

Bloque 1: Introducción práctica

1.2. Secuencias de instrucciones. Estructuras de control y repetición

©Programamos, octubre de 2015

Algunos derechos reservados. Este artículo se distribuye bajo la licencia "Reconocimiento-CompartirIgual 3.0 España" de Creative Commons, disponible en <http://creativecommons.org/licenses/by-sa/3.0/es/deed.es>

Este documento (o uno muy similar) está disponible en (o enlazado desde) <http://programamos.es>

José Ignacio Huertas Fernández

@jihuefer

Jesús Moreno León

@j_moreno1

<http://programamos.es>

info@programamos

Twitter: [@programamos](https://twitter.com/programamos)

<http://facebook.es/programamos>

¿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

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:



Crab

x: -158 y: -55 dirección: -90°

estilo de rotación: izquierda-derecha

puede ser arrastrado:

mostrar:

fijar estilo de rotación izquierda-derecha



```
al presionar tecla flecha derecha
  apuntar en dirección 90
  mover 10 pasos

al presionar tecla flecha izquierda
  apuntar en dirección -90
  mover 10 pasos
```

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

Controlar la apariencia

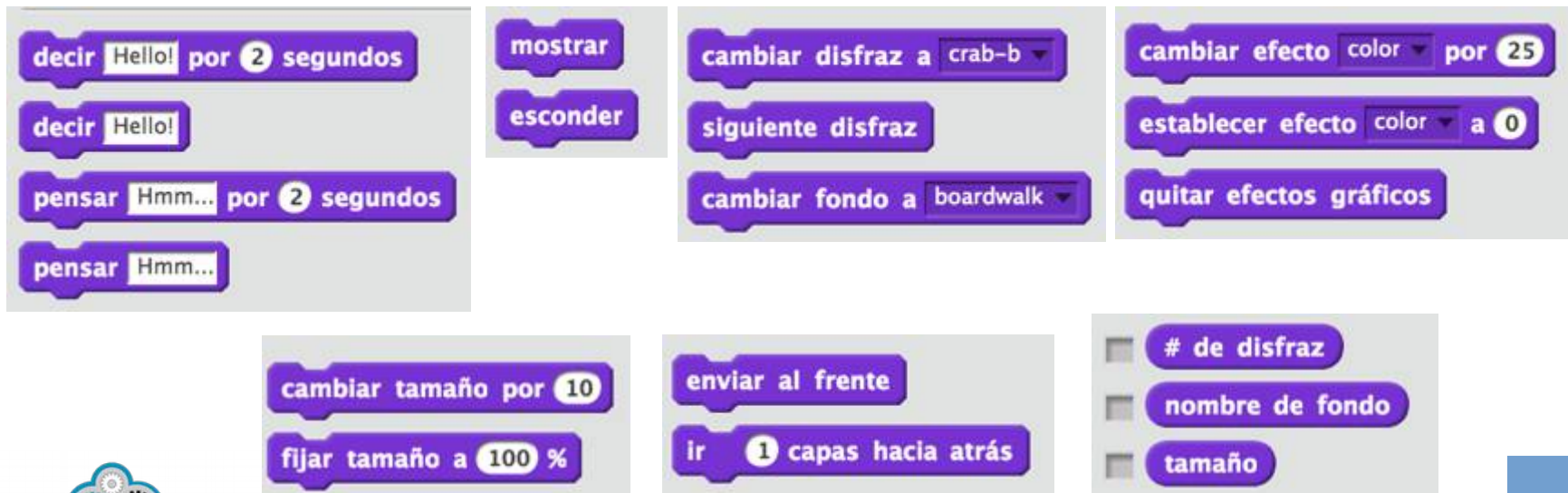
1. Distintos disfraces

- 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 image displays a collection of Scratch 'Appearance' category blocks, organized into several groups:

- Speech and Thought:** 'decir Hello! por 2 segundos', 'decir Hello!', 'pensar Hmm... por 2 segundos', 'pensar Hmm...'
- Visibility:** 'mostrar', 'esconder'
- Disguise and Background:** 'cambiar disfraz a crab-b', 'siguiente disfraz', 'cambiar fondo a boardwalk'
- Effects:** 'cambiar efecto color por 25', 'establecer efecto color a 0', 'quitar efectos gráficos'
- Size:** 'cambiar tamaño por 10', 'fijar tamaño a 100 %'
- Position:** 'enviar al frente', 'ir 1 capas hacia atrás'
- Properties Panel:** A list of properties with checkboxes: '# de disfraz', 'nombre de fondo', 'tamaño'

3. Practicamos:

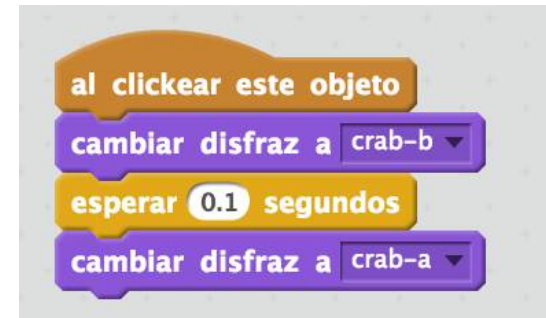
- Haciendo uso del evento “Al clickear este objeto”, controla la apariencia de un objeto para que realice algún cambio de apariencia



Controlar la apariencia

3. Practicamos:

- Una posible solución



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

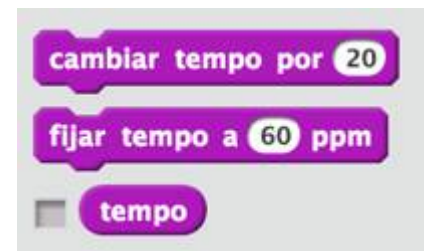
1. Distintos sonidos

- 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, tocar el tambor o un instrumento, controlar el volumen o el tempo.



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/>

4. Trabajar la música con Scratch

- Con Scratch es posible tocar partituras.



The image shows a Scratch script for playing a church organ melody. The script starts with a 'when clicked' event block, followed by 'set instrument to 20' and 'hide'. The melody is composed of four phrases, each repeated twice:

- Phrase #1:** play note 55 for 0.5 beats, play note 57 for 0.5 beats, play note 59 for 0.5 beats, play note 55 for 0.5 beats.
- Phrase #2:** play note 59 for 0.5 beats, play note 60 for 0.5 beats, play note 62 for 1 beats.
- Phrase #3:** play note 62 for 0.25 beats, play note 64 for 0.25 beats, play note 62 for 0.25 beats, play note 60 for 0.25 beats, play note 59 for 0.5 beats, play note 55 for 0.5 beats.
- Phrase #4:** play note 55 for 0.5 beats, play note 50 for 0.5 beats, play note 55 for 1 beats.

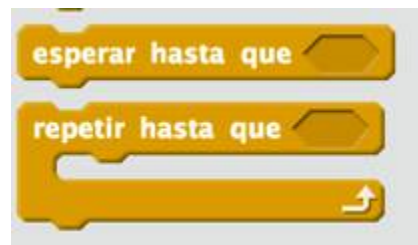


The image shows the musical notation for the song 'Frère Jacques'. The notation is written on a treble clef staff with a common time signature (C). The melody is composed of four phrases, each repeated twice:

- Phrase #1:** Frè - re Jac - ques
- Phrase #2:** Dor - mez - vous?
- Phrase #3:** Son - nez les ma - ti - nes
- Phrase #4:** Ding dingue dong

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

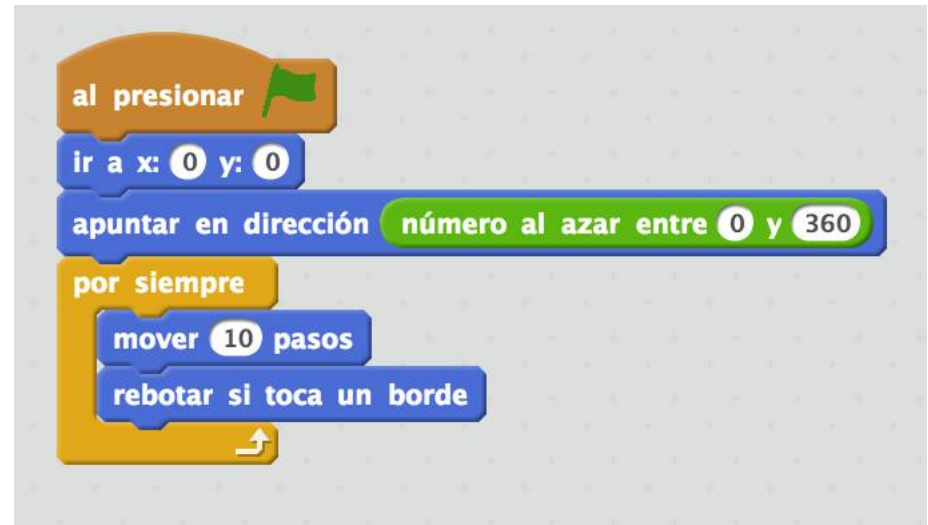
- Un caso típico es programar un objeto para que SIEMPRE realice las mismas acciones. En este caso, haremos que una pelota se mueva constantemente y rebote si toca un borde.



Bucles y condicionales

3.1 La pelota

- Una posible solución:



3.2. La base: movimiento

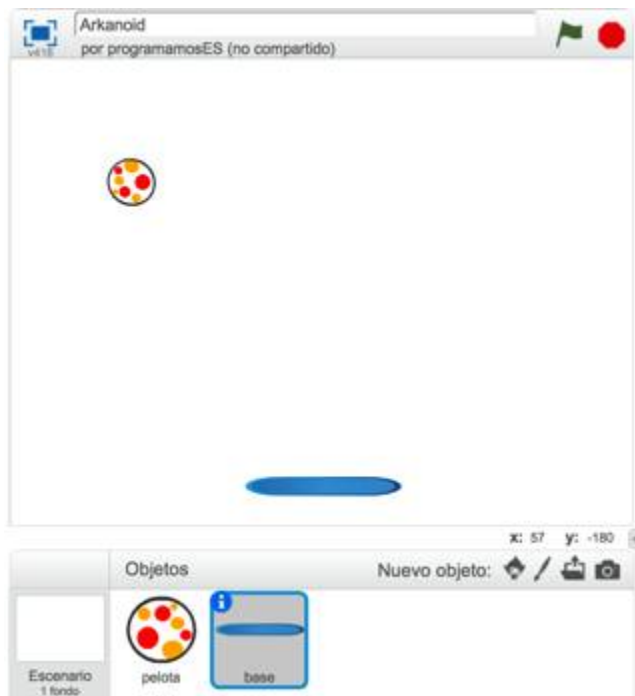
- Creamos un **objeto** que haga de **base** y controlamos su movimiento mediante el teclado (haciendo uso de los bucles y condicionales).



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.

Bucles y condicionales

3.3. Control del juego

- Una posible solución:

