

## ¿Cómo aprender a programar videojuegos con MakeCode Arcade?

Esto es una recopilación de tutoriales de Internet que los he resumido en este tutorial.



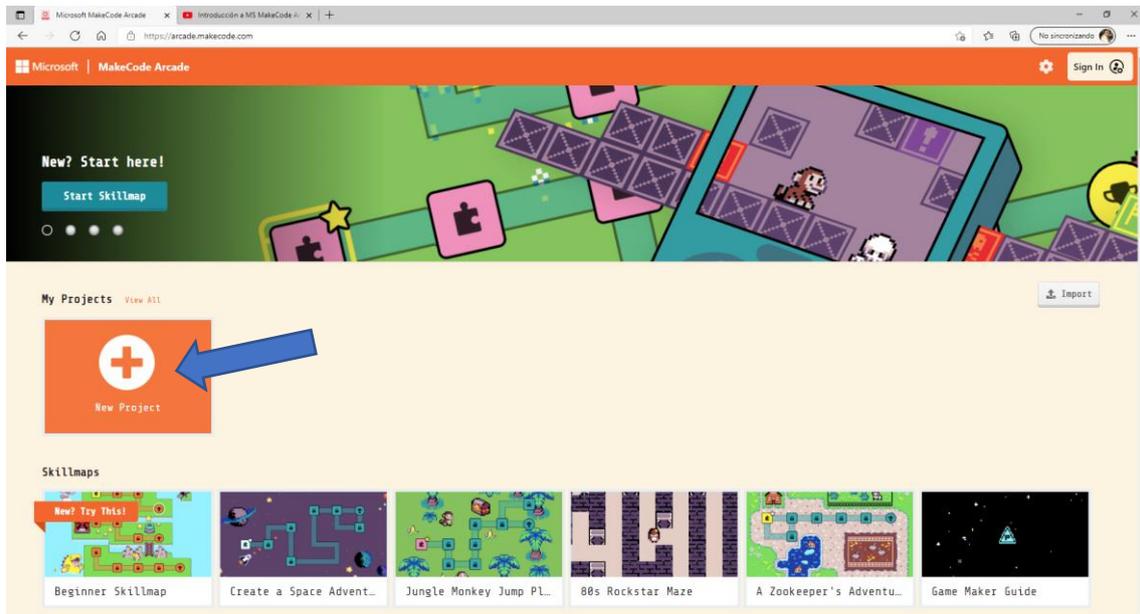
Para poder realizar las prácticas de este tutorial os adjunto el enlace:

<http://arcade.makecode.com>

Introducción a MS MakeCode Arcade a través de TEAMS.

Introducción al entorno de programación de MakeCode arcade integrado en MS TEAMS.

Una vez hemos accedido al siguiente enlace:



Seleccionaremos New Project

**Create a Project** ✕

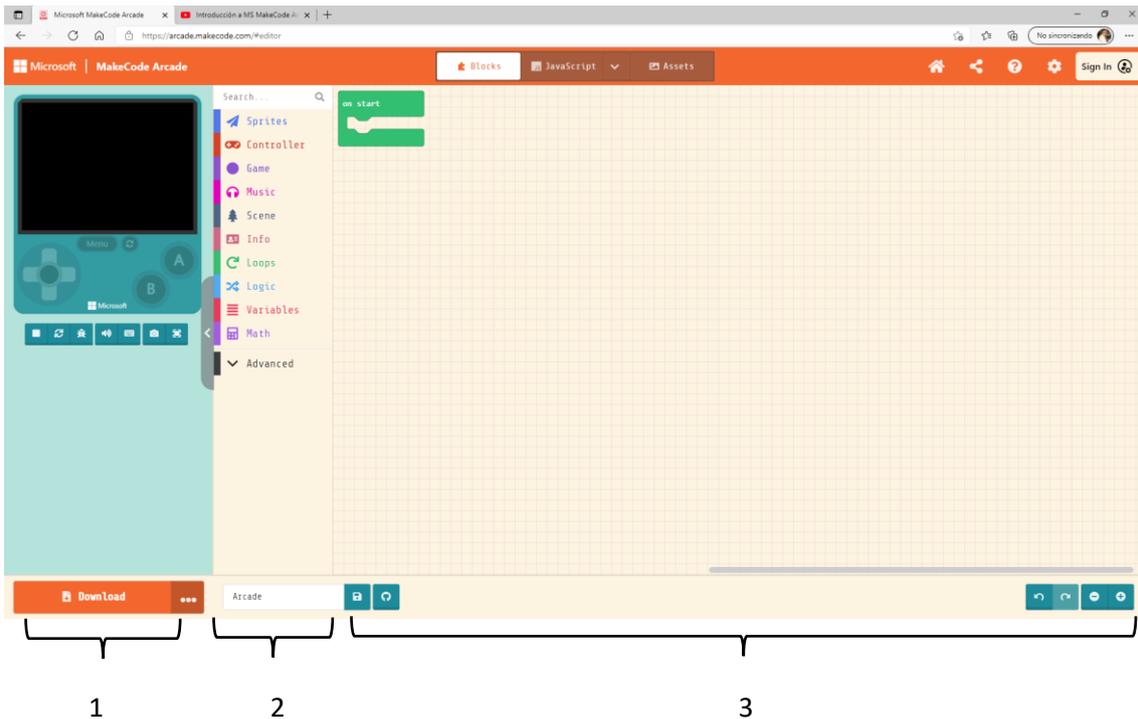
Give your project a name.

Arcade

[Code options](#)

Create

Nos pedirá que le asignemos un nombre a nuestro proyecto, yo he puesto "Arcade", seguido del botón "Create".



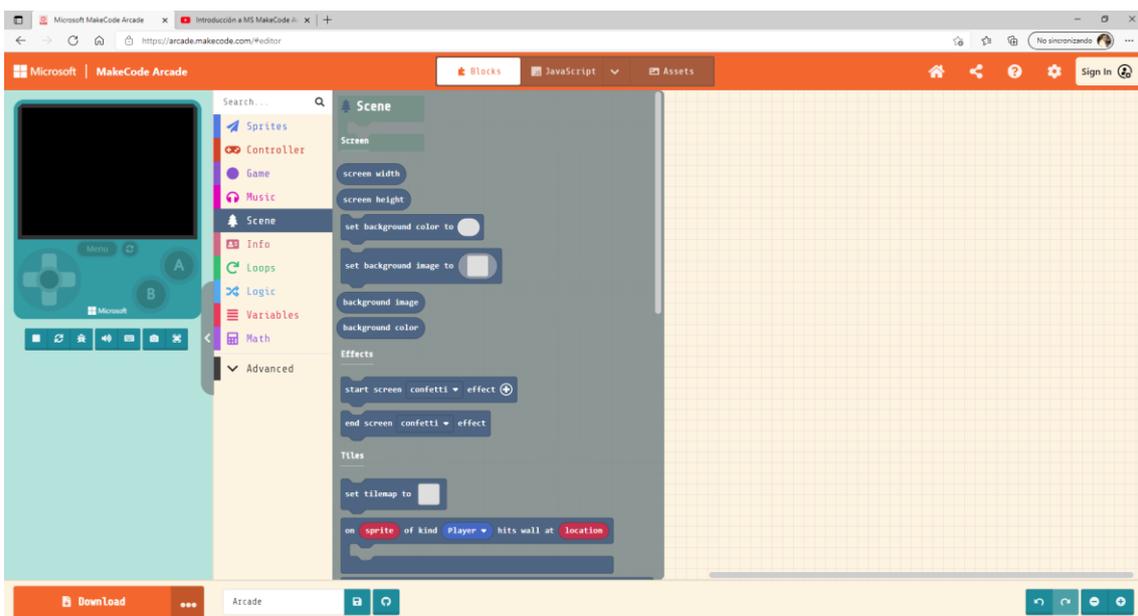
La primera columna muestra un visor donde podremos ir viendo como está funcionando nuestro proyecto.

En la segunda columna aparecen una serie de herramientas, clasificadas en categorías.

En la tercera columna tenemos el área donde programaremos.

Como primera práctica vamos a crear una escena para nuestra animación.

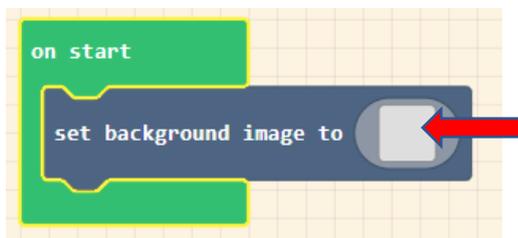
En el apartado de herramientas vamos a seleccionar Scene.



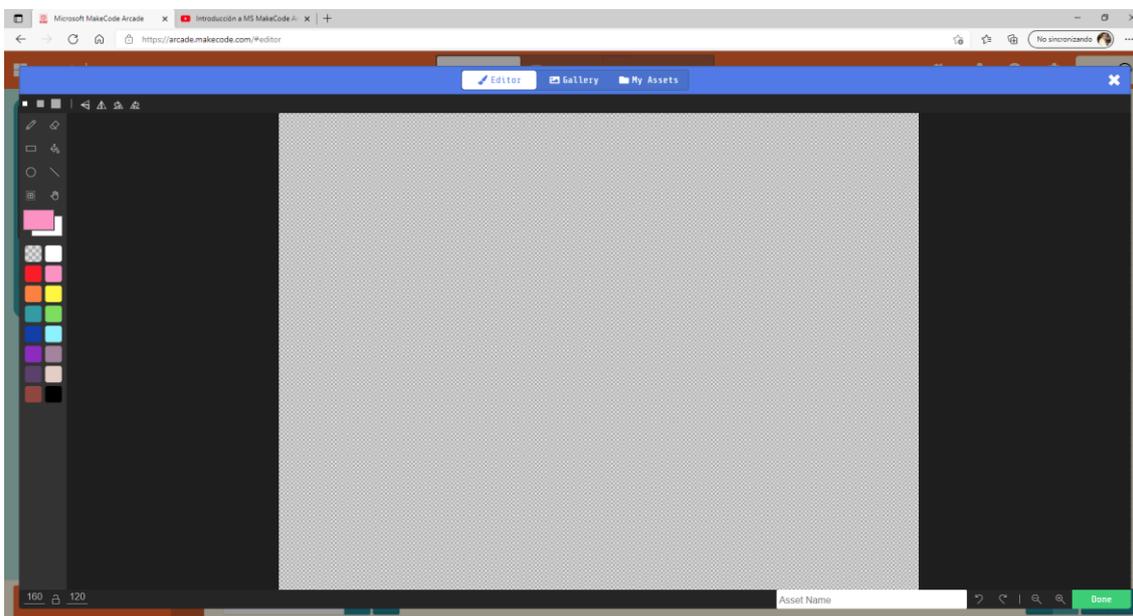
Seleccionaremos set background image to. Establecer imagen de fondo.



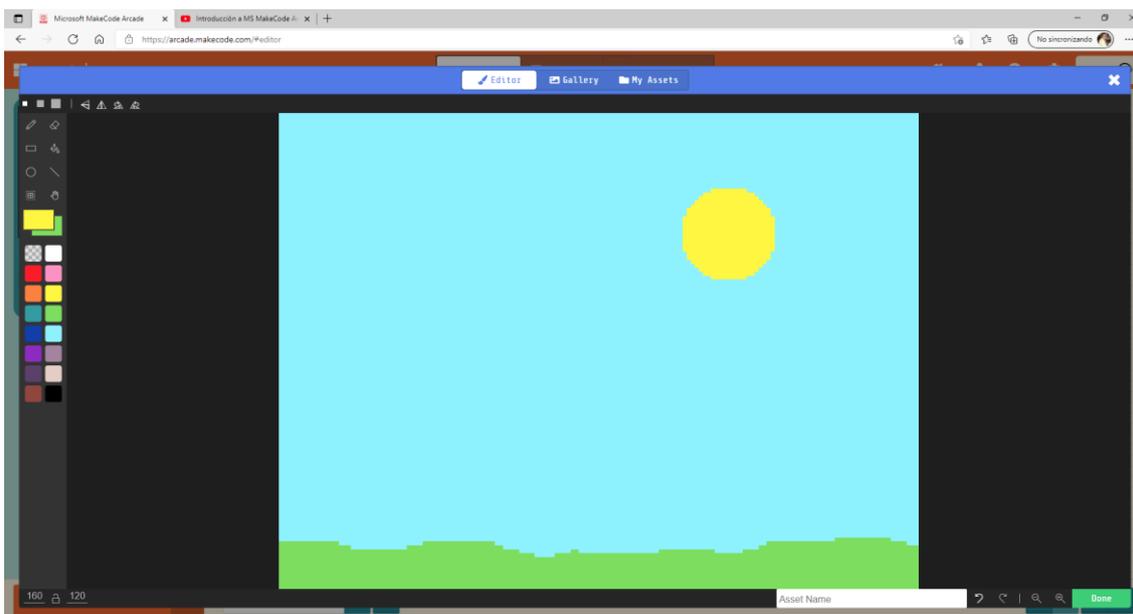
La arrastraremos al bloque que aparece al inicio "on start".



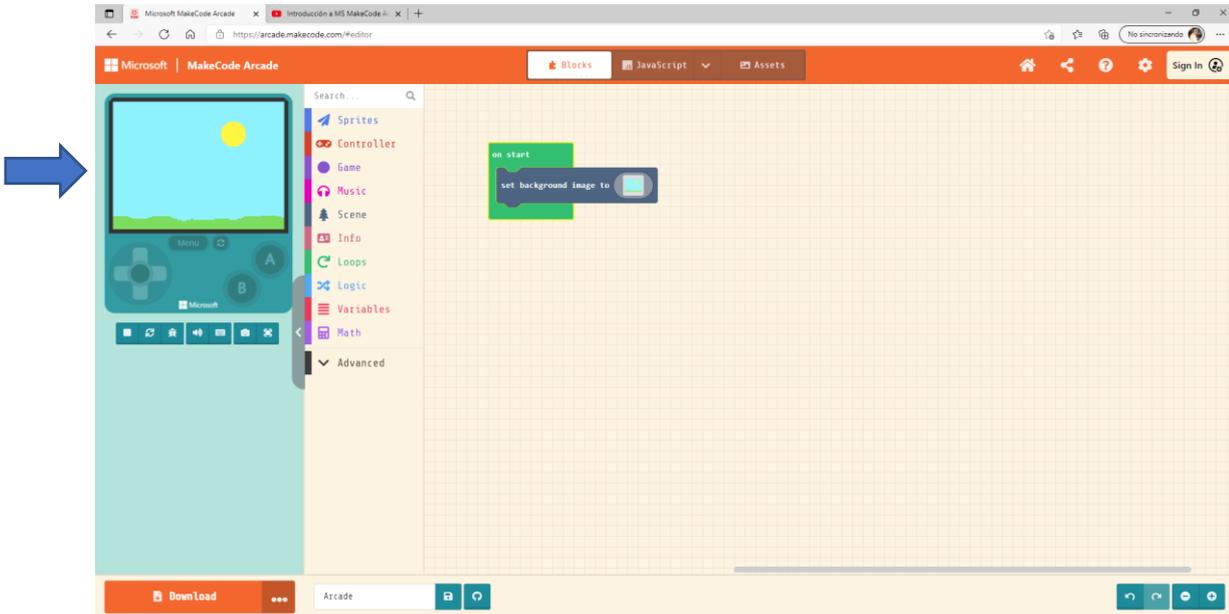
Seleccionando el cuadrado vamos a abrir el editor.



Vamos a dibujar un suelo, un cielo y el sol.

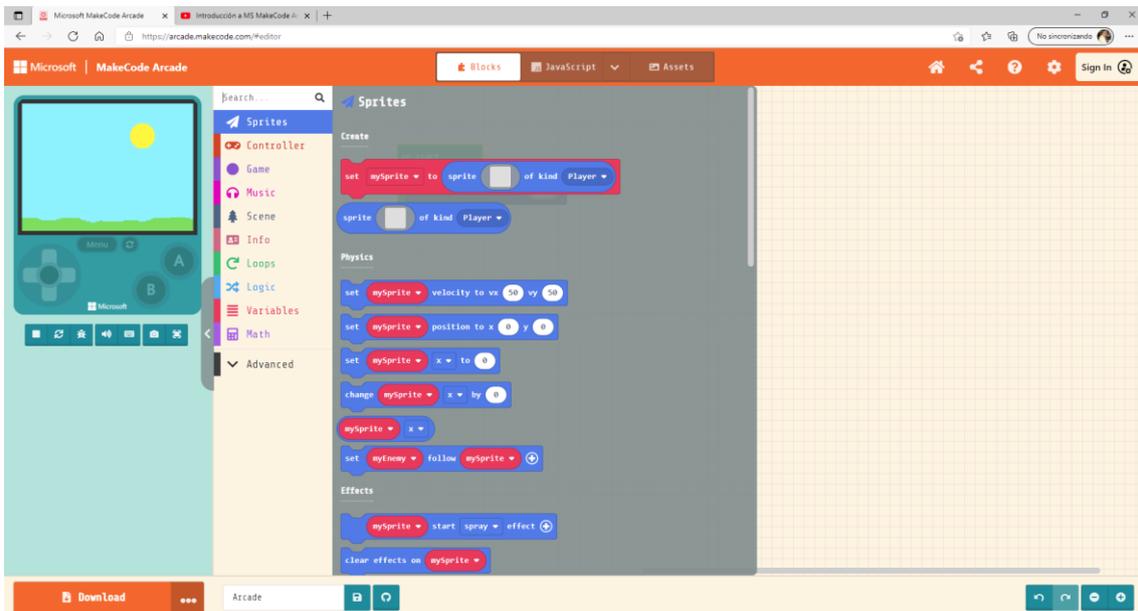


Una vez que hemos finalizado oprimimos el botón verde que se encuentra en la esquina inferior derecha llamado "Done" Realizado o hecho.



El visualizador ya ha cargado el fondo.

A continuación vamos a agregar nuestro personaje, para ello nos iremos a la categoría "Sprites".

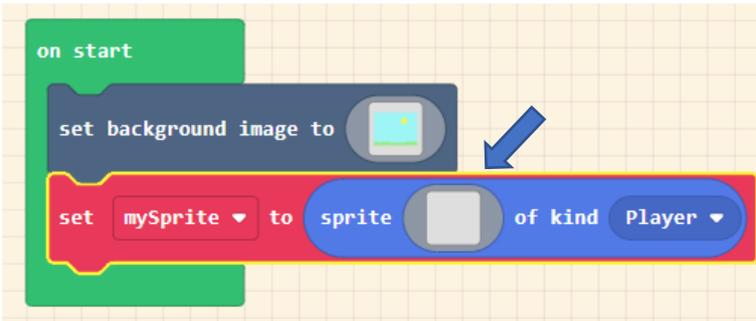


Son imágenes que se les pueden dar características de juego, movimiento, animaciones, efectos, posiciones, tamaños, etc.

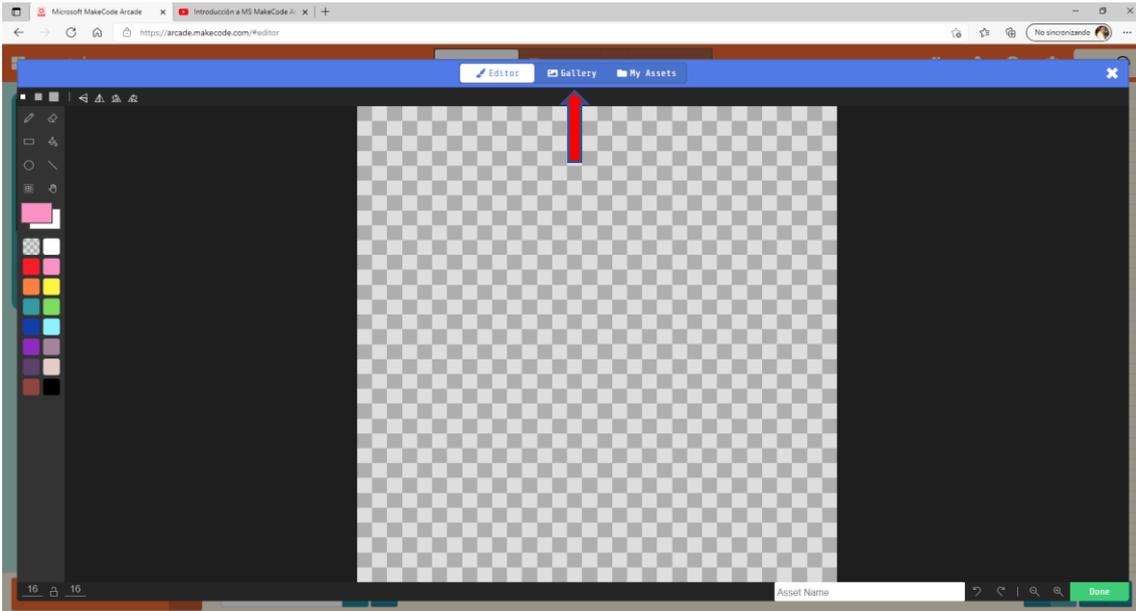
Del apartado Create seleccionaremos:



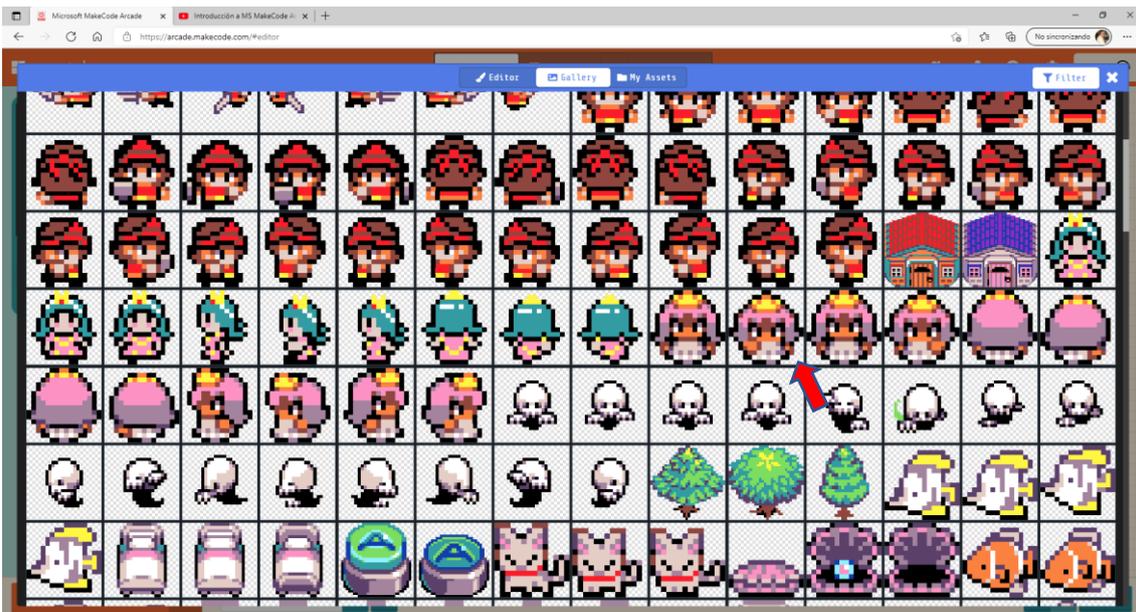
Lo agregamos dentro del bloque on start.



Vamos hacer clic en el cuadrado para abrir el editor.

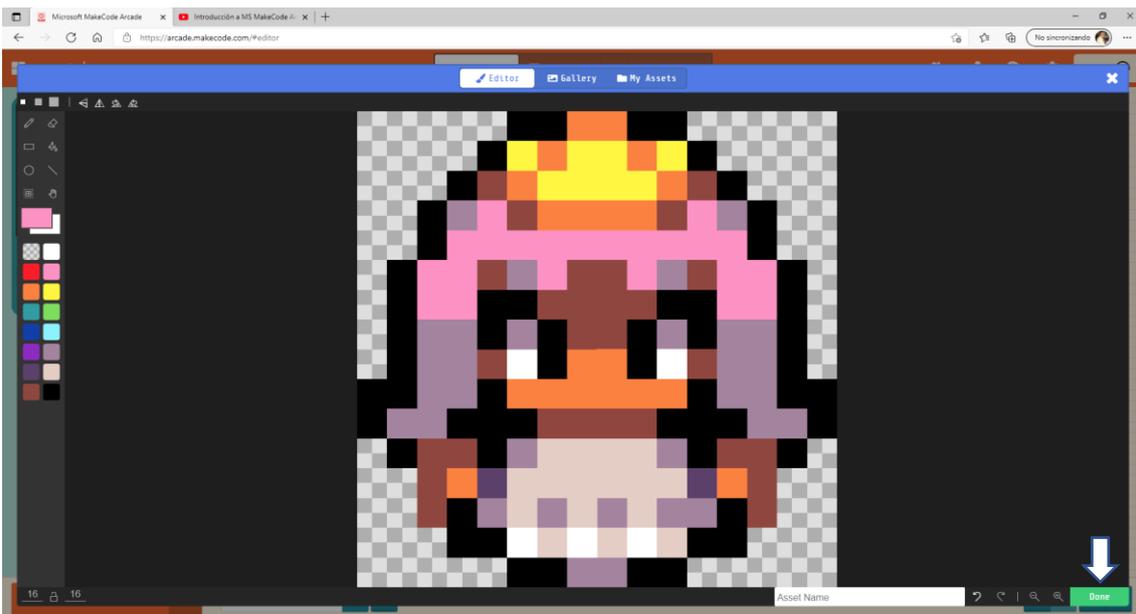


Seleccionaremos "Gallery".

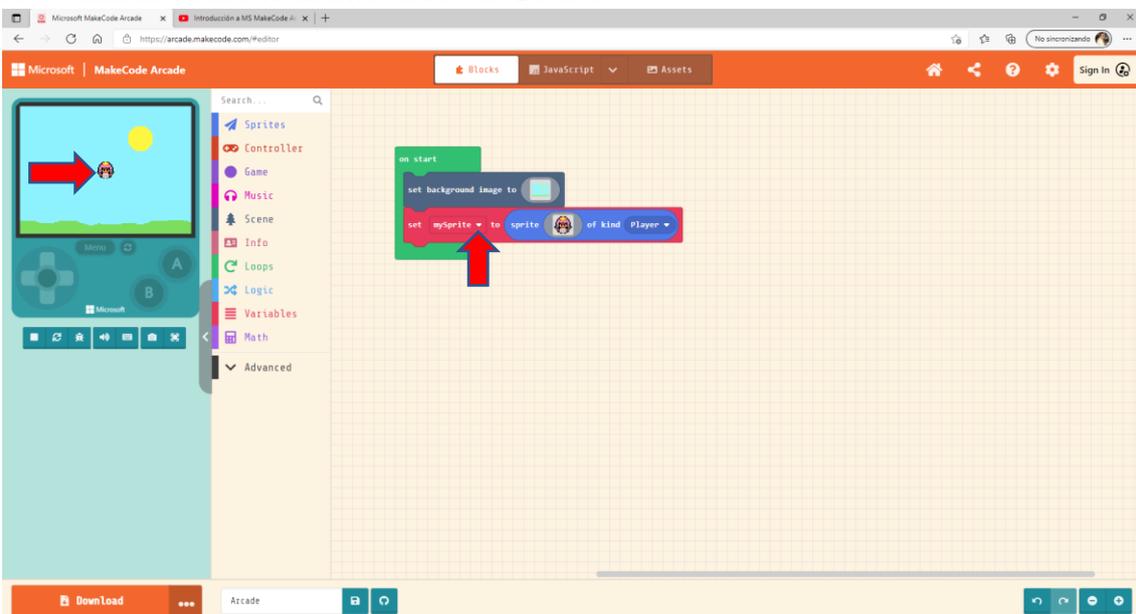


Nos mostrará una serie de elementos para seleccionar el que nosotros deseemos.

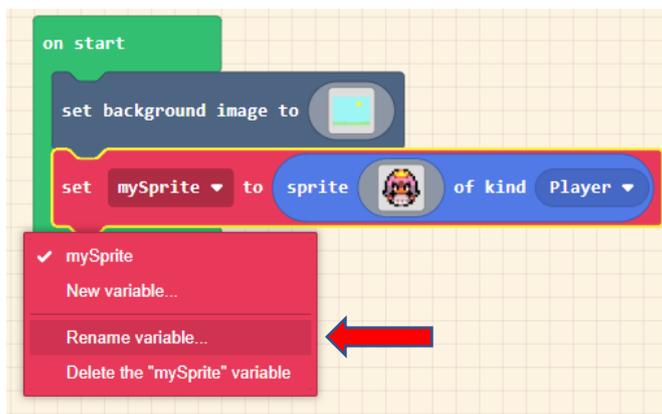
Vamos a seleccionar la princesa.



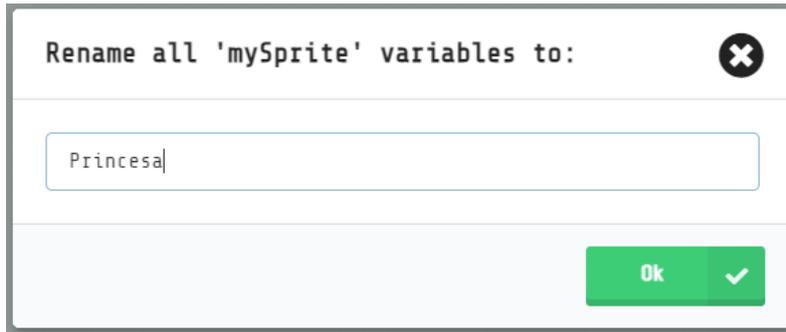
A continuación seleccionaremos el botón “Done”.



En el visor ya aparece nuestro personaje justo en el centro de la pantalla, posición X=0 e Y=0. A continuación vamos a cambiar el nombre y la posición donde queremos que aparezca.

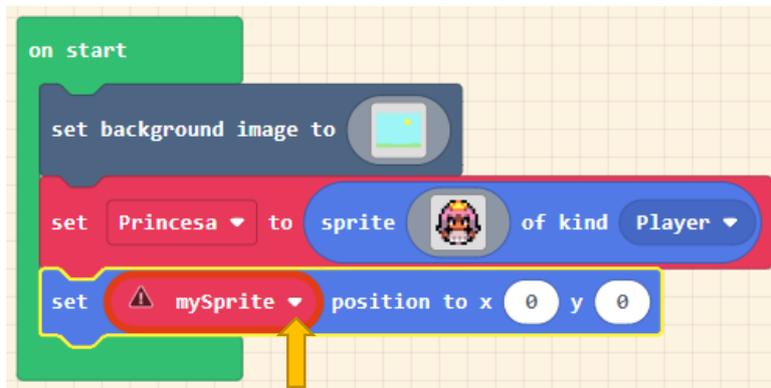
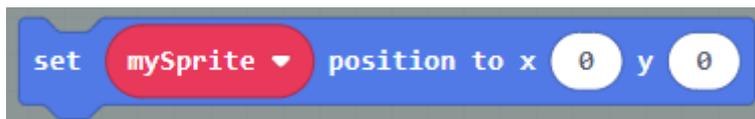


Seleccionaremos “Rename variable...”.



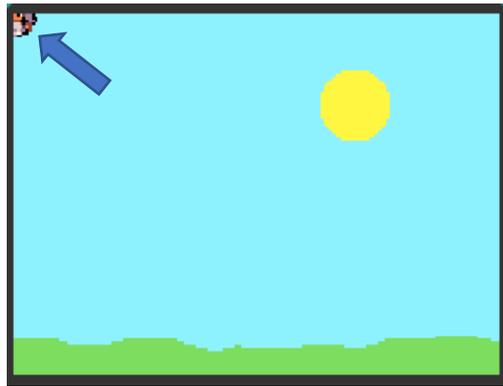
Ahora le vamos a dar la ubicación.

Seleccionamos la categoría Sprites y dentro del grupo Physics, seleccionaremos set position.

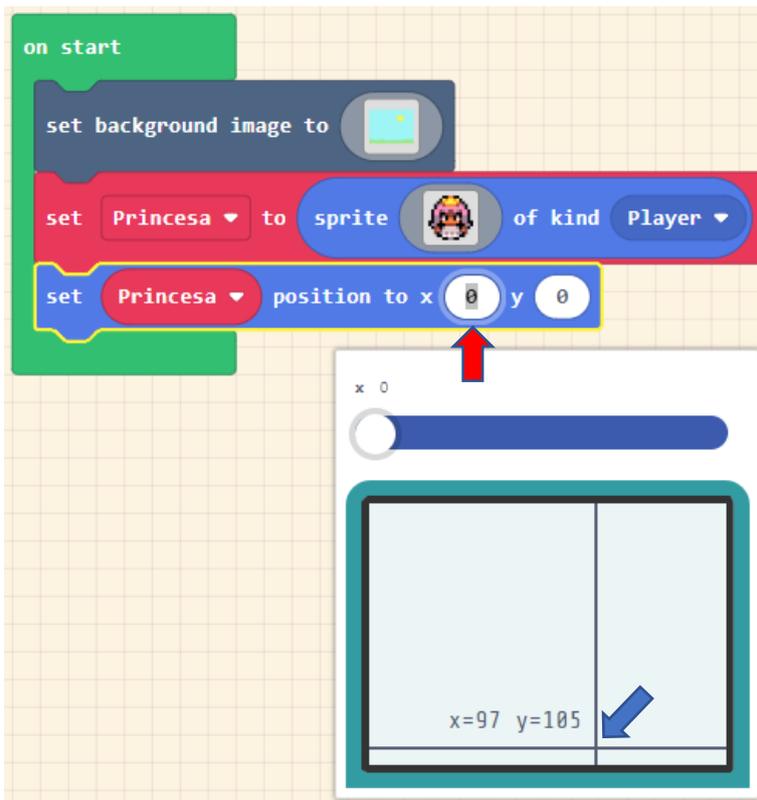


Tenemos que determinar que nuestro Sprite se llama princesa.

La coordenada x = 0, y=0.



Vamos a cambiar la ubicación modificando el valor de la X y de la Y.

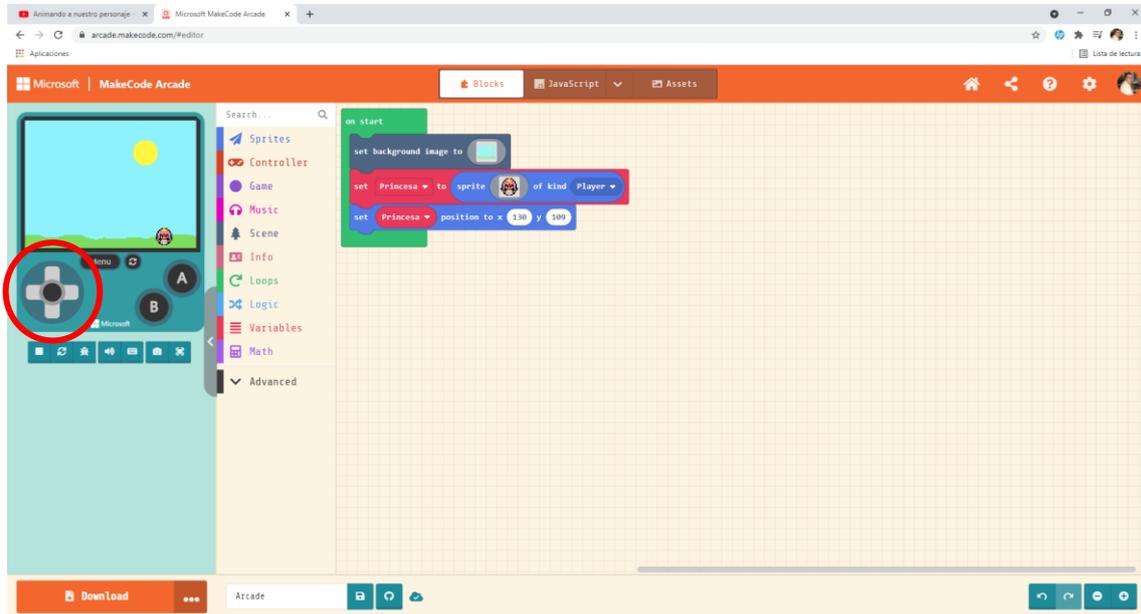


Esperamos que cargue el visor.

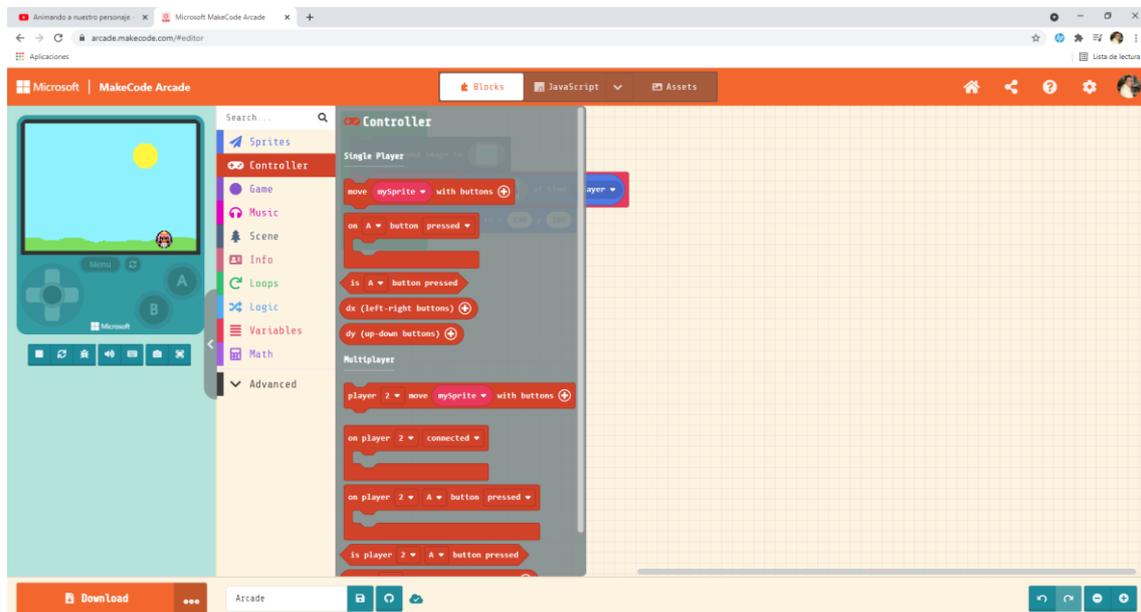


## Animando nuestro personaje

Trabajo con controles, evento de los botones y animaciones para crear el desplazamiento del personaje y el efecto de mover los pies.

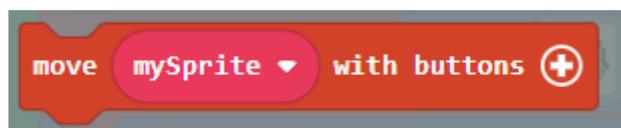


El siguiente objetivo es que con las flechas de dirección podamos mover nuestro personaje.



Seleccionamos el grupo de bloques llamado "Controller"

Seleccionaremos del apartado Single Player.

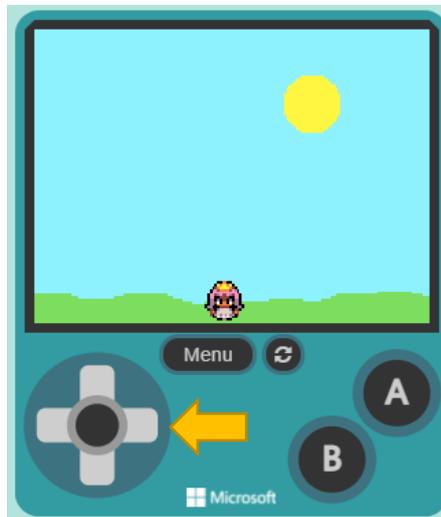


Lo agregamos al bloque "on start"



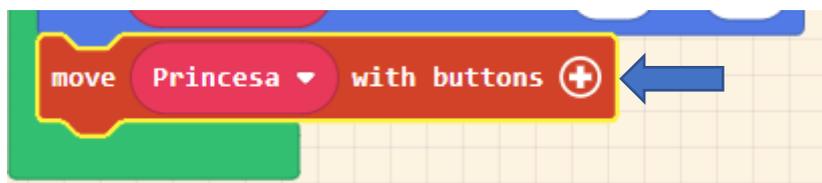
Seleccionamos "Princesa".

Ahora con las flechas de dirección ya puedes mover al personaje.

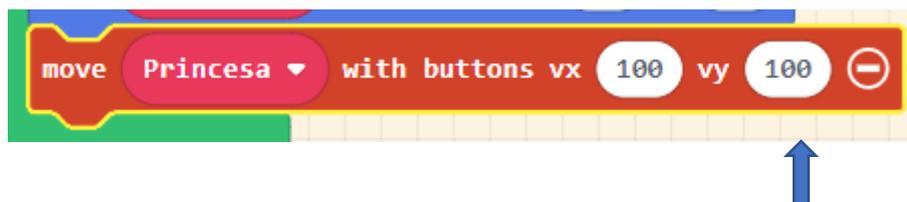


También funcionan con las teclas de dirección del teclado.

Ahora queremos que no vaya de arriba abajo.



Seleccionamos en el signo +.

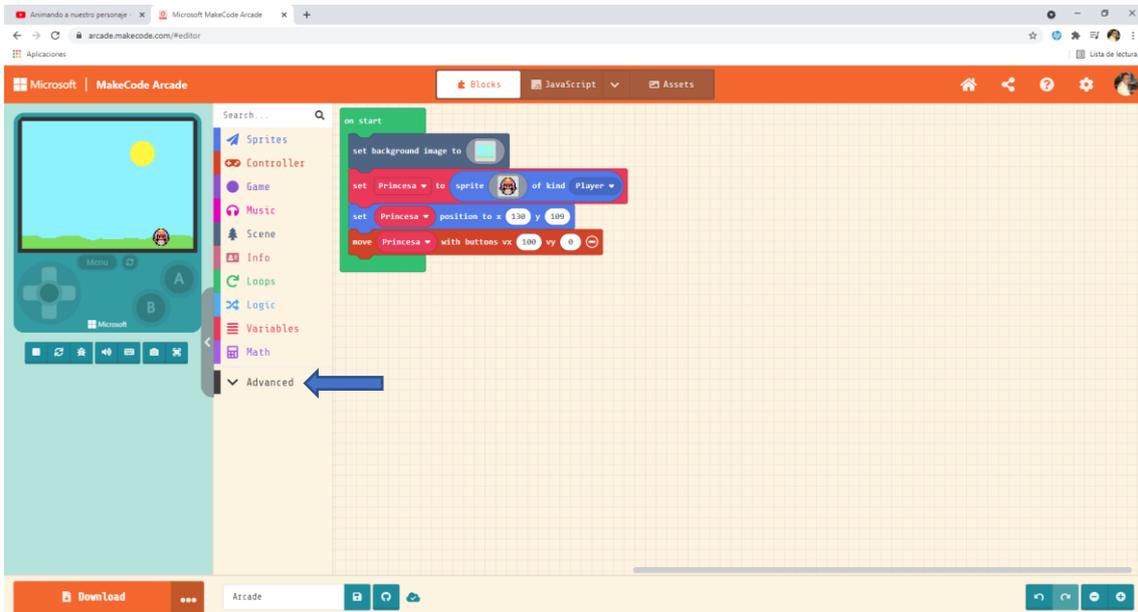


Al eje vertical le ponemos un 0.

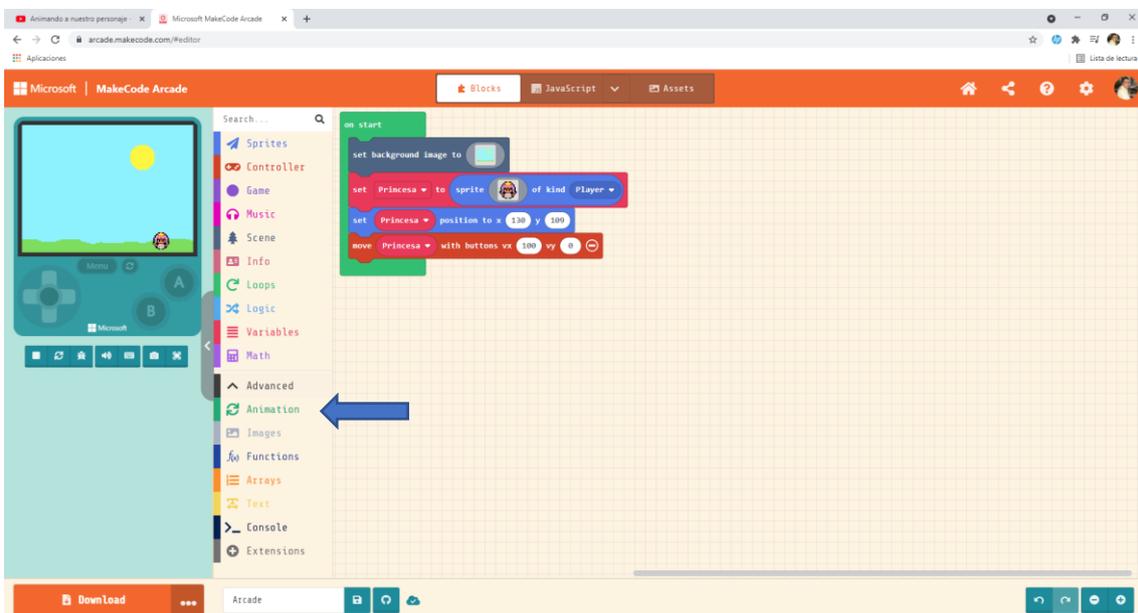


Ahora podrás observar moviendo las flechas de dirección que lo puedes desplazar de izquierda a derecha y viceversa, pero no podrás desplazarlo de arriba abajo y viceversa.

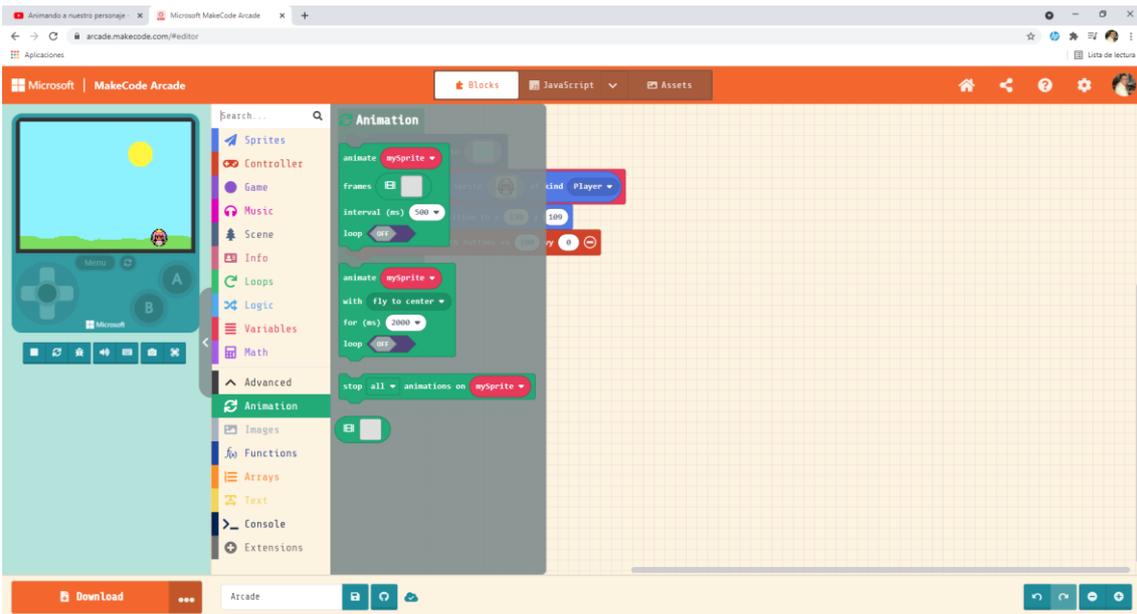
Para generar un efecto de animación:



Seleccionaremos Advanced.



Y podrás observar Animation.



Queremos que la princesa cuando se desplace esta cambie de perfil a la dirección que va.

Primero desde el grupo de bloques Controller, seleccionaremos

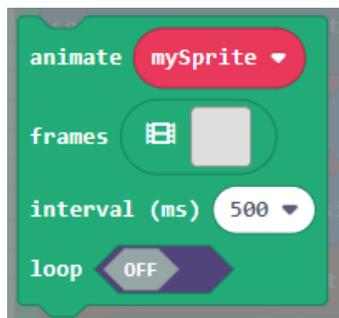


Vamos a cambiar por la tecla izquierda.

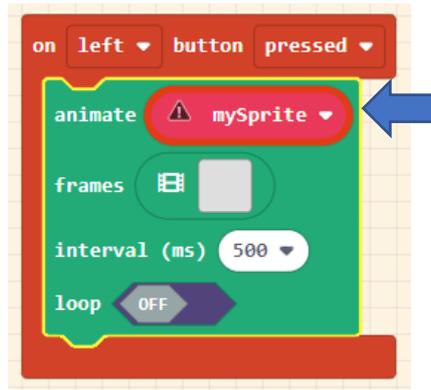


Ahora nos vamos a la categoría "Animation".

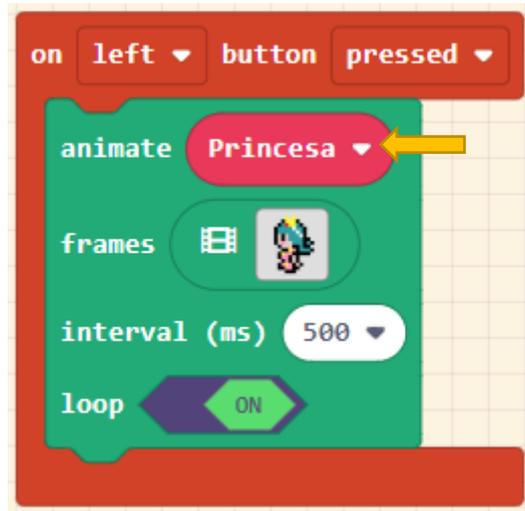
Vamos a seleccionar el bloque:



Lo arrastramos al bloque on left.

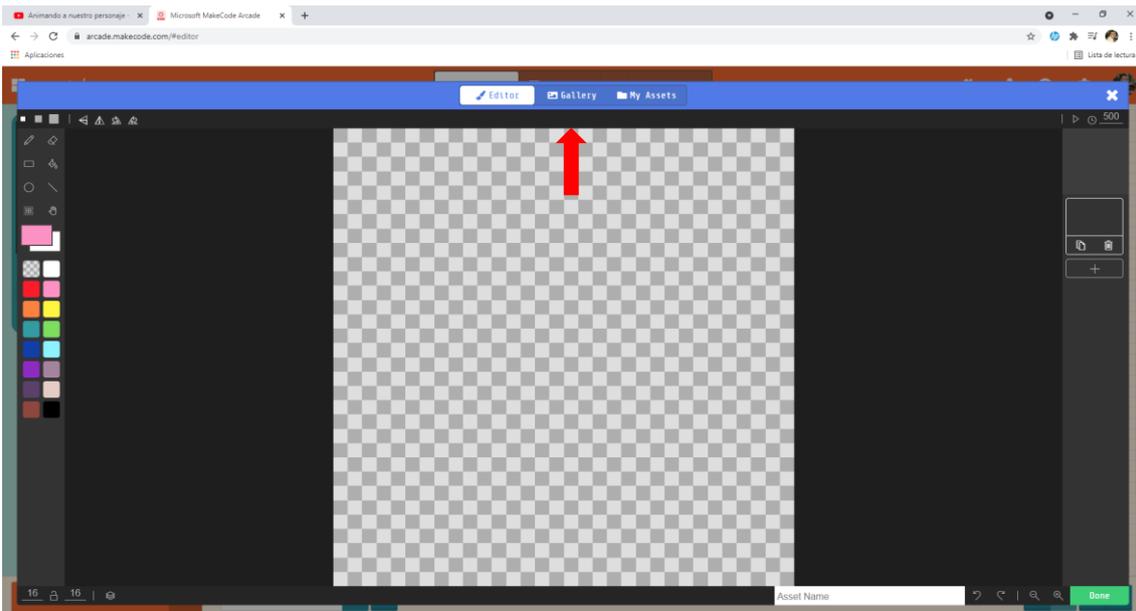


Vamos a seleccionar el nombre de nuestro personaje.

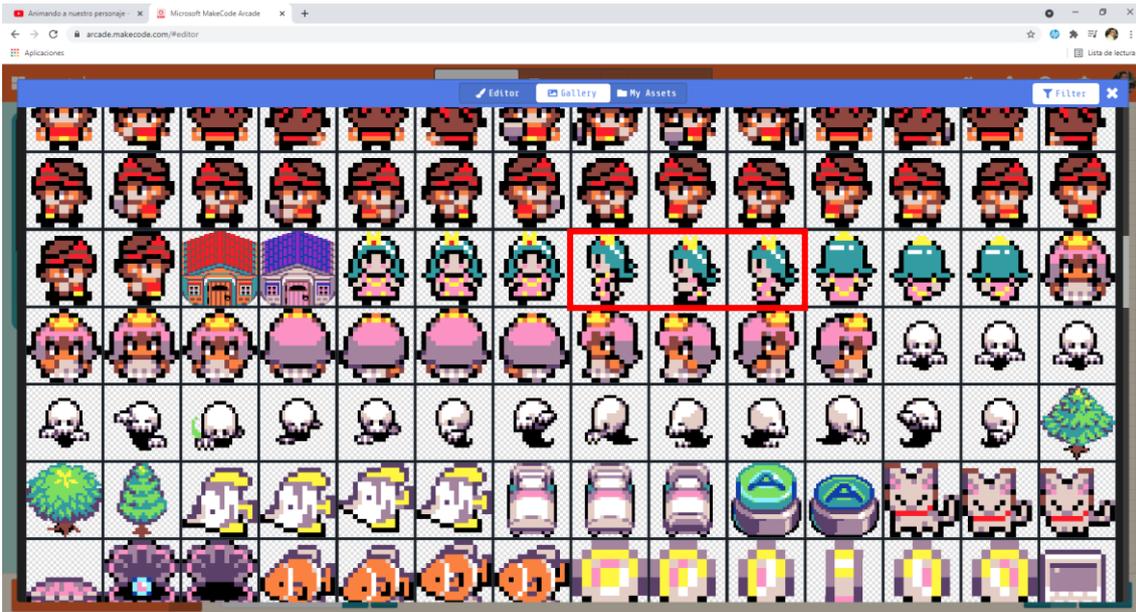


Vamos a seleccionar frames (fotogramas).

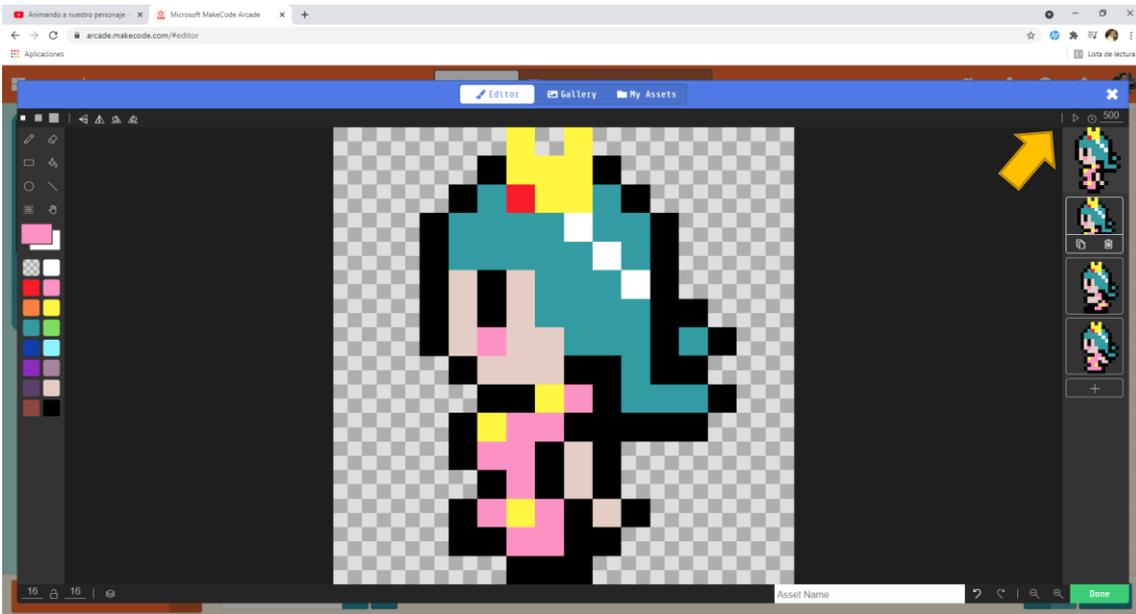
Activamos loop en ON.



Seleccionaremos Gallery.



Seleccionamos las tres imágenes que esta mirando hacia la izquierda.

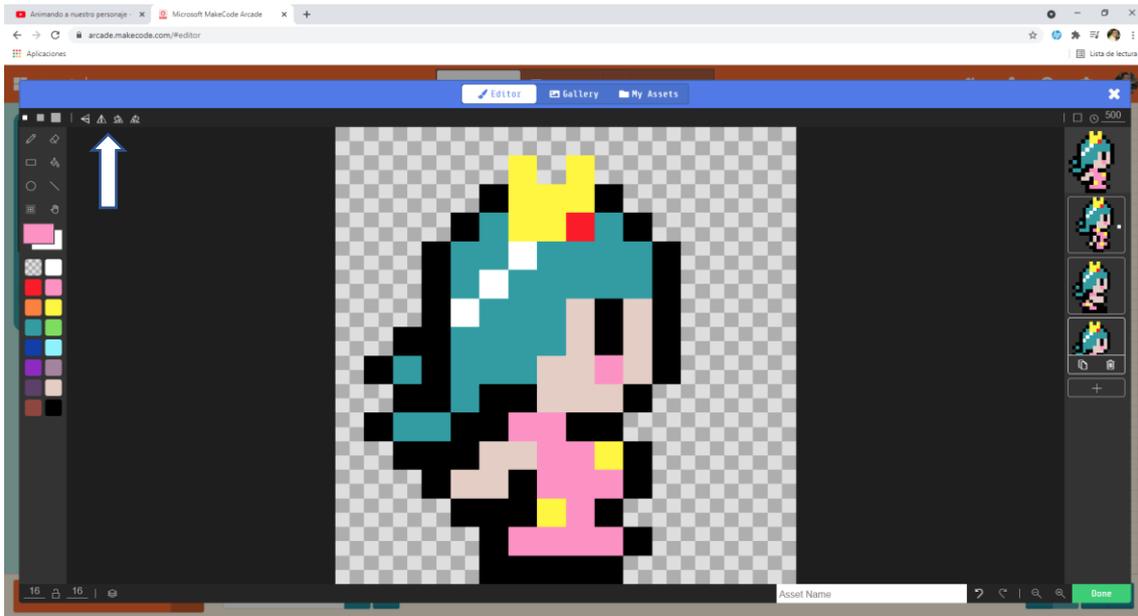


Si le damos al Play ya podemos ver la animación.

Ahora tienes que realizar los pasos necesarios para cuando le hacemos caminar hacia la derecha.

```
on right button pressed
  animate Princesa
  frames [ ]
  interval (ms) 500
  loop ON
```

Hemos cogido los mismos fotogramas pero los hemos orientado hacia la derecha.



## Mejorando la animación de nuestro personaje

Utilización del evento “reléase” que se produce en el momento que se libera una tecla con el fin de retornar a la animación de “descanso” del personaje.

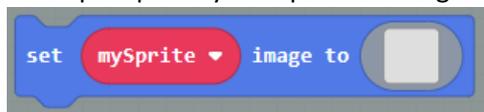
El problema es que cuando se suelta la flecha la animación sigue caminando, queremos que muestre una imagen mirando en frente como la primera que pusimos.

Seleccionamos Controller y seleccionamos el mismo bloque.

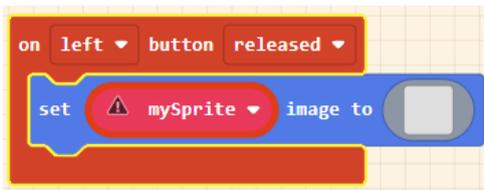


Cambiamos de presionado a liberado.

Seleccionaremos el grupo de bloque Sprites y del apartado Image seleccionaremos set image.



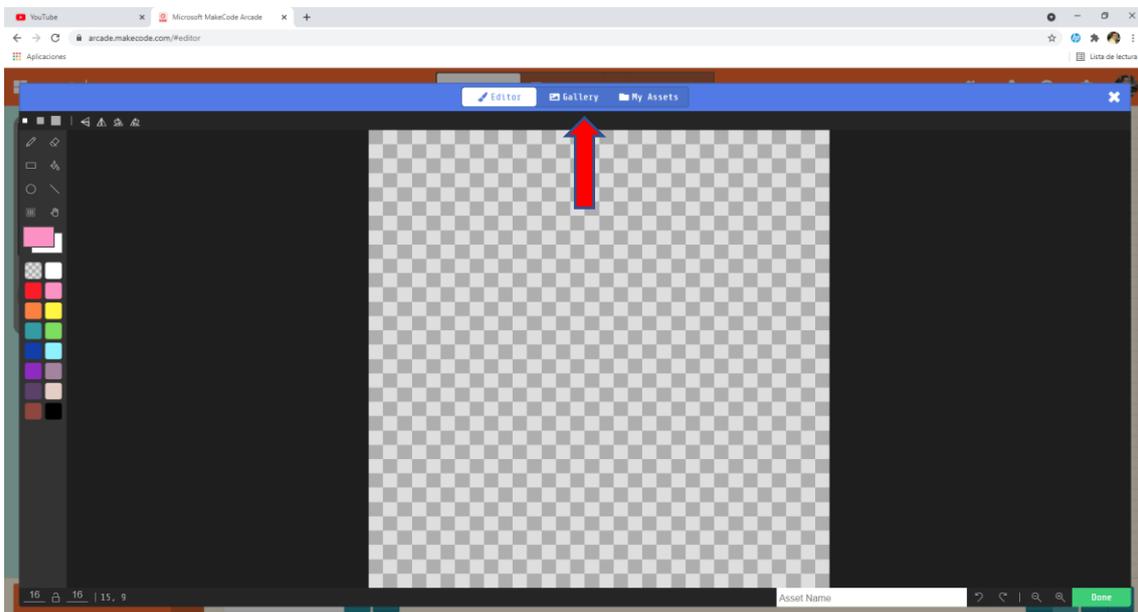
Lo insertamos en el bloque on left button released.



Seleccionaremos nuestro personaje.

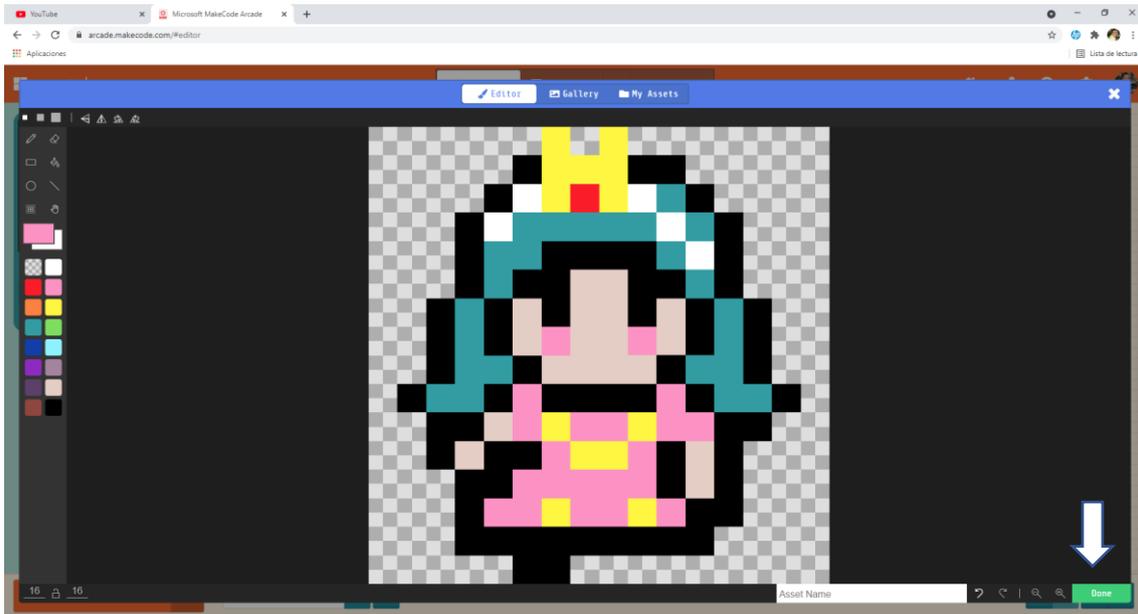


Ahora vamos a establecer la imagen que yo quiero.



Seleccionaremos Gallery.

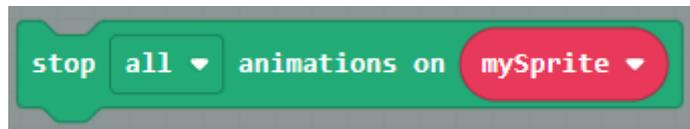
Seleccionaremos la princesa mirando de frente.



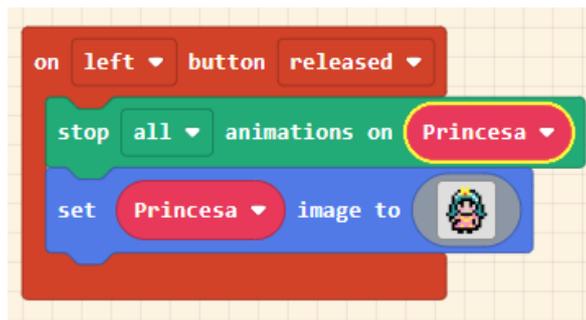
Seleccionamos el botón "Done".

Si lo pruebas verá que no acaba de funcionar como queremos.

Tenemos que ir al grupo Animation y seleccionaremos:



Ha de quedar de la siguiente forma:



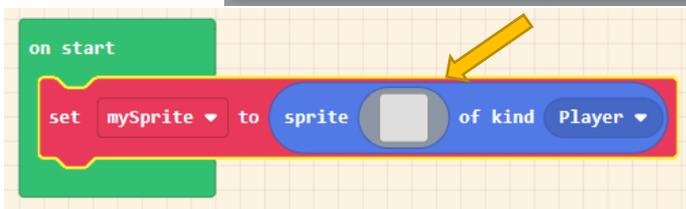
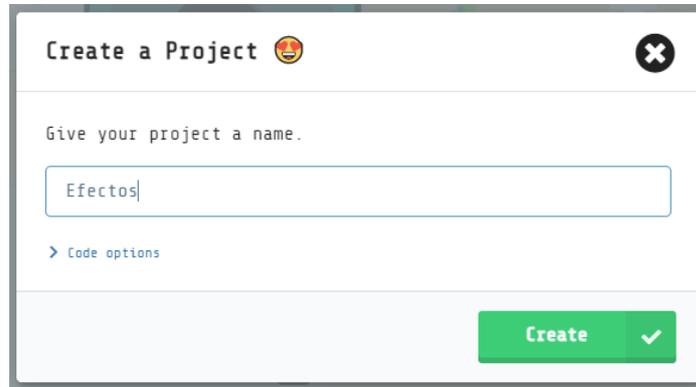
Lo mismo hay que hacer con on right button released.



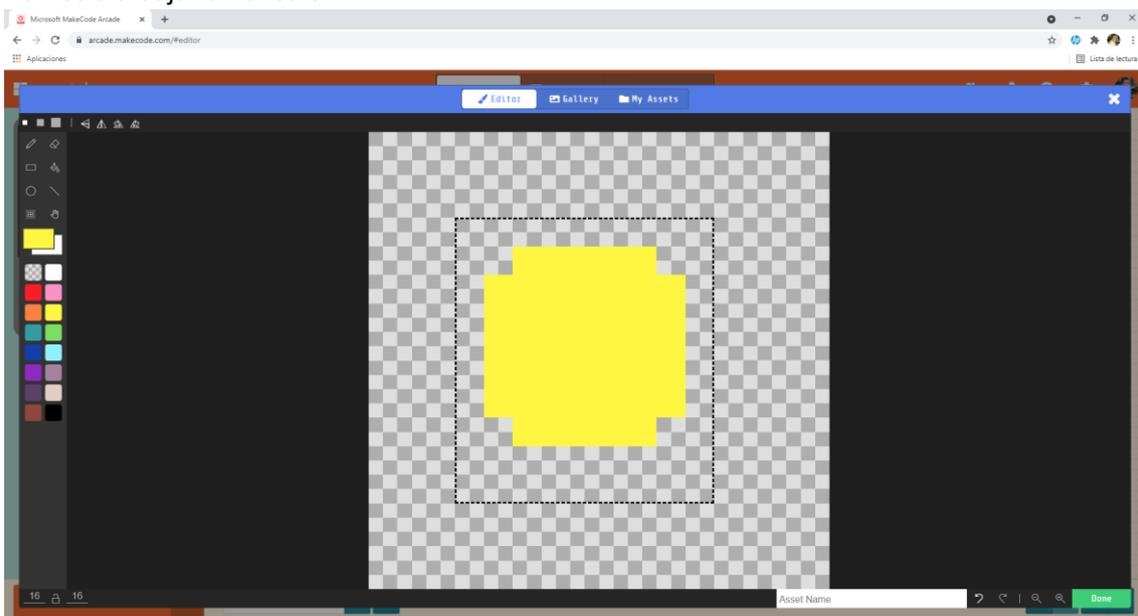
## Truco de animación

Utilización de las herramientas “animation” y “effects” de los sprites para lograr una divertida animación.

Para este apartado vamos a empezar con un nuevo proyecto.



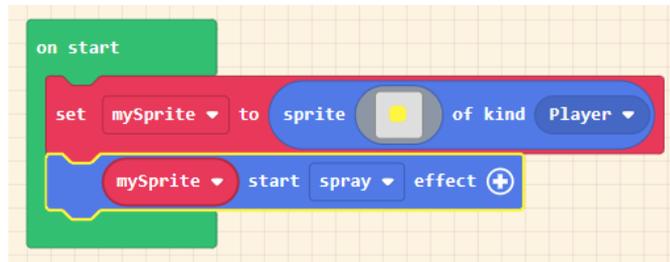
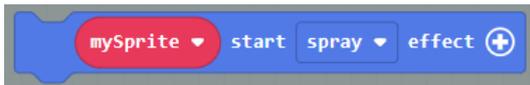
Vamos a dibujar un círculo.



Seleccionamos el botón “Done”.



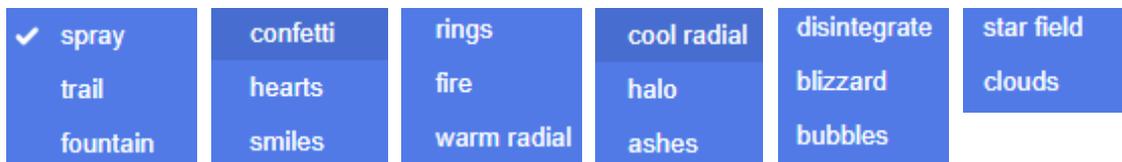
Agregamos un efecto del grupo Sprites apartado Effects.



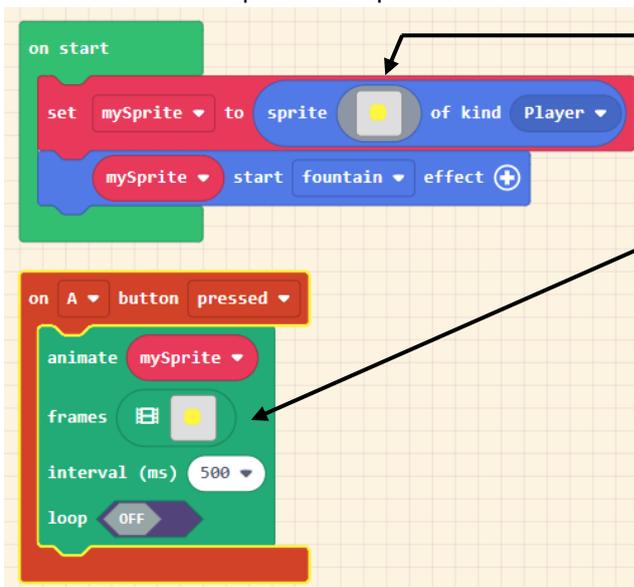
El efecto por defecto es "spray".



Hay más efectos:



Ahora queremos que en el momento de imprimir un botón A o B se realice el efecto. Seleccionaremos el primer bloque de Animation.



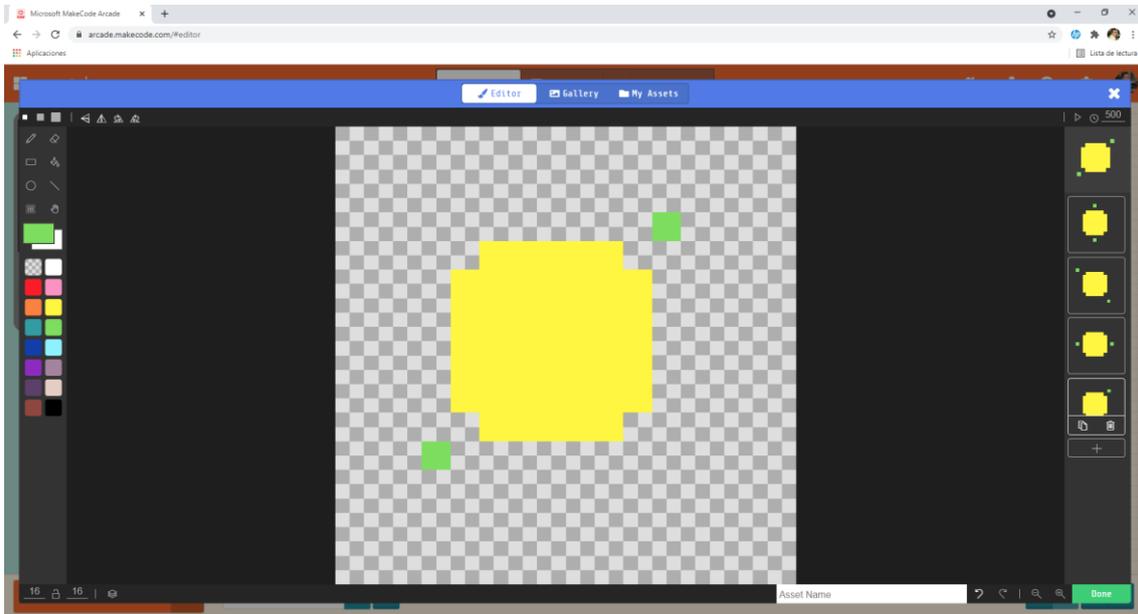
Abrimos el Sprite de arriba lo seleccionamos y hacemos Ctrl +c.  
Abrimos el Sprite de abajo y hacemos Ctrl + v.

Realizamos una modificación.

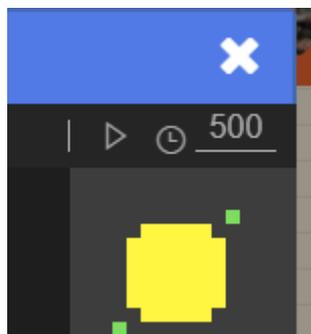
Duplicamos este fotograma y lo modificamos.

Volvemos a copiar este fotograma y lo vamos a modificar.

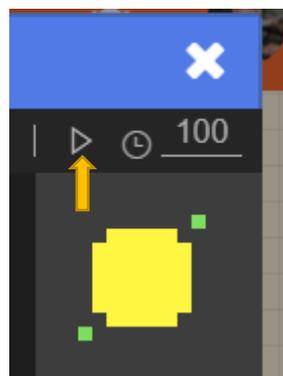
Falta el último.



Ha de quedar de la siguiente forma.



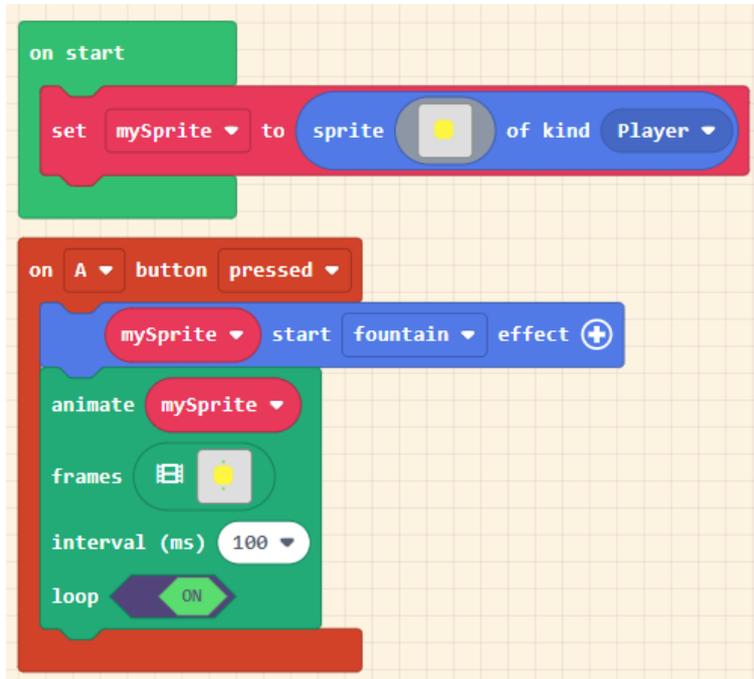
Vamos a cambiar la velocidad a 100.



Podemos reproducir para ver el efecto.

Seleccionamos el botón "Done".

Vamos a realizar el siguiente cambio.

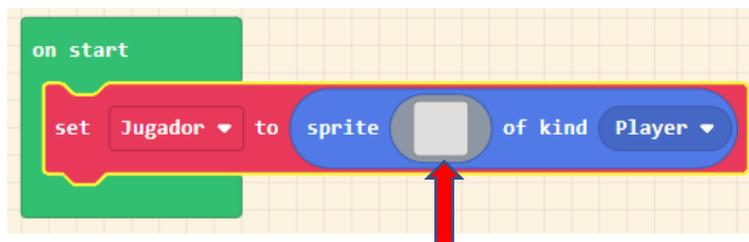
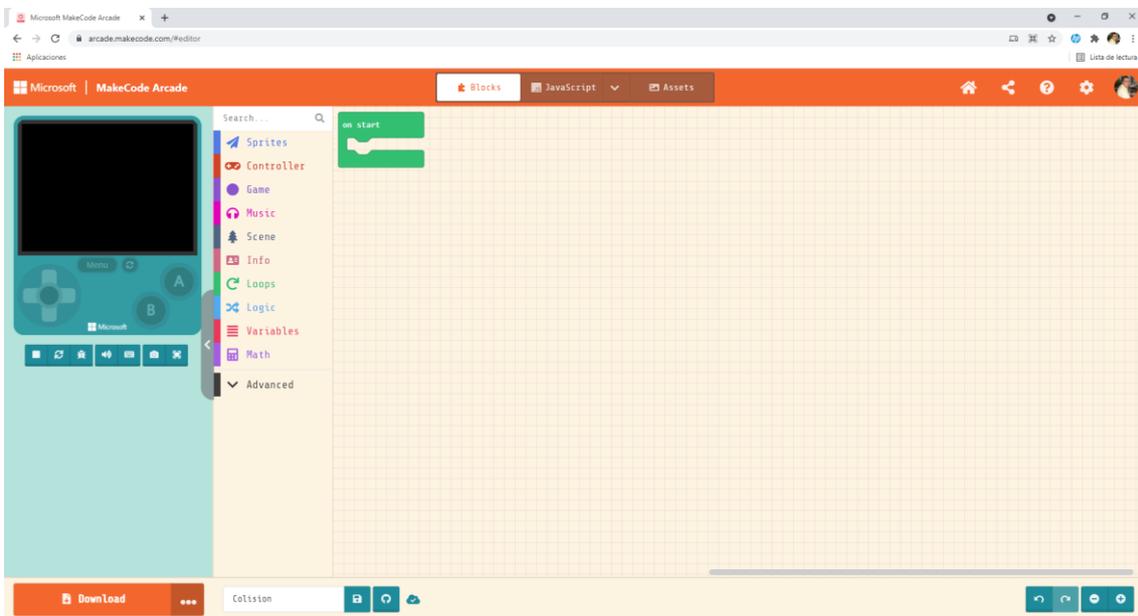
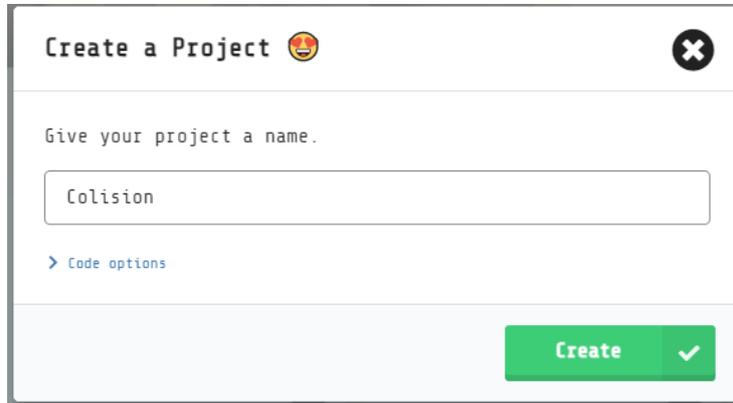


Cuando ejecutes en el visor luego tienes que presionar el botón A, ya empieza la animación.

## Programando colisiones entre sprites

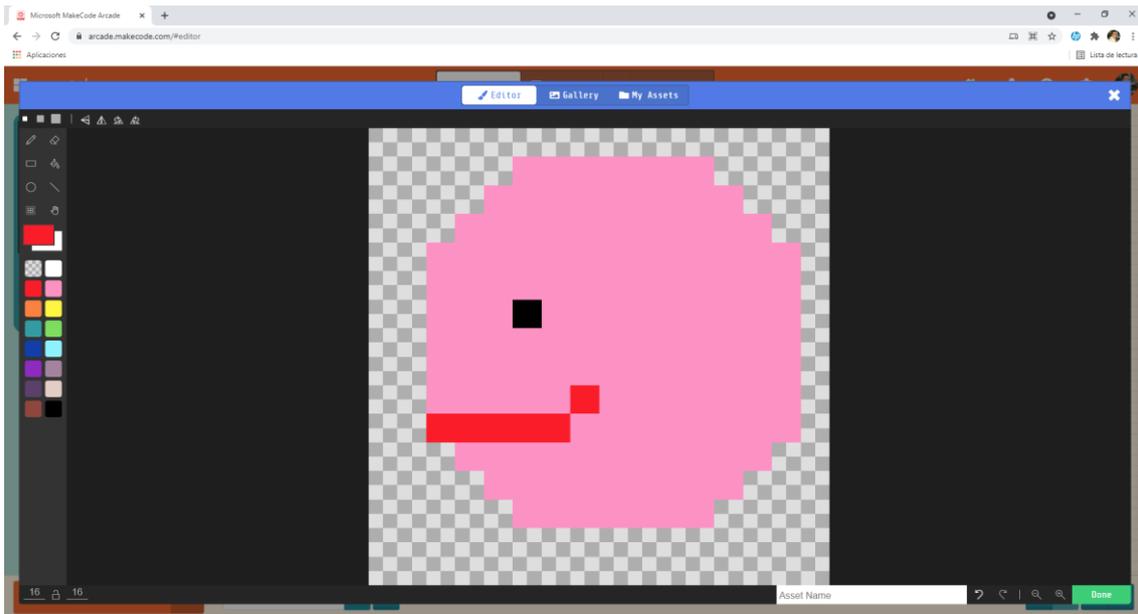
Utilizando el evento “on overlapping” para detectar la colisión entre dos objetos de una animación, y así programar una determinada consecuencia.

Vamos a crear un nuevo proyecto al que llamaremos “Colision”.

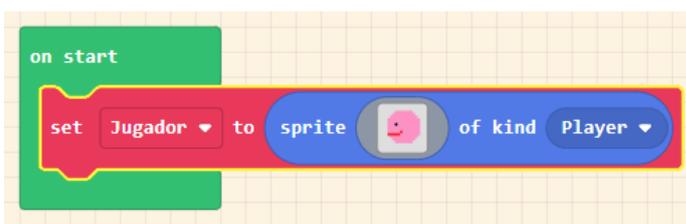
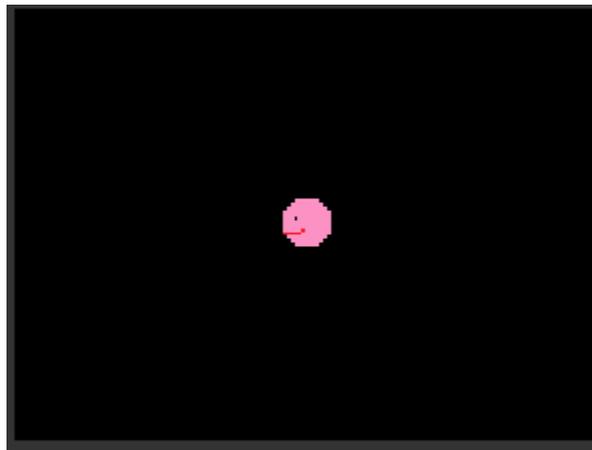


Creamos un Sprite llamado “Jugador”.

Vamos a dibujar.



Le damos al botón "Done".



Vamos a crear otro Sprite.

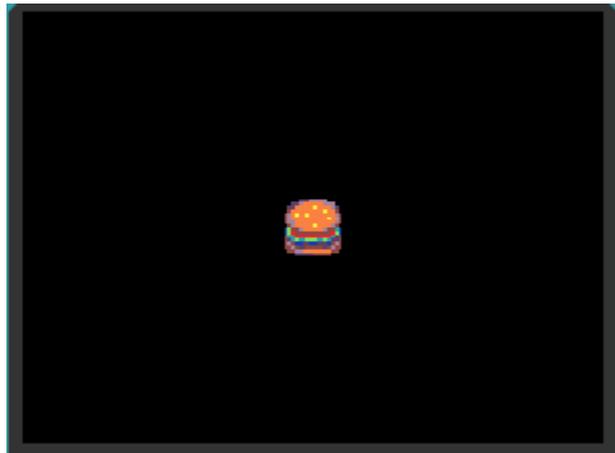


Le hemos puesto de nombre Hamburguesa y de la galería hemos cogido el dibujo.

Lo vamos a seleccionar de tipo alimento.

```

on start
  set Jugador to sprite of kind Player
  set Hamburguesa to sprite of kind Food
  
```



La hamburguesa se ha posicionado encima del jugador, vamos a mover el jugador.

En Sprites seleccionaremos:

```

set mySprite position to x 0 y 0
  
```

```

on start
  set Jugador to sprite of kind Player
  set Hamburguesa to sprite of kind Food
  set Jugador position to x 113 y 56
  
```

Cambiamos las coordenadas.



Ahora vamos a dar movimiento con Controller.

```

on start
  set Jugador to sprite of kind Player
  set Hamburguesa to sprite of kind Food
  set Jugador position to x 113 y 56
  move Jugador with buttons
  
```

Ahora verás que si utilizas la flechas de dirección el jugador se mueve.

Ahora vamos a ver que pasan si los dos sprites colisionan.

Ahora en el grupo Sprite y en la subcategoría Overlaps.

```

on sprite of kind Player overlaps otherSprite of kind Player
  
```

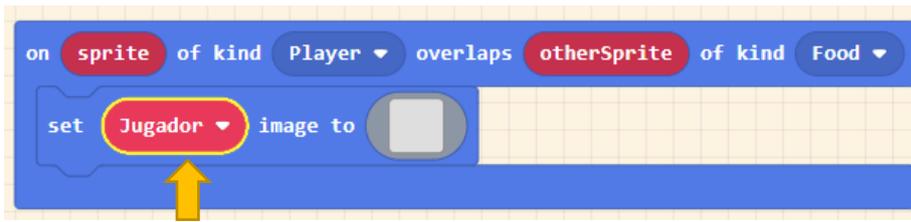


Seleccionaremos "Food" de alimento.

```

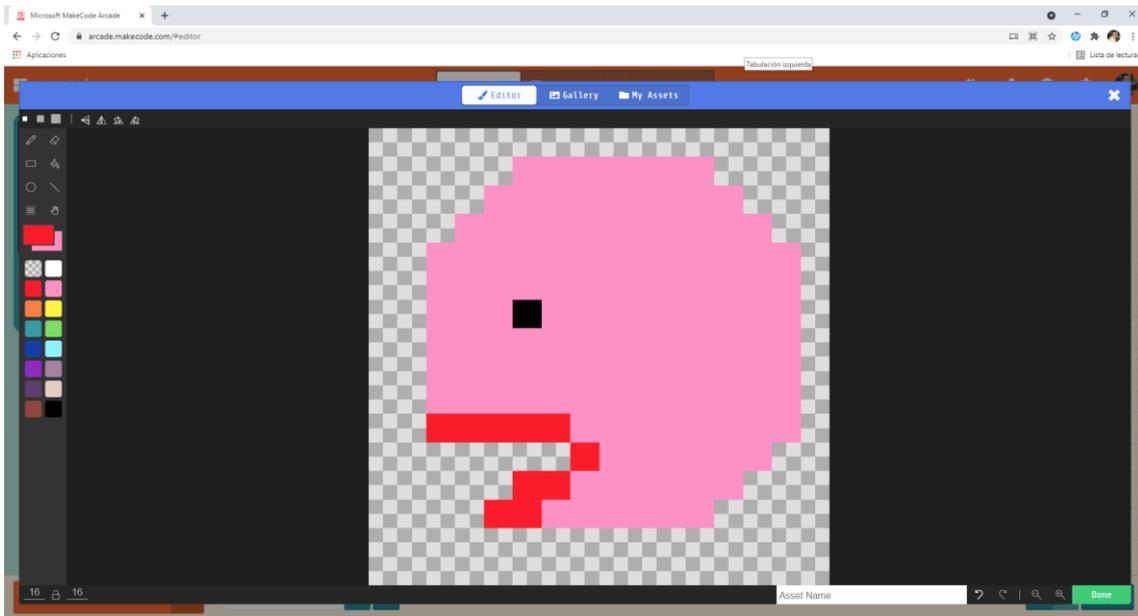
on sprite of kind Player overlaps otherSprite of kind Food
  
```

Ahora de la categoría Sprite subgrupo Image seleccionaremos set image to.



Seleccionamos Jugador.

Vamos a copiar el dibujo anterior y lo pegaremos en este y luego le haremos modificaciones.



Le hemos abierto la boca.



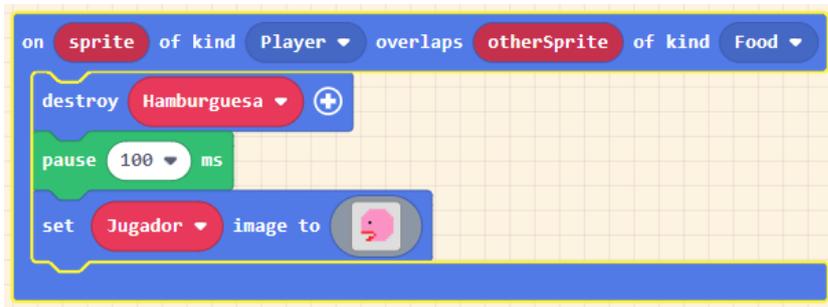
Para que sea más realista tendría que desaparecer la hamburguesa.

Del grupo Sprites en el subgrupo Effects seleccionaremos destroy.



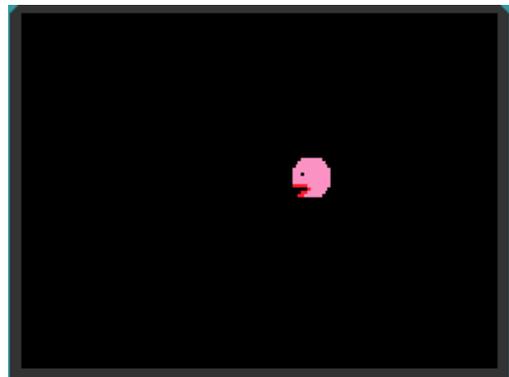
Especificamos la Hamburguesa.

Para poder una pausa vamos al grupo Loops y de este seleccionaremos pause.

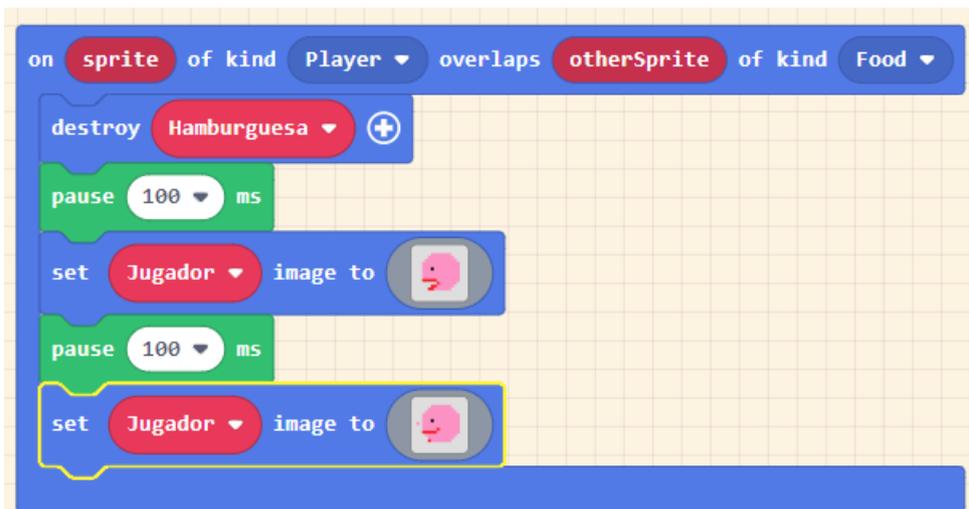


Vamos a probarlo.

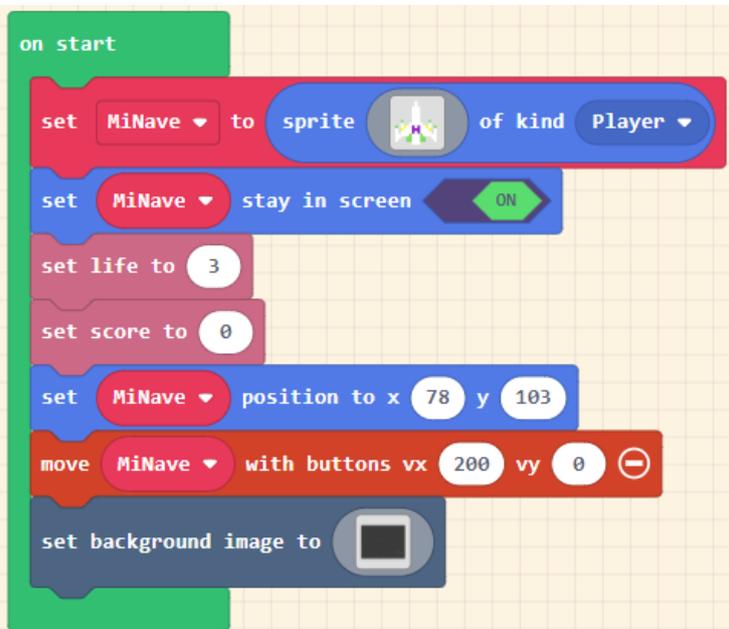
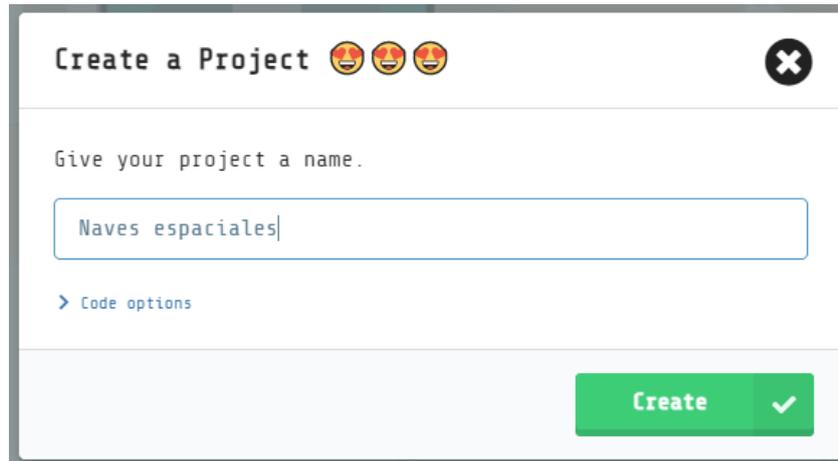
Cuando el jugador toca a la hamburguesa esta desaparece, espera unos 100 milisegundos y el jugador tiene la boca abierta.



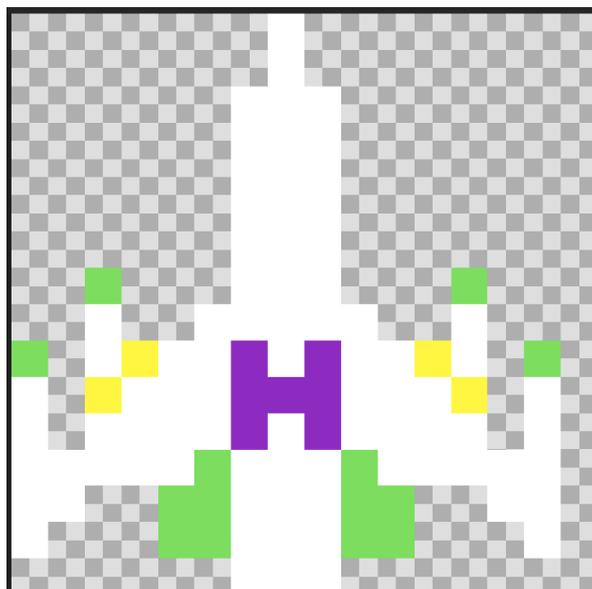
Si queremos que pasados otros 100 milisegundos vuelva a cerrar la boca.



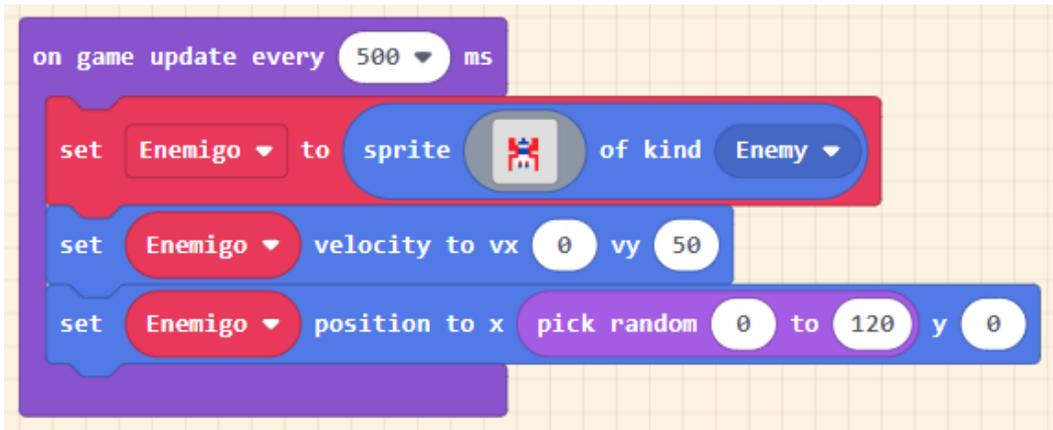
Vamos a crear un videojuego de naves espaciales



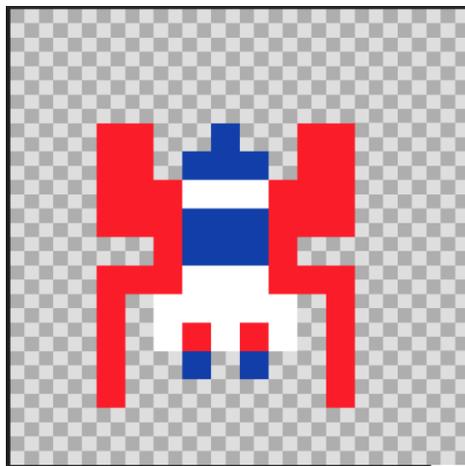
Cuando empieza el juego. Definimos un sprite con el nombre MiNave el dibujo se detalla a continuación y es de tipo Player. Controlamos que la nave cuando se mueva no salga de pantalla. Definimos 3 vidas. Definimos 0 puntos. Posicionamos MiNave a las coordenadas X: 78 y Y: 103 MiNave solo se podrá mover por las coordenadas x y no por las coordenadas y. (solo horizontal) Agregamos un fondo de color negro.



Cada 500 ms tiene que aparecer un enemigo.

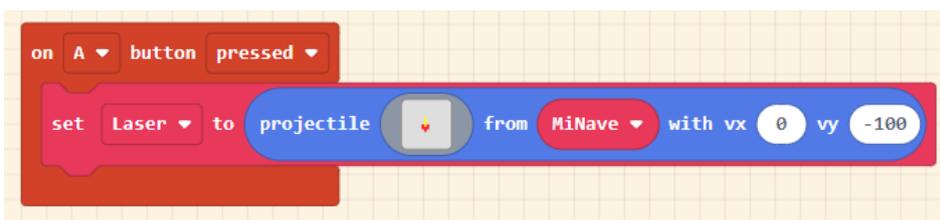


Vamos a definir el Enemigo con la siguiente imagen.



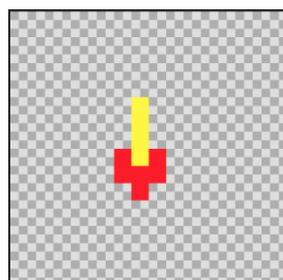
Nuestro Enemigo en las coordenadas y cogerá una velocidad de 50.

Nuestro Enemigo saldrá de una posición X: (valor aleatorio entre 0 y 2009 Y: 0.



Cuando presionamos el botón A

Definimos un Sprite llamado Laser que será un proyectil para MiNave y saldrá a una velocidad de -100.



```

on sprite of kind Player overlaps otherSprite of kind Enemy
  destroy otherSprite
  change life by -1
  
```

Si el Sprite de tipo jugador se superpone al enemigo.

Destruye al enemigo

A la variable "life" le restamos una vida.

```

on sprite of kind Projectile overlaps otherSprite of kind Enemy
  destroy otherSprite
  destroy sprite with fire effect for 100 ms
  change score by 10
  
```

Si el Sprite de tipo proyectil se superpone al enemigo.

Destruye al enemigo.

La variable "score" le incrementamos 10 puntos.

```

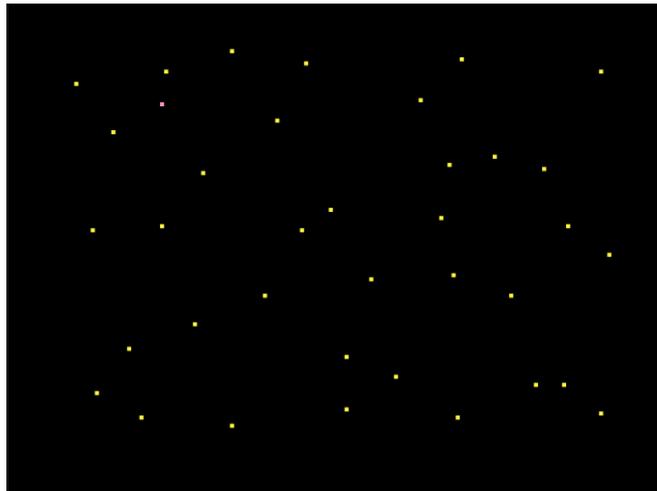
forever
  if score = 100 then
    game over WIN
  
```

Para siempre

Si la variable "score" puntuación llega a 100 entonces

Muestra en pantalla el mensaje de "game over".

Para que el cielo sea más real le vamos a poner estrellas.



Este fondo se encuentra en el bloque on start.

```
on start
  set MiNave to sprite  of kind Player
  set MiNave stay in screen ON
  set life to 3
  set score to 0
  set MiNave position to x 78 y 103
  move MiNave with buttons vx 200 vy 0
  set background image to 
```