

Primer ejercicio Logic



Descripción

En este ejercicio crearemos un videojuego estilo “Guitar Hero”, donde los elementos irán apareciendo desde la parte de arriba de la pantalla y nuestro jugador tendrá una posición asignada a cada tecla.

Para ello accederemos a [MakeCode Arcade](#) y realizaremos las operaciones necesarias.

Objetivos de programación y diseño

- Trabajar lógica mediante un control del juego en MakeCode Arcade.
- Trabajar y entender las variables en MakeCode Arcade.
- Asignar una posición del jugador a cada tecla.
- Descubrir las melodías y sonidos del programa y sus aplicaciones.
- Aumentar la dificultad cambiando la velocidad.

Programación del juego

CREACIÓN DE ASSETS

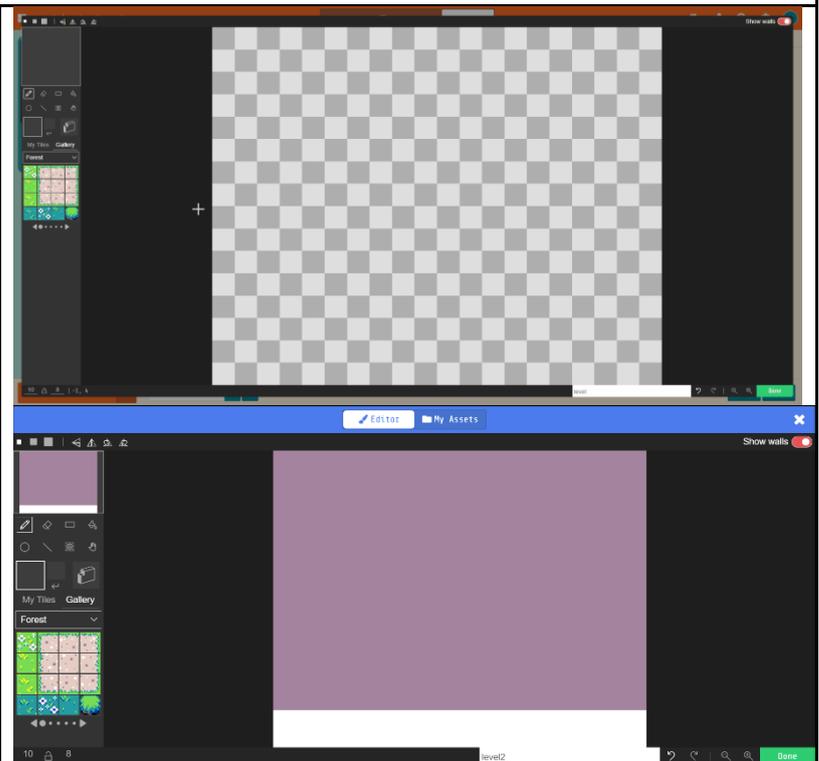
CREACIÓN SPRITE PROTAGONISTA

Te recomendamos utilizar una matriz de 16x16 px para el **Sprite** de **Steve**.



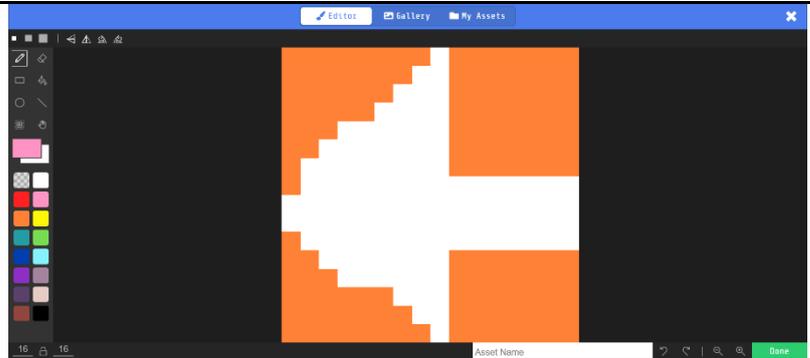
CREACIÓN TILEMAP

Te recomendamos utilizar una matriz de 10x8 px.



CREACIÓN SPRITES ADICIONALES

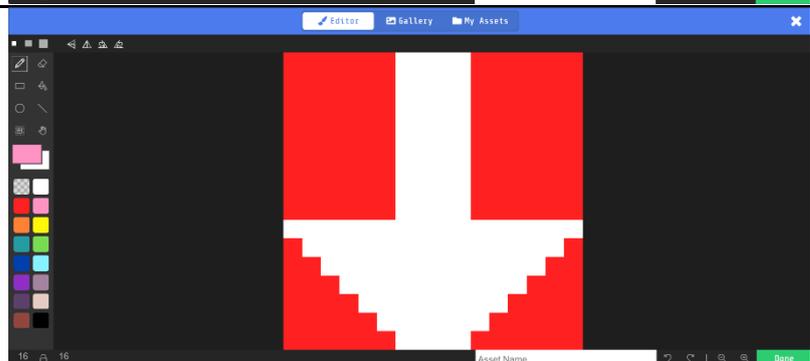
Te recomendamos utilizar una matriz de 16x16 px para el **Sprite** de **left**.



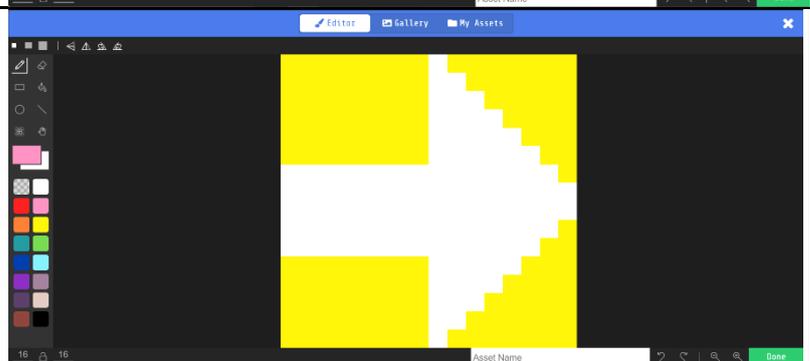
Te recomendamos utilizar una matriz de 16x16 px para el **Sprite** de **up**.



Te recomendamos utilizar una matriz de 16x16 px para el **Sprite** de **down**.



Te recomendamos utilizar una matriz de 16x16 px para el **Sprite** de **right**.



PROGRAMACIÓN PRINCIPAL

CREACIÓN INICIO DEL JUEGO

El primer paso es crear un bloque **on start** y crear nuestro **escenario**, además podemos darle un **efecto** al comenzar el juego.

El siguiente paso es crear el **sprite** de nuestro jugador, escogemos una de las imágenes de la galería y la aplicamos, después estableceremos una **posición** predeterminada para el personaje.

Añadimos la **velocidad** de las notas y lo establecemos en 40. También, añadimos un **marcador** y lo establecemos en 0 y también le damos al jugador 5 **vidas**.

Por último, añadimos el **tempo** de la reproducción de la música.

on start

set tilemap to

start screen star field effect

set Steve to sprite of kind Player

set Steve position to x 80 y 100

set speed to 40

set score to 0

set life to 5

set tempo to 120 (bpm)

CREACIÓN DE LA MECÁNICA APARICIÓN DE LAS NOTAS

Lo que haremos ahora será crear la mecánica que el juego elija aleatoriamente una de las 4 opciones cada medio segundo, establecer una posición de salida para cada flecha y aumentar la velocidad de aparición. Con el bloque "on game update every" vamos a hacer que lo que haya dentro pase cada medio segundo.

Con una **variable** que creemos "lane" estableceremos un carril de bajada para cada flecha y haremos que escoja una de las cuatro líneas al azar. Ahora, con un "if", daremos las instrucciones del carril número 1. En esta línea aparecerá el **sprite** de una flecha hacia la izquierda que será de tipo **proyectil**. Después le establecemos una **velocidad x** de 0 y en la **velocidad y** añadimos la **variable** "speed". Por último, estableceremos una **posición** para la salida de la flecha izquierda, en este caso será $x = 30$ e $y = 8$. En la línea 2, usaremos el **sprite** de la flecha hacia arriba y le damos los mismos parámetros de **velocidad**, pero cambiamos la **posición** a $x = 60$ e $y = 8$. Para las características de la línea 3 usaremos una flecha hacia abajo, estableceremos los mismos parámetros de **velocidad**, pero cambiamos la **posición** a $x = 100$ e $y = 8$. Para la última línea, usaremos una flecha hacia la derecha con los mismos parámetros de **velocidad** que el anterior, pero cambiando la **posición** a $x = 130$ e $y = 8$. Por último, si queremos añadirle dificultad, añadimos "set speed by 1", de esta forma la **velocidad** de

```

on game update every 500 ms
  set lane to pick random 1 to 4
  if lane = 1 then
    set left to sprite of kind Projectile
    set left velocity to vx 0 vy speed
    set left position to x 30 y 8
  else if lane = 2 then
    set up to sprite of kind Projectile
    set up velocity to vx 0 vy speed
    set up position to x 60 y 8
  else if lane = 3 then
    set down to sprite of kind Projectile
    set down velocity to vx 0 vy speed
    set down position to x 100 y 8
  else
    set right to sprite of kind Projectile
    set right velocity to vx 0 vy speed
    set right position to x 130 y 8
  change speed by 1
  
```

<p>salida de las flechas aumentará en uno.</p>	
<p>CREACIÓN MECÁNICA MOVIMIENTO JUGADOR</p>	
<p>Ahora es momento de darle movimiento al jugador para que pueda recoger las flechas que van cayendo, para ello, le damos la misma posición en el eje x al pulsar el botón izquierda que la que tiene la flecha izquierda y establecemos la nota en 1.</p>	<pre> on left button pressed set Steve position to x 30 y 100 set note to 1 </pre>
<p>Al pulsar el botón derecho le damos al jugador la misma posición en el eje x que la flecha derecha y establecemos la nota en 2.</p>	<pre> on right button pressed set Steve position to x 130 y 100 set note to 2 </pre>
<p>Al pulsar el botón arriba le damos al jugador la misma posición en el eje x que la flecha arriba y establecemos la nota en 3.</p>	<pre> on up button pressed set Steve position to x 60 y 100 set note to 3 </pre>
<p>Al pulsar el botón abajo le damos al jugador la misma posición en el eje x que la flecha abajo y establecemos la nota en 4.</p>	<pre> on down button pressed set Steve position to x 100 y 100 set note to 4 </pre>

CREACIÓN MECÁNICA PERDER VIDAS

Ahora, para que los bloques que no hayamos cogido no se acumulen debajo, vamos a añadir el bloque "on sprite of kind "projectile hits wall", con esto vamos a conseguir que las flechas se destruyan, podemos decorarlo con efecto de fuego y con un sonido. Además, el jugador perderá una vida cada vez que no los coja.

```

on sprite of kind Projectile hits wall at location
  destroy sprite with fire effect for 100 ms
  change life by -1
  play sound power down
  
```

CREACIÓN MECÁNICA OBTENCIÓN DE PUNTOS Y SONIDO DE LAS NOTAS

Ahora haremos un overlaps para que cuando el jugador coja una flecha se destruya la flecha con un efecto, que aumente la puntuación en 1 y les damos dos notas distintas que cogerá aleatoriamente con un volumen de 105.

```

on sprite of kind Player overlaps otherSprite of kind Projectile
  destroy otherSprite with spray effect for 100 ms
  set volume 105
  change score by 1
  set note2 to pick random 0 to 1
  
```

Vamos a darle los parámetros de la note 1 (flecha izquierda). Con un "if" estableceremos que si la note = 1 entonces ocurrirá algo, para ello introducimos un "if else" dentro del anterior "if" e introducimos la variable note2 junto con dos sonidos distintos, de esta forma, cogerá uno de los dos aleatoriamente.

```

set note2 to pick random 0 to 1
if note = 1 then
  if note2 = 0 then
    play tone at Middle C for 1/2 beat
  else
    play tone at Middle D for 1/2 beat
  
```



Ahora es el turno de la **note 2** (flecha derecha). Repetimos el proceso anterior con la **condición** dentro de otra **condición**, pero estableciendo la **C** y añadiendo dos **sonidos** distintos, de tal manera que si cogemos esta flecha sonará uno de los dos **aleatoriamente**.

```

play tone at Middle D for 1/2 beat
+
else if note = 2 then -
if note2 = 0 then
play tone at Middle E for 1/2 beat
else -
play tone at Middle F for 1/2 beat
+

```

Lo mismo con la **note 3** (flecha arriba). Añadimos otra **condición** dentro de otra **condición** y establecemos la **note = 3**, después introducimos otros dos **tonos** distintos para que escoja uno **aleatoriamente**.

```

play tone at Middle F for 1/2 beat
+
else if note = 3 then -
if note2 = 0 then
play tone at Middle G for 1/2 beat
else -
play tone at Middle A for 1/2 beat
+

```

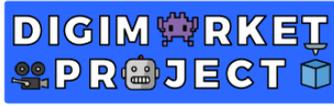
<p>Por último, la note 4 (flecha abajo). Con la condición dentro de la otra condición establecemos la note = 4, después, al igual que hicimos anteriormente, añadimos los dos tonos para que se reproduzca uno.</p>	
---	--

CREACIÓN MECÁNICA DE FINAL DEL JUEGO

<p>El último paso es colocar un "on life zero" y añadir efectos dentro para que cuando nuestra vida llegue a cero, aparezca game over y se paren los sonidos.</p>	
---	--

Con esta programación, nuestro jugador aparecerá en la pantalla y tendrá que ir moviéndose lateralmente para recoger las flechas que vayan cayendo y crear una melodía a la vez que suma puntos. Cuando no llegue a una nota perderá un punto de vida y si llega a cero, se acabará la partida.

Ahora, es tu turno de personalizarlo y añadirle contenido. Aquí te dejamos el nuestro para que te inspires un poco: <https://makecode.com/069M2R7W0dX8>



Glosario

Condicionales: Secuencia de instrucciones que se ejecuten en función del valor de una condición.

Ejemplo: If, If...Else

If: Sentencia condicional que, según el resultado de una operación lógica, ejecuta una secuencia de instrucciones o se omite.

If-Else: Sentencia condicional que si se cumple ejecuta una secuencia de instrucciones y otra que omite.

If...Else if: Secuencia de condicionales en la cual, iremos pasando, de manera ordenada, de una condición a otra hasta que se cumpla una de ella.

Operadores de comparación: Son operadores que comparan un valor a otro y se introducen dentro de una condición.

Variables: Es un espacio asociado a un identificador, en ese espacio hay un valor que puede ser modificado.

Funciones: Es un subprograma que recoge un conjunto de instrucciones y pueden ejecutarse desde el programa principal haciendo una llamada a él.

Secuencias: Secuencia o instrucción es una acción programada que realizará el ordenador

Evento: Ejecuta una secuencia de instrucciones cuando ocurre un suceso externo al sistema.

Jugador: Un participante en un juego

Aceleración: Es la variación de velocidad por unidad de tiempo.

Velocidad: Es una magnitud física que relaciona la posición con el incremento de tiempo.

Escenario: Espacio donde se desarrolla el videojuego.

Aleatoriedad: La generación de números que tienen la misma probabilidad de generarse.

Puntuación: Puntos totales que obtiene un jugador al realizar ciertas interacciones.

Vida: Recurso que tiene el jugador para poder seguir en la misma partida. Una vez que terminen las vidas, la partida se acaba.

Game Over: La partida se ha terminado. Se suele mostrar puntuaciones y te pregunta si quieres jugar otra partida.

Música: Combinación de sonidos y silencios que componen un ritmo.

Género de juego: Una clasificación de los videojuegos en función de su jugabilidad.

Efecto: Algo que se aplica sobre el escenario, objeto, personaje y más elementos para transmitir realismo o una sensación dentro del juego.

Paleta de Colores: Panel donde se dispone de una variedad de colores que permite seleccionar un color para poder aplicarlo en algún elemento del videojuego.