

Muévete con Scratch

Aplicación en Scratch para el desarrollo del Pensamiento Computacional.

Elena Álvarez Castro

MAES. Especialidad Informática, Tecnología y Procesos Industriales. It. Informática.
Complementos de Formación de la Tecnología, Procesos Industriales e Informática
Universidad de Granada
Curso 2020/2021

Contenidos

1. Introducción.....	2
2. Contexto.....	2
3. Guion de prácticas.....	2
3.1. Conceptos Generales	2
3.2. Estructura del guion	2
4. Preguntas	3
5. Conclusiones	5
6. Material utilizado	5
7. Anexo. Guion de Prácticas	6

1. Introducción

Este proyecto tiene como objetivo realizar un guion de prácticas para que el alumnado de 4º de E.S.O. realice un programa en la plataforma Scratch, relacionado con los contenidos curriculares de la asignatura Física y Química. La práctica está destinada a desarrollar el pensamiento computacional en el alumnado. En este informe, se expone el contexto y el guion para el alumnado. Por último, se responde a las preguntas planteadas.

2. Contexto

Esta actividad está contextualizada dentro de la asignatura Física y Química de 4º de E.S.O. En el currículo establecido por el BOE en el Real Decreto 1105/2014 del 26 de diciembre, se definen los contenidos curriculares correspondientes a esta materia. En específico, en el Bloque 4: El Movimiento y las fuerzas, se define entre otros contenidos curriculares los tipos de movimientos (Sec. I. Pág. 266): el movimiento. Movimientos rectilíneo uniforme, rectilíneo uniformemente acelerado y circular uniforme.

El alumnado al que va destinado este guion de prácticas, tiene un nivel medio de Scratch, conoce la interfaz y el sistema de programación por bloques (distinguen entre los diferentes tipos de bloques y utilización de variables simples). Los conceptos relacionados con estos tres movimientos, ecuaciones y características, se deben explicar antes de realizar esta práctica.

3. Guion de prácticas

3.1. Conceptos Generales

El programa desarrollado es una historia, en la que, a través de un personaje narrador y tres objetos, se muestran los tres tipos de movimiento: movimiento rectilíneo uniforme (MRU), el movimiento rectilíneo uniformemente acelerado (MRUA) y el movimiento circular uniforme (MCU). Cada objeto, está programado para seguir un tipo de movimiento.

La historia se divide en cuatro escenarios. Comienza con un escenario de inicio, dónde el narrador da la bienvenida a la historia, a continuación, se muestra un escenario dónde un objeto realiza un MRU, después otra escena, en el que aparece un objeto que sigue un MRUA y un último escenario dónde aparece de nuevo un objeto, esta vez siguiendo MCU. En todos los escenarios permanece el personaje del narrador, ya que, es el encargado de contar la historia e identificar los movimientos en sus respectivos escenarios.

El resultado final, se puede encontrar en este enlace: <https://scratch.mit.edu/projects/498393649>

Este resultado final, se mostrará al alumnado antes de comenzar la práctica, con el objetivo mostrar cómo puede quedar su historia y aumente su motivación.

3.2. Estructura del guion

El guion de prácticas está planteado para ajustarse a diferentes ritmos de aprendizaje. Se divide en dos bloques: un bloque esencial para completar la práctica y un bloque de ampliación. El alumnado promedio, completará el primer bloque en unos 40-45 minutos, dejando tiempo para que pueda empezar la ampliación. En el caso de ritmos de aprendizaje más lentos o más rápidos, este margen de tiempo permite que el grupo complete la práctica de forma satisfactoria.

Dentro de los dos bloques, el guion se divide a su vez en diferentes actividades. Las primeras actividades se presentan más guiadas y las últimas no se expresan con tanto detalle. Esto permite que el alumnado vaya desarrollando autonomía conforme avanza en las tareas.

Las actividades planteadas en el guion son:

Bloque1	<ol style="list-style-type: none">1. Programar los tipos de movimiento<ol style="list-style-type: none">1. Movimiento rectilíneo uniforme (MRU)2. Movimiento rectilíneo uniformemente acelerado (MRUA)3. Movimiento circular uniforme (MCU)2. Establecer los escenarios de la historia3. Crear a un Narrador
Bloque 2	<ol style="list-style-type: none">4. Ampliación 1: Añade música de fondo5. Ampliación 2: Comparar los movimientos MRU y MRUA

Tabla 1: Estructura del guion de prácticas.

El guion completo se adjunta como anexo a este informe. De esta forma, está lo más preparado posible para ser impreso o compartido con el alumnado de forma online, sin necesidad de extraerlo de este informe. El formato y la redacción se han ajustado apropiadamente para que resulte lo más sencillo posible de seguir y realizar.

4. Preguntas

Pregunta 1: Indique qué habilidades (y en qué sentido) del pensamiento computacional se adquirirán con el ejercicio propuesto.

A continuación, se describen las habilidades del pensamiento computacional que se desarrollan, adquieren y/o trabajan con esta práctica:

- **Abstracción:** habilidad de obviar lo irrelevante y prestar atención a los elementos esenciales. Esta habilidad se encuentra durante todo el desarrollo de la práctica, puesto que, se realizan diferentes actividades, que juntas, forman la historia completa. Separar la realización en diferentes actividades, permite que durante el desarrollo de una en concreto, el alumnado ‘se olvide’ de las demás. Un ejemplo es cómo se realiza la programación de los tipos de movimientos. Se programa cada objeto por separado, ocultando los demás.
- **Generalización y búsqueda de patrones:** habilidad de basarse en actividades anteriores para resolver una nueva. En general, el programa descrito, se basa en repetir las mismas operaciones para cada objeto, haciendo ligeros cambios (las ecuaciones), pues, los tres objetos tienen prácticamente la misma programación. Conforme se avanza en la práctica, se van eliminando progresivamente instrucciones para realizar la programación, ya que, deben basarse en las instrucciones descritas anteriormente.
- **Descomposición:** habilidad para separar un problema en problemas más pequeños. En este caso para llegar a programar la historia completa, hay que realizar cinco pasos:

1. Programar los tipos de movimiento
 - Movimiento rectilíneo uniforme (MRU)
 - Movimiento rectilíneo uniforme (MRU)
 - Movimiento rectilíneo uniformemente acelerado (MRUA)
 - Movimiento circular uniforme (MCU)
2. Establecer los escenarios de la historia
3. Crear a un Narrador
4. Añadir música de fondo
5. Comparar los movimientos MRU y MRUA

Además, el primer punto, Programar los Movimientos, se divide a su vez en tres apartados, para facilitar su comprensión, ya que, es el primer paso y la parte más relacionada con los contenidos curriculares.

- **Evaluación:** el alumnado puede saber si está realizando correctamente los ejercicios, iniciando el programa en Scratch. En este sentido la plataforma permite realizar esto sin ningún problema, puesto que, siempre permite que se ejecute el programa (no da errores). Esto es esencial para que el alumnado vea cómo avanza su programa y las partes en las que falla.
- **Pensamiento algorítmico:** esta habilidad está presente a lo largo de toda la práctica por el hecho de realizar una programación por bloques, en la que cada paso hacia el resultado es un bloque más. Sin embargo, destaca en la programación de los movimientos, cuando hay que introducir la ecuación de cada movimiento en Scratch. En esta tarea, es necesario ir añadiendo bloques de operación para lograr llegar a resolver la ecuación completa. Por tanto, el alumnado debe ir añadiendo las operaciones en el orden correcto para encontrar la ecuación resultante.

En resumen, esta práctica cumple su objetivo de fomentar, trabajar o desarrollar el pensamiento computacional. A través de ella, el alumnado adquiere en mayor o menor medida diferentes habilidades del pensamiento computacional.

Pregunta 2: Indique qué otras competencias se pueden adquirir con el desarrollo del ejercicio y explíquelo brevemente.

En el desarrollo de la práctica, también se trabajan ciertas competencias clave como son la Comunicación Lingüística (CCL), mediante la redacción del texto del narrador. Este texto debe ser breve y concreto para explicar en poco tiempo y poco texto (capacidad de síntesis), los conceptos clave de los tres tipos de movimiento. También está presente la Competencia Digital (CD), se trabaja con una herramienta online y una programación por bloques. Para terminar con las competencias clave, por su puesto la Competencia Matemática y Competencias Básicas en Ciencia y Tecnología (CMCT), que están presentes en todo el desarrollo de la práctica.

Por otro lado, también se trabajan competencias relacionadas con la autonomía y la creatividad. La práctica se desarrolla de forma individual y el alumnado puede escoger los fondos de la historia y los objetos de dos de los movimientos (el primero se realiza con el gato de Scratch para que se centren en la programación antes de distraerse eligiendo un objeto).

Competencias STEAM, Ciencias, Tecnología, Ingeniería, Artes y Matemáticas (*Science, Technology, Engineering, Arts and Mathematics*). Las matemáticas se trabajan en el cálculo de las posiciones finales a través de ecuaciones, calcular los segundos de espera para dar tiempo a leer la historia, etc. Ciencias, tecnología e ingeniería, se trabajan mediante la programación por bloques, programar la

secuencia de escenarios, introducir bucles, incluir condiciones para que los objetos aparezcan y desaparezcan. Las artes, aunque no se desarrollan al nivel de las demás, se pueden incluir los fondos escogidos, los objetos, los sonidos, además Scratch al ser tan visual, fomenta esta competencia.

Por último, refuerza los contenidos curriculares relacionados con los movimientos MRU, MRUA y MCU. El ejercicio de crear una variable para cada elemento de la ecuación, incrementar el tiempo en un bucle y verificar que, introduciendo la ecuación los objetos realizan los movimientos, ayuda a asimilar los conceptos y entender lo que representan las ecuaciones

5. Conclusiones

La historia realiza en Scratch, no presenta un nivel avanzado de programación, sólo incluye tres bucles y todo se basa en implantar una serie de condiciones. La complejidad reside en realizar la secuencia de la historia, establecer exactamente que necesitas para cada escena. Este planteamiento consigue que, aunque el alumnado no tenga un gran nivel de programación e incluso no comprenda del todo los contenidos curriculares, pueda realizar la práctica. Considero que este aspecto es muy importante para fomentar la motivación y el interés por aprender, que al fin y al cabo son la clave para que el alumnado progrese, tanto en su vida académica como en su vida personal.

6. Material utilizado

Material de la asignatura disponible en la plataforma PRADO de la Universidad de Granada.

Real Decreto 1105/2014 del 26 de diciembre, en el que se establecen los contenidos curriculares para Educación Secundaria Obligatoria y Bachillerato.

Proyectos y Tutoriales disponibles en la plataforma Scratch:

<https://scratch.mit.edu/explore/projects/tutorials/>

7. Anexo. Guion de Prácticas

A continuación, se adjunta el guion de prácticas para el desarrollo de la historia el Scratch.

Muévete con Scratch

Introducción

En esta práctica vamos a realizar en scratch un resumen de los tres tipos de movimiento que hemos visto en clase: el movimiento rectilíneo uniforme (MRU), el movimiento rectilíneo uniformemente acelerado (MRUA) y el movimiento circular uniforme (MCU). Para esto crearemos una historia, en la que aparecerán tres objetos. Cada objeto realizará un tipo de movimiento. Un narrador será el encargado de presentar y explicar cada tipo de movimiento.

Para completar la historia seguiremos esta secuencia de actividades:

1. Programar los tipos de movimiento
 - a. Movimiento rectilíneo uniforme (MRU)
 - b. Movimiento rectilíneo uniformemente acelerado (MRUA)
 - c. Movimiento circular uniforme (MCU)
2. Establecer las escenas de la historia
3. Crear a un Narrador
4. Ampliación 1 - Añadir música de fondo
5. Ampliación 2 - Comparar los movimientos MRU y MRUA

1. Programar los tipos de movimiento

Movimiento rectilíneo uniforme M.R.U

En esta primera actividad vamos a programar el gato de Scratch para que siga un movimiento rectilíneo uniforme. Recuerda la ecuación del movimiento rectilíneo uniforme:

$$x = x_0 + vt$$

Dónde

- x es la posición final del objeto
- x_0 es la posición inicial
- v es la velocidad
- t es el tiempo

Para calcular la posición final de un objeto, necesitamos saber: su posición inicial, la velocidad y el tiempo.

Estas son las tres variables que tenemos que definir en nuestro programa. Para crear estas variables seguimos estos pasos:

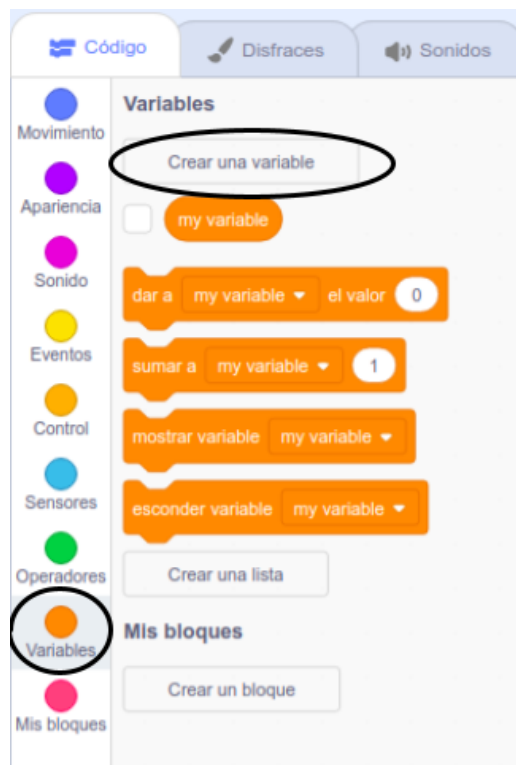
Paso 1: Seleccionamos el gato y pinchamos en el apartado de variables.

Paso 2: Selecciona 'Crear una variable'

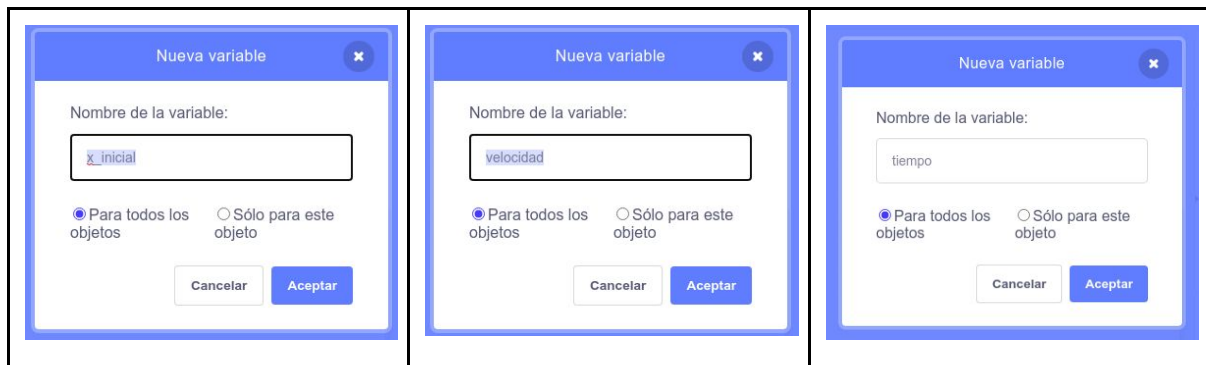
Paso 3: Introduce el nombre de las variables:

- x_inicial (será nuestra posición inicial)
- velocidad
- tiempo

Puedes crear estas variables para todos tus objetos o sólo para el objeto con el que estás trabajando. En este caso, vamos a crearlas para todos nuestros objetos y le damos a aceptar.

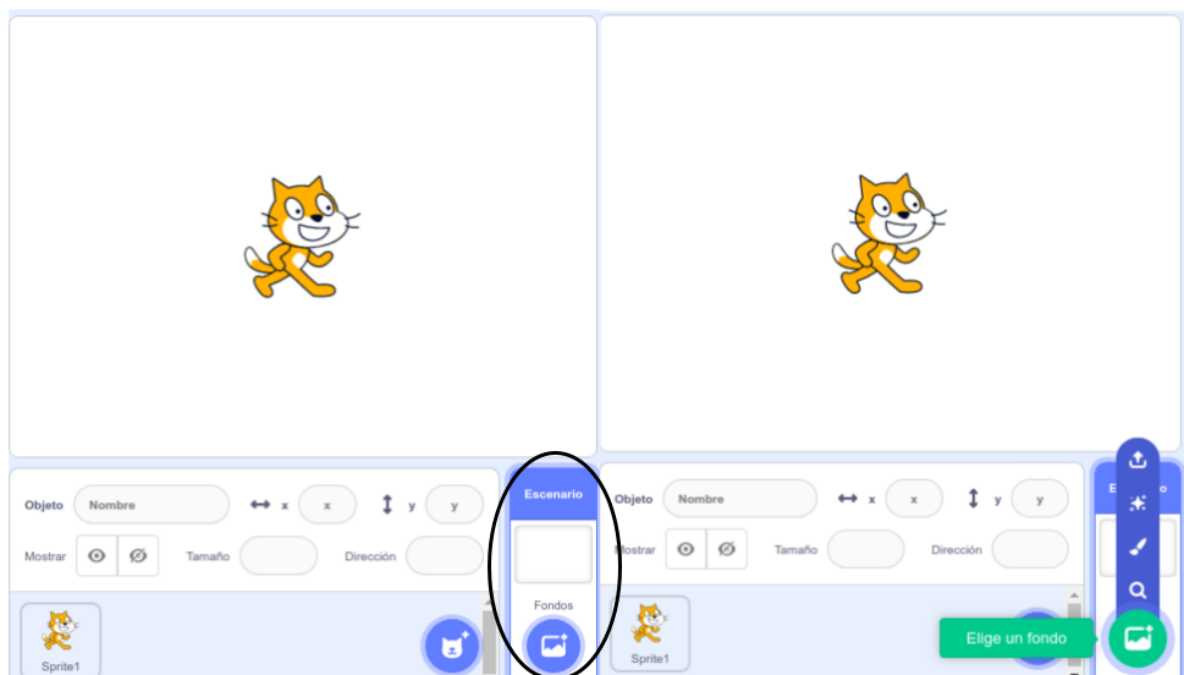


Muévete con Scratch

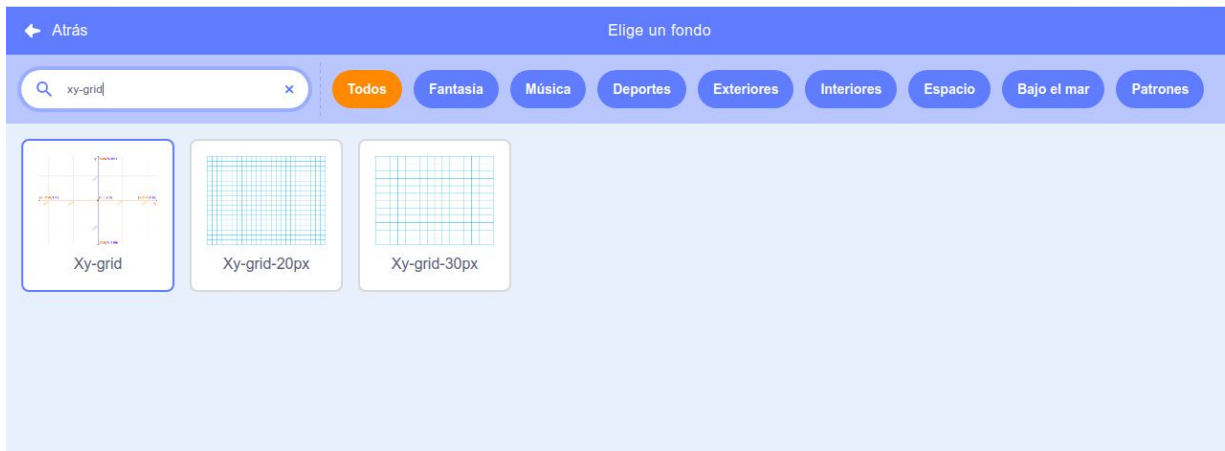


Una vez que ya tenemos nuestras variables, vamos a establecer un fondo. Vamos a poner un fondo que nos muestre los ejes cartesianos, para poder ver las diferentes posiciones que toma nuestro objeto.

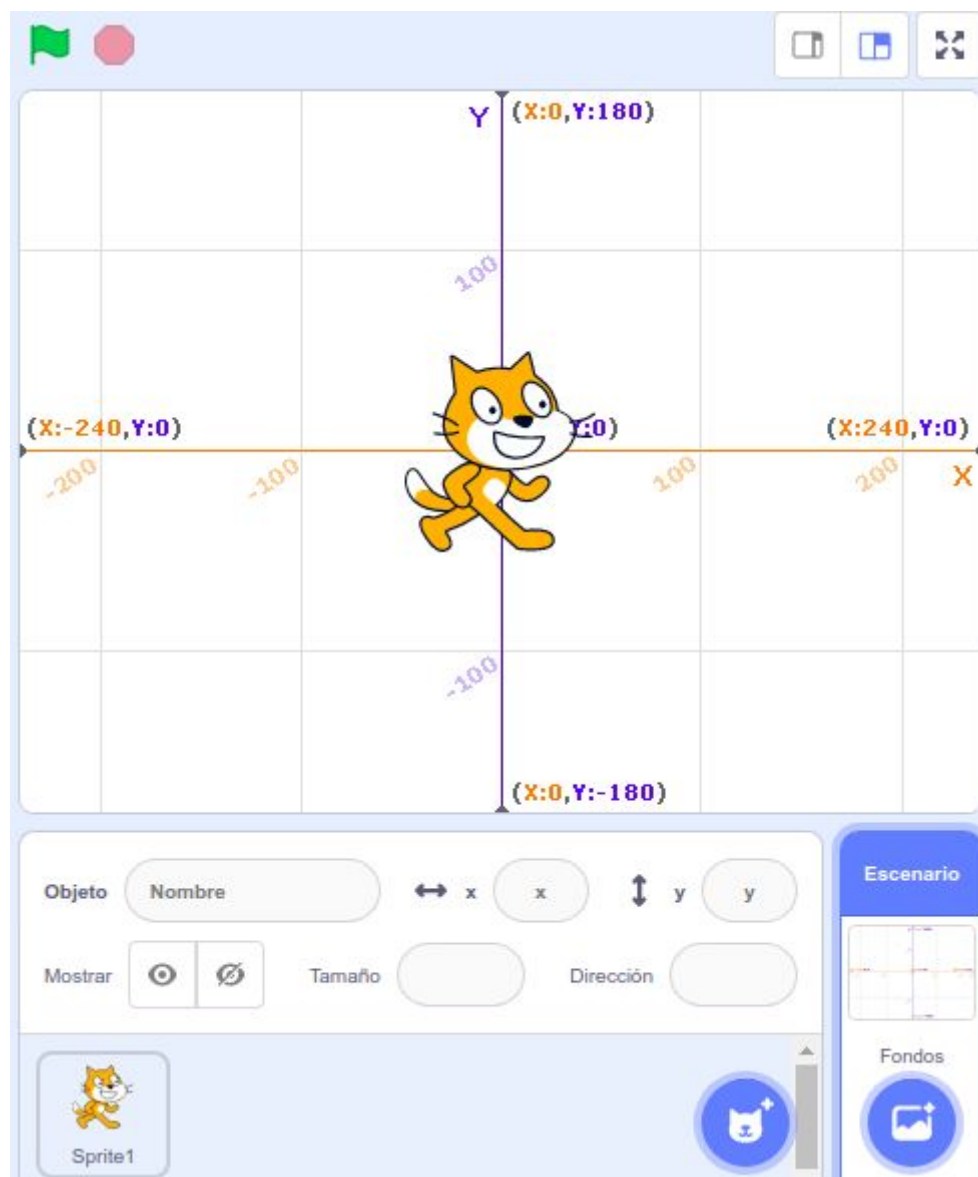
En la galería de scratch hay un fondo que muestra los ejes **X** e **Y**. En el apartado de Escenarios, selecciona **añadir un fondo** y buscan en la galería el fondo **xy-grid**



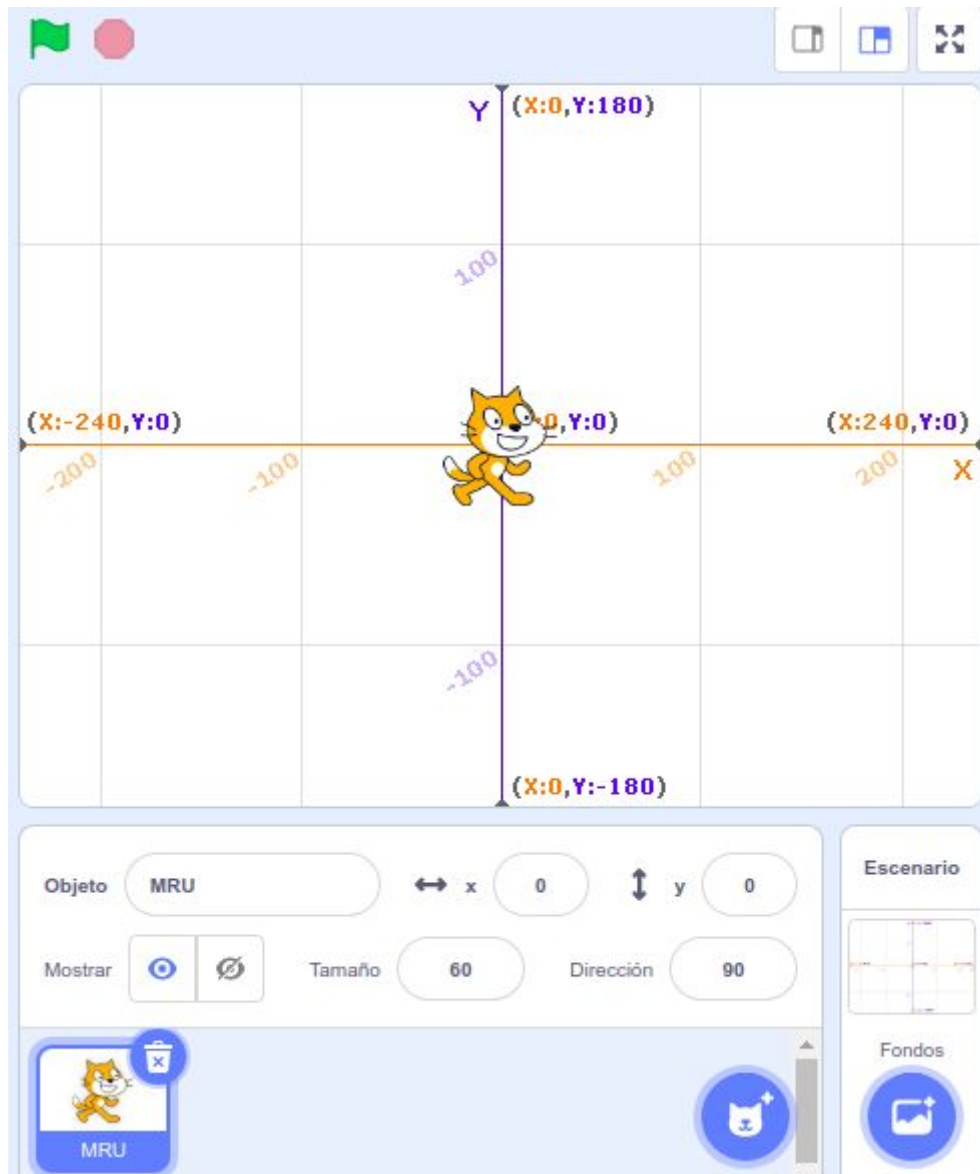
Muévete con Scratch



Cuando selecciones el fondo, volverás a la pantalla de inicio y el gato aparecerá en los ejes cartesianos.



Ajustamos el tamaño del gato para que cuadre con el fondo y le cambiamos el nombre a MRU para identificarlo mejor, ya que será el objeto que siga el movimiento rectilíneo uniforme.



Ahora vamos a programar el gato para que siga un movimiento rectilíneo uniforme. Para ello seguimos los pasos:

Paso 1: Establecemos valores iniciales para nuestras variables a través de los bloques de variable:

- **x_inicial = - 200**, ya que es el punto de nuestro eje x más a la izquierda. Poniendo este valor, nuestro objeto tendrá toda la pantalla para moverse hacia la derecha
- **tiempo=0**, partimos de que el tiempo es igual a cero

- **velocidad=10.** La unidad de medida de las distancias en scratch son píxeles. La velocidad se refiere a los píxeles por segundo a los que se mueve nuestro objeto.



Paso 2: Para asegurarnos de que nuestro objeto aparezca ya situado en los ejes, al comenzar el programa, le establecemos su posición en el eje **X** y en el eje **Y**, justo al comenzar. Introducimos un bloque de movimiento con la posición inicial.

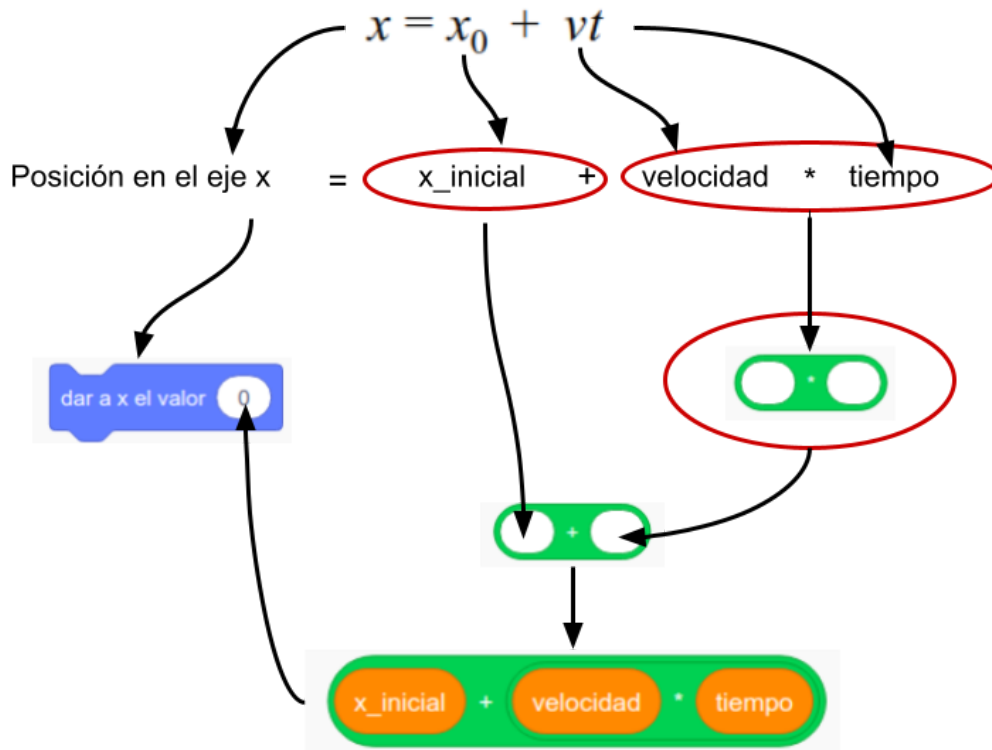


Paso 3: Para lograr que nuestro objeto se mueva en función del tiempo, tenemos que ir dándole valores a esta variable progresivamente. Para esta tarea utilizamos un **bloque de control**, un bucle, en el que vamos aumentando el valor del tiempo una unidad, cada vez que se repite el bucle.



Paso 4: Para que el gato siga un MRU, tenemos que introducir nuestra ecuación del MRU en el valor de X. De esta forma, cada vez que aumente el tiempo, se vuelve a calcular una posición final. La ecuación se construye a través de **bloques operadores** y nuestras propias **variables**.

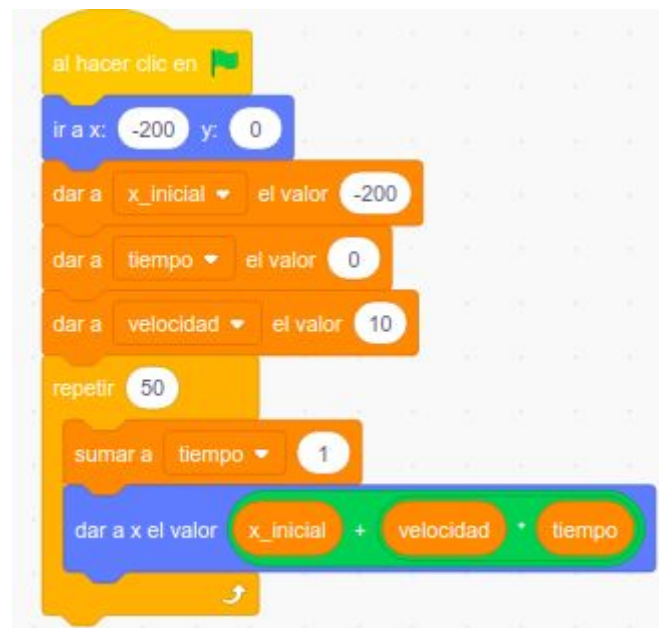
Crea la ecuación del MRU en scratch siguiendo este esquema:



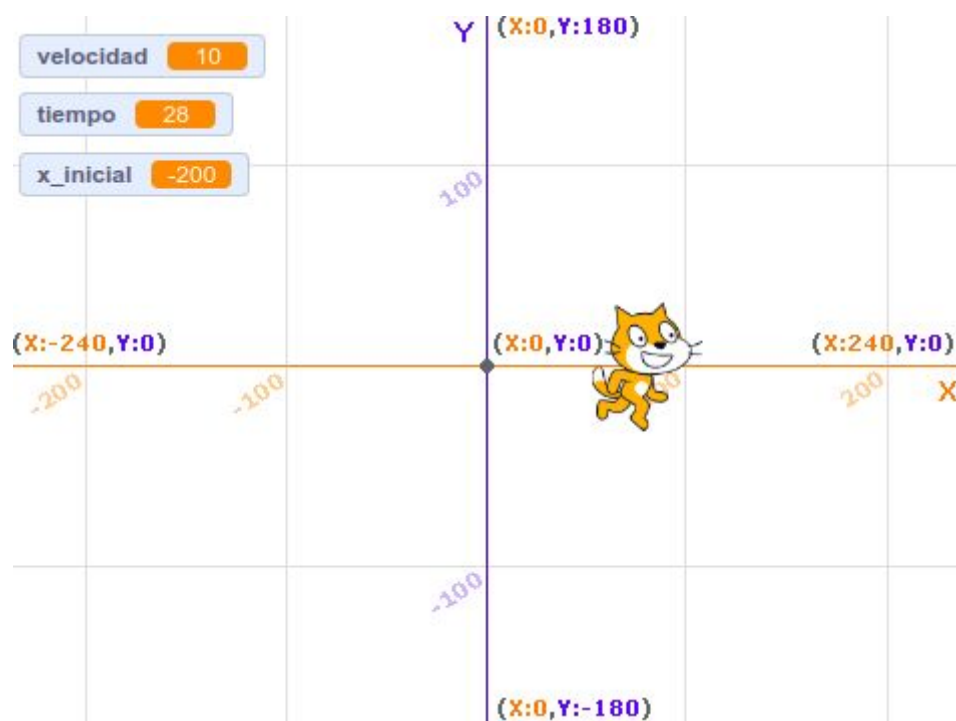
El bloque resultante para calcular la posición en el eje X será este:



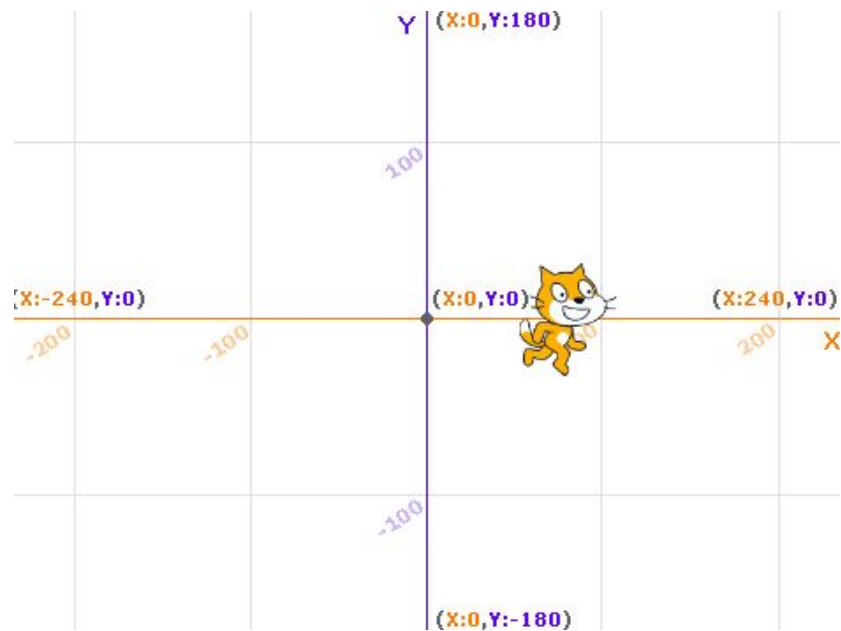
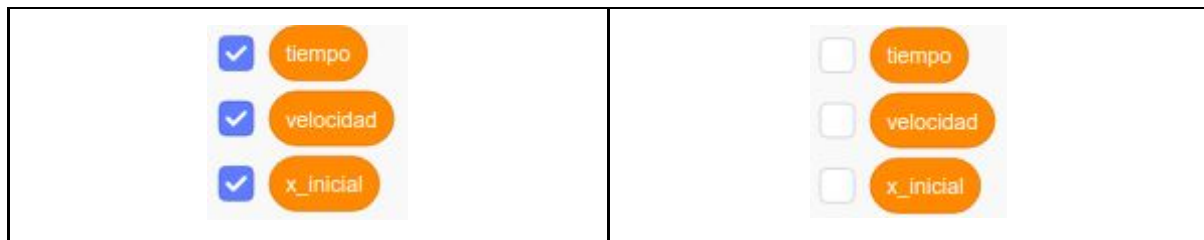
En el bucle repetir, ponemos 50, esto significa que las instrucciones que hay dentro se repetirán 50 veces. Este número de repeticiones da tiempo a que el gato llegue al final de la pantalla. El programa total para que nuestro gato realice un movimiento rectilíneo uniforme es:



Con este programa nuestro objeto sigue un movimiento rectilíneo uniforme. si hacemos click en la bandera, se moverá de izquierda a derecha con un MRU



El valor de las variables, se muestran por defecto en el fondo. Para ocultarlas, tienes que desactivar el tick de la izquierda en el apartado de variables:



Movimiento Rectilíneo Uniformemente Acelerado M.R.U.A

Una vez realizado el MRU para el gato, vamos a programar otro objeto para que siga un MRUA. Los pasos son los mismos que para el MRU, sólo hay que ajustarlo para cumplir la ecuación del movimiento uniformemente acelerado.

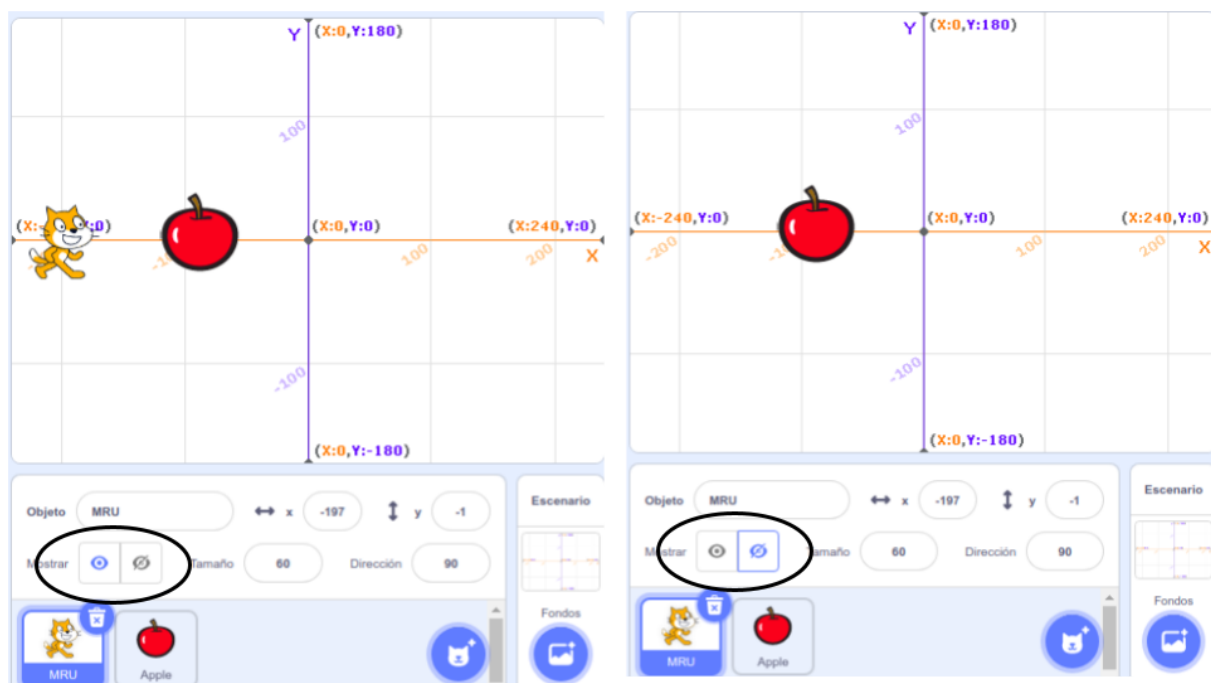
Elegimos otro objeto. Por ejemplo, hemos cogido una manzana, tú puedes escoger el que quieras de la galería.

Muévete con Scratch

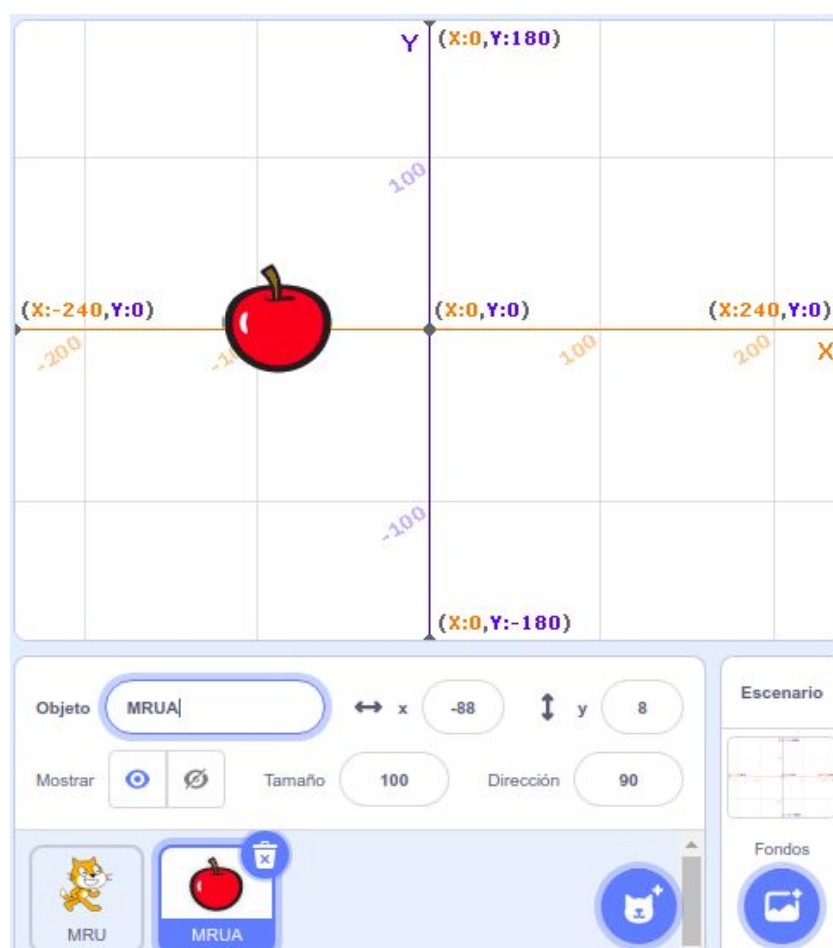


Mientras estamos programando este objeto, **oculta el gato**, así no nos molestará mientras programamos este nuevo objeto. Para ocultarlo, selecciona el ojo tachado abajo a la derecha, mientras está seleccionado el gato

Muévete con Scratch



Ponle el tamaño que mejor se ajuste al fondo. En el caso de manzana, la vamos a dejar a 100.



Ahora, seleccionamos la manzana y vamos a programarla para que siga un movimiento rectilíneo uniformemente acelerado. Recordamos la ecuación:

$$x = x_0 + v_0 t + 0.5 a t^2$$

Dónde

- x es la posición final
- x_0 es la posición inicial
- v_0 es la velocidad inicial
- t es el tiempo
- a es la aceleración

Necesitamos introducir dos variables más para este objeto: la **velocidad inicial** y la **aceleración**. En este caso, al crearlas, vamos a seleccionar crearlas **solo para este objeto**, ya que es una particularidad del MRUA.



Una vez que tenemos nuestras variables, seguimos los mismos pasos que para el MRU:

Paso 1: Establecer valores iniciales para nuestras variables: La velocidad inicial es cero porque el objeto inicialmente está parado.

Paso 2: Para asegurarnos de que nuestro objeto aparezca ya situado en los ejes, al comenzar el programa, le establecemos su posición los ejes **X** e **Y**, justo al comenzar. Introducimos un bloque de movimiento con la posición inicial.

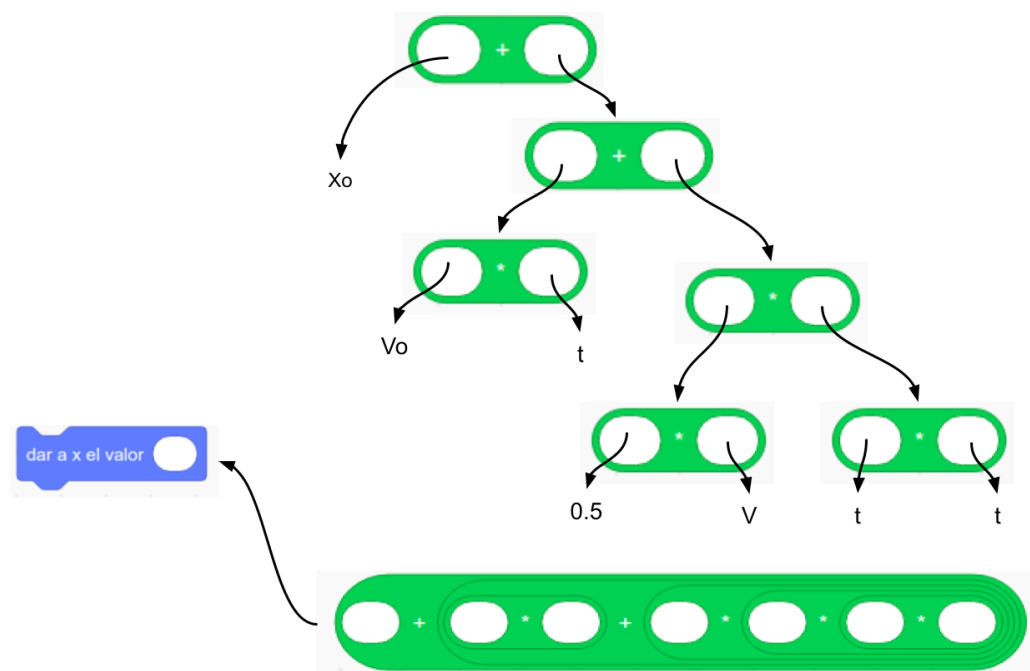


Paso 3: Ponemos el bucle para ir aumentando el valor del tiempo



Paso 4: Introducimos la ecuación del MRUA. Mediante bloques de operación como la anterior ecuación.

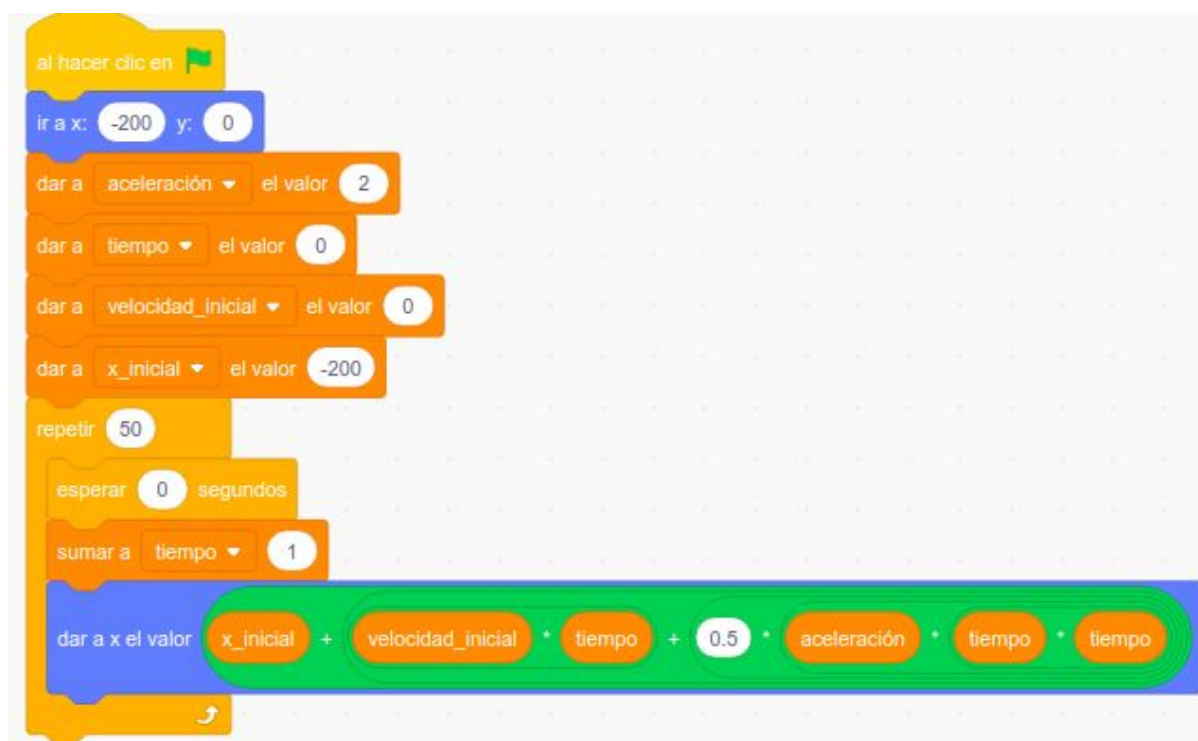
$$x = x_0 + v_0 t + 0.5 * a * t * t$$



El bloque resultante es:

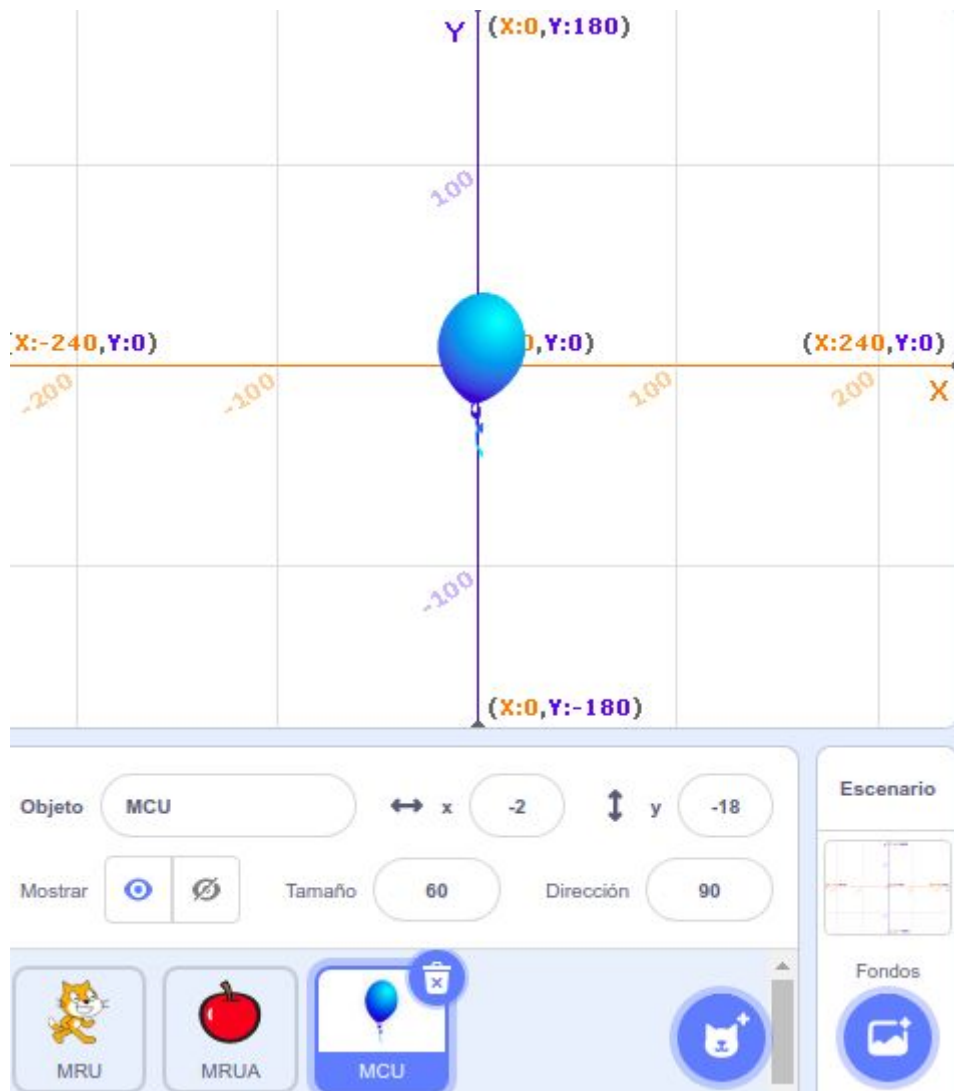


El programa completo queda como el de la siguiente imagen. Presiona la bandera y verás el movimiento.



Movimiento circular uniforme M.C.U.

Elige otro un objeto nuevo para programar este movimiento y oculta los dos anteriores. Puedes ponerle un nombre reconocible y ajustar su tamaño a la pantalla si es necesario. En este ejemplo hemos escogido un globo.



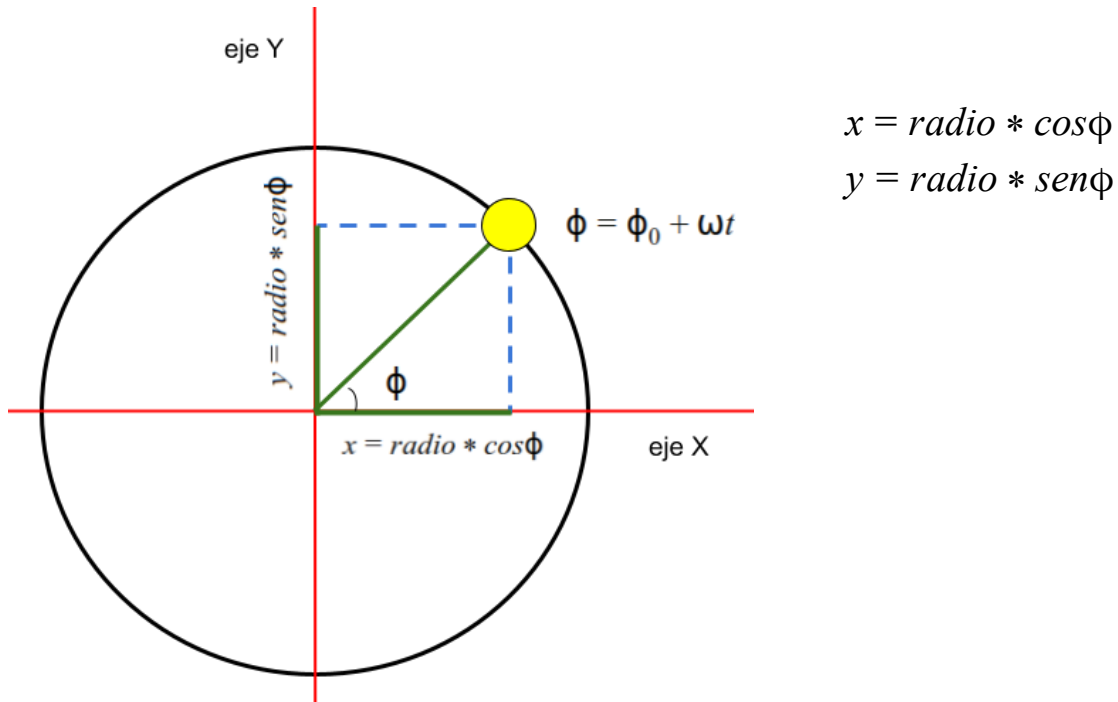
En el movimiento circular, nos movemos tanto en el eje **X** como en el eje **Y**, por tanto, tenemos que calcular la posición en cada eje.

La ecuación del movimiento circular uniforme es:

$$\phi = \phi_0 + \omega t$$

Dónde ω es la velocidad angular, ϕ es el ángulo y ϕ_0 es el ángulo inicial. Para calcular la posición en el eje **X**, y, la posición en el eje **Y** de nuestro objeto,

descomponemos el ángulo, multiplicando su radio por el seno y el coseno del ángulo de su posición, tal y como aparece en el esquema.



En este caso tenemos, **dos variables nuevas**, el **radio** y el **ángulo**. Creamos estas dos variables sólo para este objeto.



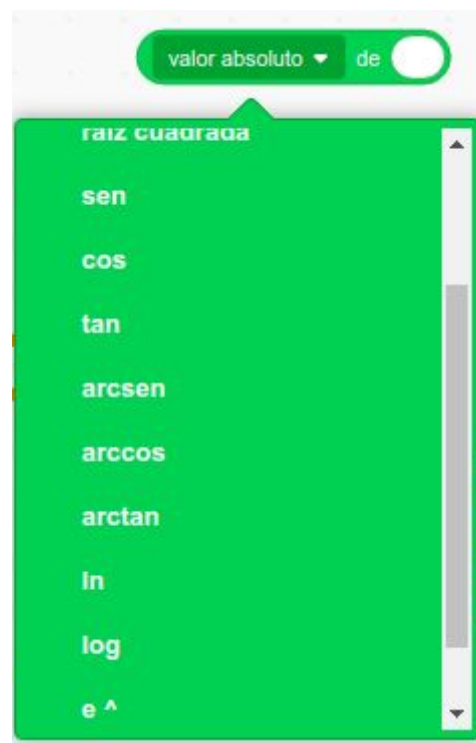
Seguimos los pasos para programar un movimiento circular:

Paso 1: Dar valor a las variables

Paso 2: Establecer una posición inicial



Paso 3: Crear el bucle y el cálculo de la posición en el eje **X** y en el eje **Y**. Para calcular el coseno y el seno del ángulo, utilizamos este **bloque de operadores**, donde puedes escoger la operación que necesites.

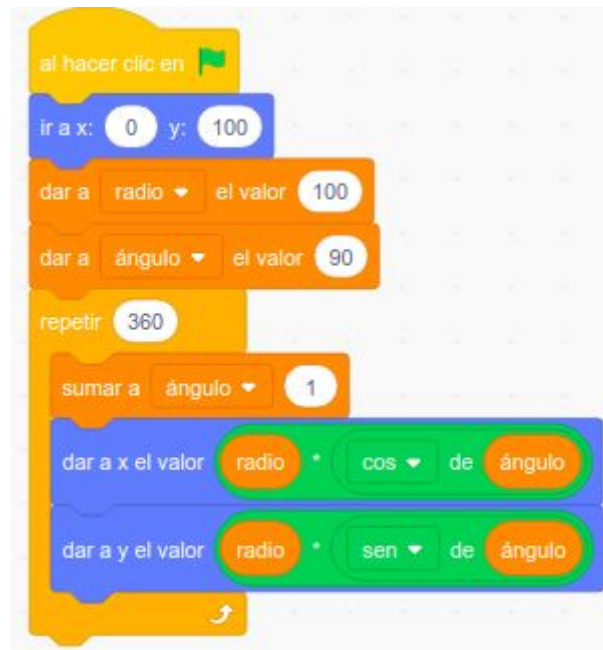


La ecuación resultante, tiene un valor para el eje **X** y un valor para el eje **Y**.



En este caso, en el bucle no aumentamos el tiempo, ya que, nuestras ecuaciones para la posición en los ejes de coordenadas no dependen del tiempo, sino del ángulo. Por tanto, **la variable que tenemos que aumentar en el bucle es el ángulo**.

Puesto que, una vuelta completa son 360° , para que nuestro objeto de una vuelta completa, necesitamos que el bucle se repita 360 veces, ya que, cada vez que el bucle se repite, el ángulo suma una unidad. El resultado de la programación para el movimiento circular es el siguiente:



Si pulsas en la bandera, verás a tu objeto realizar un movimiento circular uniforme.

Establecer las escenas de la historia

¡Ya tenemos nuestros tres movimientos!. Ahora vamos a crear una historia en la que vayan apareciendo de uno en uno. Es decir, una secuencia para mostrar nuestros movimientos uno detrás de otro.

Vamos a crear una pantalla de inicio de la historia. Para esto, ve a la parte de escenarios y añade un fondo. En este ejemplo hemos elegido blue sky.

Escena 1	Escena 2	Escena 3	Escena 4
Pantalla de Inicio	MRU	MRUA	MCU

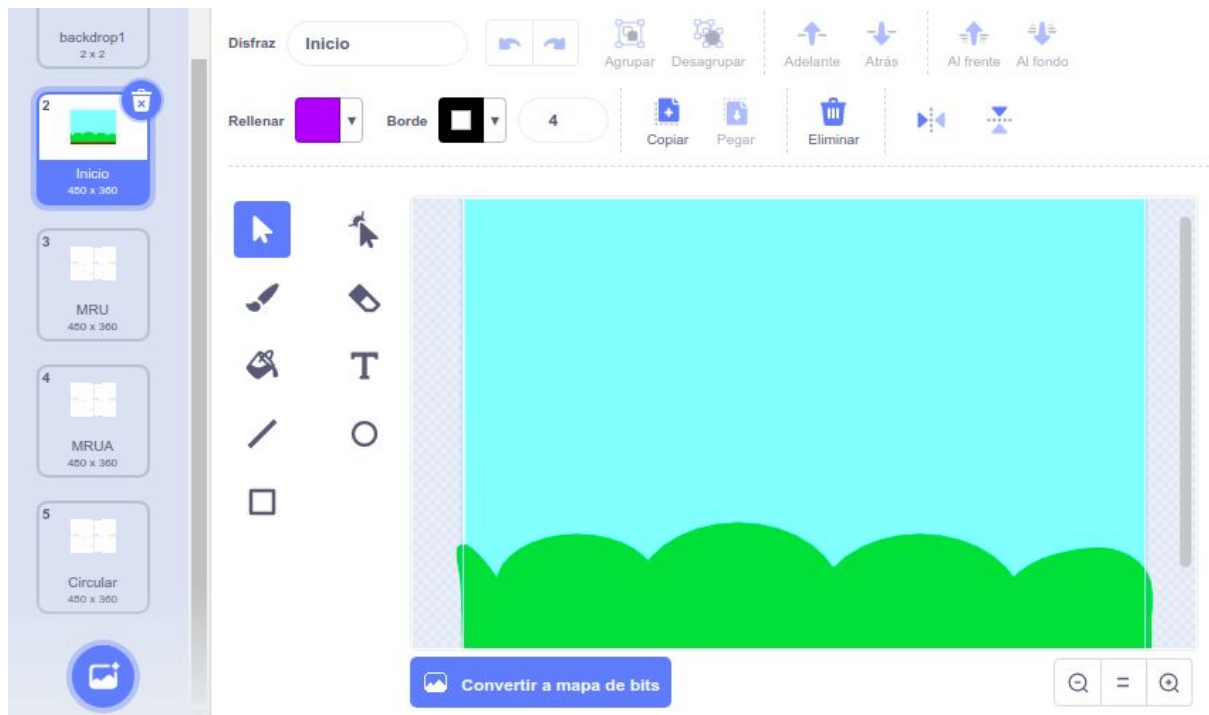
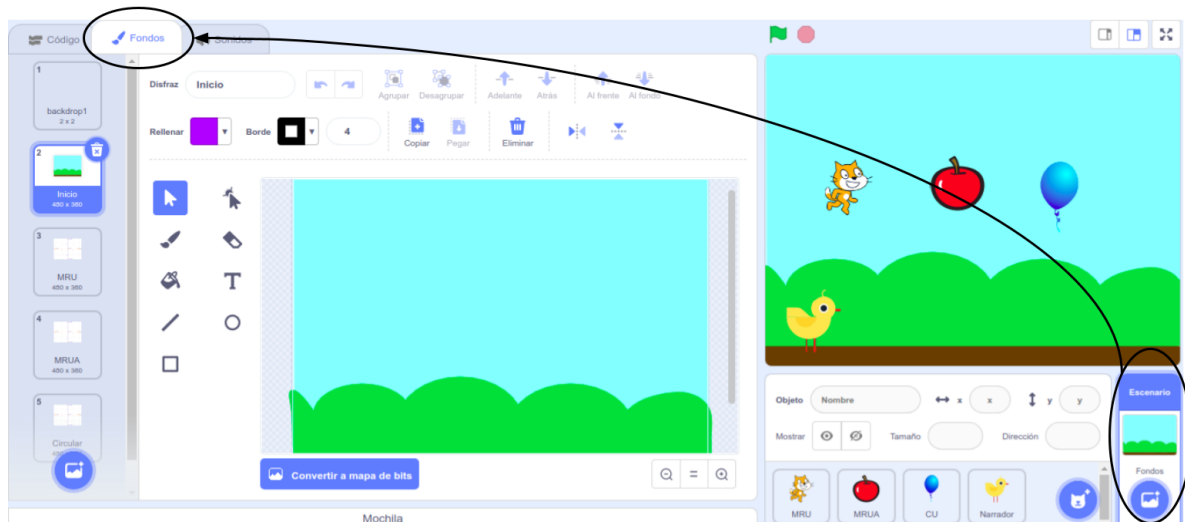
Ahora, creamos un fondo específico para cada movimiento, puede ser el mismo o puedes ir cambiando de fondos.

Cada fondo será una escena en nuestra historia. Al seleccionar **Escenarios**, entramos en la pestaña de **Fondos** (arriba a la izquierda) y establecemos una secuencia de fondos:

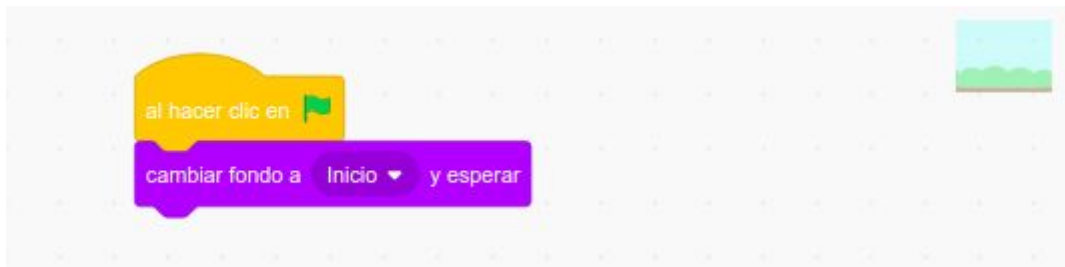
Muévete con Scratch

- Fondo para inicio
- Fondo para MRU
- Fondo para MRUA
- Fondo para circular

De tal forma que tendremos cuatro fondos. Puedes cambiarle el nombre a cada fondo para que sea más fácil identificarlos.



Vuelve a la pestaña de código y selecciona Escenario (abajo a la derecha). **Establece que cuando empiece el programa al hacer clic en la bandera, el fondo se cambie al fondo que has establecido para el inicio.**



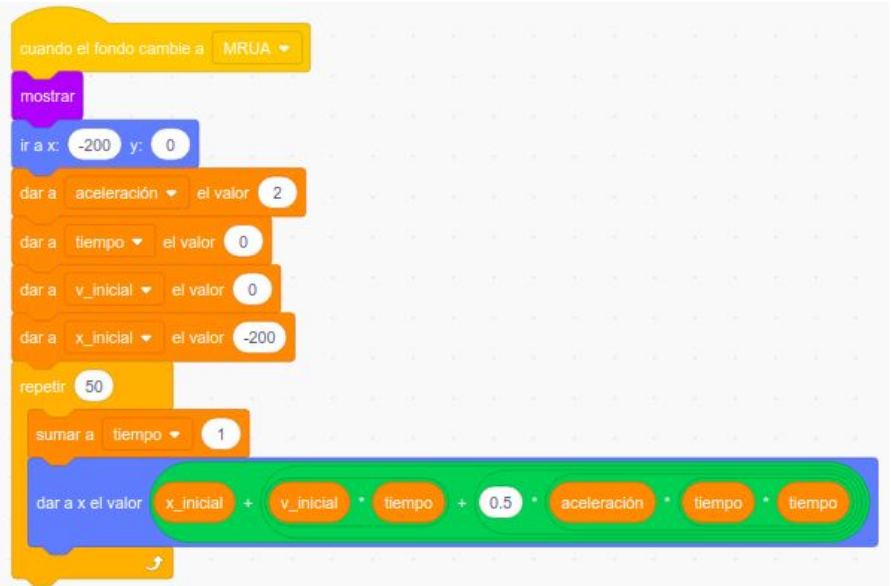

Ahora vamos a ajustar nuestros objetos para que también aparezcan en la ventana de inicio. En el código de cada objeto, incluye que **al hacer clic en la bandera, se muestran en una posición determinada para el inicio**. En este ejemplo los hemos puesto centrados en la pantalla.

MRU	MRUA	Circular

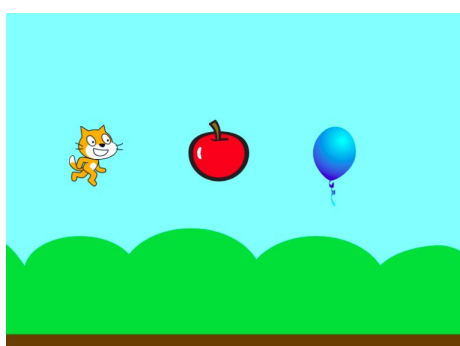
Para controlar cuando empiezan a realizar los movimientos, tenemos que poner otra condición que no sea hacer clic en la bandera, pues si hacemos eso, empezarán a realizar los movimientos en la pantalla de inicio.

Puesto que cada movimiento tiene su fondo. Vamos a **incluir la condición de que empiecen a realizar su movimiento cuando se cambie a su fondo establecido**, quedando bloques como los de la imagen para cada objeto.

MRU	
-----	--

<p>MRUA</p>	 <pre> cuando el fondo cambie a MRUA mostrar ir a x: -200 y: 0 dar a aceleración el valor 2 dar a tiempo el valor 0 dar a v_inicial el valor 0 dar a x_inicial el valor -200 repetir 50 sumar a tiempo 1 dar a x el valor x_inicial + v_inicial * tiempo + 0.5 * aceleración * tiempo * tiempo </pre>
<p>Circular</p>	 <pre> cuando el fondo cambie a Circular mostrar ir a x: 0 y: 100 dar a radio el valor 100 dar a ángulo el valor 90 repetir 360 sumar a ángulo 1 dar a x el valor radio * cos de ángulo dar a y el valor radio * sen de ángulo </pre>

Incluyendo estos bloques en cada objeto, al hacer click en la bandera, nos aparecerá la imagen de inicio con los tres objetos en las posiciones iniciales que hemos establecido y en el fondo que hemos elegido para inicio.



Para cambiar de fondo y que aparezcan nuestros objetos realizando los movimientos, vamos a añadir un narrador que cuente la historia.

Crear un personaje Narrador

Para crear una historia, vamos a añadir un personaje narrador, que aparecerá en todas las escenas y nos explicará en qué consiste cada movimiento.

Añade un nuevo objeto al programa y sitúalo en una esquina de la pantalla. Este personaje nos explicará el movimiento en cada pantalla. Antes de ponerle lo que nos dirá, vamos a programar sus tareas.

Tareas del Narrador

- Tiene que aparecer en la pantalla de inicio cuando empiece el programa, decir un texto, que por ahora es hola, y cambiar al siguiente fondo, que será el del MRU.



- Cuando está en la pantalla de MRU dirá algo relacionado con este movimiento



- Y así con los otros dos movimientos



¡Perfecto, ya tenemos la programación base de la historia!

Ahora vamos a ir objeto a objeto, para establecer cuando se tiene que mostrar y cuando no

- Objeto del MRU
 - Se muestra al inicio
 - Se muestra en la pantalla de MRU con su programación (ya lo tenemos establecido de antes)
 - No se muestra en la pantalla de MRUA
 - No se muestra en la pantalla del movimiento circular



- Objeto del MRUA
 - Se muestra al inicio
 - No se muestra en la pantalla de MRU
 - Se muestra en la pantalla de MRUA con su programación (ya lo tenemos establecido de antes)
 - No se muestra en la pantalla del movimiento circular

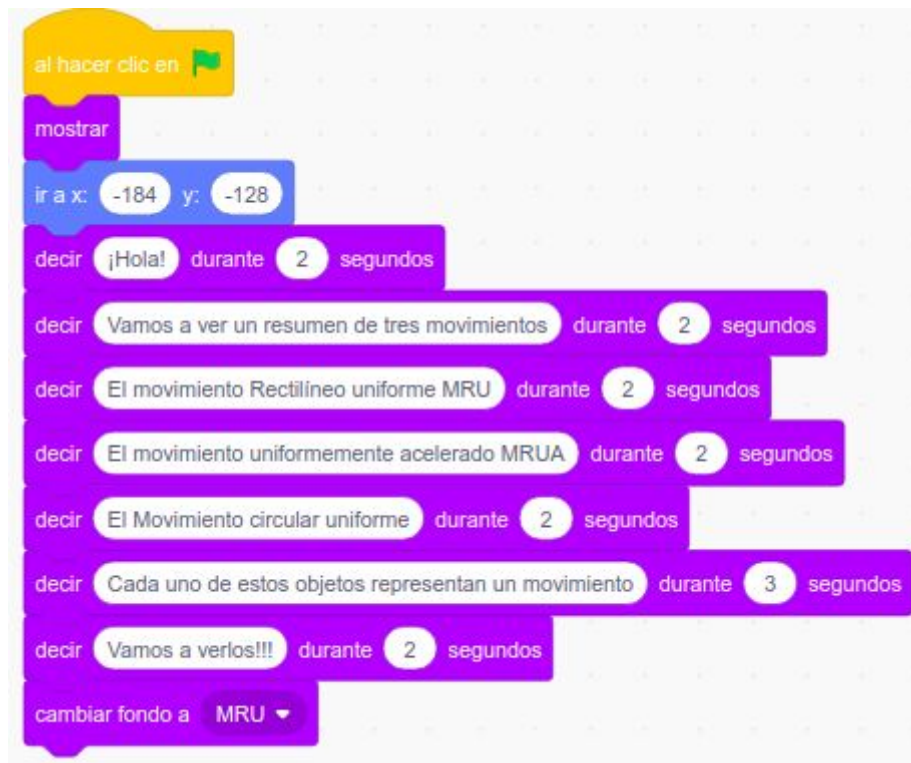


- Objeto del movimiento circular
 - Se muestra al inicio
 - No se muestra en la pantalla de MRU
 - No se muestra en la pantalla del MRUA
 - Se muestra en la pantalla del movimiento circular con su programación (ya lo tenemos establecido de antes)



Ya podemos crear la historia. Para esto, vamos a darle texto a nuestro narrador y establecer cuándo cambiamos de fondo.

El primer cambio de fondo lo hace nuestro narrador después de dar la bienvenida. Establece un texto de bienvenida para tu narrador e indícale que cambie al fondo del MRU cuando acabe:



En el siguiente fondo tenemos el MRU. Vamos a establecer que nuestro narrador diga algo sobre este movimiento y después, nuestro objeto realizará el movimiento.

Para esto, añadimos un bloque de control esperar en la programación del objeto que realiza el MRU. **Tendrá que esperar el tiempo que el narrador esté hablando.**

Cuando termine de hacer el MRU, podemos pasar al siguiente fondo, así que añadimos también en la programación del objeto la instrucción correspondiente

Narrador en MRU



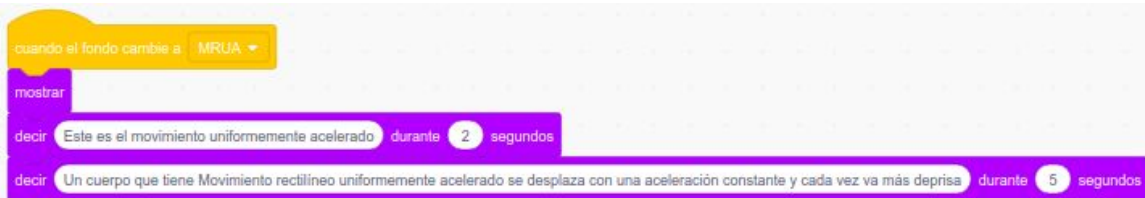
Objeto MRU + esperar y pasar al siguiente



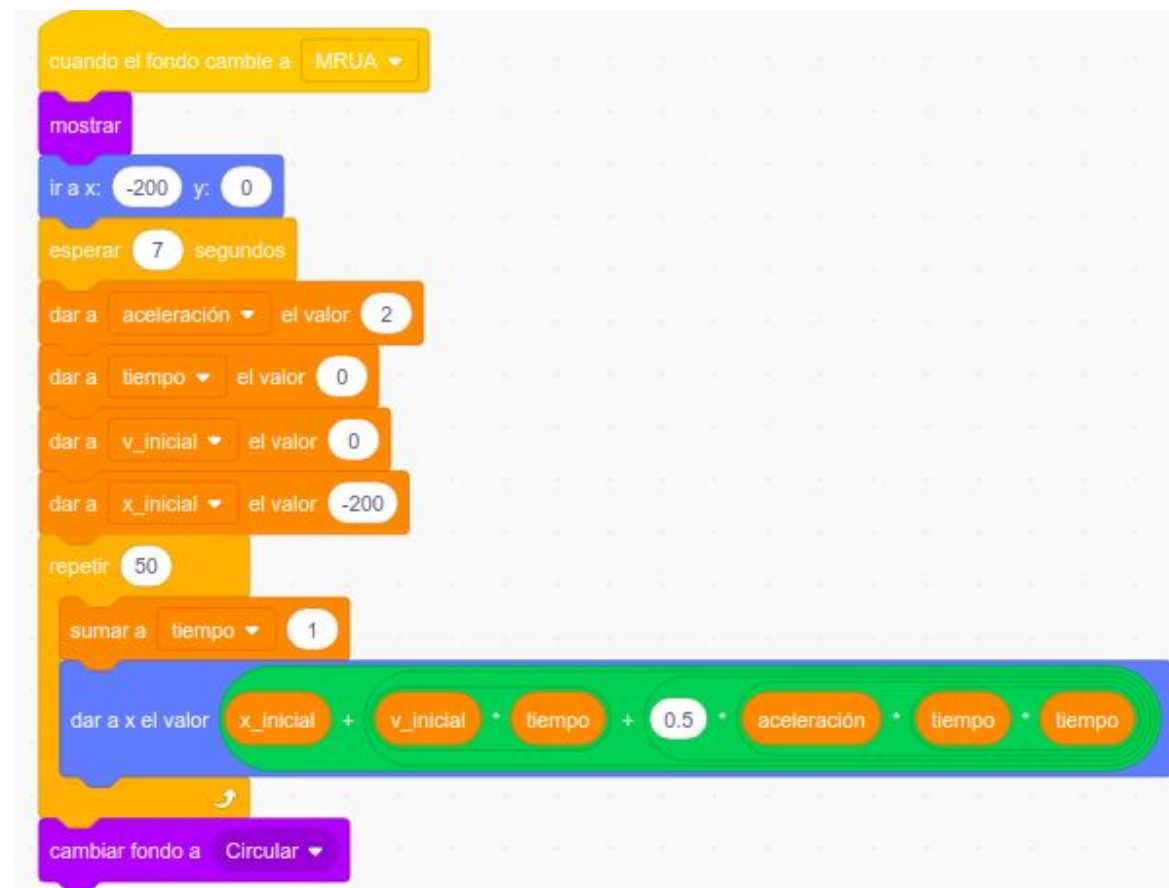
Puedes añadir esperar unos segundos antes de cambiar, así evitas que el cambio sea demasiado brusco

Ahora nos vamos al MRUA. Establecemos lo que dice nuestro narrador de este movimiento y le decimos al objeto que realiza el movimiento que cambie de pantalla cuando acabe

Narrador en MRUA



Objeto MRUA+pasar al siguiente



Ahora nos queda el movimiento circular

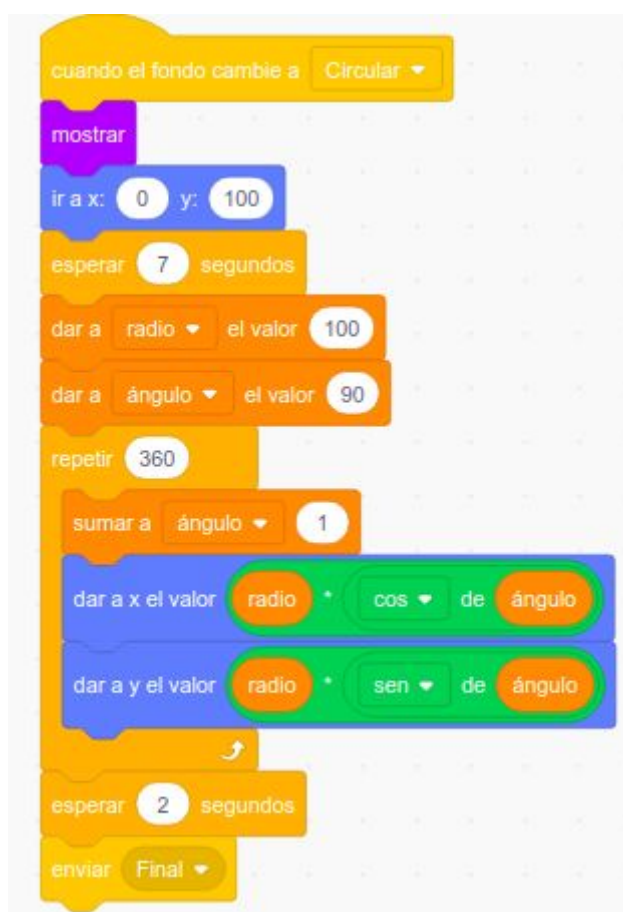
Narrador en Movimiento circular



Puesto que el movimiento circular es el último, al terminar volverá a la pantalla de inicio. Para avisar de esto a todos los demás, vamos a crear un mensaje.

En el bloque de control 'enviar mensaje', seleccionamos nuevo mensaje y le ponemos 'final'. Cuando el movimiento circular termine, enviaremos el mensaje final.

Movimiento circular + esperar + pasar al siguiente



Ya sólo queda decirle a todos los demás objetos que cuando reciban el mensaje final, vuelvan a la pantalla de inicio. Nuestro narrador, cambiará el fondo y se despedirá y terminaremos la historia. Establece una posición para cada objeto en la despedida. Puede ser la misma que la inicial.

Narrador recibe el mensaje final



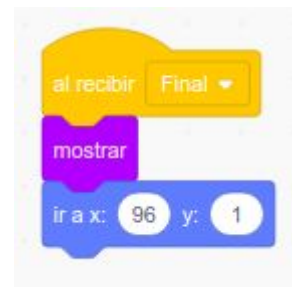
Objeto MRU recibe el mensaje final



Objeto MRUA recibe el mensaje final



Objeto movimiento circular recibe el mensaje final



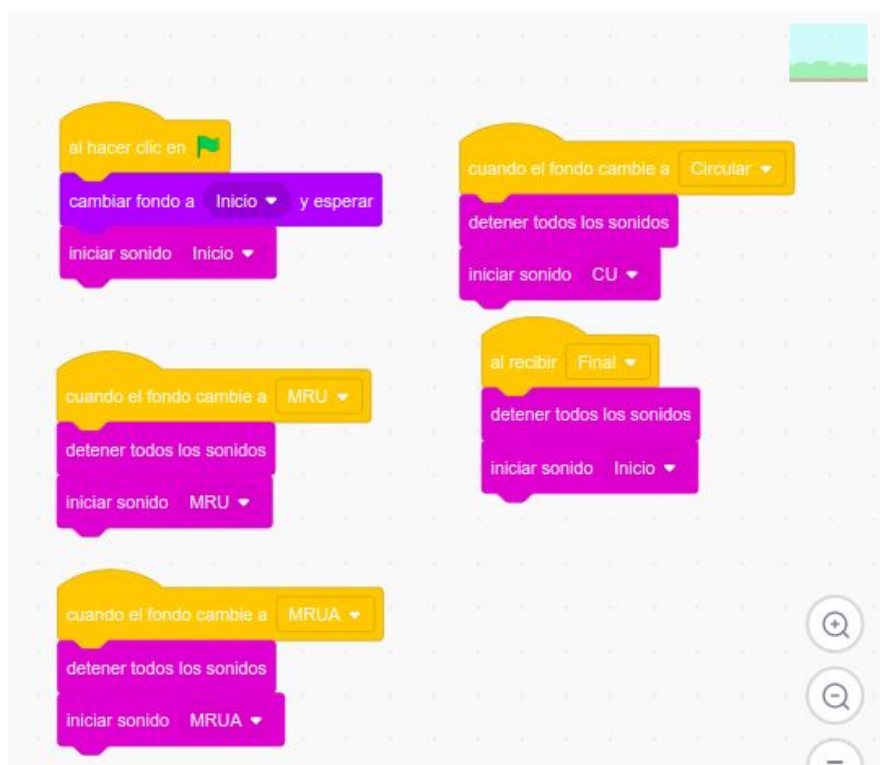
Ahora pulsa la bandera y verifica que la historia sigue su curso. Personaliza los textos para explicar los tres tipos de movimientos con tus palabras.

Ampliación 1 - Añadir Música de fondo

Para que sea más ameno, vamos a poner música en cada movimiento. Selecciona Escenarios y en la pestaña de sonido (arriba a la derecha), selecciona cuatro sonidos, uno para cada movimiento.



Introduce una música diferente en cada movimiento. A través de Escenarios, establece que cambie de sonido cada vez que cambie de fondo.



Ampliación 2 - Comparar MRU y MRUA

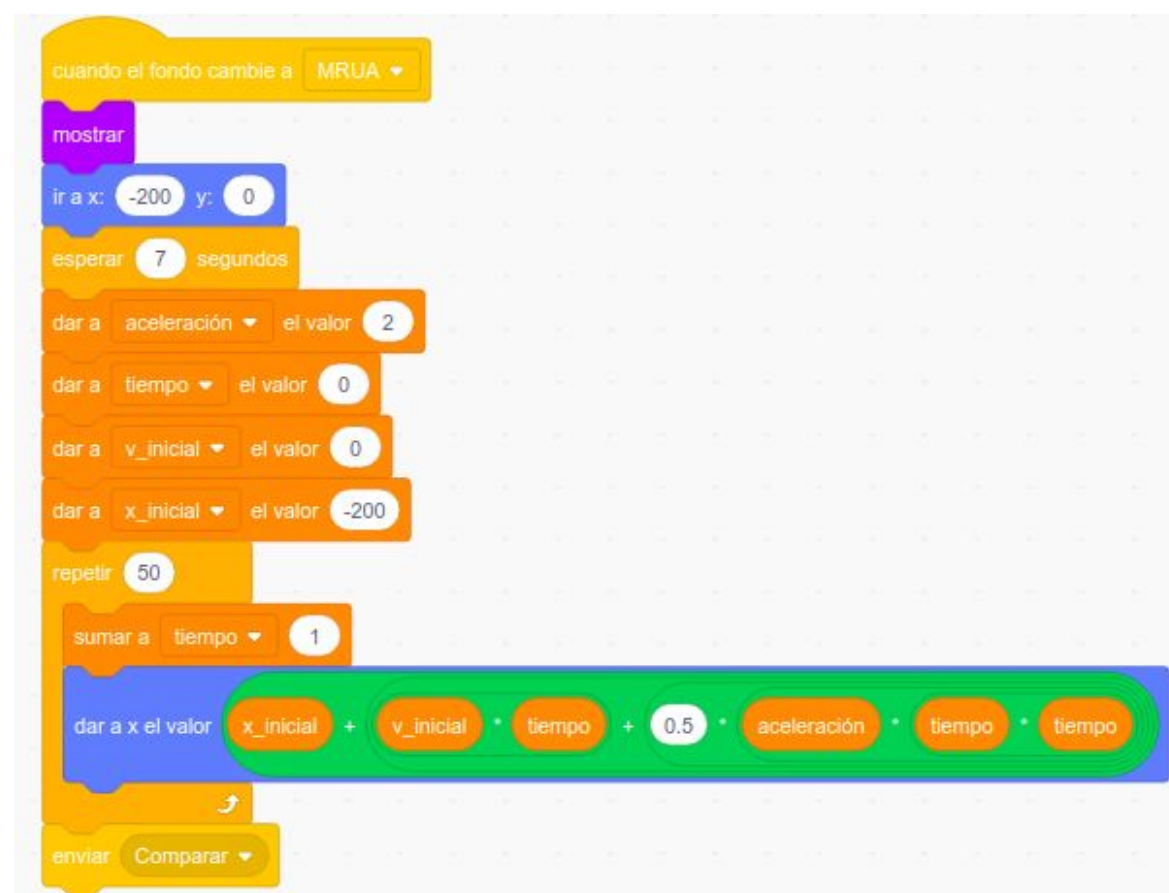
Si ya has terminado todo lo anterior, vamos a realizar otro reto: Añadir a la historia una escena en la que se muestran el MRU y el MRUA al mismo tiempo, para poder comparar la diferencia entre ellos.

Para realizar esto, vamos a utilizar de nuevo la instrucción de enviar un mensaje. Cuando lleguemos al MRUA, enviaremos un mensaje a nuestro narrador y al MRU, así podremos mostrar la comparación.

Sigue los siguientes pasos:

- Al terminar el MRUA, en vez cambiar de fondo, envía el mensaje comparar

MRUA



- Avisa a tu narrador para que explique lo que pasará

Narrador recibe comparar



- El MRU recibe comparar y realiza su movimiento. añade un nuevo sistema de bloques duplicando la programación, así no perderás lo anterior y se añadirá como una ventana más de la historia

MRU recibe comparar.

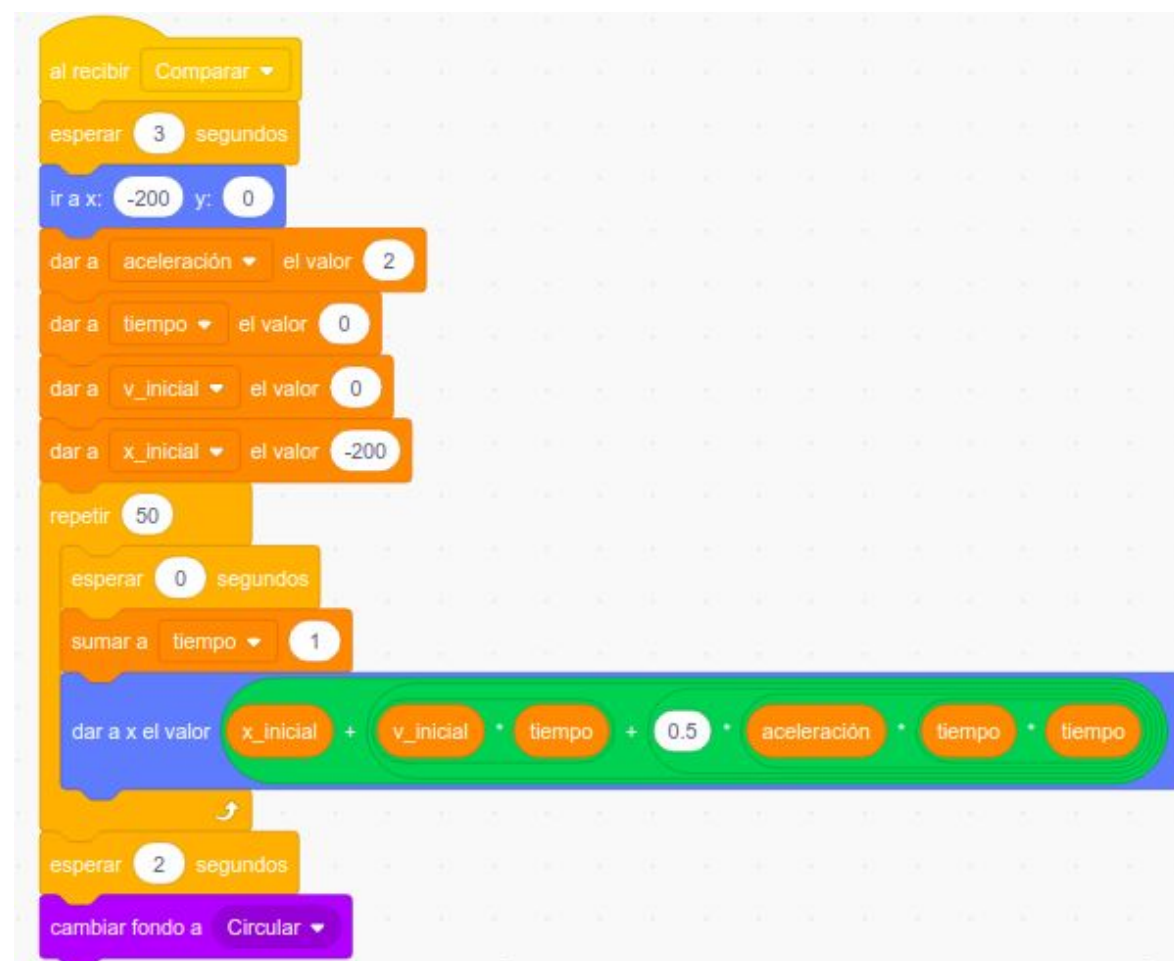


Como puedes ver, la posición en la que empieza es diferente a la del programa principal, tiene la $y=100$, esto es para que aparezca encima del MRUA y se puedan comparar correctamente.

- MRUA recibe el mensaje comparar y realiza su movimiento. Recuerda crear un nuevo sistema de bloques para este cometido y poner el mismo tiempo de espera que en MRU, así empezarán a la vez

Una vez que terminan de realizar su movimiento, la historia continúa, así que para seguir, indica la instrucción de pasar la fondo del movimiento circular

MRUA recibe comparar.



De esta forma, hemos añadido una nueva pestaña a nuestra historia. Pulsa la bandera y visualiza el resultado.