

Bloque 1: Introducción práctica

1.2. Secuencias de instrucciones. Estructuras de control y repetición

¿Qué vamos a ver?

- 1. Tipos de bloques en Scratch**
- 2. Controlar el movimiento**
- 3. Controlar la apariencia**
- 4. Controlar el sonido**
- 5. Bucles y condicionales**

Tipos de bloques en Scratch

1. Tipos

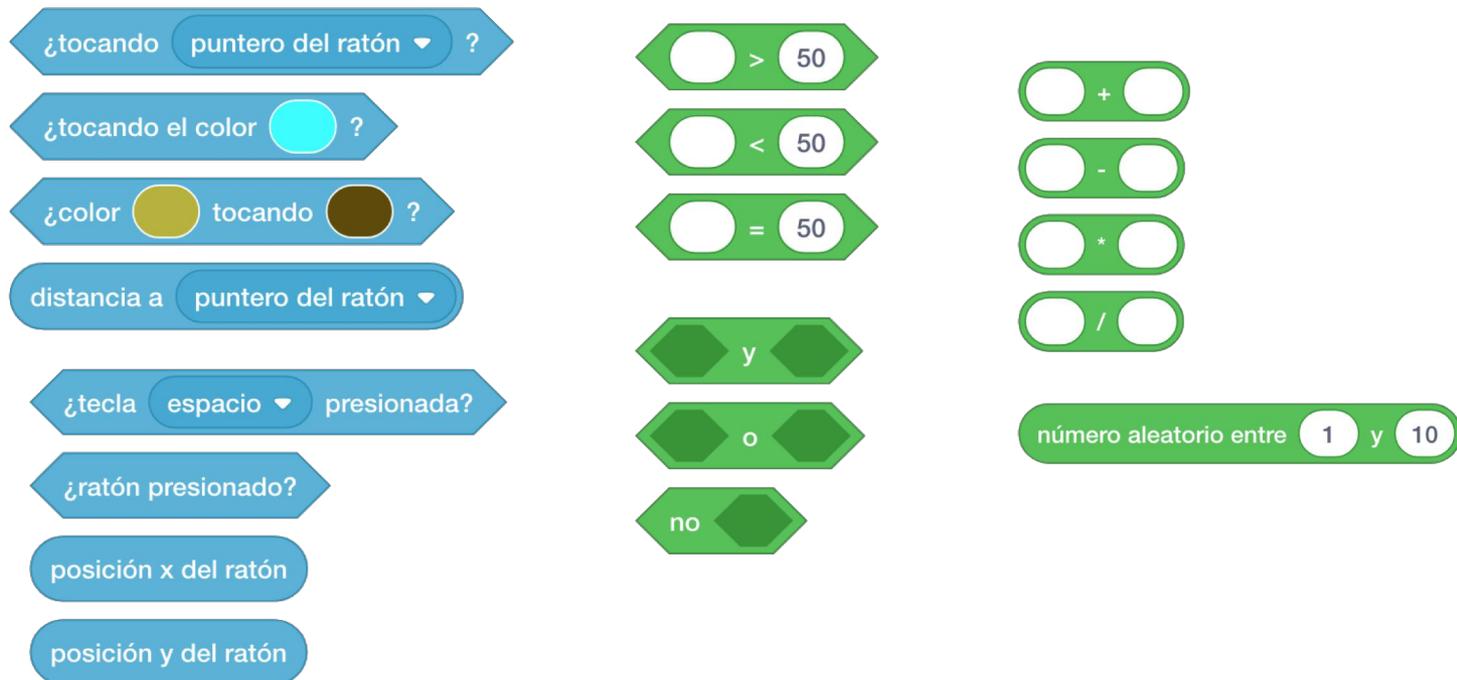
- **Inicio de programa:** Al producirse un evento comienzan la ejecución.
- **Instrucciones:** permiten realizar distintas acciones: movimiento, control de la apariencia, tocar un sonido...
- **Estructuras:** permiten agrupar distintas acciones con distintos propósitos: repetirlas, ejecutarlas si cumple una condición, ...



Tipos de bloques en Scratch

1. Tipos (cont.)

- Valores y condiciones.



Controlar el movimiento

1. Bloques de movimiento

- Permiten controlar la posición y la dirección del objeto



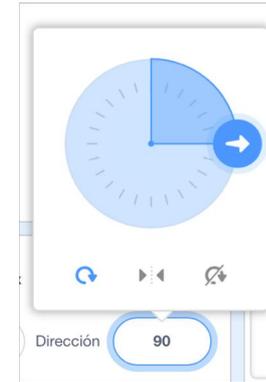
Scratch 'Movimiento' block palette:

- Movimiento:** mover 10 pasos, girar 15 grados, girar 15 grados.
- Control:** ir a posición aleatoria.
- Sensores:** ir a x: -30 y: 53.
- Operadores:** deslizar en 1 segs a posición aleatoria, deslizar en 1 segs a x: -30 y: 53.
- Mis bloques:** apuntar en dirección 90.



Scratch stage properties for 'Pufferfish':

- Objeto: Pufferfish
- x: -30
- y: 53
- Tamaño: 100
- Dirección: 90



Scratch stage directional dial:

- Dirección: 90

Controlar el movimiento

1. Bloques de movimiento

- Permiten controlar la posición y la dirección del objeto

Movimiento	Eventos
Apariencia	Control
Sonido	Sensores
Lápiz	Operadores
Datos	Más Bloques



yo

x: 13 y: -51 dirección: 90°

estilo de rotación:

puede ser arrastrado:

mostrar:

mover 10 pasos

girar 15 grados

girar 15 grados

apuntar en dirección 90

apuntar hacia

ir a x: 87 y: 10

ir a puntero del ratón

deslizar en 1 segs a x: 87 y: 10

cambiar x por 10

fijar x a 0

cambiar y por 10

fijar y a 0

rebotar si toca un borde

fijar estilo de rotación izquierda-de

posición x

posición y

dirección

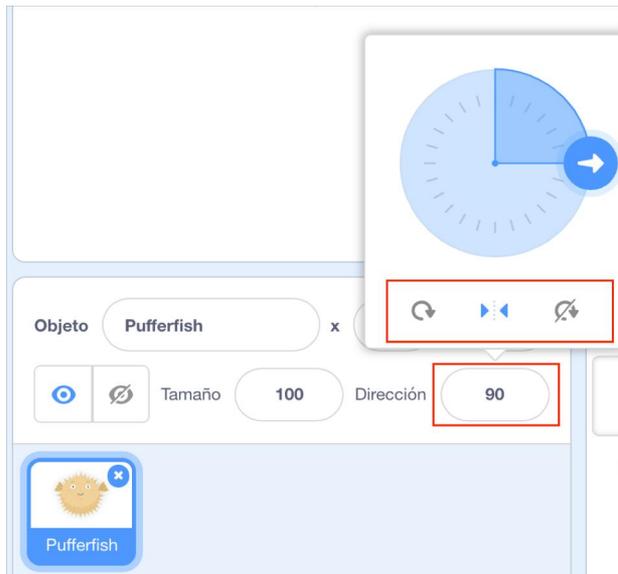
2. Practicamos: Move it!

- Haciendo uso del evento “Al presionar tecla”, permite que se pueda controlar el movimiento de un personaje mediante el teclado.

Controlar el movimiento

2. Practicamos: Move it!

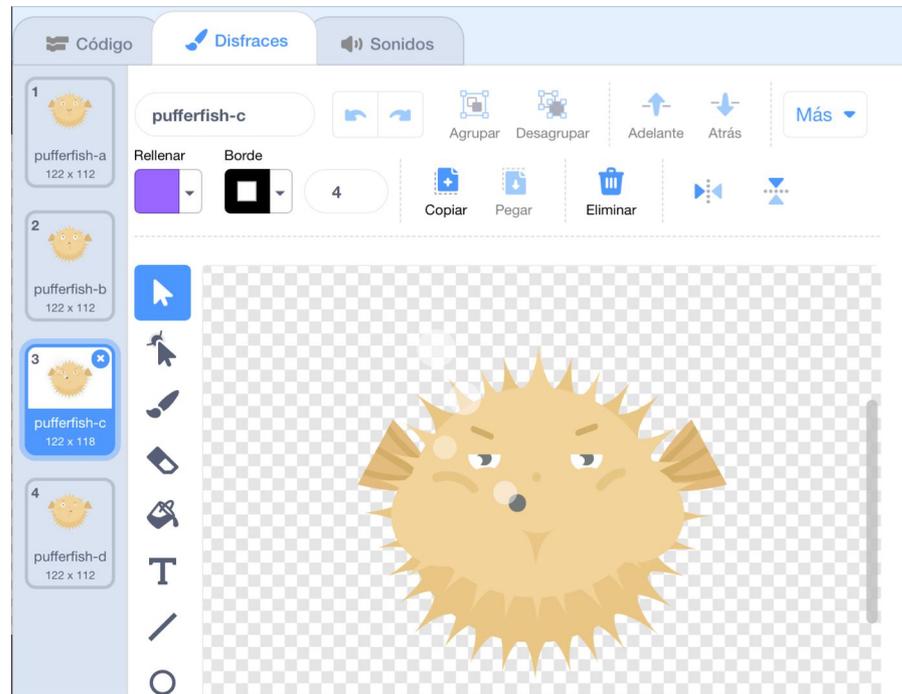
- Una posible solución:



<http://scratch.mit.edu/projects/22455713/>

1. Distintos disfraces

- Tanto los objetos como el escenario pueden tener más de una apariencia o “disfraz”.



2. Instrucciones de apariencia

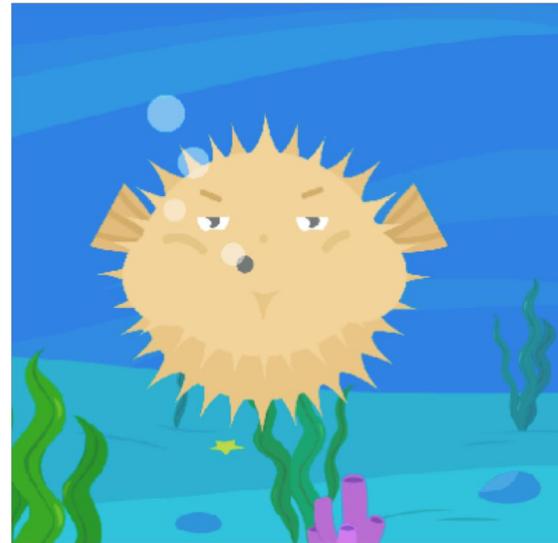
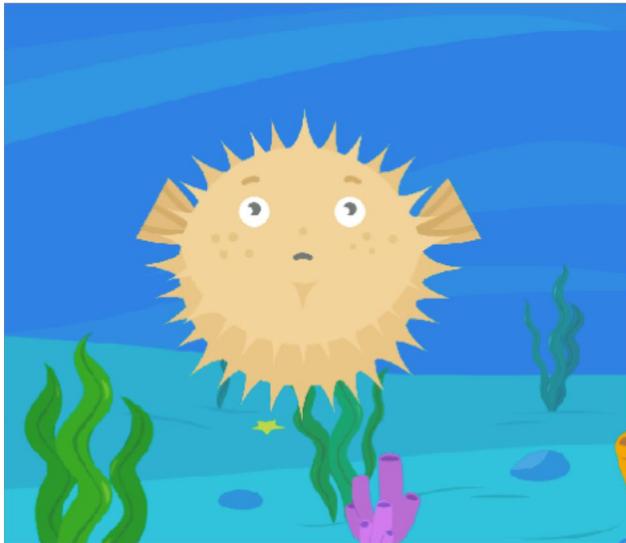
- Permiten controlar la apariencia del objeto/escenario: viñetas para decir o pensar, mostrarlo o esconderlo, disfraz, tamaño, efectos, posición con respecto a otros objetos (frente), ...



The screenshot shows the Scratch code editor interface. The top navigation bar includes 'Código', 'Disfraces', and 'Sonidos'. The left sidebar lists various categories: Movimiento, Apariencia (selected), Sonido, Eventos, Control, Sensores, Operadores, Variables, and Mis bloques. The main workspace displays a sequence of 'Apariencia' blocks: 'decir ¡Hola! durante 2 segundos', 'decir ¡Hola!', 'pensar Umm... durante 2 segundos', 'pensar Umm...', 'cambiar disfraz a pufferfish-a', 'siguiente disfraz', 'cambiar fondo a fondo1', 'siguiente fondo', 'cambiar tamaño por 10', and 'fijar tamaño al 100 %'.

3. Practicamos:

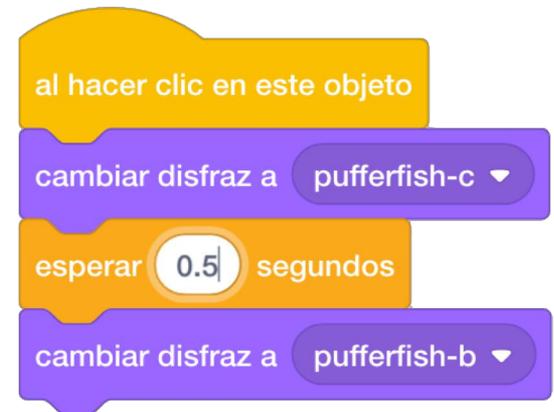
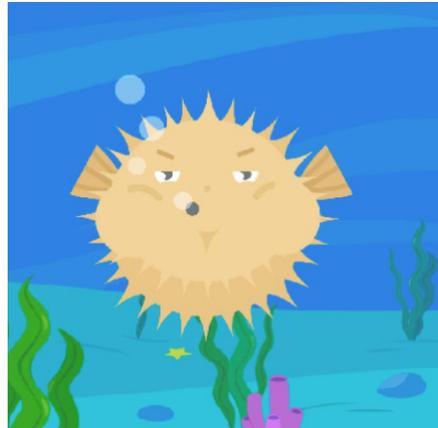
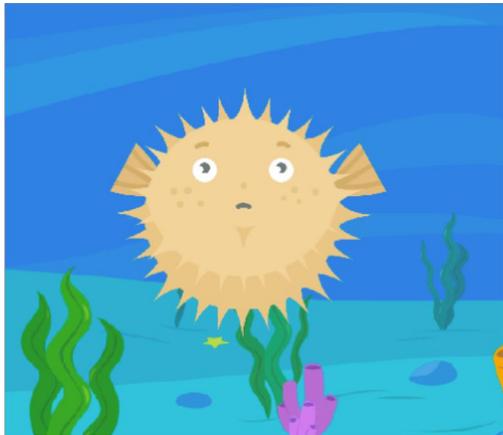
- Haciendo uso del evento “Al clickear este objeto”, controla la apariencia de un objeto para que realice algún cambio de apariencia



Controlar la apariencia

3. Practicamos:

- Una posible solución



<http://scratch.mit.edu/projects/22455713/>

1. Distintos sonidos

- Tanto los objetos como el escenario pueden tener más de un sonido que puede ser “tocado” mediante las instrucciones correspondientes del mismo.



1. Bloques de sonido

- Permiten controlar el sonido del objeto/escenario: tocar un sonido del objeto/escenario, controlar el volumen o añadir efectos.

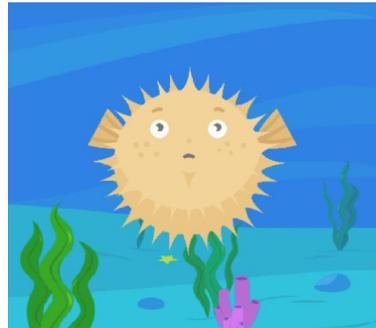


The screenshot shows the 'Sonido' (Sound) block palette in Scratch. The palette is organized into categories on the left: Movimiento, Apariencia, Sonido (highlighted), Eventos, Control, Sensores, Operadores, Variables, and Mis bloques. The 'Sonido' category contains the following blocks:

- tocar sonido (pop) hasta que termine
- iniciar sonido (pop)
- detener todos los sonidos
- sumar al efecto (altura) 10
- dar al efecto (altura) el valor 100
- quitar efectos de sonido
- cambiar volumen por -10
- fijar volumen al 100 %
- volumen

2. Practicamos 1:

- Modificamos el proyecto anterior para que toque un sonido al clicar sobre el objeto.



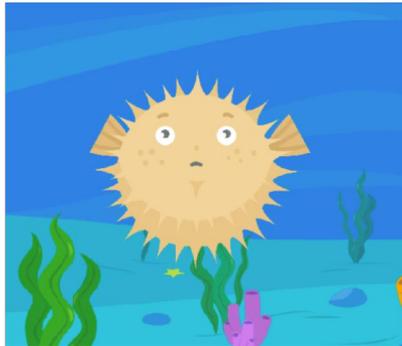
3. Practicamos 2:

- Añade un sonido de fondo.
- Si tienes micrófono, graba un sonido y haz que lo diga el objeto.

Controlar el sonido

2. Practicamos 1:

- Una posible solución:



<http://scratch.mit.edu/projects/22455713/>

Controlar el sonido

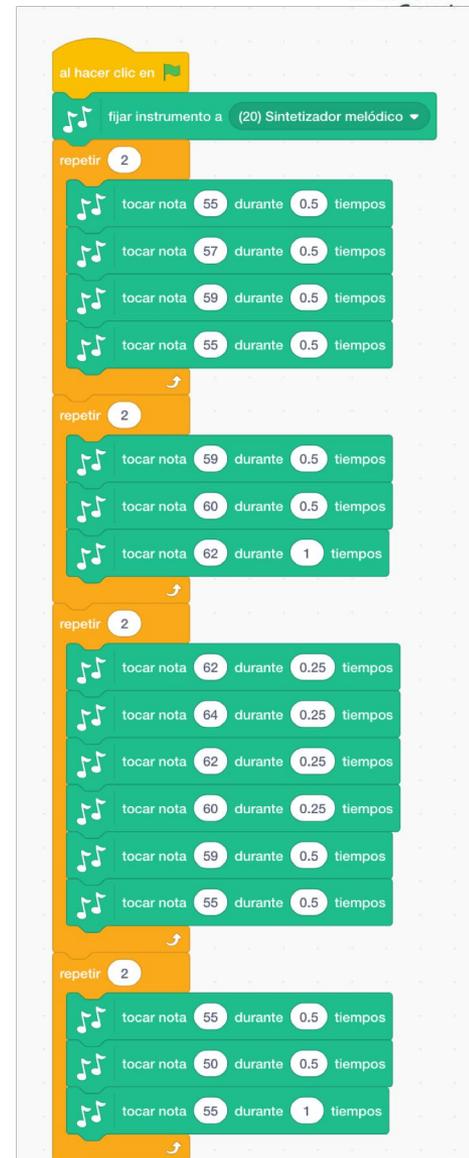
4. Trabajar la música con Scratch

- Con Scratch es posible tocar partituras.

FRÈRE JACQUES



Frè - re Jac - ques Dor - mez - vous?
Son - nez les ma - ti - nes Ding ding Dong



al hacer clic en

fijar instrumento a (20) Sintetizador melódico

repetir 2

- tocar nota 55 durante 0.5 tiempos
- tocar nota 57 durante 0.5 tiempos
- tocar nota 59 durante 0.5 tiempos
- tocar nota 55 durante 0.5 tiempos

repetir 2

- tocar nota 59 durante 0.5 tiempos
- tocar nota 60 durante 0.5 tiempos
- tocar nota 62 durante 1 tiempos

repetir 2

- tocar nota 62 durante 0.25 tiempos
- tocar nota 64 durante 0.25 tiempos
- tocar nota 62 durante 0.25 tiempos
- tocar nota 60 durante 0.25 tiempos
- tocar nota 59 durante 0.5 tiempos
- tocar nota 55 durante 0.5 tiempos

repetir 2

- tocar nota 55 durante 0.5 tiempos
- tocar nota 50 durante 0.5 tiempos
- tocar nota 55 durante 1 tiempos

1. Bucles

- Nos permiten agrupar una serie de instrucciones para que se ejecuten un número (in)determinado de veces.



2. Condicionales

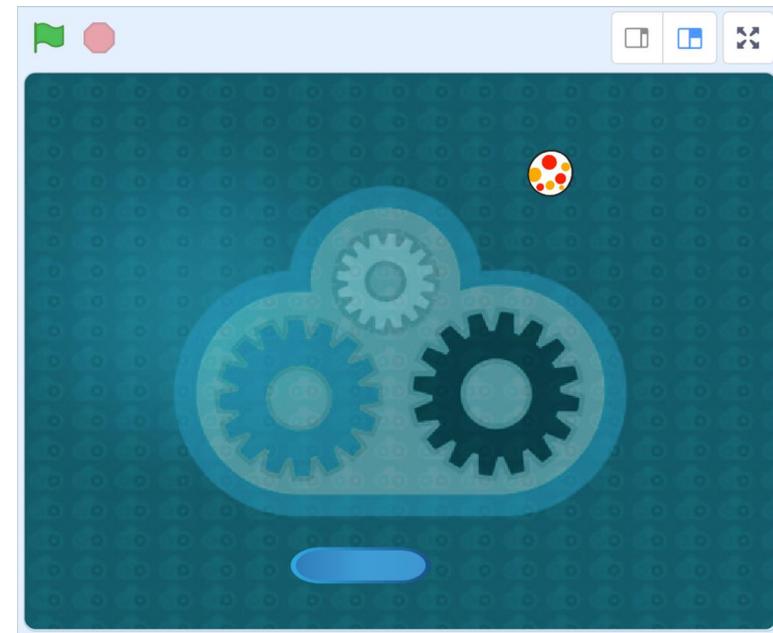
- Nos permiten agrupar una serie de instrucciones para que se ejecuten sólo si se cumple una condición.



Bucles y condicionales

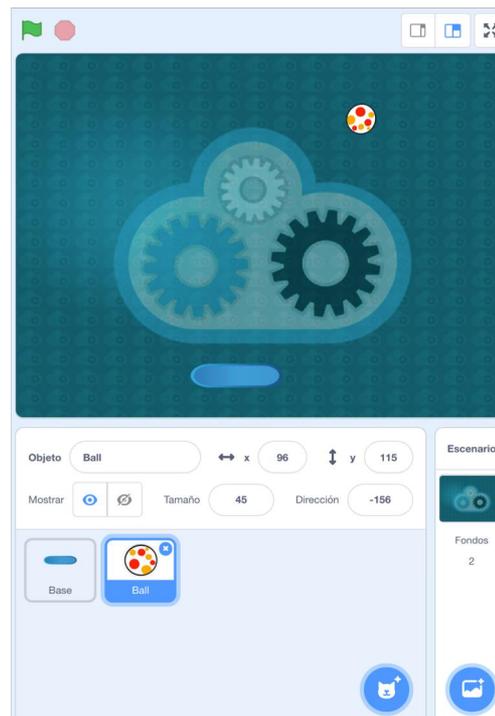
3. Practicamos: “El Arkanoid”

- La base se podrá mover a derecha e izquierda mediante teclado.
- La bola comenzará en el centro y saldrá en una dirección aleatoria. Se irá movimiento continuamente y rebotará en todos los bordes.
- Si la bola toca el borde inferior termina la partida. Si toca la base rebotará hacia arriba..



3.1 La pelota

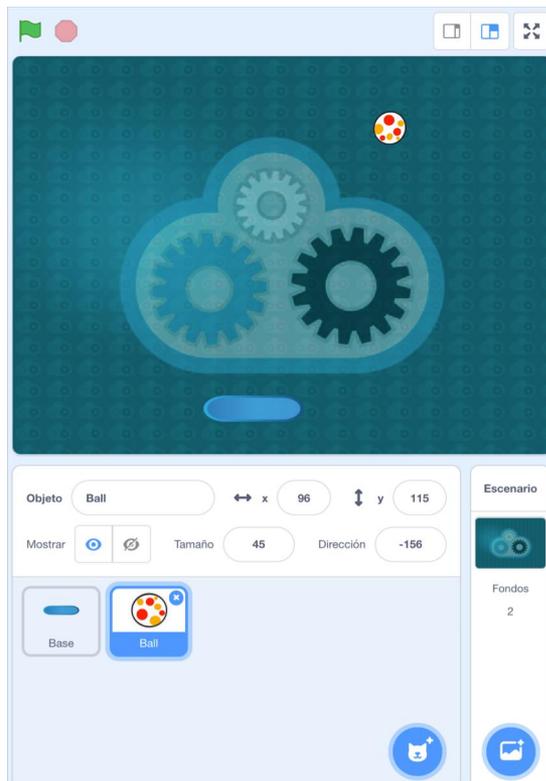
- Un caso típico es programar un objeto para que SIEMPRE realice las mismas acciones. En este caso, haremos que una pelota se mueva constantemente y rebote si toca un borde.



Bucles y condicionales

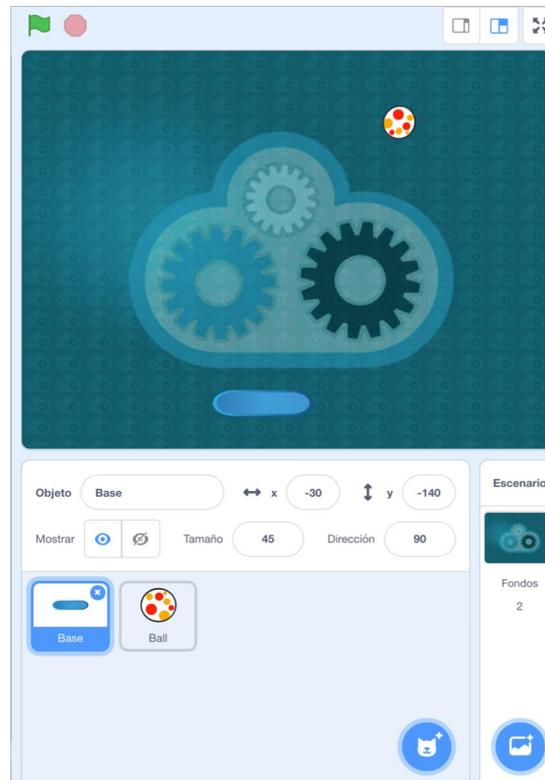
3.1 La pelota

- Una posible solución:



3.2. La base: movimiento

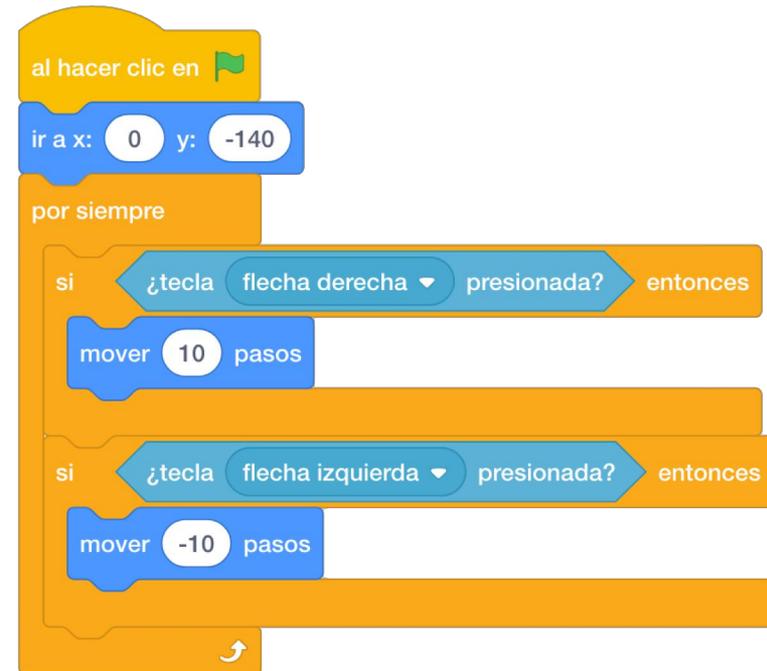
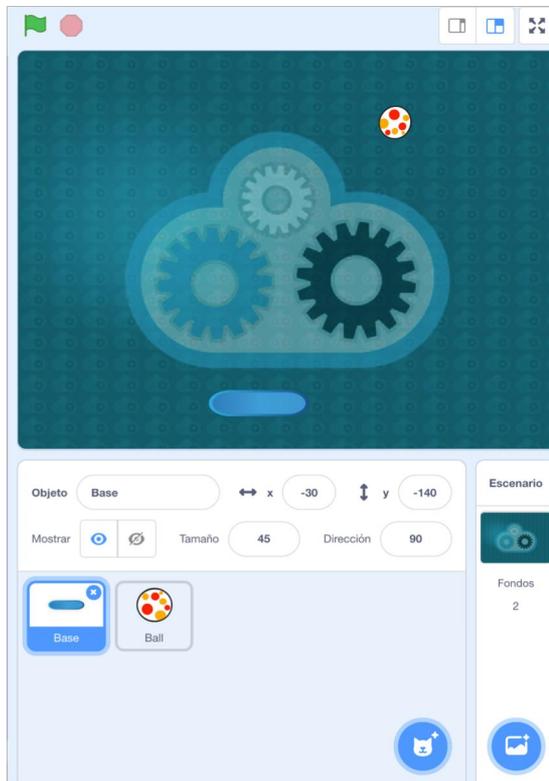
- Creamos un **objeto** que haga de **base** y controlamos su movimiento mediante el teclado (haciendo uso de los bucles y condicionales).



Bucles y condicionales

3.2. La base: movimiento

- Una posible solución.



3.3. Control del juego

- Dentro del programa de la pelota controlaremos dos cosas:
 - Si toca la base debe rebotar.
 - Si toca el borde inferior se termina la partida.

3.3. Control del juego

- Una posible solución:

