

¿Podemos programar otros personajes?

Introducción a Scratch

¿Podemos elegir los personajes y los escenarios de nuestros proyectos? ¿Cómo construimos programas interactivos? ¿Qué tan diferentes son los entornos de programación Scratch y Pilas Bloques?

Esta secuencia consiste en una aproximación a Scratch como entorno didáctico de programación. La estrategia es presentar desafíos acotados que inviten a las y los estudiantes a explorar la nueva herramienta para conocer sus novedades (el trabajo con los personajes y su aspecto visual y la posibilidad de interactuar con el usuario) e identificar ideas y herramientas que ya conocen que pueden transferir a este nuevo entorno (como la repetición, los procedimientos o la alternativa condicional).

Actividad 1

Las y los estudiantes exploran proyectos existentes en la plataforma online de Scratch y abordan una serie de desafíos puntuales con el objetivo de realizar una exploración guiada de la herramienta. El objetivo es que conozcan los bloques de las categorías Movimiento, Apariencia y Eventos.

Actividad 2

Las y los estudiantes abordan un conjunto de desafíos que requieren aplicar en el nuevo entorno herramientas de programación vistas en secuencias anteriores: repetición, alternativa condicional y procedimientos.

Actividad 3

Se propone una instancia de trabajo libre en la que las y los estudiantes podrán programar un pequeño juego y personalizarlo como primer paso hacia el trabajo en entornos abiertos (a diferencia de los entornos orientados a desafíos). El objetivo es que construyan un proyecto que sientan propio y que, además, requiera del uso de algunas de las herramientas de programación conocidas.

Datos curriculares

Nivel: Primaria, segundo ciclo; Secundaria, ciclo básico

Área: Programación

Eje: Lenguajes de programación

- Herramientas de lenguajes de programación: procedimientos, repeticiones, alternativa condicional, sensores, eventos.

Eje: Soluciones a problemas computacionales

- Diseño de soluciones computacionales: estrategia, legibilidad.

Objetivos de aprendizaje

- Conocer aspectos básicos y herramientas disponibles en el uso de Scratch.
- Transferir nociones de programación para resolver desafíos breves en un nuevo entorno.
- Introducir la noción de evento como mecanismo para elaborar programas interactivos.

Saberes previos de CC

Área: Programación

Eje: Lenguajes de programación

- Herramientas de lenguajes de programación: comandos primitivos, procedimientos, repeticiones, alternativa condicional, sensores.

Eje: Soluciones a problemas computacionales

- Diseño de soluciones computacionales: estrategia, legibilidad, reutilización del código.

Materiales necesarios

- Dispositivos con Scratch 3 instalado o acceso a su versión en línea <https://scratch.mit.edu.ar/>

Acerca de esta iniciativa

Desde el sitio curriculum.program.ar tenemos por objetivo acompañar a la comunidad docente de habla hispana en el desafío de llevar las Ciencias de la Computación al aula. Para ello, construimos un repositorio que reúne diversos recursos para el aula que desde la Iniciativa Program.AR de la Fundación Sadosky impulsamos desde 2013.

Organizados a partir de los saberes a promover con nuestras y nuestros estudiantes y los conceptos de la disciplina presentados en la [Propuesta curricular para la inclusión de las Ciencias de la Computación \(CC\) en el aula](#), encontrarán en curriculum.program.ar proyectos, secuencias didácticas y actividades desarrollados por una diversidad de autores y docentes en conjunto con instituciones y universidades de América Latina.

Estos materiales, que han sido desarrollados para responder a necesidades de diferentes contextos y países y que son heterogéneos en su formato y extensión, comparten un mismo propósito: integrar las Ciencias de la Computación en la escolaridad obligatoria para promover en el conjunto de las y los estudiantes la construcción de saberes que les permitan comprender, apropiarse y transformar la tecnología digital y computacional y así participar de manera crítica del mundo contemporáneo.

Perspectiva de género

La Fundación Sadosky busca propiciar una experiencia educativa inclusiva y promotora de la equidad de género. Sabemos que existe una fuerte desigualdad de género en el acceso al uso de recursos tecnológicos y a conocimientos de ciencias de la computación. Uno de los motivos de esta brecha tiene que ver con que socialmente es considerada como una disciplina de varones. Por eso es imprescindible que, como docentes, podamos contribuir a desnaturalizar prejuicios y generar estrategias para incentivar especialmente el trabajo de estudiantes mujeres y de identidades de género trans y no binarias.

En el documento [Enseñar computación desde la mirada de la Educación Sexual Integral \(ESI\)](#) es posible encontrar orientaciones para crear aulas más inclusivas y respetuosas para estudiantes y docentes de todos los niveles educativos.

Cómo utilizar este recurso

Esta secuencia es parte de una colección que se encuentra disponible en el sitio curriculum.program.ar

Se integran actividades “desenchufadas” o en papel, con otras en plataformas especialmente diseñadas para la enseñanza de la programación, como Pilas Bloques o Scratch.

Créditos

Autores: Javier Castrillo, Tomás Caballero

Coordinación autoral: Julián Dabbah

Coordinación editorial: Inés Roggi

Edición: Florencia N. Acher Lanzillotta

Diseño: Fabio Viale

Cómo citar este documento

Fundación Sadosky (2024), “¿Podemos programar otros personajes? Introducción a Scratch”, en *Actividades para aprender a Program.AR*. Disponible en: <https://curriculum.program.ar/>



Listado de secuencias que componen esta colección

Primitivas, procedimientos y repetición

1. ¿Qué es programar?
2. Definimos nuestros bloques
3. Programamos en papel cuadriculado
4. Programamos estrategias en Pilas Bloques
5. Creamos desafíos de repetición
6. Seguimos programando estrategias en Pilas Bloques
7. Creamos desafíos de procedimientos

Alternativa condicional

8. ¿Cómo se resuelven problemas cambiantes?
9. Resolvemos recorridos cambiantes
10. Programamos estrategias para problemas cambiantes
11. Creamos desafíos cambiantes

Interactividad y variables

12. ¿Podemos programar otros personajes?
13. Programamos el personaje de un videojuego
14. Guardamos información
15. Programamos nuestro videojuego

Repetición condicional

16. Un videojuego que no sabemos cuándo termina

Actividad 1

Primeros pasos en Scratch

Se reparten tarjetas que invitan a las y los estudiantes a resolver desafíos puntuales y acotados en Scratch con el objetivo de que exploren el entorno y los modos de uso de esta herramienta, enfocándose especialmente en los bloques de las categorías Apariencia y Eventos.

Objetivo >

Se espera que las y los estudiantes se aproximen al entorno de Scratch y a la construcción de programas interactivos.

Inicio >

El **propósito de este momento** es motivar la introducción de Scratch como una herramienta para construir programas visuales e interactivos.

Orientaciones

Organizamos la clase en grupos heterogéneos pequeños¹. Les pedimos que ingresen a Scratch². A modo de motivación para conocer de qué se trata esta nueva herramienta de programación y qué tipo de programas nos permite construir, invitamos a las y los estudiantes a explorar la galería de proyectos de Scratch. Para acceder a la galería de proyectos, pueden ver los proyectos que se encuentran en la página principal como “Proyectos destacados”, o bien, ingresar a la opción “[Explorar](#)” de la parte superior de la página principal.

Proponemos que recorran libremente la galería y observen en particular algún proyecto que les interese. Podrán ejecutar el programa presionando la bandera verde para interactuar como usuarios/as.

Al finalizar la exploración, invitamos a las y los estudiantes a compartir sus hallazgos sobre el entorno y describir el proyecto que eligieron explorar.

¹ Se pueden consultar dinámicas lúdicas para el armado de grupos heterogéneos en el documento [Enseñar computación desde la mirada de la Educación Sexual Integral \(ESI\)](#).

² Para usar Scratch pueden hacerlo con la versión sin conexión (previamente instalada, disponible en: <https://scratch.mit.edu/download>) o acceder a la versión en línea (<https://scratch.mit.edu>). En Linux también se puede usar el fork Scratux (<https://github.com/scratux/scratux?tab=readme-ov-file>).

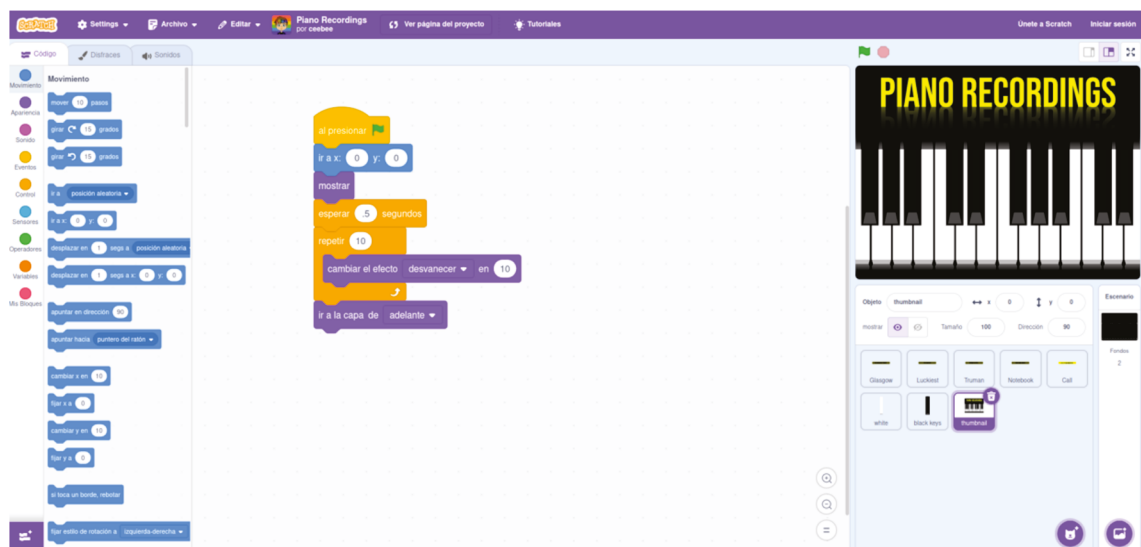


¿Cuál es el objetivo o desafío del proyecto? ¿De qué manera se interactúa con él (por ejemplo, se mueve un personaje con el teclado, se presiona una tecla para avanzar, etc.)?

A continuación, les proponemos que presionen el botón “Ver Dentro”, situado arriba y a la derecha de la página del proyecto, y vean cómo está programado el proyecto.



Página de un proyecto de Scratch. El botón de la bandera verde lo ejecuta (1) y el de “Ver dentro” muestra cómo está construido (2).



Vista de un proyecto de Scratch “por dentro”.

Una vez que todo el grupo ha seleccionado y entrado en su proyecto, hacemos una puesta en común.



¿Cómo se construye un proyecto en Scratch? ¿Qué elementos identifican? ¿Encontraron similitudes con Pilas Bloques?

El objetivo de esta exploración es reconocer similitudes con PilasBloques y elementos que corresponden a conceptos que ya conocen. Por ejemplo, en ambos entornos, los programas se construyen con bloques y es posible identificar algunos comandos y anticipar qué realizan, como la alternativa condicional o la repetición. La novedad de Scratch es que en la construcción del programa se pueden elegir el escenario y los personajes.

Si queremos promover una exploración más profunda, podemos alentar a los grupos a que modifiquen el programa que analizaron a partir de tareas simples como cambiar un disfraz, alguna funcionalidad menor o algún comportamiento particular, que se desprenda de la exploración hecha hasta el momento.

Al finalizar, las y los estudiantes podrán registrarse en Scratch para poder almacenar sus trabajos y compartirlos. Esto se hace desde el botón "Únete a Scratch" en <https://scratch.mit.edu/join> En caso de no poder hacer ese registro, las soluciones se pueden descargar en cada computadora. Independientemente de si trabajan en línea o no, es importante acordar una manera de guardar los trabajos.

Desarrollo >

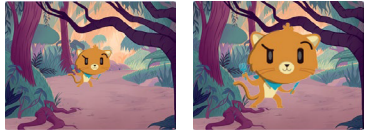
El propósito de este momento es aproximarse al proceso de creación de un programa en *Scratch* desde cero a través de desafíos acotados.

Orientaciones

Entregamos a los grupos las tarjetas disponibles para esta actividad en el [Anexo](#), formadas por tres desafíos. El recorrido propuesto guía la creación de un nuevo proyecto (elegir un escenario y un personaje), y explorar los bloques de eventos del teclado o del clic del mouse y otros de las categorías de Apariencia y Movimiento. Buscamos que atraviesen el proceso de crear un programa en Scratch desde cero sin perder el foco del objetivo de programación. Es probable que necesitemos asistir en el manejo del tiempo para que los grupos resuelvan rápidamente el aspecto visual y cuenten con el mayor tiempo posible para explorar y ensayar en el entorno la solución al desafío.

Cambiar tamaño

Desafío 1:
Hacer que el tamaño del personaje aumente cuando se presione la **tecla G**.




Indicaciones:

1. Elegí un fondo.
2. Agregá un personaje.
3. Hacé que su tamaño aumente cada vez que presiones la **tecla G**.

Probá con estos bloques:

cambiar tamaño por 10
al presionar tecla espacio

Desafío 2:
Hacer que el personaje vuelva a su tamaño original al ejecutar el programa desde el inicio (haciendo clic en ).

Probá con estos bloques:

fixar tamaño al 100 %
al hacer clic en

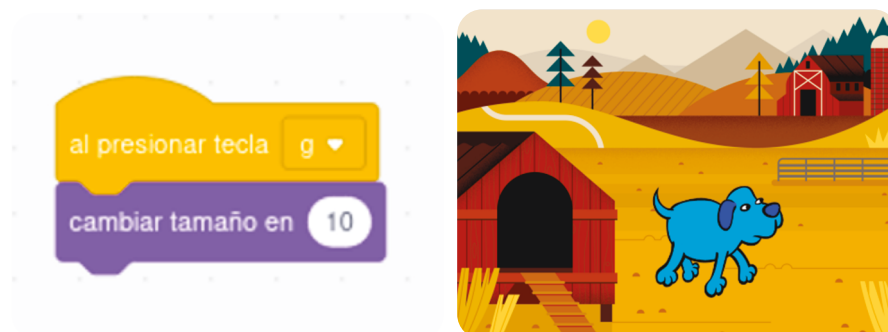
Desafío 3:
Hacer que el personaje reduzca su tamaño cuando se presione la **tecla P**.

Pista: Probá cambiando el valor del bloque cambiar tamaño. ¿Admite valores negativos?

Ejemplo de una de las tarjetas a entregar.

Brindamos un espacio para que los grupos avancen en la resolución del primer desafío con autonomía. Recorremos los puestos de trabajo y realizamos puestas en común cuando consideremos pertinente que compartan los hallazgos entre todos los grupos.

Cuando todos los grupos hayan resuelto el primer desafío, promovemos una puesta en común para que compartan cómo fue la manera en que llegaron a la solución del desafío y las categorías de bloques que exploraron.



Ejemplo de resolución del primer desafío de la tarjeta de ejemplo.

Luego de concluir la primera puesta en común, invitamos a las y los estudiantes a completar los siguientes desafíos. Es una oportunidad para

recordar la importancia de ir guardando los proyectos (ya sea en la cuenta en línea o en la computadora) para poder reutilizar (y resguardar) el trabajo previo.



Posible resolución de los tres desafíos de la tarjeta de ejemplo.

Cierre >

El **propósito de este momento** es explicitar las nociones de interactividad, evento y objeto en el trabajo con Scratch.

Orientaciones

Recuperamos el trabajo con las tarjetas y retomamos los intercambios orales sobre parecidos y diferencias entre Scratch y Pilas Bloques. A partir de lo que los grupos cuenten sobre los desafíos que resolvieron y cómo lo hicieron, nos interesa reforzar que:

- tanto en Pilas Bloques como en Scratch se utilizan bloques para construir el programa.
- en Scratch, los bloques son más variados y aparecen organizados en categorías, por ejemplo, las categorías Apariencia (que refiere a acciones que impactan en cómo se ven los personajes), Movimiento (que permite desplazar los personajes en el escenario) y Eventos (que define la interacción de la usuaria o el usuario con el programa a través del teclado o el mouse).

- Scratch permite crear **programas interactivos utilizando eventos**, es decir, programas que permiten considerar las acciones de quien está usando el programa (por ejemplo, si presiona una tecla o hace clic) durante la ejecución.
- los eventos son una herramienta que provee el entorno de Scratch para establecer qué debe suceder cuando el usuario interactúa con el programa, por ejemplo, son eventos los bloques **Al presionar tecla []** o **Al hacer clic en este objeto**.
- Scratch nos permite agregar personajes u objetos al escenario y cada uno de ellos se programa por separado.

Actividad 2

Herramientas conocidas

A partir de un nuevo conjunto de tarjetas, las y los estudiantes resuelven desafíos que requieren recuperar las herramientas de programación que ya conocen (procedimientos, repetición simple y alternativa condicional).

Objetivos >

Se espera que las y los estudiantes:

- Empiecen a reconocer que la mayoría de los lenguajes de programación **tienen características en común** y la importancia de poder transferir lo que se sabe de un lenguaje a otro.
- Recuperen los conceptos de procedimiento, repetición simple y alternativa condicional para aplicarlos en la creación de programas interactivos.

Inicio >

El **propósito de este momento** es recuperar modos de trabajo en Scratch para desarrollar programas interactivos.

Conversamos con las y los estudiantes acerca de lo que experimentamos en la **Actividad 1** para recuperar la noción de programa interactivo, los bloques de evento y las nuevas categorías de bloques que permiten programar a los personajes (Movimiento, Apariencia, etc.).

Desarrollo >


El **propósito de este momento** es resolver desafíos que requieran transferir herramientas de programación conocidas al entorno Scratch.

Presentamos un nuevo conjunto de tarjetas (ver "Tarjetas para la actividad 2", [Anexo](#)) y repartimos una a cada grupo.

Los desafíos de estas tarjetas requieren el uso de la alternativa condicional o de la repetición simple combinadas con la definición de procedimientos.

5


Desafío 1: Hacer que el fantasma intente asustar a las elfas cuando pase sobre ellas.




Indicaciones:

1. Elegí un escenario.
2. Agregá un fantasma (*Ghost*) y tres elfas (*Elf*).
3. Hacé que el fantasma se mueva a la derecha al presionar la tecla →.
4. Hacé que asuste (disfraz *ghost-c*) si está sobre una elfa o vuelva a la normalidad (disfraz *ghost-a*).

Probá con estos bloques:






Desafío 2: Crear y utilizar los bloques:



Una de las tarjetas a entregar.

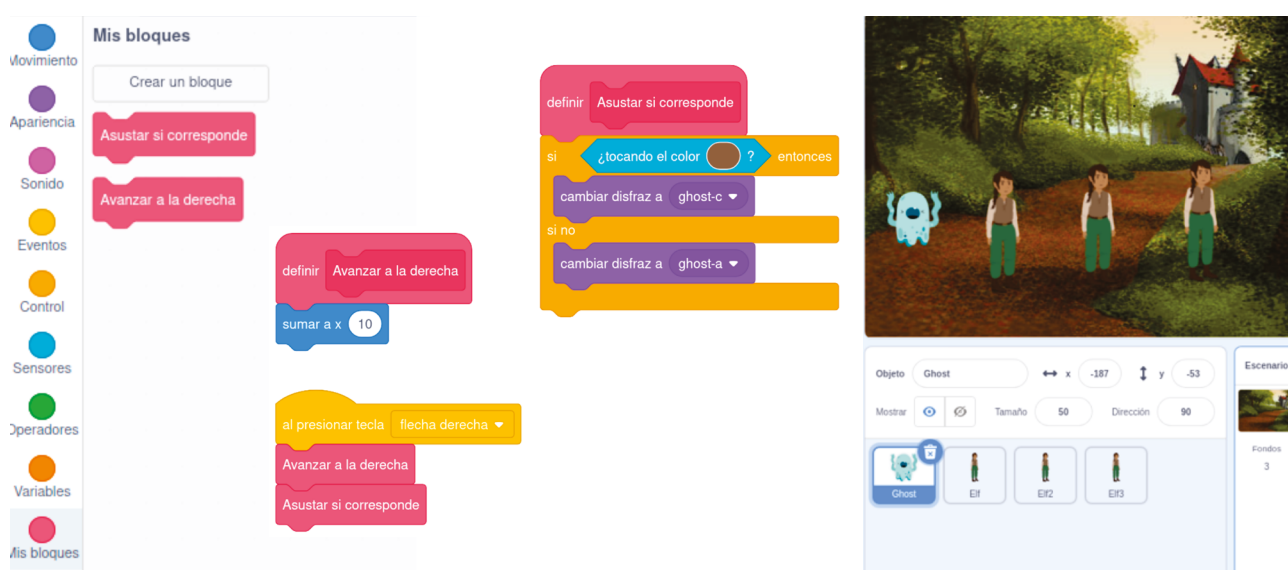
Proponemos a los grupos que comiencen a trabajar en la resolución de sus tarjetas y recorreremos los puestos de trabajo mirando los progresos e interviniendo solo si fuera necesario.

En nuestras orientaciones reforzamos la importancia de la recuperación de los saberes previos y la identificación de similitudes entre Scratch y otros entornos conocidos para motivar y acompañar los procesos de transferencia. Nos interesa reforzar con estos desafíos que, a pesar de no haber trabajado ampliamente con Scratch, ya conocen muchas cosas sobre su modo de funcionamiento, pues es similar a otros entornos de programación. En este sentido, si bien es una de las primeras exploraciones, no están partiendo de cero.

Solución posible al desafío 1 de la tarjeta 5.

Una dificultad que puede presentarse al resolver el segundo desafío es la diferencia en las denominaciones entre Pilas Bloques y Scratch. Scratch denomina "Mis Bloques" a lo que en PilasBloques se llama "Procedimientos". Podemos relacionar la elección de este nombre con el hecho de que los procedimientos nos permiten definir bloques propios a partir de los bloques que provee el entorno o el desafío (por ejemplo, como vimos en la secuencia "[Definimos nuestros bloques](#)" de esta colección).



Solución al desafío 2 de la tarjeta 5.

Cierre >

El **propósito de este momento** es reconocer que, en este nuevo entorno, están presentes las mismas herramientas que abordaron en Pilas Bloques y explicitar los procesos de transferencia que hicieron al resolver los desafíos.

Armamos una puesta en común en la que cada grupo relata qué desafío tuvo que resolver, qué herramientas de programación requirió para eso, cómo las utilizó y cómo las utilizaban en otros entornos conocidos, como Pilas Bloques. Además de identificar las similitudes, es importante explicitar los procesos de transferencia que les permitieron usar las herramientas conocidas en un nuevo contexto.

Para concluir esta primera aproximación al nuevo entorno y reforzar la idea de que algunas de las herramientas conocidas están presentes en Scratch, enumeramos algunas e invitamos a categorizarlas según pertenezcan a Pilas Bloques (u otro entorno conocido previamente) o Scratch o ambos. Podemos realizarlo oralmente y anotar en una tabla en el piza-

rrón o distribuir papeles para que las y los estudiantes anoten los aspectos y los peguen en un afiche en el sector correspondiente.

Algunos aspectos posibles con los que trabajar son: Programación por bloques, Comandos básicos, Procedimientos, Repetición, Alternativa condicional, Sensores, Eventos, Escenario, Agregar/quitar objetos, Modificar tamaño, Mover personajes, Cambiar disfraces, Modificar escenario, Añadir pistas, Información de objetivo logrado.

Pilas Bloques	Scratch
Pistas	Programación por bloques
Información de objetivo logrado	Comandos básicos
	Procedimientos
	Repetición
	Alternativa condicional
	Sensores
	Escenario
	Mover personajes
	Eventos
	Agregar/quitar objetos
	Modificar tamaño
	Disfraces
	Modificar/elegir escenario

Una clasificación posible de los aspectos de cada entorno de programación.

Durante la clasificación o con la tarea ya resuelta en el pizarrón, aprovechamos para señalar que:

- muchos aspectos fundamentales (las herramientas de programación aprendidas, la existencia del escenario, el uso de bloques, por ejemplo) pertenecen a ambos entornos; reconocer y aprovechar esas similitudes agiliza el dominio de un nuevo entorno.
- si algo no aparece exactamente cómo lo conocemos, no necesariamente significa que no exista o que no podamos aprovechar lo que ya sabemos en el nuevo entorno (por ejemplo, "Mis bloques" en Scratch cumple la misma función que los procedimientos en PilasBloques).
- los eventos son algo exclusivo de Scratch; Scratch nos habilita la posibilidad de crear programas interactivos, es decir, que contemplen las acciones de un/a usuario/a durante su funcionamiento.
- el escenario y los personajes en Pilas Bloques vienen dados por el desafío, mientras que en Scratch podemos elegirlos y modificarlos; utilizaremos Scratch para crear programas interactivos con los personajes y escenarios que queramos o necesitemos.

Actividad 3

Personalizamos un desafío

A modo de cierre de la secuencia, habilitamos un espacio para que las y los estudiantes continúen trabajando sobre los proyectos que empezaron, poniendo el foco en usar las herramientas que ya conocen para incorporar a sus programas un comportamiento dado (por ejemplo, que el personaje regrese al borde al tocar un objeto). Guiamos una instancia de trabajo más libre en la que puedan aprovechar las posibilidades que brinda un entorno abierto para personalizar (y apropiarse de) los proyectos que están construyendo.

Objetivos >

Se espera que las y los estudiantes:

- Recuperen los conceptos de procedimiento, repetición simple, alternativa condicional, sensores y/o eventos para construir comportamientos preestablecidos.
- Se familiaricen con la creación de programas interactivos.

Inicio >

El **propósito de este momento** es reforzar la idea de que existen similitudes entre los entornos o los lenguajes de programación y, por lo tanto, se puede aprovechar lo que sabe de unos en otros.

Orientaciones

Comenzamos con una breve puesta en común para recuperar las similitudes que encontraron con Pilas Bloques y los hallazgos en Scratch.

Invitamos a los grupos a que recuperen alguno de los programas que construyeron en las actividades anteriores de la secuencia para aumentarlo según sus preferencias en línea con algunas consignas que les brindaremos.

Reforzamos que, en esta actividad, podrán aprovechar las posibilidades que brinda un entorno abierto como Scratch para personalizar e incorporar sus preferencias al proyecto.

Desarrollo >

El **propósito de este momento** es que las y los estudiantes realicen una primera experiencia de trabajo en un entorno abierto en el que se combine el cumplimiento de un objetivo de programación con la posibilidad de personalización del programa.

Orientaciones

Se trabajará con una consigna común y luego con otras que pueden variarse de acuerdo con el tiempo disponible, cantidad de estudiantes, grado de comprensión, etc.

El desarrollo de esta actividad se hará en dos partes. La primera parte atenderá **una consigna común** a todos los grupos: "Permitir que el usuario controle el movimiento de un personaje, por ejemplo, hacer que se mueva con las flechas del teclado".

Para la segunda parte, podemos asignar las tarjetas con las consignas o dejar que los grupos las elijan (ver "Tarjetas para la actividad 3", [Anexo](#)). Estas consignas se refieren a comportamientos que debe tener el programa (cómo debe moverse un personaje, cuándo debe cambiar el escenario, etc.), pero no dicen nada sobre el aspecto del programa (qué personaje, qué escenario, etcétera).

Cada consigna por separado debe apuntar a obtener una nueva versión del programa con una función añadida. Buscaremos lograr que obtengan más de una versión en vistas de construir programas cada vez más complejos, como pequeños videojuegos.

La idea es que puedan cumplir las consignas apelando a las herramientas de programación que ya conocen. Es importante que registren, para compartir luego con los demás grupos, qué parte del programa resuelve cada consigna, qué bloques eligieron para eso y por qué. Pueden sistematizarlo durante la resolución del desafío o recuperarlo con el programa terminado.

Nuestras orientaciones deben apuntar a agilizar la etapa de la personalización (elección de los fondos, los personajes, los disfraces, etc.) para que se concentren en la programación. Para resolver la etapa de programación, apelamos a experiencias previas, tanto durante esta secuencia como durante el trabajo previo en otros entornos. La dinámica de trabajo dependerá mucho de la familiaridad de los grupos con la herramienta y la facilidad con la que identifiquen los bloques que necesitan para cumplir con cada consigna. Es importante que brindemos acompañamiento, pero teniendo presente que el objetivo de esta instancia es que puedan experimentar un mayor grado de autonomía. Si detectamos dificultades comunes, podemos realizar interrupciones para resolverlas entre todas y todos.

Cierre >

El **propósito de este momento** es reforzar la idea de entorno abierto, reconocer herramientas de programación comunes y explicitar acciones de transferencia de saberes entre los distintos entornos de programación.

Promovemos un intercambio entre los grupos para que compartan sus producciones. Podemos proyectar los programas para que los vea todo el curso o habilitar la circulación entre los grupos a modo de feria y que cada grupo presente su programa.



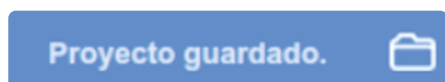
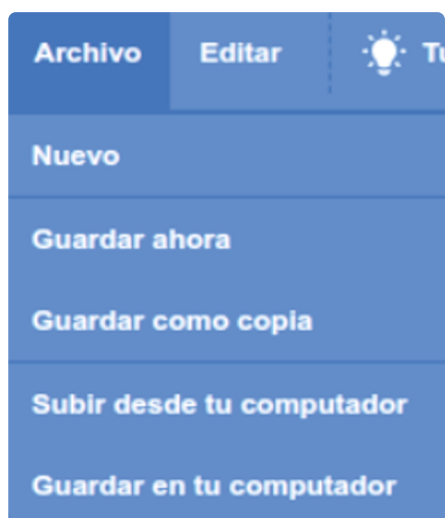
¿Qué consignas resolvieron? ¿En qué orden las abordaron? ¿Qué bloques usaron para resolver cada una? ¿Podemos identificar qué parte del programa resuelve cada consigna? ¿Por qué?

En el diálogo entre los grupos, es importante que identifiquen en el proceso de construcción cuándo recurrieron a saberes previos y de qué manera los adaptaron (es decir, que expliciten el **proceso de transferencia**). Por ejemplo, "Para conseguir que un personaje cambie de aspecto cuando pasa sobre otro, colocamos una alternativa condicional a continuación del movimiento. Pensamos en esta solución porque había dos casos posibles (que estuviera o no estuviera el objeto) al igual que en los desafíos de Pilas Bloques cuando podíamos encontrar o no una lata para recoger. Además, igual que en Pilas Bloques, usamos un sensor para construir la condición y colocamos los bloques correspondientes a cada caso dentro de cada rama de la alternativa".

Abstraer el proceso de construcción y explicitar los razonamientos probablemente requiera de una moderación intensa del relato. Sin embargo, **es sumamente valioso explicitar estas acciones de transferencia**, ya que, de otra manera, pueden pasar desapercibidas y no formar parte de los aprendizajes logrados durante la secuencia.

Anexo

Proceso de registro en Scratch

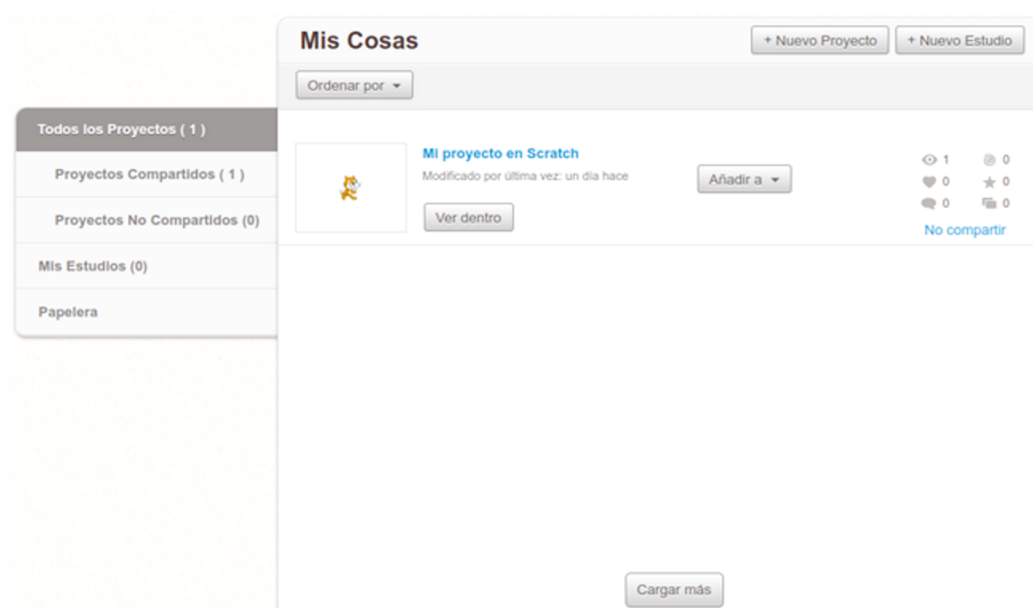


1. Una vez que presionamos "Únete a Scratch", comenzaremos el proceso de registro en esta plataforma.

2. Allí se deberán ingresar ciertos datos personales como usuario, contraseña, país, correo electrónico, etc. El correo servirá para confirmar estos datos y dar el alta a la cuenta. Luego, usando el botón "Guardar Ahora" podrán almacenar sus trabajos y compartirlos con el resto.

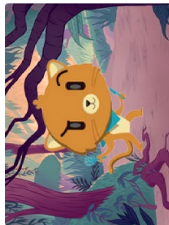
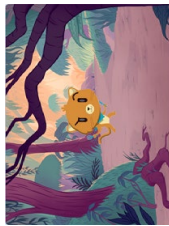
3. Cuando el proyecto esté guardado podremos ver la confirmación

4. Y en "Mis cosas" estarán los trabajos que hayan guardado con esa modalidad.



Cambiar tamaño

Desafío 1:
Hacer que el tamaño del personaje aumente cuando se presione la **tecla G**.



Indicaciones:
1. Elegí un fondo.
2. Agregá un personaje.
3. Hacé que su tamaño aumente cada vez que presiones la **tecla G**.

Próba con estos bloques:

Cambiar tamaño por 10

al presionar tecla espacio

Desafío 2:
Hacer que el personaje vuelva a su tamaño original al ejecutar el programa desde el inicio (haciendo clic en).

Próba con estos bloques:

fijar tamaño al 100 %

al hacer clic en

Desafío 3:
Hacer que el personaje reduzca su tamaño cuando se presione la **tecla P**.

Pista: Probá cambiando el valor del bloque **cambiar tamaño**. ¿Admite valores negativos?

Aplicar efecto visual

Desafío 1:
Hacer que el tamaño del personaje se pixele cuando se presione la **tecla P**.



Indicaciones:
1. Elegí un fondo.
2. Agregá un personaje.
3. Hacé que el personaje se pixele más y más cada vez que presiones la **tecla P**.

Próba con estos bloques:

sumar al efecto color 25

al presionar tecla espacio

Desafío 2:
Hacer que el personaje recupere su aspecto original al ejecutar el programa desde el inicio (haciendo clic en).

Próba con estos bloques:

dar al efecto color el valor 0

al hacer clic en

Desafío 3:
Hacer que el personaje revierta el pixelado cuando se presione la **tecla D**.

Pista: Probá cambiando el valor del bloque **sumar al efecto**. ¿Admite valores negativos?

Girar

Desafío 1:
Hacer que el personaje gire en sentido horario cuando se presione la **tecla →**.



Indicaciones:
1. Elegí un fondo.
2. Agregá un personaje.
3. Hacé que el personaje gire en sentido horario cada vez que presiones la **tecla →**.

Próba con estos bloques:

girar 15 grados

al presionar tecla espacio

Desafío 2:
Hacer que el personaje vuelva a su orientación original al ejecutar el programa desde el inicio (haciendo clic en).

Próba con estos bloques:

apuntar en dirección 90

al hacer clic en

Desafío 3:
Hacer que el personaje gire en sentido antihorario cuando se presione la **tecla ←**.

Pista: Probá con el bloque **girar** 15 grados

Cambiar disfraz y hablar

Desafío 1:

Hacer que el disfraz del personaje cambie cuando se hace clic sobre él.



Indicaciones:

1. Elegí un fondo.
2. Agregá un personaje.
3. Hacé que su disfraz cambie a otro cuando hagás clic sobre él.

Probá con estos bloques:

cambiar disfraz a dinosaur4-d

al hacer clic en este objeto

Desafío 2:

Hacer que el personaje vuelva al disfraz original al ejecutar el programa desde el inicio (haciendo clic en).

Probá con estos bloques:

cambiar disfraz a dinosaur4-d

al hacer clic en

Desafío 3:

Hacer que el personaje, además de cambiar de disfraz, diga algo, como una caricatura.

Pista:

decir (HOLA) durante 2 segundos

Mover aleatoriamente e ir a posición

Desafío 1:

Hacer que el personaje vaya a posición aleatoria en el escenario cuando se hace clic sobre él.



Indicaciones:

1. Elegí un fondo.
2. Agregá un personaje.
3. Hacé que el personaje se mueva a una posición aleatoria en el escenario cada vez que hagás clic sobre él.

Probá con estos bloques:

desplazar en 1 segs a posición aleatoria

al hacer clic en este objeto

Desafío 2:

Hacer que el personaje aparezca en el centro del escenario al ejecutar el programa desde el inicio (haciendo clic en).

Probá con estos bloques:

ir a x: 0 y: 0

al hacer clic en

Desafío 3:

Hacer que el personaje aparezca en una posición aleatoria antes de moverse.

Pista:

ir a posición aleatoria

Mover horizontalmente

Desafío 1:

Hacer que el personaje se mueva a la derecha cuando se presione la tecla .



Indicaciones:

1. Elegí un fondo.
2. Agregá un personaje.
3. Hacé que el personaje se mueva a la derecha cada vez que presiones la tecla .

Probá con estos bloques:

sumar a x 10

al presionar tecla espacio

Desafío 2:

Hacer que el personaje aparezca en el centro horizontal al ejecutar el programa desde el inicio (haciendo clic en).

Probá con estos bloques:

dar a x el valor 0

al hacer clic en

Desafío 3:

Hacer que el personaje se mueva a la izquierda cuando se presione la tecla .

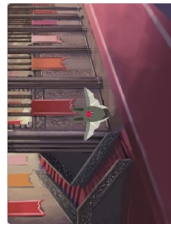
Pista:

Probá cambiando el valor del bloque sumar a x. ¿Admite valores negativos?

Mover verticalmente

Desafío 1:

Hacer que el personaje se mueva hacia arriba cuando se presione la **tecla I**.



Indicaciones:

1. Elegí un fondo.
2. Agregá un personaje.
3. Hacé que el personaje se mueva hacia arriba cada vez que presiones la **tecla I**.

Probá con estos bloques:



Desafío 2:

Hacer que el personaje aparezca en el centro vertical al ejecutar el programa desde el inicio (haciendo clic en **▶**).

Probá con estos bloques:



Desafío 3:

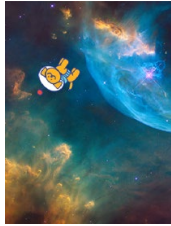
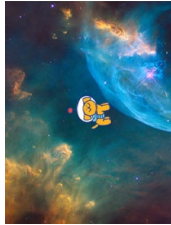
Hacer que el personaje se mueva hacia abajo cuando se presione la **tecla J**.

Pista: Probá cambiando el valor del bloque **sumar a y**.
¿Admite valores negativos?

Mover en una dirección

Desafío 1:

Hacer que el personaje se mueva en diagonal cuando se presione la **tecla I**.



Indicaciones:

1. Elegí un fondo.
2. Agregá un personaje.
3. Hacé que el personaje se mueva en diagonal cada vez que presiones la **tecla I**.

Probá con estos bloques:



Desafío 2:

Hacer que el personaje aparezca en el centro del escenario al ejecutar el programa desde el inicio (haciendo clic en **▶**).

Probá con estos bloques:



Desafío 3:

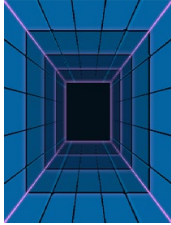
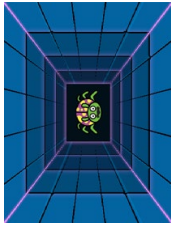
Hacer que el personaje en la dirección contraria a la original cuando se presione la **tecla V**.

Pista: Probá cambiando el valor del bloque **apuntar en dirección**.

Mostrar y esconder

Desafío 1:

Hacer que el personaje desaparezca cuando se hace clic sobre él.



Indicaciones:

1. Elegí un fondo.
2. Agregá un personaje.
3. Hacé que el personaje desaparezca cuando hagas clic sobre él.

Probá con estos bloques:



Desafío 2:

Hacer que el personaje vuelva a aparecer en el escenario al ejecutar el programa desde el inicio (haciendo clic en **▶**).

Probá con estos bloques:



Desafío 3:

Hacer que el personaje piense algo antes de desaparecer.

Pista: Probá con el bloque **pensar Um... durante 2 segundos**.

Tarjetas para la Actividad 2


1

Desafío 1:

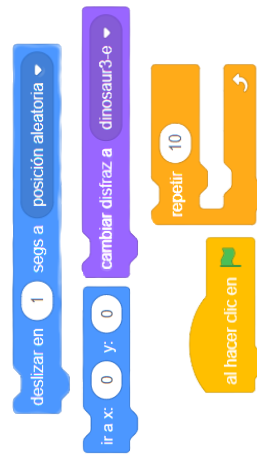
Hacer que el pterodáctilo haga un berrinche y, luego, se calme.



Indicaciones:

1. Elegí un escenario.
2. Agregá un pterodáctilo (*Dinosaur3*).
3. Hacé que se enoje (*disfraz dinosaur3-d*) y vuele a 10 posiciones aleatorias al hacer clic sobre .
4. Hacé que luego se calme (*disfraz dinosaur3-e*) en el centro del escenario.

Probá con estos bloques:



Desafío 2:

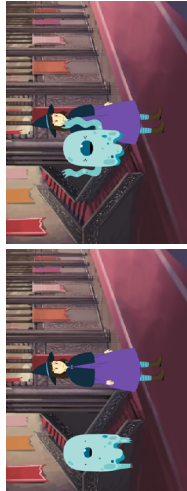
Crear y utilizar los bloques:



2

Desafío 1:

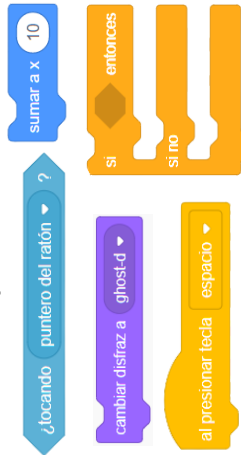
Hacer que el fantasma intente asustar a la bruja cuando pase sobre ella.



Indicaciones:

1. Elegí un escenario.
2. Agregá un fantasma (*Ghost*) y una bruja (*Witch*).
3. Hacé que el fantasma se mueva a la derecha al presionar la tecla \rightarrow .
4. Hacé que asuste (*disfraz ghost-c*) si está sobre la bruja o vuelva a la normalidad (*disfraz ghost-a*).

Probá con estos bloques:



Desafío 2:

Crear y utilizar los bloques:



3

Desafío 1:

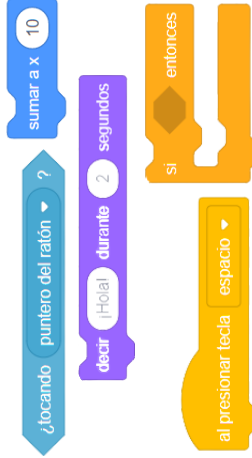
Hacer que el fantasma intente asustar a la bruja cuando pase sobre ella.



Indicaciones:

1. Elegí un escenario.
2. Agregá un mago (*Wizard*) y una varita (*Wand*).
3. Hacé que el mago se mueva a la derecha al presionar la tecla \rightarrow .
4. Hacé que diga un hechizo, por ejemplo "¡Avada kadabra!", si está sobre la varita mágica.

Probá con estos bloques:



Desafío 2:

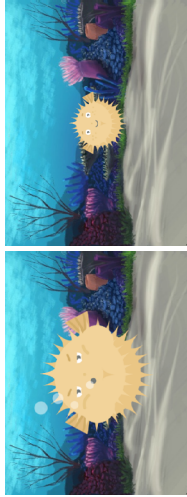
Crear y utilizar los bloques:




4

Desafío 1:

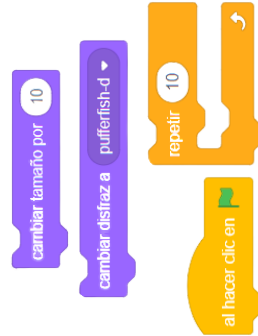
Hacer que el pez globo se enoje y luego se tranquilice.



Indicaciones:

1. Elegí un escenario.
2. Agregá un pez globo (*Pufferfish*).
3. Hacé que se enoje (*disfraz pufferfish-c*) y aumente su tamaño 10 veces al hacer clic en .
4. Hacé que luego reduzca su tamaño 10 veces y se tranquilice (*disfraz pufferfish-a*).

Probá con estos bloques:



Desafío 2:

Crear y utilizar los bloques:



5

Desafío 1:

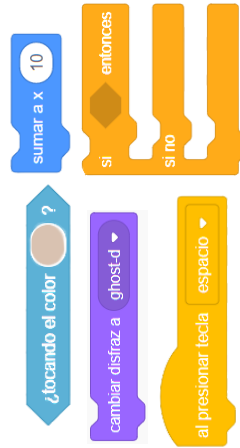
Hacer que el fantasma intente asustar a las elfas cuando pase sobre ellas.



Indicaciones:

1. Elegí un escenario.
2. Agregá un fantasma (*Ghost*) y tres elfas (*Elf*).
3. Hacé que el fantasma se mueva a la derecha al presionar la tecla \rightarrow .
4. Hacé que asuste (*disfraz ghost-c*) si está sobre una elfa o vuelva a la normalidad (*disfraz ghost-a*).

Probá con estos bloques:



Desafío 2:

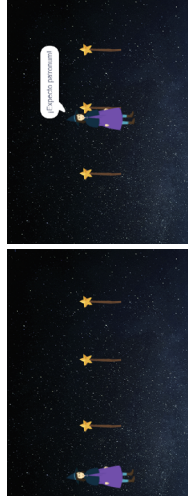
Crear y utilizar los bloques:



6

Desafío 1:

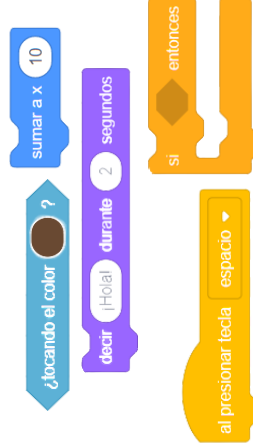
Hacer que la bruja lance un hechizo cuando pase sobre una varita mágica.



Indicaciones:

1. Elegí un escenario.
2. Agregá una bruja (*Witch*) y tres varitas (*Ward*).
3. Hacé que la bruja se mueva a la derecha al presionar la tecla \rightarrow .
4. Hacé que diga un hechizo, por ejemplo "¡Expecto patronum!", si está sobre una varita mágica.

Probá con estos bloques:



Desafío 2:

Crear y utilizar los bloques:



Tarjetas para la Actividad 3

Ubicaciones al azar

Hacer que al comenzar el programa alguno/s de los objetos o personajes se ubiquen aleatoriamente en el escenario.

Pista: Explorar la categoría Eventos y Movimiento.

Se encuentran dos personajes

Cambie el fondo y/o suene un sonido.

Pista: Definan procedimientos para no reutilizar partes del programa.

Un obstáculo

Si el personaje se encuentra con un obstáculo, debe volver a la esquina superior izquierda. Para el obstáculo, pueden agregar un objeto nuevo o usar uno que ya esté en el programa.

Pista: Definir un procedimiento o bloque que se llame "Volver si chocó".

Alcanzar un objetivo

Hacer que si un personaje alcanza un objeto u otro personaje cambie de disfraz, indicando que cumplió su objetivo.

Pista: Pueden definir procedimientos para no repetir partes del programa.

Una pared

Trazar una línea en el escenario (una pared) y hacer que, si un personaje la toca, vuelva a su posición inicial.

Pista: Dibujen la línea de un color que no aparezca en ningún otro lugar del escenario.

Movimiento permanente

Algún personaje u objeto (agregado o existente) debe moverse solo durante toda la ejecución del programa.

Pista: Revisar los bloques de la categoría Control.

"Objetivo: llegar arriba"

Hacer que cuando un personaje toca el borde superior cambie de disfraz, indicando que cumplió con su objetivo.

Pista: Explore los sensores. ¿Pueden saber qué borde está tocando un objeto? ¿Qué movimientos pueden hacer que un objeto toque el borde superior?

Dar la vuelta

Hacer que cuando un personaje toca el borde derecho, aparezca por el borde izquierdo.

Pista: Explore las coordenadas y los bloques asociados a ellas.

Pista: Explore los sensores. ¿Pueden saber qué borde está tocando un objeto? ¿Qué movimientos pueden hacer que un objeto toque el borde derecho?

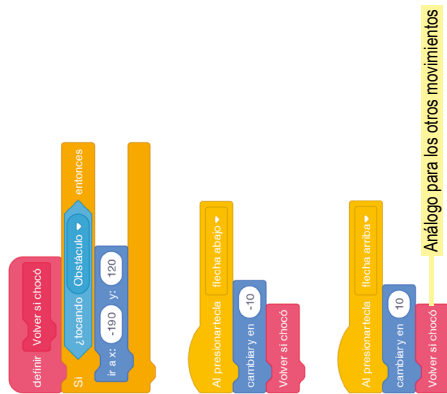
Controlar otro personaje

Permitir que el usuario mueva a otro objeto o personaje.

Pista: Puede usar otras teclas (por ejemplo) A, S, D, W.

Soluciones posibles para cada tarjeta de la Actividad 3

Un obstáculo



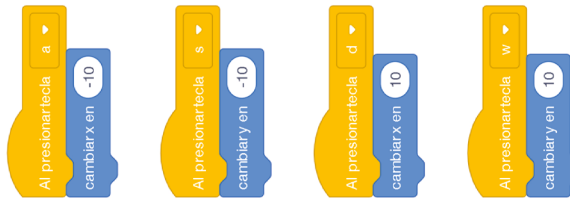
Definimos un procedimiento con una alternativa condicional para que, cada vez que se mueve el personaje, evaluemos si chocó con el obstáculo y vuelva a su lugar.

Movimiento permanente



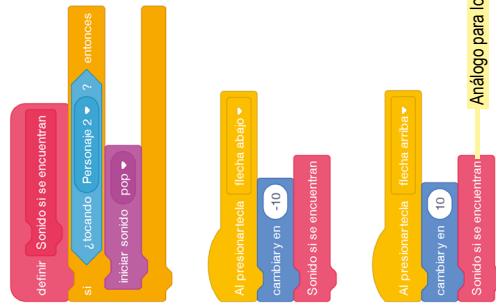
Usamos el bloque "Por siempre" con bloques de movimiento adentro.

Controlar otro personaje



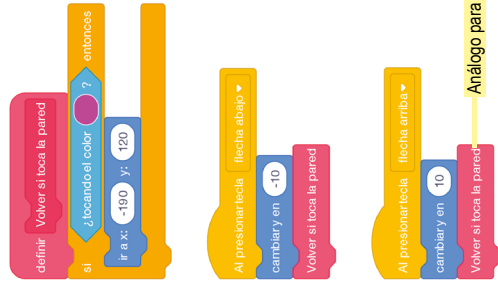
Definimos nuevos eventos del teclado en otro objeto del videojuego.

Se encuentran dos personajes



Definimos un procedimiento con una alternativa condicional para evaluar si uno de los personajes chocó con el otro y lo usamos en sus 4 eventos de movimiento.

Una pared



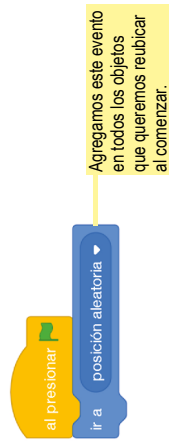
Dibujamos una línea fucsia para asegurarnos de que fuera lo único fucsia en el escenario. Luego, definimos un procedimiento con una alternativa condicional para detectar si el personaje se había movido sobre la línea usando el sensor ¿Tocado color?

Dar la vuelta



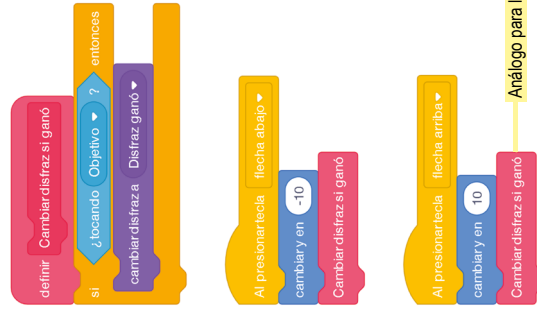
Cambiamos solo la coordenada X del personaje para desplazarlo al otro lado del escenario. Usamos una alternativa condicional con el evento ¿Tocado borde? solo en el evento de moverse a la derecha.

Ubicaciones al azar



Usamos el evento "Al presionar bandera verde" y el bloque de movimiento "Ir a posición aleatoria".

Alcanzar un objetivo



Definimos un procedimiento que evalúa si el personaje tocó al objetivo con un sensor y una alternativa condicional. Lo utilizamos en los cuatro eventos de movimiento del personaje.

Objetivo: llegar a arriba



Usamos una alternativa condicional con el sensor ¿tocando borde?; lo colocamos solo en el evento de mover hacia arriba al personaje para identificar solo cuando alcanza el borde de arriba.