini en la dirección opuesta. ¿Qué hizo Mini esta vez?

Si descompusimos la declaración condicional de este programa en una oración, se traduciría aproximadamente a lo siguiente:

- Si el giroscopio de orientación es superior a 25 (grados por segundo), gire el LED principal a un color aleatorio. Si la guiñada es cualquier otra cosa, apague el LED principal.

CONVERSAR:

- ¿Puedes relacionar cada parte de la oración con un bloque de código en la imagen a continuación?



Consejo para €

Cuando los estudiantes estén haciendo girar Mini, deben colocarlo en el suelo y girarlo con la mano como si fuera una peonza. Considere mostrarles a los estudiantes cómo hacer esto antes de ejecutar sus programas.

Si los estudiantes tienen problemas para girar Mini, pueden quitarle el caparazón y debería girar mucho más fácilmente para que el programa lo note.

Es posible que sus alumnos sean nuevos en el concepto de condicionales. Intentar relacionar el código con una oración es una forma de ayudar a los estudiantes a comprender qué hace cada bloque y por qué lo necesitan.



Desarrollo de habilidades: más bloques

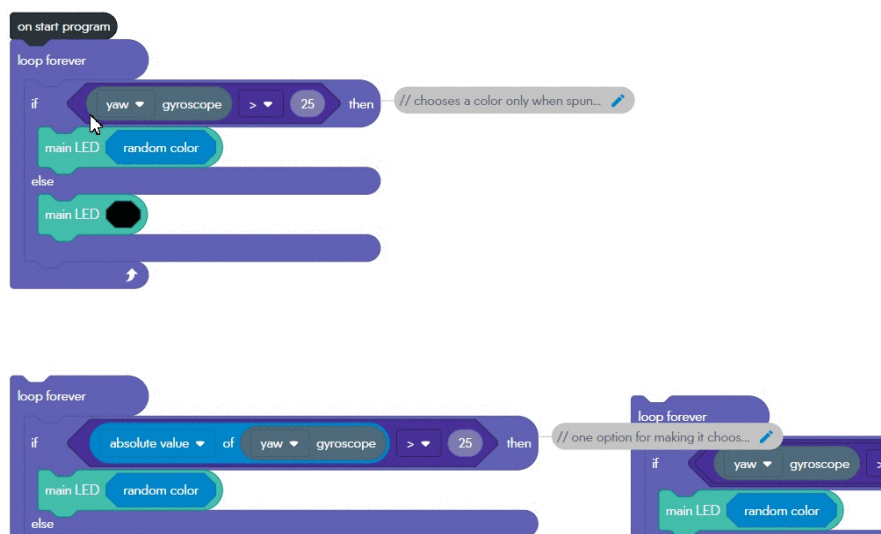


BLOQUE INTERMEDIO

¿Qué pasa si quieres que tu Mini cambie de color cuando lo giras en ambas direcciones?

¡Puede hacer esto de dos maneras! Solo podemos mirar el valor de guiñada, y no si es positivo o negativo, o podemos agregar a nuestro condicional para que el color cambie cuando la guiñada sea menor que -25.

1. Arrastre sus bloques fuera del bloque de inicio y reemplácelos con uno de los bloques de código a continuación.
2. Ejecute el programa y gire su Mini en ambas direcciones, ¡y observe cómo cambia de color! (Retire la carcasa si tiene problemas para girar Mini)
3. Ahora escriba una declaración condicional que describa el ejemplo de código que eligió.
4. Repita los pasos 1 a 3 para el otro ejemplo de código. Nota: Al programar el símbolo || significa O



Consejo para el educador:

Ejemplo para el primer bloque de código: si el valor absoluto de la guiñada es mayor que 25, cambie el LED principal a un color aleatorio. Dependiendo de la edad y las habilidades matemáticas de sus estudiantes, es posible que necesiten una introducción rápida al valor absoluto. El valor absoluto de un número es su distancia al cero, siempre representado como valor positivo. El valor absoluto de -25 es 25. El valor absoluto de 25 también es 25. Ejemplo para el segundo bloque de código: si el valor absoluto de la guiñada es mayor que 25 o menor que -25, cambie el LED principal a un color aleatorio. Nota: Este bloque de código utiliza el operador lógico, O, representado en el código por el símbolo ||.

Para los estudiantes que necesitan un desafío adicional, existe otra forma de modificar la declaración condicional para que el programa funcione de la misma manera. Si el giroscopio de orientación no es igual (!=) a cero, entonces el LED principal cambia a un color aleatorio. Desafíelos a crear esta declaración en su código.



Desafío

Ahora que ha utilizado declaraciones condicionales, ¡es hora de escribir las suyas propias!

5. Arrastre todo su código fuera del **al iniciar el programa**, y hacia un lado
6. Colocar un nuevo **bucle para siempre** bajo la **al iniciar el programa**
7. Colocar un nuevo **si no, bloquear** dentro de **bucle para siempre** bloquear
8. Elija una de las oraciones a continuación y modifique su programa para que el **SI MÁS** El bloque refleje la oración.
 - **SI** la guñada es inferior a 20, **ENTONCES** el LED principal es negro, **DEMÁS** el LED principal cambiará a un color aleatorio.
 - **SI** Mini está al revés, **ENTONCES** cambia a un color aleatorio, **DEMÁS** El LED principal es negro.
9. Pruebe su programa para ver si Mini se comporta como espera.
10. Si su programa no funciona como se esperaba, modifíquelo y vuelva a intentarlo.

Consejo para el educador:

La segunda oración pretende ser más desafiante para los estudiantes que tienen más experiencia con la programación o con Mini. Para estudiantes menos experimentados, puede recomendarles que elijan la primera oración.



Desafío extendido

Ahora da un paso más. ¡Escribe tu propia declaración condicional! Usando las dos oraciones del último paso como ejemplo, escribe tu propia declaración condicional sobre el comportamiento de Minis y un programa que coincida.

Considere usar **desvanecimiento** o **luces estroboscópicas** y **sonidos** en tu programa

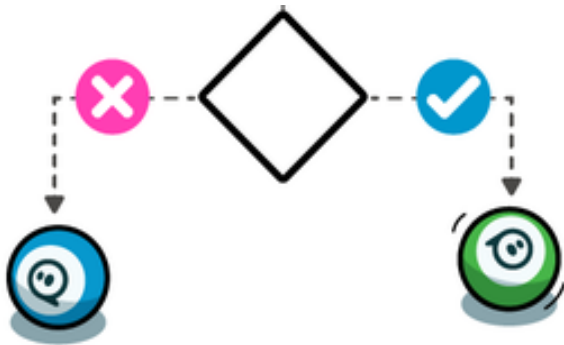


Lección 4: La parte superior que cambia de color

Su Sphero Mini puede hacer aún más cuando agrega diferentes bloques de programación. Usar condicionales es una forma de hacer que tu Mini se comporte de manera diferente en determinadas situaciones. ¡Coge tu Mini y comencemos a programar!

Objetivos de aprendizaje

1. Puedo controlar las luces de Mini con una declaración condicional.
2. Puedo programar Mini usando bloques if else



Código



Exploración:

Las declaraciones condicionales se encuentran en lenguaje, matemáticas y programación. Por lo general, están estructurados en la forma Si _____, entonces _____. Por ejemplo: si está lloviendo, entonces necesito un paraguas o si afuera está oscuro, entonces es hora de dormir.



Desarrollo de habilidades:

Abra el programa asociado con esta actividad. Mira el código adjunto al **al iniciar el bloque de programa**. ¿Qué crees que hará el programa cuando lo ejecutes? Inicie el programa y haga girar su Mini como si fuera un trompo en una dirección. ¿Qué hizo Mini? Ahora gira tu Mini en la dirección opuesta. ¿Qué hizo Mini esta vez?

¿Qué pasa si quieres que tu Mini cambie de color cuando lo giras en ambas direcciones?

¡Puede hacer esto de dos maneras! Solo podemos mirar el valor de guiñada, y no si es positivo o negativo, o podemos agregar a nuestro condicional para que el color cambie cuando la guiñada sea menor que -25.



Desafío:

Ahora que ha utilizado declaraciones condicionales, ¡es hora de escribir las suyas propias! Siga la sección "Desafío" de esta actividad utilizando el enlace del código QR



Desafío extendido:

Ahora da un paso más. ¡Escribe tu propia declaración condicional! Usando las dos oraciones del último paso como ejemplo, escribe tu propia declaración condicional sobre el comportamiento de Minis y un programa que coincida.



Lección 5: Laberinto Caos

Descripción general

Programa Sphero mini para navegar por tu propio laberinto original. Para completar este desafío, debes recopilar datos sobre la mejor ruta a través del laberinto y descubrir cómo crear un programa para que Sphero pueda navegar con éxito a través del caos.

Nivel de programación

Bloque Intermedio

Objetivos de aprendizaje

1. Puedo evaluar un laberinto para encontrar la solución más rápida y eficiente.
2. Puedo crear un programa para navegar por Sphero a través de un laberinto usando Bloques y el Lienzo de Bloques.

Estándares

- Medir y estimar longitudes en unidades estándar.
- Resolver problemas que involucren medición y estimación de intervalos de tiempo, volúmenes de líquidos y masas de objetos.

Conexiones de contenido

- Ciencias de la Computación
- Matemáticas

Materiales

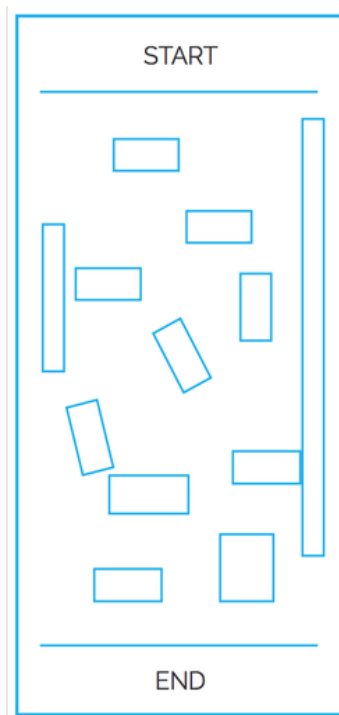
- Mini robots Sphero completamente cargados
- Programar dispositivos con la aplicación Sphero Edu instalada
- Materiales para trazar un laberinto.

- Cronómetro o cronómetro

Preparación

- Planifique áreas de trabajo para los estudiantes alrededor del espacio de aprendizaje. Cada grupo de estudiantes necesitará al menos un área de 5' por 5' de espacio.
- Haga que los estudiantes creen un laberinto usando varios materiales que se encuentran en su salón de clases.

Configuración de ejemplo



Código QR del programa

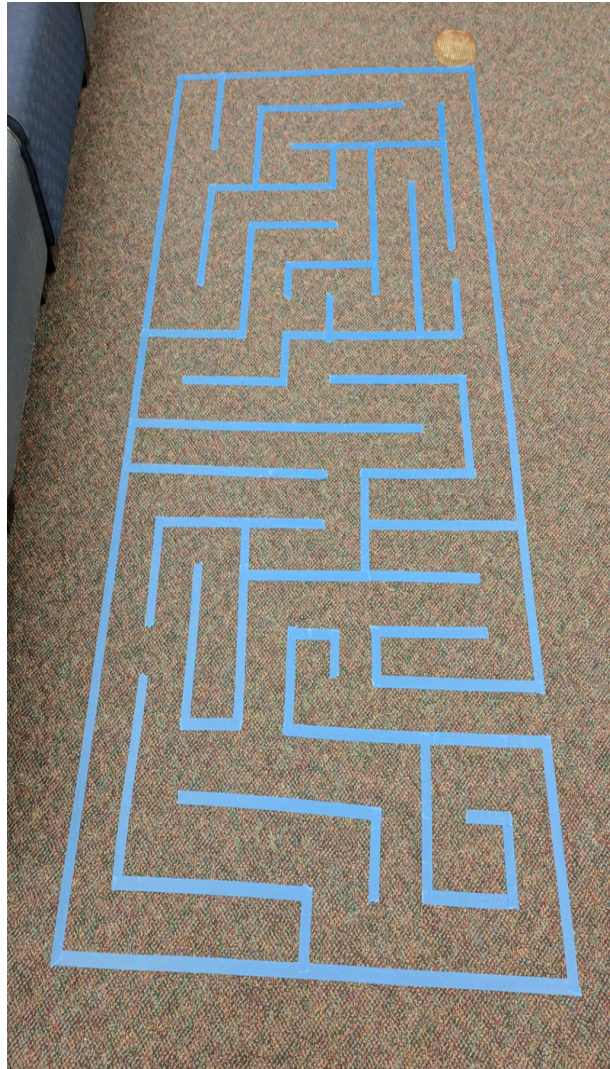




Exploración

¿Qué se necesita para que Sphero navegue por un laberinto? Definitivamente es más que simplemente conducirlo. En esta actividad recopilars datos y programars el camino más eficiente a través de un laberinto.

A continuación se muestra un vídeo y una imagen con ejemplos de posibles laberintos.



Consejo para el educador:

Introducción: (15 minutos)

Divida a los estudiantes en grupos de 3 o 4 estudiantes y presente Sphero y este Desafío.

Sugerimos que los estudiantes asuman roles específicos para los Pasos 3 y 5 (Recopilación de datos y programación), como la Parte 3: medidor, registrador, dibujante/cajonera, calculadora, Parte 5: lector (lee las instrucciones escritas en la parte 3 al programador), traductor, programador y probador



Desarrollo de habilidades

Tu laberinto puede ser tan simple como una cinta azul en el suelo. No se necesita mucho para crear un camino para que Sphero navegue. Si tienes espacio, toma algunos objetos de la habitación para crear paredes y un camino para Sphero. Cosas como libros, piezas de Lego, cajas, zapatos o incluso papeleras funcionarán.

A medida que creas este laberinto, agrega algunos obstáculos para hacer el camino más difícil. Los pasajes estrechos también pueden ser complicados.

Consejo para el educador:

Edificio Laberinto: (15 minutos)

Si incluye múltiples caminos que Sphero podría usar para atravesar el laberinto, los equipos deberán determinar qué ruta será la más fácil de navegar o si pueden programar a Sphero para que se mueva más rápido.

Como clase, construyan el laberinto por el que los estudiantes programarán sus Spheros para atravesarlo.

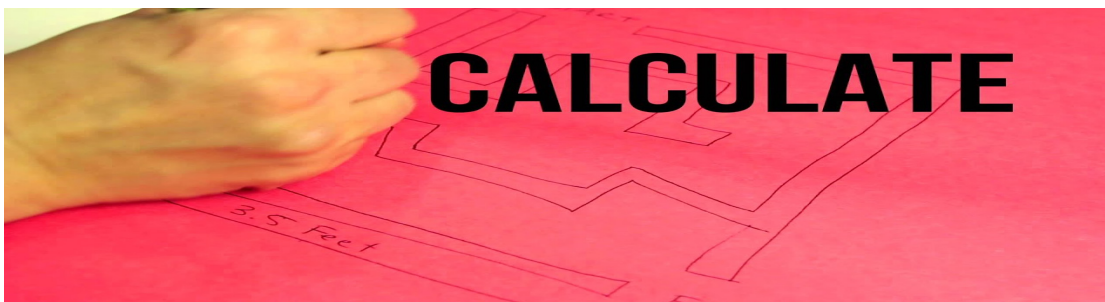


Opcionalmente, puedes construir el laberinto antes de que entren los estudiantes (lo que acelerará la actividad).

Desarrollo de habilidades: ¿hasta dónde viaja Sphero?

Para navegar por el laberinto de forma rápida y eficiente, querrás determinar qué tan lejos viaja Sphero. Necesitará saber qué distancia recorre Sphero en un período de tiempo determinado (por ejemplo, en 1 segundo).

Crea un programa que te ayude a tomar esta medida. (SUGERENCIA: comience con un simple bloqueo rodante, establezca la velocidad y establezca la duración en 1 segundo).



Consejo para el educador:

Para determinar la velocidad de Sphero, programe Sphero para que se mueva hacia adelante durante 1 segundo a toda velocidad. Luego mide la distancia que viajó Sphero.



Desarrollo de habilidades: navegando por el laberinto

Utilizando los datos que recopiló en el Paso 3, escriba las instrucciones de lo que desea que haga Sphero. Dibuja el laberinto en una hoja de papel, determina el camino de Sphero y toma medidas de distancias y ángulos. Algo como esto:

- Seguir recto 40 cm.
- Detener
- Gira a la izquierda 90 grados
- Seguir recto 20 cm.

Al lado de cada instrucción, escriba qué bloque necesitaría para completar esa instrucción. Cuando hayas terminado, tendrás instrucciones paso a paso para que Sphero se mueva por el laberinto.



Desafío

¡Es hora de poner toda la planificación a trabajar y comenzar a programar!

Pruébe su programa sobre la marcha. Modifique el programa según sea necesario. Recuerde que está programando para lograr rapidez y eficiencia.

SUGERENCIA: Si descubre que Sphero no gira de manera constante o tan nítida como le gustaría, explore el bloque Delay en "Controles".

Reflexión

Escribe tus reflexiones sobre esta actividad.

- ¿Qué funcionó y qué no?
- ¿Cómo harías las cosas de manera diferente en el futuro?
- ¿Qué ruta funcionó mejor?
- ¿Cuál fue la parte más complicada del laberinto?
- ¿Cuál fue la parte más desafiante de la actividad?



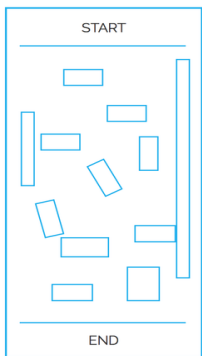
Lección 5: Laberinto Caos

Programa Sphero mini para navegar por tu propio laberinto original. Para completar este desafío, debes recopilar datos sobre la mejor ruta a través del laberinto y descubrir cómo crear un programa para que Sphero pueda navegar con éxito a través del caos.

Objetivos de aprendizaje

- 1. Puedo evaluar un laberinto para encontrar la solución más rápida y eficiente.
- 2. Puedo crear un programa para navegar por Sphero a través de un laberinto usando Bloques y el Lienzo de Bloques.

Ejemplo de configuración



Código QR del programa



Exploración:

¿Qué se necesita para que Sphero navegue por un laberinto? Definitivamente es más que simplemente conducirlo. En esta actividad recopilarás datos y programarás el camino más eficiente a través de un laberinto.



Desarrollo de habilidades:

Tu laberinto puede ser tan simple como una cinta azul en el suelo. No se necesita mucho para crear un camino para que Sphero navegue. Si tienes espacio, toma algunos objetos de la habitación para crear paredes y un camino para Sphero.

Para navegar por el laberinto de forma rápida y eficiente, querrás determinar qué tan lejos viaja Sphero. Necesitará saber qué distancia recorre Sphero en un período de tiempo determinado (por ejemplo, en 1 segundo).



Desafío:

¡Es hora de poner toda la planificación a trabajar y comenzar a programar!

Pruébe su programa sobre la marcha. Modifique el programa según sea necesario. Recuerde que está programando para lograr rapidez y eficiencia.

Reflexión:

- ¿Qué funcionó y qué no?
- ¿Cómo harías las cosas de manera diferente en el futuro?
- ¿Qué ruta funcionó mejor?
- ¿Cuál fue la parte más complicada del laberinto?
- ¿Cuál fue la parte más desafiante de la actividad?



Lección 6: **Evita al Minotauro**

Descripción general

Nuestro héroe, Sphero mini, queda atrapado por una criatura de la mitología griega, ¡el Minotauro! En la mitología griega, muchos héroes usaban su cerebro en lugar de su fuerza física para burlar a sus enemigos. Junto con tu equipo, crea un programa para mostrar cómo Sphero mini podría burlar al Minotauro y escapar del laberinto.

Nivel de programación

Bloque avanzado

Objetivos de aprendizaje

1. Aprenderé a controlar Sphero usando un código preciso.
2. Ilustraré cómo decidir qué bloques hacer que Sphero alcance un objetivo.
3. Crearé y ejecutaré un programa que llevará a Sphero por un camino cambiante.
4. Analizaré el programa con hechos que lo respalden; reflexionar sobre el aprendizaje.

Estándares

- Desarrollar y utilizar un modelo para describir el papel de la gravedad en los movimientos dentro de las galaxias y el sistema solar.
- Evaluar soluciones de diseño competitivas utilizando un proceso sistemático para determinar qué tan bien cumplen con los criterios y limitaciones del problema.
- Analice datos de pruebas para determinar similitudes y diferencias entre varias soluciones de diseño para identificar las mejores características de cada una que se pueden combinar en una nueva solución para cumplir mejor con los criterios de éxito.

Conexiones de contenido

- Ciencias Sociales
- Ciencias de la Computación

Materiales

- Mini robots Sphero completamente cargados
- Programar dispositivos con la aplicación Sphero Edu instalada
- Recorte de cartón que representa al Minotauro.
- Algo para crear un laberinto simple.

Preparación

- Planifique áreas de trabajo para los estudiantes alrededor del espacio de aprendizaje. Cada grupo de estudiantes necesitará al menos un área de 5' por 5' de espacio.
- Haga que los estudiantes creen un laberinto usando varios materiales que se encuentran en su salón de clases.

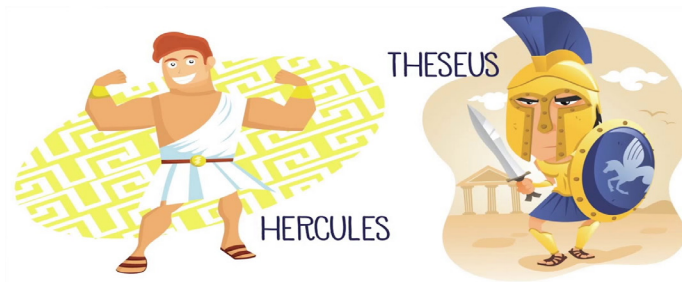
Código QR del programa





Exploración

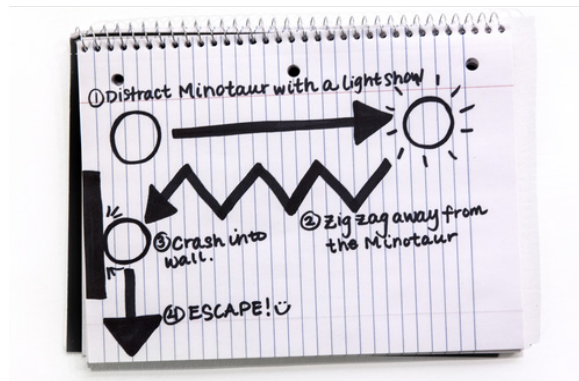
¡Oh, no! ¡Sphero mini está atrapado en el laberinto del Minotauro! Mira este vídeo para descubrir cómo puedes ayudarlo a escapar: <https://youtu.be/1pnmvGkuH7o>



Para escapar, tú y tus compañeros de equipo debéis diseñar un programa que haga que Sphero mini realice algunos movimientos sofisticados para confundir al Minotauro.

Empiece por pensar en lo que quiere que haga su Sphero para lograrlo.

Planifíquelo escribiendo o dibujando sus ideas, así:



¿Cómo debería ser el laberinto de los Minotauros? Dibuje algunas ideas en papel y evalúe los materiales disponibles para usted.

Utilizando el lienzo de bloques, experimente con el código para hacer que Sphero mini se mueva de diferentes maneras, ilumine luces, etc. Cuanto más creativo, mejor. ¡Tu objetivo es confundir al Minotauro para poder escapar!

Escribe cada bloque que usaste y lo que hizo que Sphero mini hiciera; usarás esta lista para construir tu programa para escapar del Minotauro.



Desarrollo de habilidades

Usando los materiales disponibles, construye el laberinto alrededor de la lógica de tu programa de bloques en el que Sphero mini se encontrará con el Minotauro.

¡Mira este vídeo como una posibilidad para inspirar tu escapada! <https://youtu.be/N7ihLhL4RN8>



Desafío

Usando el lienzo de bloques, desarrolle código que mueva a Sphero de manera errática pero decidida; recuerde, ¡todavía tiene que salir del laberinto!

Pruebe su programa.

¿Puedes mejorarlo? ¿Cómo?



Desafío - Consulta tu programa

Ejecute su programa. ¿Esfero escapó del Minotauro? ¿Qué cambios puedes hacer?

- Escribe tus reflexiones sobre esta actividad y discútelas con la clase:
- ¿Qué funcionó y qué no?
- ¿Cómo harías las cosas de manera diferente en el futuro?
- ¿Qué código funcionó mejor?
- ¿Cuál fue la parte más difícil o más divertida del desafío?



Lección 6: Evita al Minotauro

Nuestro héroe, Sphero mini, queda atrapado por una criatura de la mitología griega, ¡el Minotauro! En la mitología griega, muchos héroes usaban su cerebro en lugar de su fuerza física para burlar a sus enemigos. Junto con tu equipo, crea un programa para mostrar cómo Sphero mini podría burlar al Minotauro y escapar del laberinto.

Objetivos de aprendizaje

1. Aprenderé a controlar Sphero usando un código preciso.
2. Ilustraré cómo decidir qué bloques hacer que Sphero alcance un objetivo.
3. Crearé y ejecutaré un programa que llevará a Sphero por un camino cambiante.
4. Analizaré el programa con hechos que lo respalden; reflexionar sobre el aprendizaje.



Código QR del programa



Exploración:

¡Sphero mini está atrapado en el laberinto del Minotauro! Tú y tus compañeros de equipo debéis diseñar un programa que haga que Sphero mini realice algunos movimientos sofisticados para confundir al Minotauro y escapar del laberinto.



Desarrollo de habilidades:

Usando los materiales disponibles, construye el laberinto alrededor de la lógica de tu programa de bloques en el que Sphero mini se encontrará con el Minotauro.



Desafío:

Usando el lienzo de bloques, desarrolle código que mueva a Sphero de manera errática pero decidida; recuerde, todavía tiene que salir del laberinto.



Desafío extendido:

Ejecute su programa. ¿Esfero escapó del Minotauro? ¿Qué cambios puedes hacer?



Lección 7: ¡Qué personaje!

Descripción general

Sphero mini tiene una historia que contar. O tal vez sí. De cualquier manera, ¡vamos a contarla!

Crea un programa para animar a Sphero mini para que actúe y cuente una historia para que todos la escuchen. Aprovecha todo lo que Sphero tiene para ofrecer, desde movimientos simples hasta luces y sonidos.

Nivel de programación

Bloque avanzado

Objetivos de aprendizaje

1. Puedo escribir o volver a contar una historia para que Sphero la cuente.
2. Puedo crear un guión gráfico para guiar mi trabajo.
3. Puedo programar animaciones originales para mejorar mi historia.

Estándares

- Diseñar, modificar y seguir algoritmos simples que involucren secuencias de pasos, bifurcaciones e iteraciones (repeticiones).
- Definir y descomponer problemas del mundo real teniendo en cuenta los requisitos funcionales y las limitaciones económicas, ambientales, sociales, técnicas y de usabilidad.
- Implementar y modificar programas con interfaces de usuario que impliquen bifurcación, iteración y funciones en un lenguaje de programación de propósito general.

Conexiones de contenido

- Ciencias de la Computación
- Artes del lenguaje

Materiales

- Mini robots Sphero completamente cargados

- Dispositivos de programación con la aplicación Sphero Edu instalada
- Materiales artesanales para construir personajes.
- Software de edición de video y pantalla verde (opcional)



Código QR del programa





Exploración

La personificación es cuando le das una característica humana a algo no humano. Utilizarás esta técnica literaria para ayudarte a contar tu historia a través de Sphero mini.

Le darás rasgos y características que normalmente se le darían a un personaje humano. Empieza a pensar en cómo harás esto con un robot y Sphero Edu.



Exploración - Rasgos de carácter

Ya sea que esté volviendo a contar una historia o escribiendo algo nuevo, considere cómo programará Sphero mini para mostrar emociones, comunicarse con los demás y mostrar su personalidad.

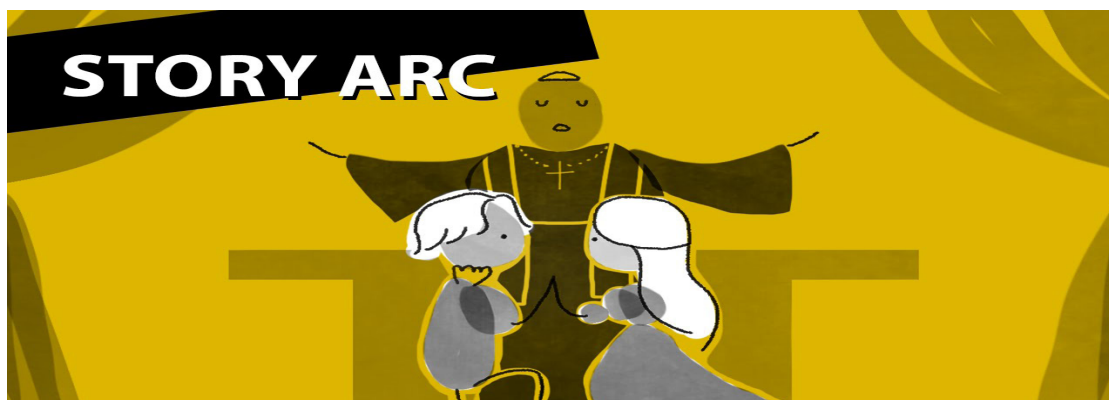
Haga una lluvia de ideas sobre algunas ideas en algún papel. Para guiarlo, considere algunas de las siguientes preguntas:

- ¿Cómo se puede utilizar Sphero? **LED principal** y bloques de programación asociados?
- ¿Cómo puedes usar **Sonido** y **Hablar** bloques?
- A lo largo de una historia, los personajes expresan estados de ánimo y emociones. ¿Cómo puedes programarlos en la historia?
- ¿Cómo puedes mostrar la personalidad del personaje de Sphero de una manera que atraiga a tu audiencia?



Exploración - Arco argumental

Toda historia básica tiene un principio, un desarrollo y un final. Suelen seguir un arco narrativo predecible. El siguiente vídeo explica qué es un arco narrativo utilizando el ejemplo clásico de Romeo y Julieta. Enlace: <https://youtu.be/BaPR0y89s6Y>



Mientras eliges tu historia (nueva o contada), piensa en los diferentes diálogos y acciones que sucederán durante cada parte de tu historia. Estas cosas se pueden transmitir utilizando las luces, movimientos y/o sonidos de Sphero.



Desarrollo de habilidades

¡Es hora de trabajar en tu historia!

Cuando comiences a desarrollar (o volver a contar) la trama y el desarrollo del personaje, utiliza un guión gráfico como ayuda para planificar. Puede utilizar el guión gráfico para diagramar momentos importantes y planificar animaciones, movimientos, luces y sonidos específicos. Utilice el storyboard.pdf adjunto o tome un trozo de papel y dóblelo en nueve partes iguales. enlace: <https://sphero-media-sphero-prod.s3.amazonaws.com/cwist/picturesteps/99/cf/storyboard.pdf>

Tome una fotografía de su(s) guión(s) gráfico(s) y adjúntela al final de este paso.

Storyboard



1	2	3
4	5	6
7	8	9

Consejo para el educador:

Puede optar por imprimir el storyboard.pdf o simplemente mostrarles a los estudiantes cómo dividir una hoja de papel en nueve partes iguales.



Desarrollo de habilidades - Érase una vez un Sphero mini

Has planificado tu historia y cómo quieres contarla. Ahora es el momento de involucrar a Sphero mini.

Recuerda que puedes programar el LED principal de Sphero, su movimiento y añadir sonidos y habla. Utilizando lo que sabes sobre los bloques y una combinación de bloques, comienza a crear los estados de ánimo, las emociones y las acciones de Sphero.

Si necesitas un poco de ayuda, echa un vistazo al programa adjunto a esta actividad. Incluye algunas funciones de narración que puedes usar o consultar en busca de inspiración.

Enlace al programa: <https://edu.sphero.com/program/960407/edit>

Si has elegido decorar o construir alrededor de Sphero mini con materiales para manualidades, asegúrate de practicar tus programas con estos. La masa agregada afectará el modo en que se mueve Sphero mini.



Desafío

¡Escuchemos tu historia! Prepara Sphero mini para el gran espectáculo.

Practica tu historia varias veces para reducir el tiempo. Una vez que estés listo, busca un compañero para filmarlo todo. Comienza tu programa y cuenta tu historia.

Como desafío adicional, considere usar una pantalla verde. Con una simple edición de vídeo, puedes llevar a tu audiencia a tierras lejanas. Pregúntale a tu maestro sobre esto.

Consejo para el educador:

Si está usando iPads y tiene acceso a iMovie, debería ser bastante sencillo usar el efecto de pantalla verde para colocar a los estudiantes y a Sphero en lugares lejanos mientras su historia se graba en video.



Lección 7: ¡Qué personaje!

Sphero mini tiene una historia que contar. O tal vez sí. De cualquier manera, ¡vamos a contarla!

Crea un programa para animar a Sphero mini para que actúe y cuente una historia para que todos la escuchen. Aprovecha todo lo que Sphero tiene para ofrecer, desde movimientos simples hasta luces y sonidos.

Objetivos de aprendizaje

1. Puedo escribir o volver a contar una historia para que Sphero la cuente.
2. Puedo crear un guión gráfico para guiar mi trabajo.
3. Puedo programar animaciones originales para mejorar mi historia.



Código QR del programa



Exploración:

Le darás a Sphero mini rasgos y características que normalmente se le darían a un personaje humano. Empieza a pensar en cómo harás esto con un robot y Sphero Edu.



Desarrollo de habilidades:

A medida que comiences a desarrollar la trama y el desarrollo de los personajes, utiliza un guión gráfico como ayuda para planificar. Puede utilizar el guión gráfico para diagramar momentos importantes y planificar animaciones, movimientos, luces y sonidos específicos.



Desafío:

¡Escuchemos tu historia! Prepara Sphero mini para el gran espectáculo.

Practica tu historia varias veces para reducir el tiempo. Una vez que estés listo, busca un compañero para filmarlo todo. Comienza tu programa y cuenta tu historia.



Desafío extendido:

Como desafío adicional, considere usar una pantalla verde. Con una simple edición de vídeo, puedes llevar a tu audiencia a tierras lejanas. Pregúntale a tu profesor sobre esto en el programa principal.



Lección 8: Funciones divertidas y divertidas

Descripción general

En esta actividad, utilizará Sphero para construir una calculadora. En este proceso, aprenderá cómo crear e invocar funciones.

Nivel de programación

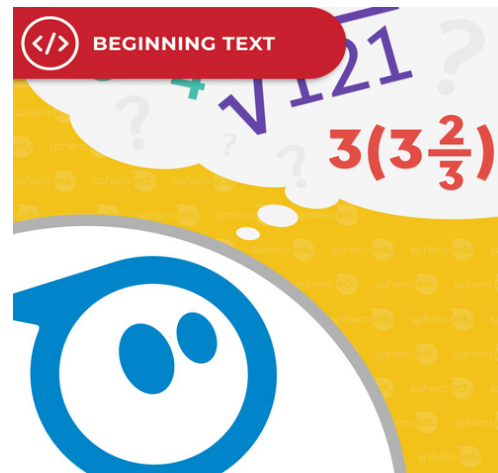
Texto inicial

Objetivos de aprendizaje

1. Puedo practicar los estándares de la industria, incluyendo:
 - Mejoramiento
 - Depuración
 - Pseudocódigo
2. Puedo definir y utilizar los fundamentos de la informática, incluidos:
 - Tipos de datos
 - Asíncrono / Sincrónico
 - API
 - Argumentos de parámetros
3. Puedo usar JavaScript para crear un programa.

Materiales

- Mini robots Sphero completamente cargados
- Dispositivos de programación con la aplicación Sphero Edu instalada



Código QR del programa



Vocabulario

- **javascript** : El lenguaje de programación de texto que ejecuta el entorno de programación de bloques de Sphero.
- **Lienzo de texto** : El editor de la aplicación Sphero Edu para crear programas JavaScript
- **Declaración** : Instrucciones de programación en JavaScript
- **Depurar** : Localizar y solucionar errores en un programa

Estándares

- Probar y depurar (identificar y corregir errores) un programa o algoritmo para garantizar que se ejecute según lo previsto.
- Probar y perfeccionar sistemáticamente programas utilizando una variedad de casos de prueba.

Conexiones de contenido

- Ciencias de la Computación



Exploración

Probablemente hayas notado que cada programa de Sphero comienza con la siguiente función: **función asíncrona `startProgram()` { // realizar la operación }**. En esta actividad, aprenderá a utilizar funciones que reciben entradas y realizan operaciones que devuelven un valor.

Piensa en todas las diferentes **tipos de datos** has visto hasta ahora. Defina lo siguiente:

- Booleano
- Cadena
- Número

Una función puede aceptar estos tipos de datos (y otros) como entradas en una función. Los parámetros van dentro del paréntesis. **()** en una función. Las operaciones ocurren dentro de las llaves. **{ }**. Necesitas incluir un punto y coma. **;** al final de su declaración de devolución.

Si tuvieras que escribir una calculadora sencilla, se vería así:

```
1 function add(num1, num2) {  
2     return num1 + num2;  
3 }
```

A **invocar** la función (hacerla ejecutar), la llamarías así. Función **argumentos** son los valores reales pasados (y recibidos por) la función.

```
1 add(1, 3) // returns 4  
2 add(5, 7) // returns 8
```

Es posible que hayas notado que la función anterior no incluye la palabra **asíncrono**. JavaScript es **síncronico**, lo que significa que las acciones pueden ocurrir al mismo tiempo. Solo necesita incluir la palabra `async` cuando crea una función que necesita esperar a que finalice una operación antes de continuar. En el caso de Sphero, esto suele implicar interacciones con el robot.

Las interacciones con el robot son asincrónicas porque un **API** se llama. API significa Interfaz de programación de aplicaciones y se refiere a métodos de comunicación entre diferentes componentes de software (en este caso, su dispositivo que ejecuta la aplicación Sphero Edu y el robot).

Notarás que esas interacciones, como moverse o hablar, incluyen la palabra **esperar**. El operador de espera le dice al compilador (el programa que convierte JavaScript en código de máquina) que el método asíncrono no puede continuar más allá de ese punto hasta que se complete el proceso asíncrono esperado.



Desarrollo de habilidades

Responda las siguientes preguntas antes de continuar:

1. ¿Qué es una función asíncrona? ¿Cuándo necesitas usarlos?
2. ¿Qué es una API?
3. ¿Qué significa **declarar** ¿Una función?
4. ¿Qué significa **invocar** ¿Una función?

Crea una función llamada **sustraer** e invócalo. Haz un diagrama que identifique y nombre las siguientes partes:

- Los parametros
- Los argumentos

Adjunte una imagen o captura de pantalla de su diagrama para que la vea su maestro.



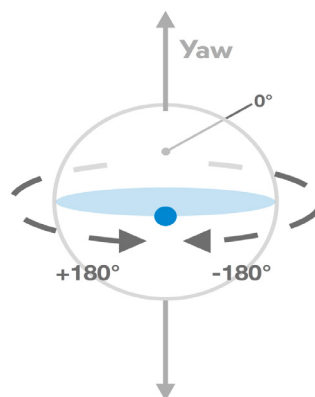
Desafío

Crearás un contador que suma y resta números dependiendo de en qué dirección gires el Sphero mini. Mire el código en la página siguiente. La función **obtenerOrientación().guiñada** recupera el ángulo de giro, de -180 a 180.

Dependiendo del ángulo al que mire su Sphero, sumará, restará o hablará el conteo actual. Puedes realizar un seguimiento de la dirección en la que giras Sphero dibujándolo en un papel (usando la imagen a continuación) o mirando los colores mientras giras el robot (verde = sumar, rojo = restar, azul = decir el conteo actual).

Utilice los comentarios en el código siguiente como guía. Puede utilizar las funciones que creó en el último paso.

PISTA : Considere cómo funciona una calculadora. Querrá utilizar count como uno de los argumentos en sus funciones.



```
1 async function startProgram() {
2   let count = 0;
3
4   // declare your add function here
5   // declare your subtract function here
6
7   while (true) {
8     setStabilization(false);
9     setBackLed(255);
10    const direction = getOrientation().yaw;
11
12    if (direction > 45) {
13      await strobe({ r: 0, g: 255, b: 0 }, 1, 1);
14      count = // invoke your add function here
15
16    } else if (direction < -45) {
17      await strobe({ r: 255, g: 0, b: 0 }, 1, 1);
18      count = // invoke your subtract function here
19
20    } else {
21      await strobe({ r: 0, g: 0, b: 255 }, 1, 1);
22      await speak("the total is " + count.toString());
23    }
24    await delay(1);
25  }
26 }
```



Desafío - Reflexión

Reflexiona sobre lo que aprendiste respondiendo las siguientes preguntas:

- ¿Cuál es la diferencia entre declarar e invocar una función?
- ¿Por qué fue necesario establecer la estabilización en falso en este código?
- Si quisieras mover el código para cambiar el color de Spheros a las funciones de sumar o restar, necesitarías convertirlas en funciones asíncronas. Explicar por qué.

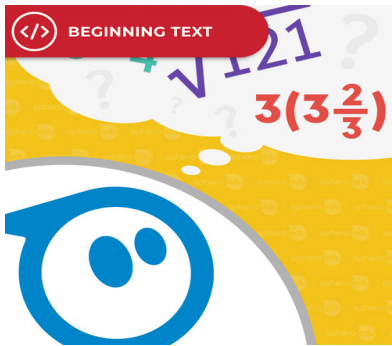


Lección 8: Funciones divertidas y divertidas

En esta actividad, utilizará Sphero para construir una calculadora. En este proceso, aprenderá cómo crear e invocar funciones.

Objetivos de aprendizaje

1. Puedo definir y utilizar los fundamentos de la informática.
2. Puedo usar JavaScript para crear un programa.



Código QR del programa



Exploración:

En esta actividad, aprenderá a utilizar funciones que reciben entradas y realizan operaciones que devuelven un valor.



Desarrollo de habilidades:

Crea una función llamada **sustraer** e invocarla. Haz un diagrama que identifique y nombre las siguientes partes:

- Los parametros
- Los argumentos



Desafío:

Crea un contador que suma y resta números dependiendo de en qué dirección gires el Sphero mini.



Reflexión del desafío

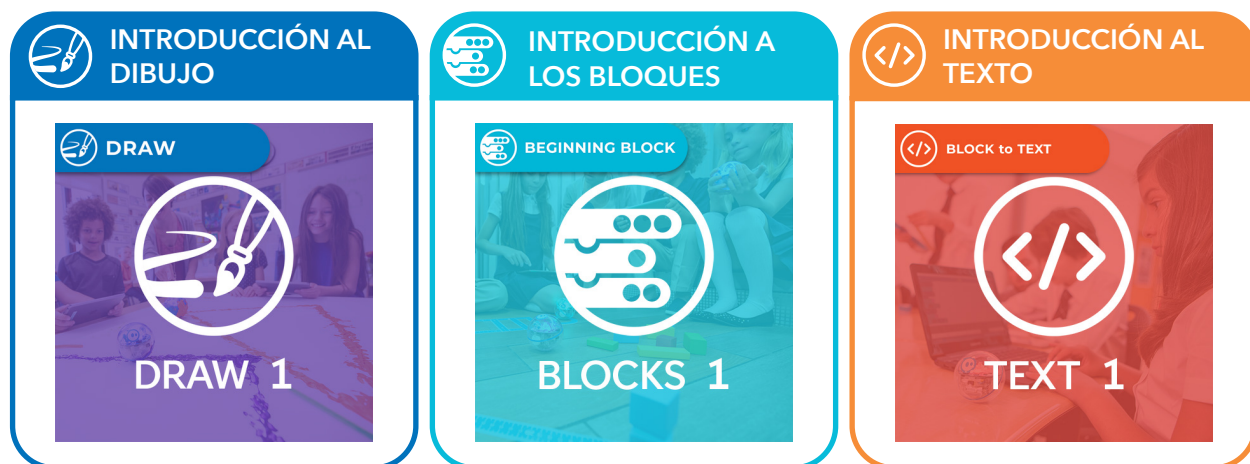
Reflexiona sobre lo que aprendiste respondiendo las siguientes preguntas:

- ¿Cuál es la diferencia entre declarar e invocar una función?
- ¿Por qué fue necesario establecer la estabilización en falso en este código?
- Si quisieras mover el código para cambiar el color de Spheros a las funciones de sumar o restar, necesitarías convertirlas en funciones asíncronas. Explicar por qué.

Ir más lejos

Después de que sus alumnos hayan completado algunas o todas las actividades de clase de esta guía. Estarán listos para seguir aprendiendo con otras actividades en la aplicación Sphero Edu, tanto creadas por Sphero como por nuestra comunidad de educadores.

Para continuar enseñando los fundamentos de la programación, considera las siguientes actividades adecuadas al nivel de tus alumnos.



Ofertas curriculares adicionales

Tapete de código Sphero

Un Code Mat ofrece una forma sencilla y accesible de aprender codificación basada en bloques, principios matemáticos básicos y resolución colaborativa de problemas con el conjunto de mini robots de su clase. Cada Code Mat de dos caras viene con tres juegos de 10 tarjetas de codificación de doble cara que brindan lecciones de codificación prácticas y guiadas. Elija el tapete Sphero City y Golf Code o el Sphero Space y Soccer Code Mat para energizar su salón de clases Sphero.

<https://sphero.cc/CODEMAT>



Recursos

Sphero está inspirando a los creadores del mañana. No podríamos estar más entusiasmados con el futuro de la educación y el papel que todos desempeñamos. Para obtener más información sobre Sphero y participar en nuestra comunidad, puede encontrar enlaces a recursos adicionales a continuación.

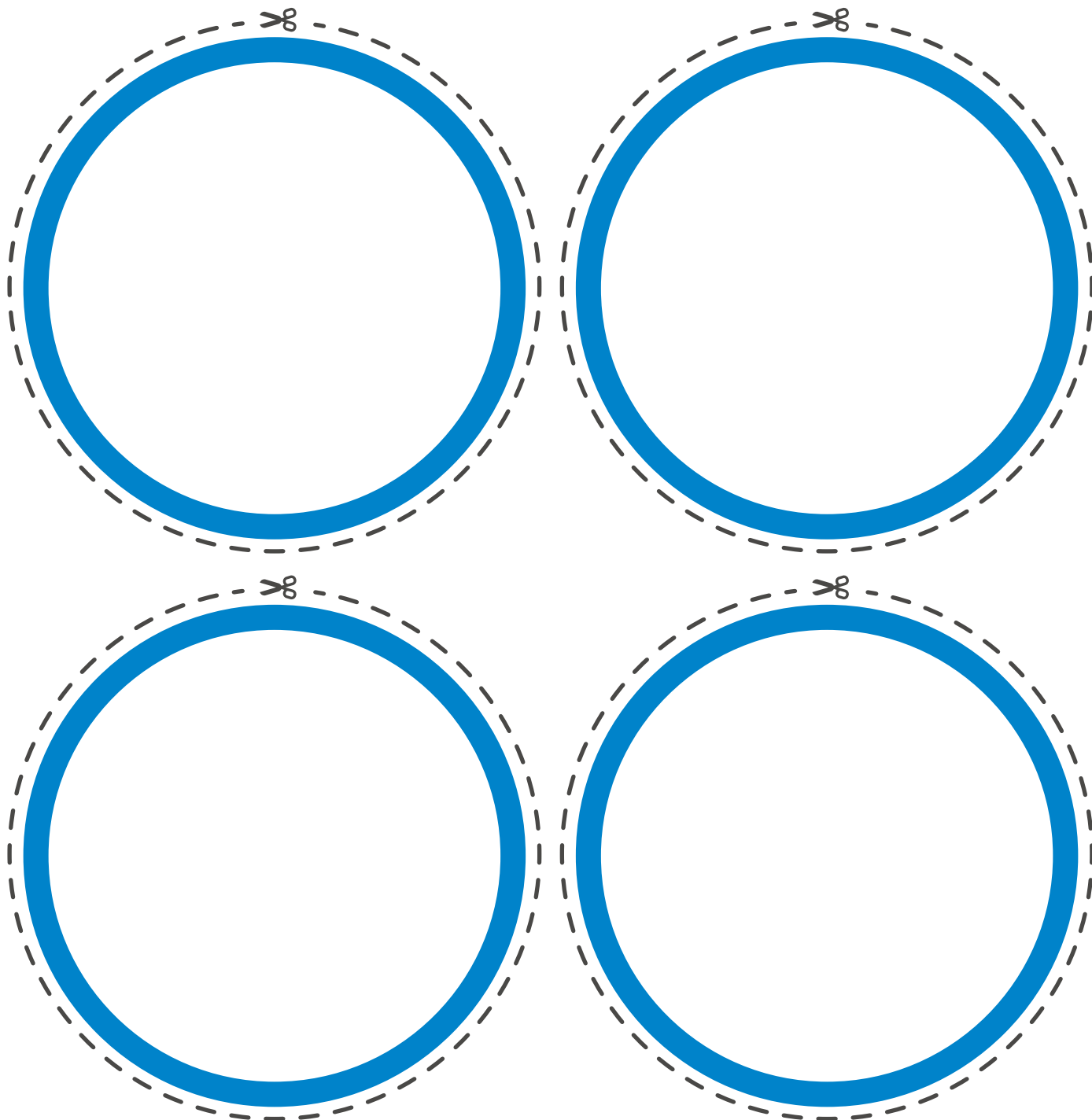
- Blog de Esfero: sphero.com/blogs/news
- Wiki de programación de Sphero: <https://sphero.cc/JS-help>
- Apoyo: soporte.sphero.com
- Recursos de autoservicio: sphero.com/pages/sphero-self-service-resources
- Contáctenos: sphero.com/pages/contact-us
- Facebook: facebook.com/GoSphero
- Gorjeo: twitter.com/sphero
- Instagram: instagram.com/sphero

Seguridad y Privacidad

Estamos dedicados a garantizar que mini y la aplicación Sphero Edu sean seguros de usar. Nuestros esfuerzos de seguridad y privacidad incluyen pruebas de terceros, auditorías anuales y un programa de recompensas por errores. Cumplimos con COPPA, hemos firmado el Compromiso de Privacidad Estudiantil y publicamos todos nuestros acuerdos de prácticas de privacidad en línea. Visita <https://www.sphero.com/privacy> para más información.

Objetivos

Haga copias de esta página según sea necesario para las actividades de la guía. Corte los objetivos a lo largo de las líneas de puntos y etiquételos según las instrucciones de la actividad individual. Acceda a una versión digital para imprimir en esphero.cc/targets.



Glosario

Dibujar lienzo : El editor de la aplicación Sphero Edu para crear programas dibujando líneas

Ejecutar : Para ejecutar o completar un bloque o programa

Bloque de lienzo : El editor en la aplicación Sphero Edu para crear programas de bloques

Aporte : Lugar para agregar valores numéricos o de texto en un bloque de programación o declaración de texto

Velocidad : Entrada para programar qué tan rápido (o lento) se moverá un mini

Título : Entrada para programar la dirección en la que apunta un mini durante o antes de un giro, entre 0° y 360°

Duración : Entrada para programar la cantidad de tiempo que mini ejecutará un movimiento como un giro o un giro en segundos

Asincrónico : Programación que ejecuta múltiples bloques o comandos al mismo tiempo

Sincrónico : Programación que ejecuta un bloque o comando a la vez

Variable : Un fragmento de información, como un valor numérico o una cadena, que puede cambiar durante un programa.

Cadena : Frases que combinan letras, números y/u otros caracteres

Control : Una herramienta de programación utilizada para hacer que ciertas partes de un programa se repitan o ejecuten bajo ciertas condiciones.

Bucle : Un control de programación que realiza una acción repitiéndose una y otra vez.

Función : Grupos de bloques que se pueden llamar o reutilizar a lo largo de un programa.

Sensor : Información recopilada por mini, incluida la ubicación, orientación y aceleración.

Orientación de guiñada : El ángulo de giro (giro) de mini de -180° a 180°

Entero : Conjunto de números enteros positivos y negativos y 0

Pseudocódigo : Código escrito informalmente que simplemente pretende trazar y describir el código real.

Comentario : Notas que los programadores dejan en su código que explican su pensamiento; no interfiere con el código

javascript : El lenguaje de programación de texto que ejecuta el entorno de programación de bloques de Sphero.

Lienzo de texto : El editor de la aplicación Sphero Edu para crear programas JavaScript

Declaración : Instrucciones de programación en JavaScript

Depurar : Localizar y solucionar errores en un programa.

Biblioteca de bloques

Movimientos



Rollo: Combina rumbo, velocidad y duración para hacer que el robot ruede.



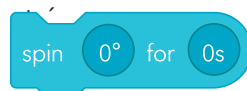
Detener: Establece la velocidad en 0 para detener el robot cuando se utiliza un bloque de velocidad.



Velocidad: Establece la velocidad objetivo del robot de -255 a 255. La velocidad positiva es hacia adelante, la velocidad negativa es hacia atrás y 0 se detiene.



Título: Establece la dirección objetivo en la que rueda el robot. Suponiendo que el robot apunta con la luz trasera azul hacia usted, entonces 0° es hacia adelante, 90° es hacia la derecha, 270° hacia la izquierda y 180° hacia



Girar: Hace girar el robot una cantidad determinada de grados a lo largo del tiempo, donde 360° es una rotación única.



Motor crudo: Este comando desactiva la estabilización y hará que el robot salte cuando ambos motores estén configurados a máxima potencia. Controla la energía eléctrica enviada a los motores izquierdo y derecho de forma independiente, de -255 a 255, durante unos segundos.



Estabilización: Enciende o apaga el sistema de estabilización. La estabilización está activada de forma predeterminada para mantener el robot en posición vertical.



Restablecer objetivo: Restablece el ángulo de calibración del rumbo (objetivo) para utilizar la dirección frontal actual del robot como 0°.

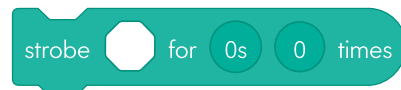
Luces



LED principal: Cambia el color de los LED principales. Configúrelo usando la rueda de color y el control deslizante de brillo, o los valores RGB exactos (rojo, verde, azul), de 0 a 255.



Desteñir: Cambia los LED principales de un color a otro, durante unos segundos. Este comando funciona como un comando independiente y no se puede usar para atenuar los LED principales mientras se usa un comando de giro.

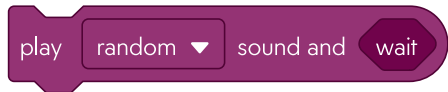


Estroboscópico: Hace parpadear los LED principales durante un período de segundos (que incluye luces encendidas y apagadas) y un conteo de ciclos. Un período corto producirá un parpadeo rápido mientras que un período largo producirá un parpadeo

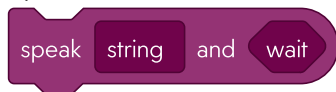


LED trasero: Cambia el color del LED trasero. Enciende el LED trasero. Esto se limita únicamente al azul. Puede ajustar el brillo tocando el número.

sonidos



Sonido: Reproduce un sonido desde su dispositivo de programación (no desde el robot) que seleccione de la lista. Alterna la opción Aleatorizar para generar un sonido aleatorio o selecciona uno de la biblioteca de Sphero.

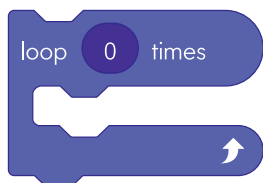


Hablar: Habla cadenas (números y palabras) desde tu dispositivo de programación. Escribe lo que quieres que diga tu robot.

Control S



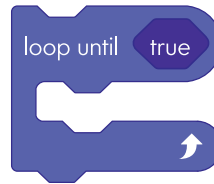
Demora: Retrasa la ejecución del siguiente bloque durante unos segundos.



Bucle: Repite los bloques dentro del número de bucles especificados, también conocido como bucle "for".



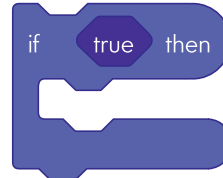
Bucle para siempre: Repite los bloques para siempre, también conocido como bucle "mientras 1". Puede utilizar esto para evaluar constantemente las condiciones.



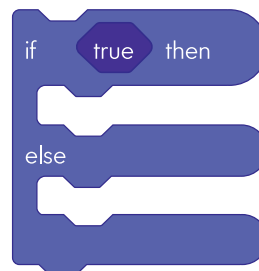
Bucle hasta: Repite los bloques internos hasta que se cumple una condición, también conocido como bucle "while". Arrastre cualquier comparador al campo "verdadero" para crear una condición. Utilice "y" y "o" para combinar o excluir condiciones, que deben agregarse antes que otros comparadores.



Salir del programa: Detiene todo el código y finaliza el programa; lo mismo que presionar el botón Detener.

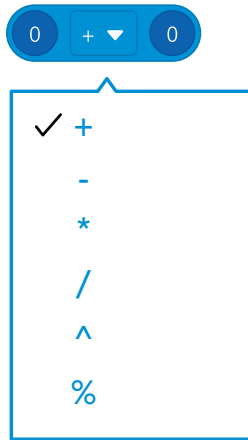


Si entonces: Llama a los bloques "si" si la condición dada es verdadera. Utilice "y" y "o" para combinar o excluir condiciones, que deben agregarse antes que otros comparadores.



Si entonces más: Llama a los bloques "si" si la condición dada es verdadera; de lo contrario llama a los bloques "else". Utilice "y" y "o" para combinar o excluir condiciones, que deben agregarse antes que otros comparadores.

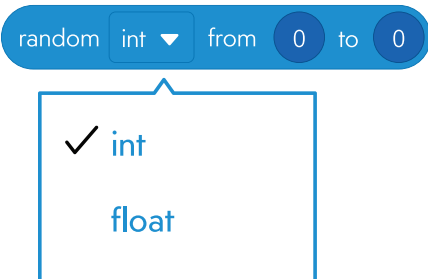
Operadores



Operadores Básicos: Agrega una acción matemática para calcular o evaluar un conjunto de datos o un valor dado.



Cadena de compilación: Combina múltiples valores en una sola cadena. Esos valores pueden ser números (variables, parámetros o sensores), cadenas, valores booleanos o colores.



Int aleatorio: Genera un valor entero aleatorio (número entero más cercano) dentro del mínimo y máximo dados.

Flotación aleatoria: Genera un valor flotante aleatorio (incluidos decimales) dentro del mínimo y máximo dados.



Canal de colores: Obtiene el valor del canal de color rojo, verde o azul para un valor RGB determinado.

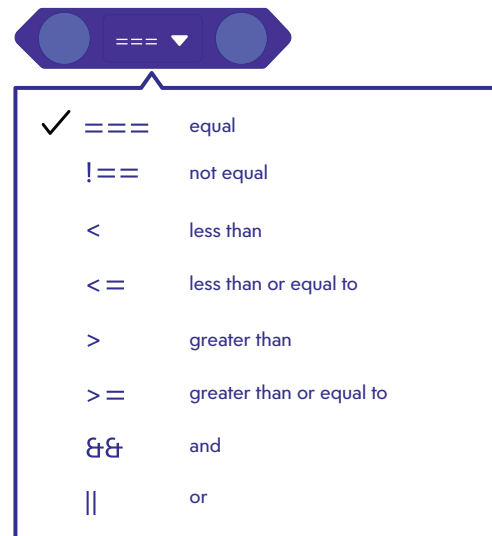


Mezclador de colores: Devuelve un nuevo color modificando un único canal (rojo, verde, azul) de un color determinado.

random color

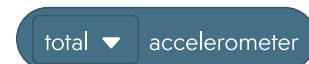
Color aleatorio: Establece un color aleatorio cuando se coloca en un bloque de color.

Comparadores



Comparadores: Los comparadores son bloques de código que comparan dos cosas (datos de sensores, operadores, cadenas, etc.). Los comparadores nos permiten crear comprobaciones lógicas que, cuando sean "verdaderas", permitirán que el programa realice una determinada acción o evento. Si una comparación es verdadera y lógica, entonces se ejecutará el código.

Sensores



Acelerómetro - Total: El sensor acelerómetro mide el cambio de fuerza en el robot.

- Total: La aceleración vectorial combinada de los tres ejes, de 0 a 14 g.
- Eje X: La aceleración de izquierda a derecha, de -8 a 8 g.
- Eje Y: La aceleración de adelante hacia atrás, de -8 a 8 g.
- Eje Z: La aceleración hacia arriba y hacia abajo, de -8 a 8 g.

pitch ▼ orientation

Orientación: El sensor de orientación mide la orientación del robot.

- Inclinación: el ángulo de inclinación hacia adelante o hacia atrás, de -180° a 180° .
- Rollo: el ángulo de inclinación hacia la izquierda o hacia la derecha, de -90° a 90° .
- Yaw: El ángulo de giro (giro), de -180° a 180° .

pitch ▼ gyroscope

Giroscopio: El sensor giroscópico mide la velocidad de giro del robot en grados por segundo.

- Inclinación: La velocidad de giro hacia adelante o hacia atrás, de -2000° a 2000° por segundo.
- Rollo: La velocidad de giro hacia la izquierda o hacia la derecha, de -2000° a 2000° por segundo.
- Yaw: La velocidad de giro lateral, de -2000° a 2000° por segundo.

total ▼ velocity

Velocidad - Total: El sensor de velocidad mide la velocidad en centímetros por segundo.

- Velocidad Total: La velocidad vectorial combinada de ambos ejes que siempre será un valor positivo, en centímetros por segundo.
- Velocidad - X: La velocidad derecha (+) o izquierda (-), en centímetros por segundo.
- Velocidad - Y: La velocidad hacia adelante (+) o hacia atrás (-), en centímetros por segundo.

total ▼ location

Ubicación - Total: La distancia desde el lugar de inicio del programa, que siempre será un valor positivo, en centímetros.

distance

Distancia: La distancia total recorrida, en centímetros.

speed

Velocidad: Valor de velocidad objetivo, de -255 a 255 .

heading

Título: Ángulo direccional objetivo del robot. Suponiendo que apunta el robot con la luz trasera azul hacia usted, entonces el rumbo de 0° es hacia adelante, 90° es hacia la derecha, 180° es hacia atrás y 270° es hacia la izquierda.

main LED

LED principal: El color RGB de los LED principales, de 0 a 255 para cada canal de color.

time elapsed

Tiempo transcurrido: La cantidad de tiempo con el que se ha ejecutado el programa (en segundos).

Eventos

on collision

En colisión: Lógica condicional llamada cuando el robot choca con un objeto.

on landing

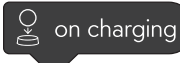
Al aterrizar: Se llama a la lógica condicional cuando el robot aterriza después de un evento de caída libre. No es necesario definir un evento de caída libre para que el robot experimente un evento de aterrizaje, pero el robot debe cumplir las condiciones de caída libre antes de aterrizar.

on freefall

En caída libre: Se llama lógica condicional cuando la gravedad es la única fuerza que actúa sobre el robot, lo que se consigue dejándolo caer o arrojándolo.

on gyro max

En Gyro Max: Lógica condicional llamada cuando el robot excede los límites de la velocidad de rotación medible de -2000° a 2000° por segundo.



Al cargar: Lógica condicional que se llama cuando el robot comienza a cargar su batería. Esto se puede activar colocando el robot en su base de carga cuando la base está enchufada (Mini excluido).



Sobre no cargar: Lógica condicional llamada cuando el robot deja de cargar su batería. Esto se puede activar retirando el robot de su base de carga o desenchufándolo (Mini excluido).

variables



Variable numérica: Un valor numérico almacenado que se puede asignar y operar. Los valores pueden ser números enteros (números enteros) o valores de punto flotante (números con decimales).



Número de conjunto: Establece el valor numérico para la variable numérica seleccionada.



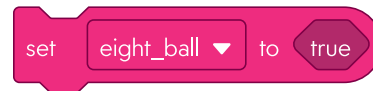
Variable de cadena: Un valor de cadena almacenado que se puede asignar y operar. Los valores pueden ser palabras u oraciones, como "¡Hola mundo!"



Establecer cadena: Establece el valor de cadena para la variable de cadena seleccionada.



Variable booleana: Un valor booleano almacenado que se puede asignar y operar. Los valores pueden ser verdaderos o falsos.



Establecer booleano: Establece el valor booleano para la variable booleana seleccionada.



Variables de color: Un valor de color almacenado que se puede asignar y operar. Los valores se componen de canales rojo, verde y azul que van de 0 a 255.



Establecer color: Establece el valor de color para la variable de color seleccionada.

Funciones



Función: Las funciones le permiten definir un grupo reutilizable de bloques y luego llamar esos bloques desde cualquier lugar dentro del programa principal. Los parámetros se pueden usar como variables locales de la función para cambiar el comportamiento cada vez que se llama a la función.

