

Lección 23

Funciones

Propósito

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

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

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

Secuencia para el aprendizaje

Conocimiento inicial (10 min)

Ampliación del conocimiento (60 min)

Transferencia del conocimiento (10 min)

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

Vocabulario

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

Código

- Define a function
- Call a function

Objetivos

Los estudiantes serán capaces de:

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

Estrategia de aprendizaje

Conocimiento inicial (10 min)

Instrucciones de alto nivel vs. Nivel bajo

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

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

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

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

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

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

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

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

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

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

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

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

Ampliación del conocimiento (60 min)

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

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

Vocabulario

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

Funciones de llamada

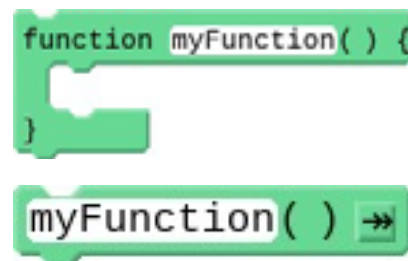
Las funciones permiten construir tus propios bloques y decidir qué código va dentro de ellos. Este es el comando que le permite crear una nueva función.

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

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

Introduciendo funciones

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



Reordenando el código

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

Funciones de llamada en Ciclos de Dibujo

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

Llamar a las funciones varias veces

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

Hacer cambios en las funciones

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

Transferencia del conocimiento (10 min)

Pregunte:

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

¿Por qué es esto algo que quisiéramos hacer?

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

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

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

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

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

Por qué usar las funciones

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

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

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

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

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

Sugerencias para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

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