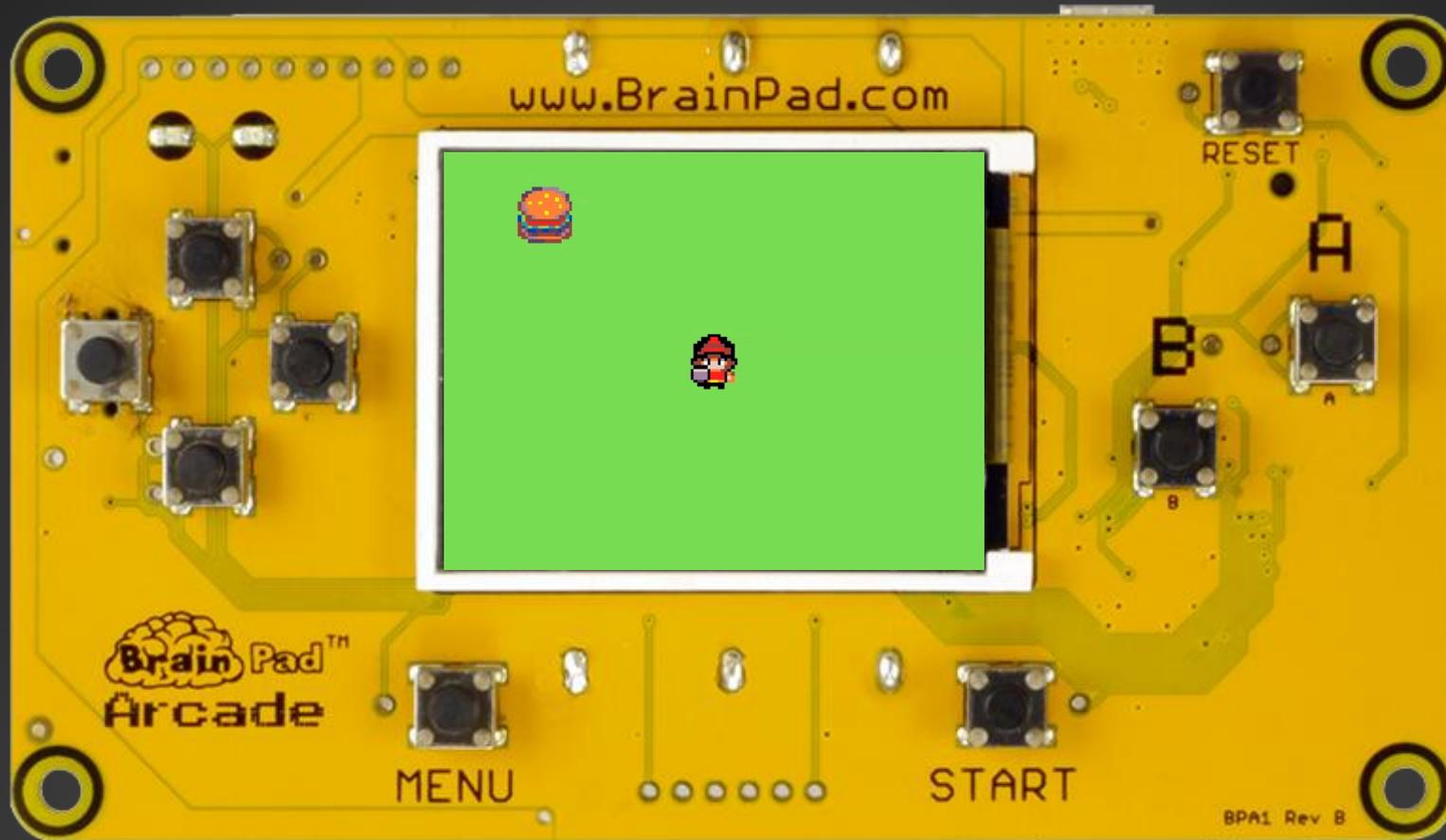


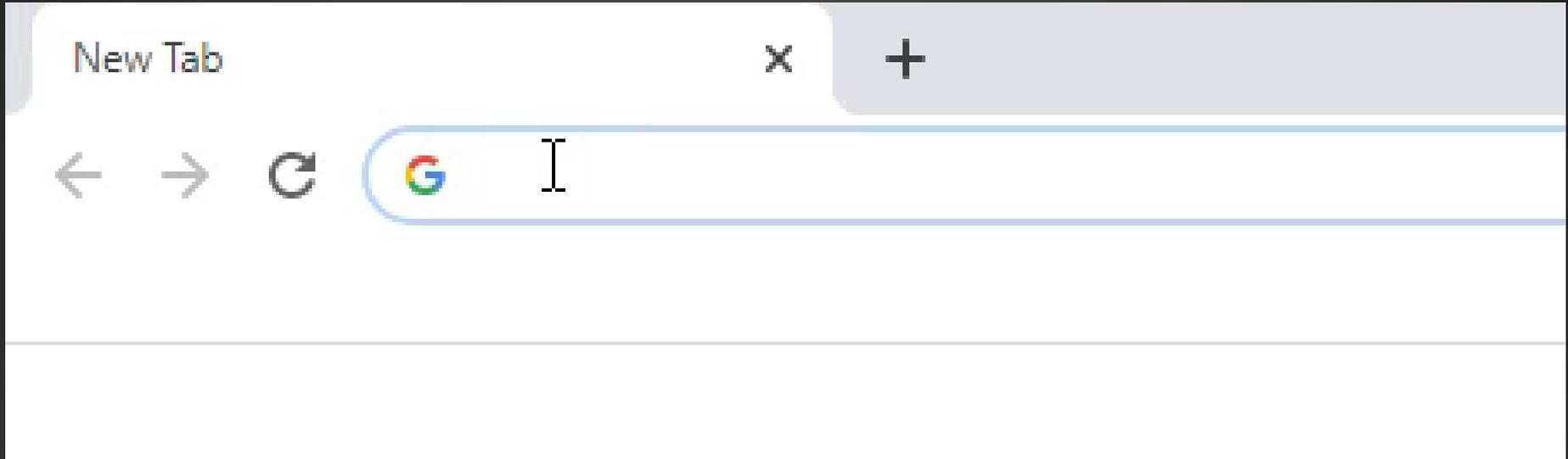


2-D ADVENTURE – EPISODE 01

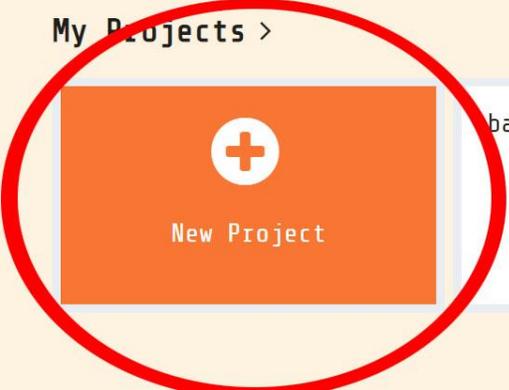
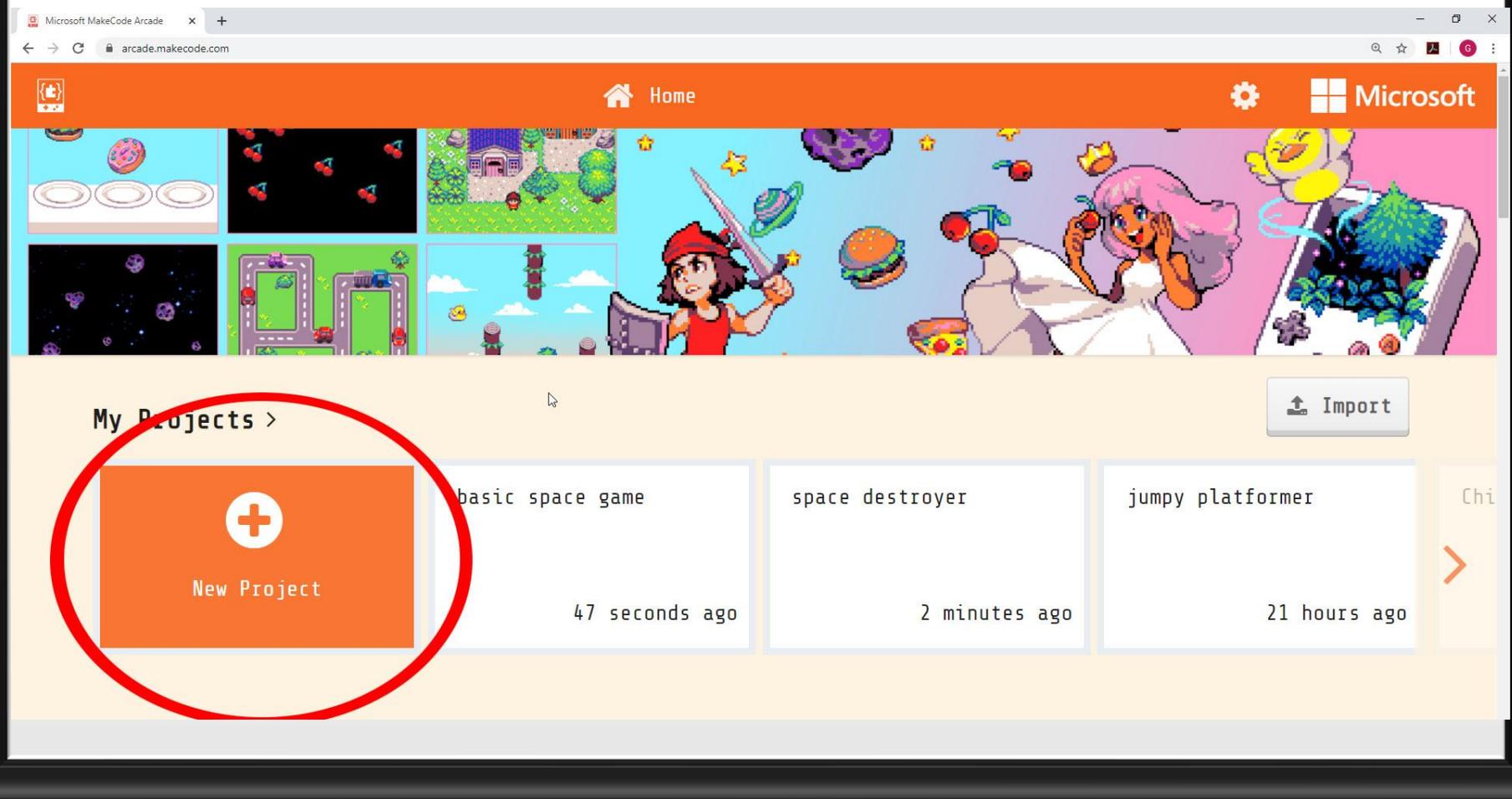


Let's create a 2-D Adventure game.

Open a web browser and navigate to...



arcade.makecode.com





Your project needs a name 😊😊😊

Give your project a name.

Create ✓

My Projects >

New Project

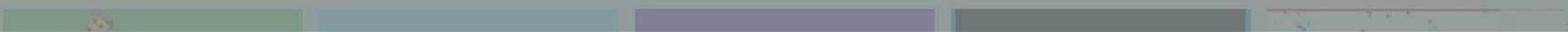
20 Ad
39 minutes ago

asic space gam...
5 days ago

space
5 days ago

space
5 days ago

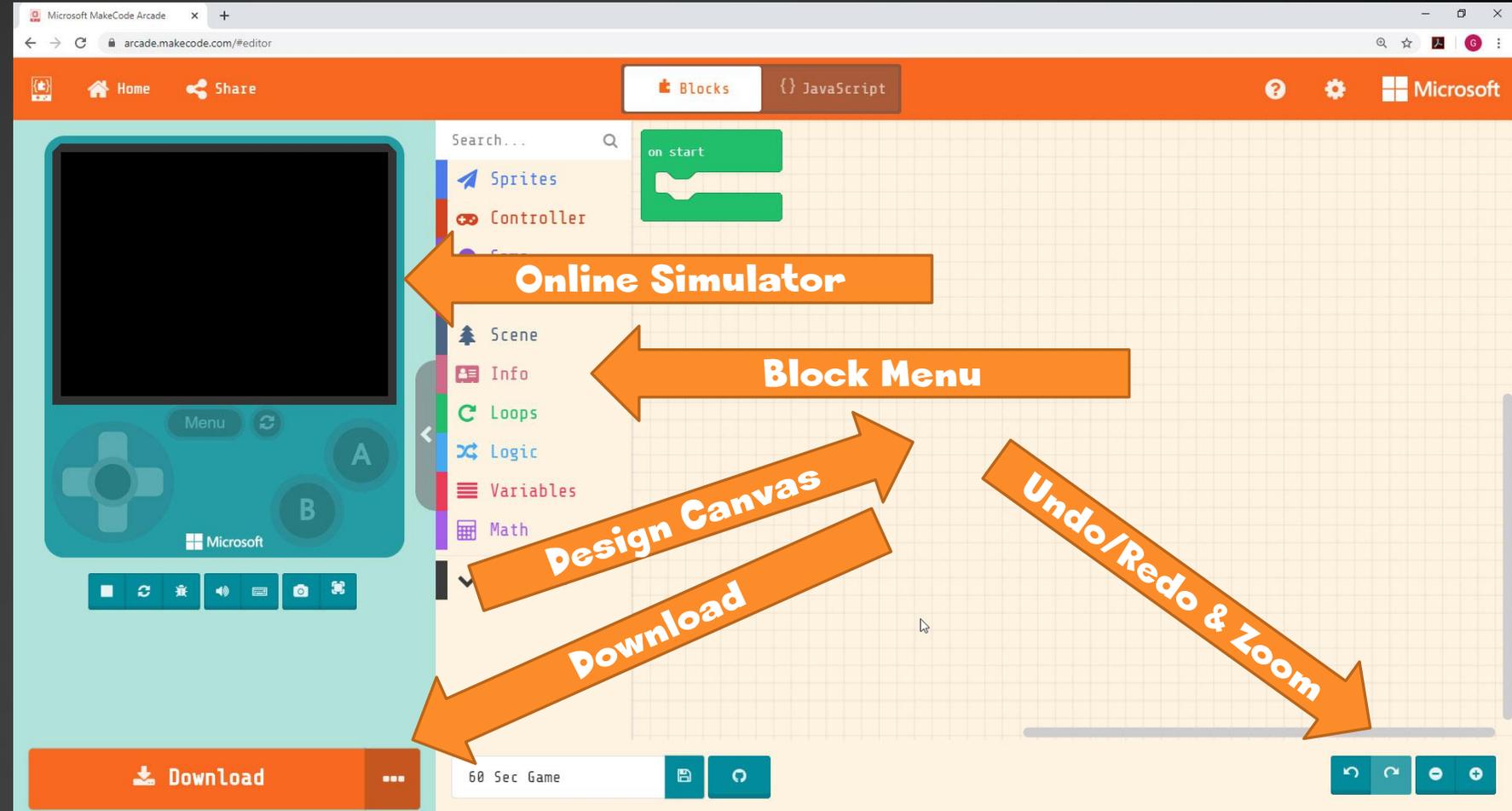
Tutorials



DESIGN SURFACE

This is where the game you create will come to life!

- Online Simulator
- Block Menu
- Design Canvas
- Download (for later)
- Undo/Redo & Zoom





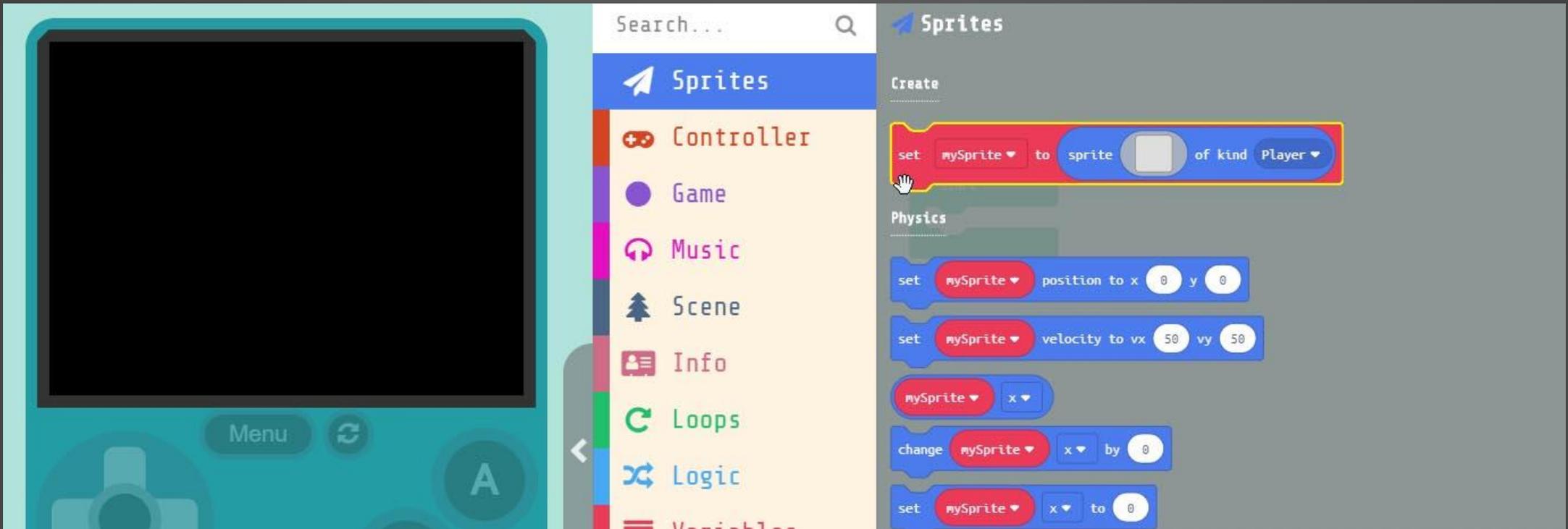
Computer Term:

SPRITE

A sprite is a two-dimensional image that is integrated into a larger scene, most often in a 2D video game.

EXAMPLE:





Grab the  block

and drag it into the  block

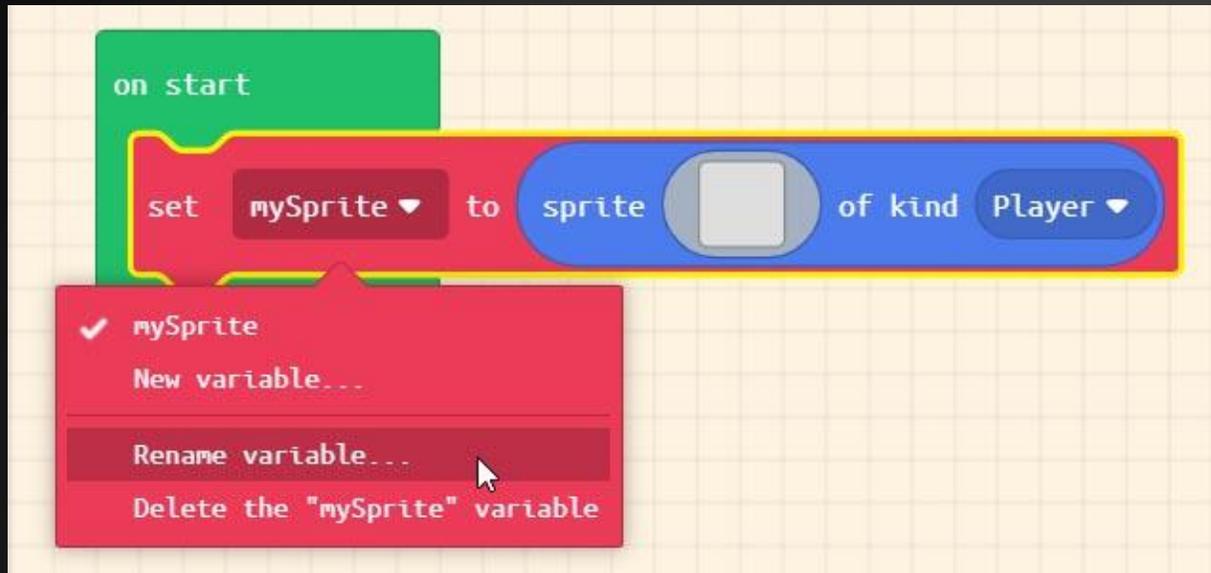


Computer Term:

VARIABLE

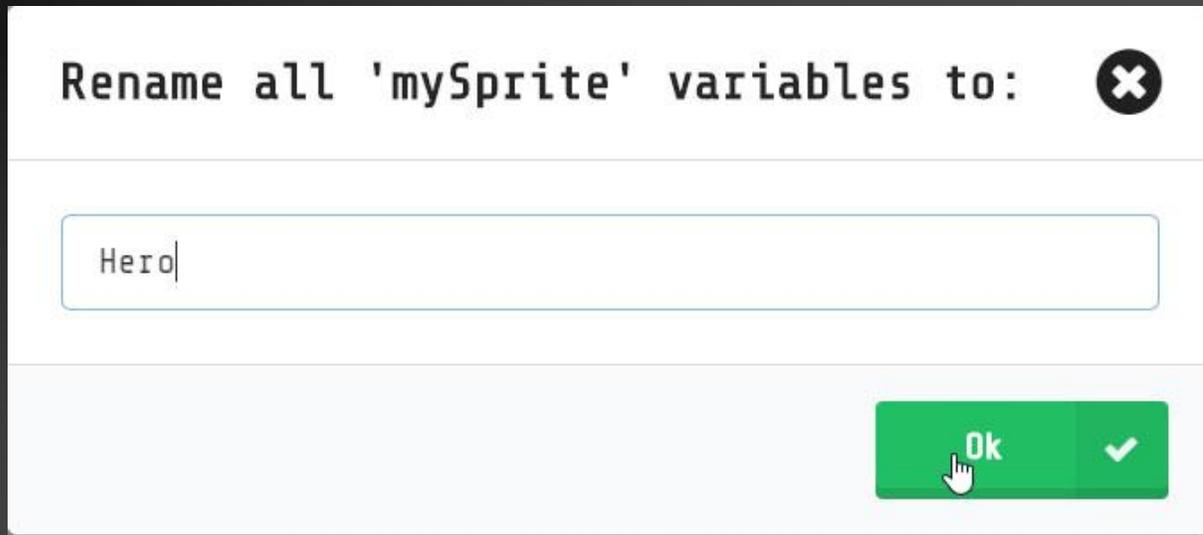
A variable is a letter or word, such as “x” or “score” that represents a changing value. Variable can be named anything but should be meaningful. To make code easier to read.

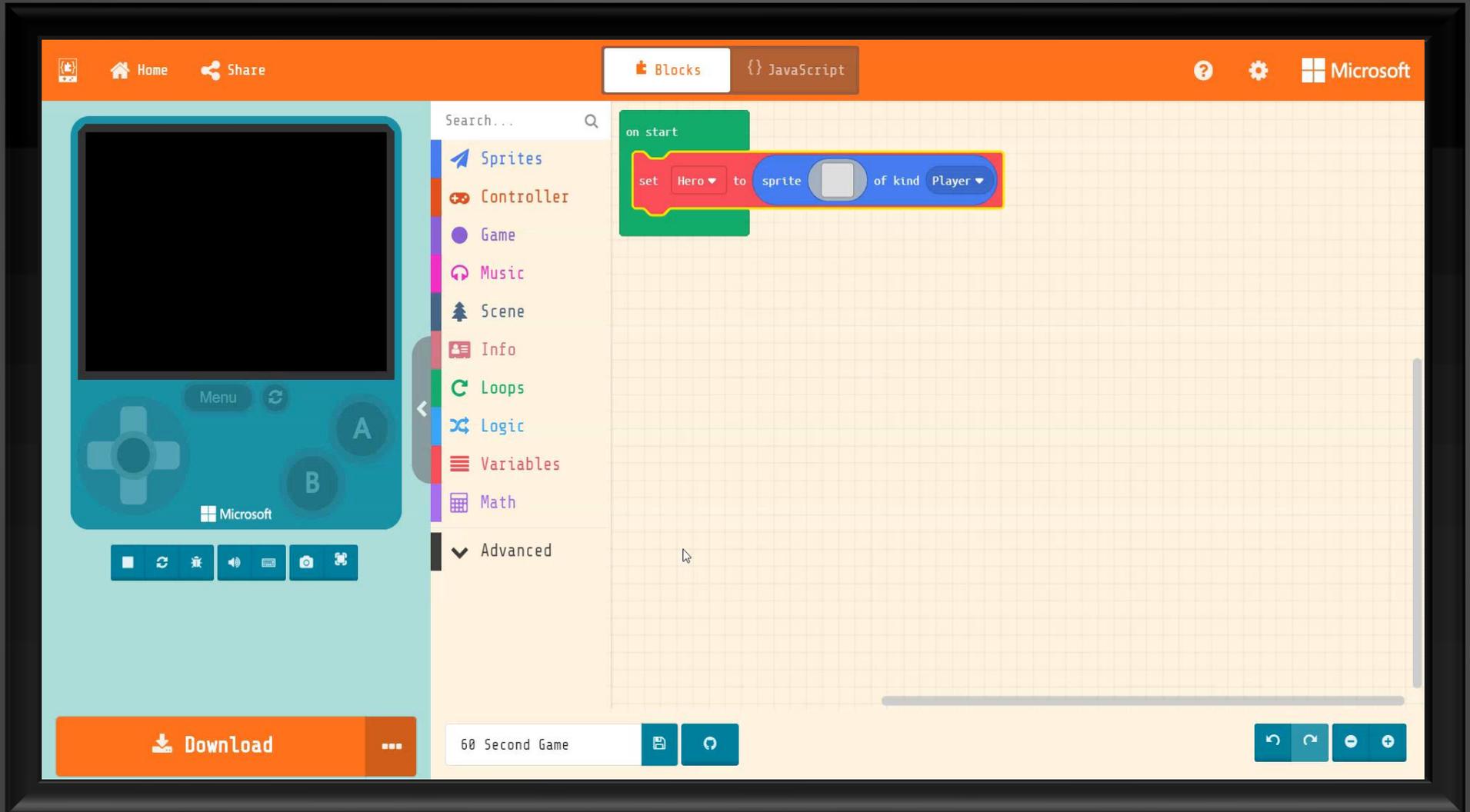
EXAMPLE: `score=score + 1`

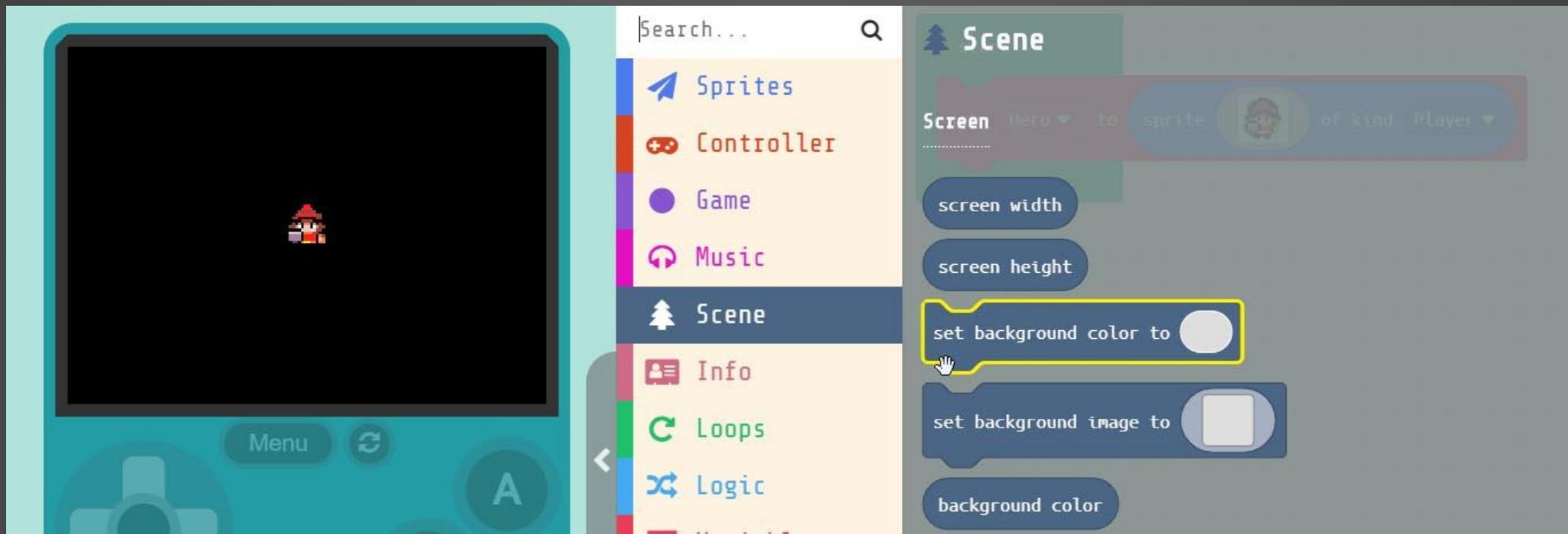


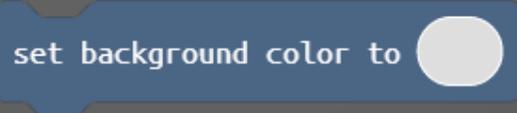
Naming our first 'variable'

Rename 'mySprite'
to 'Hero'

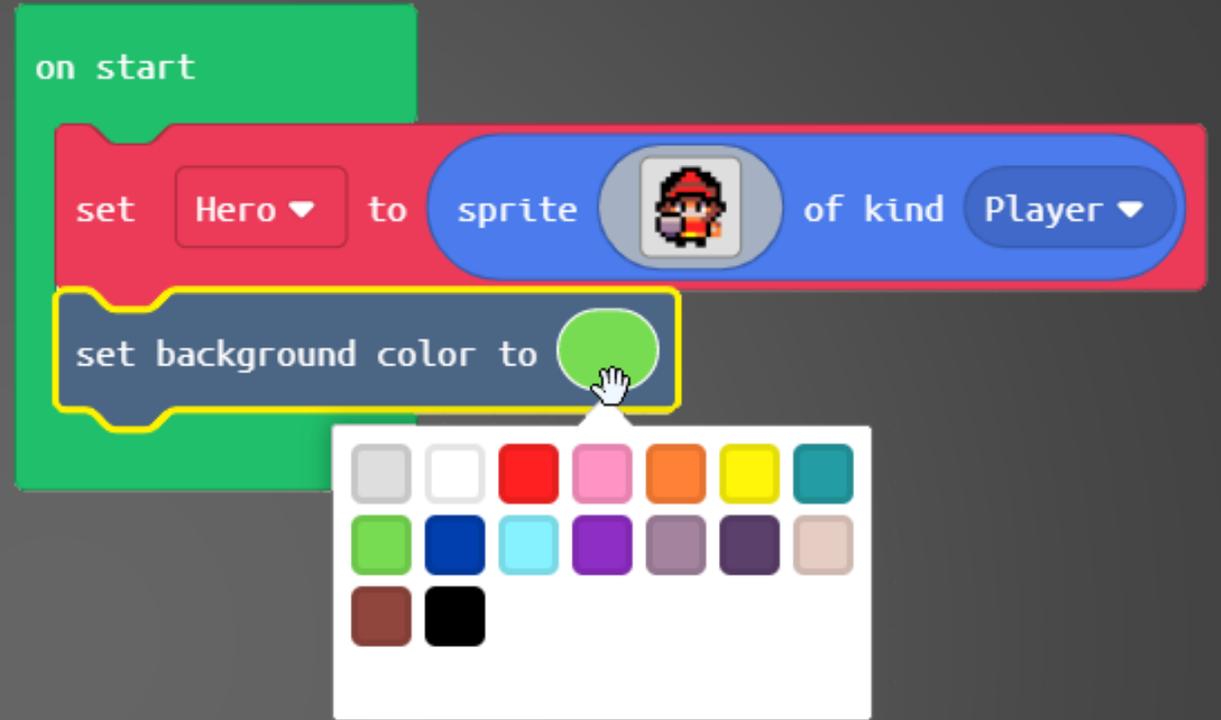
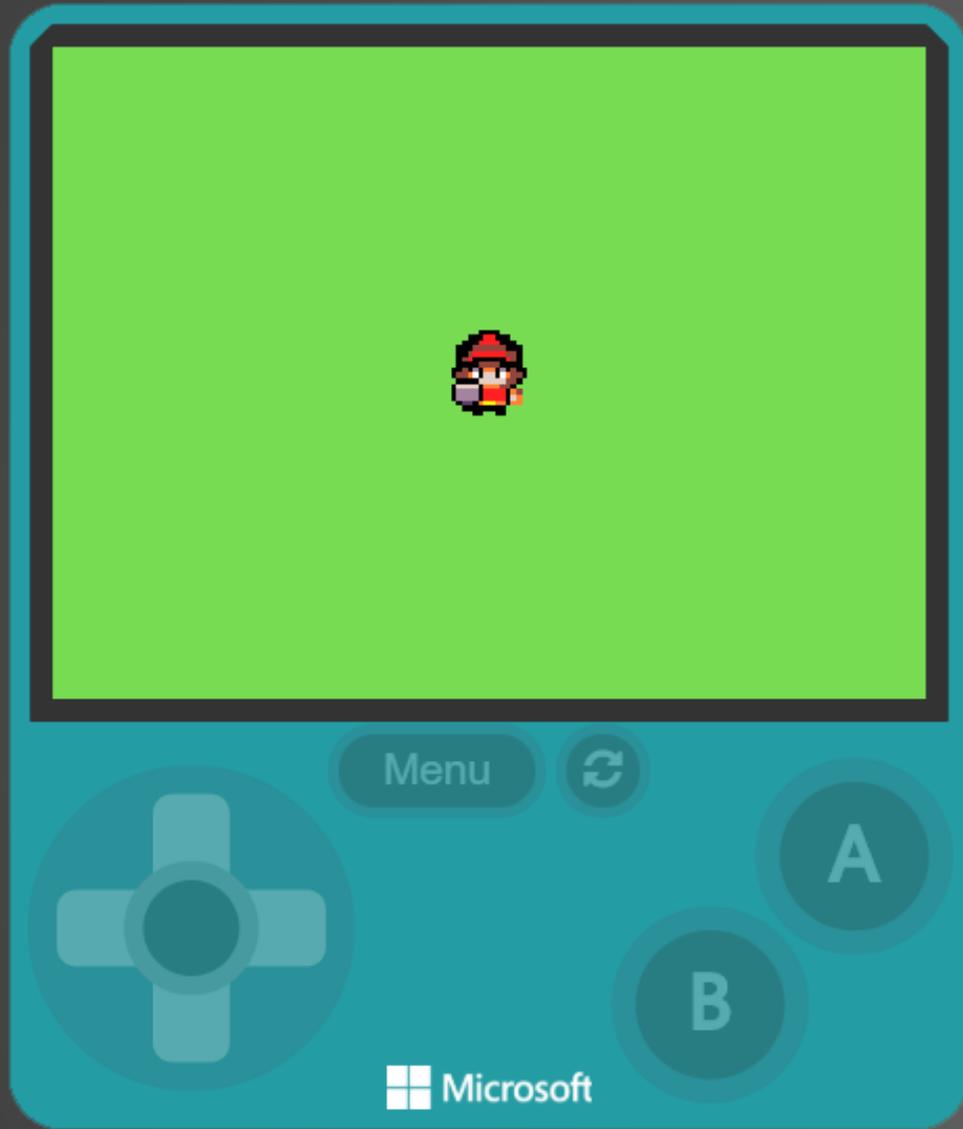




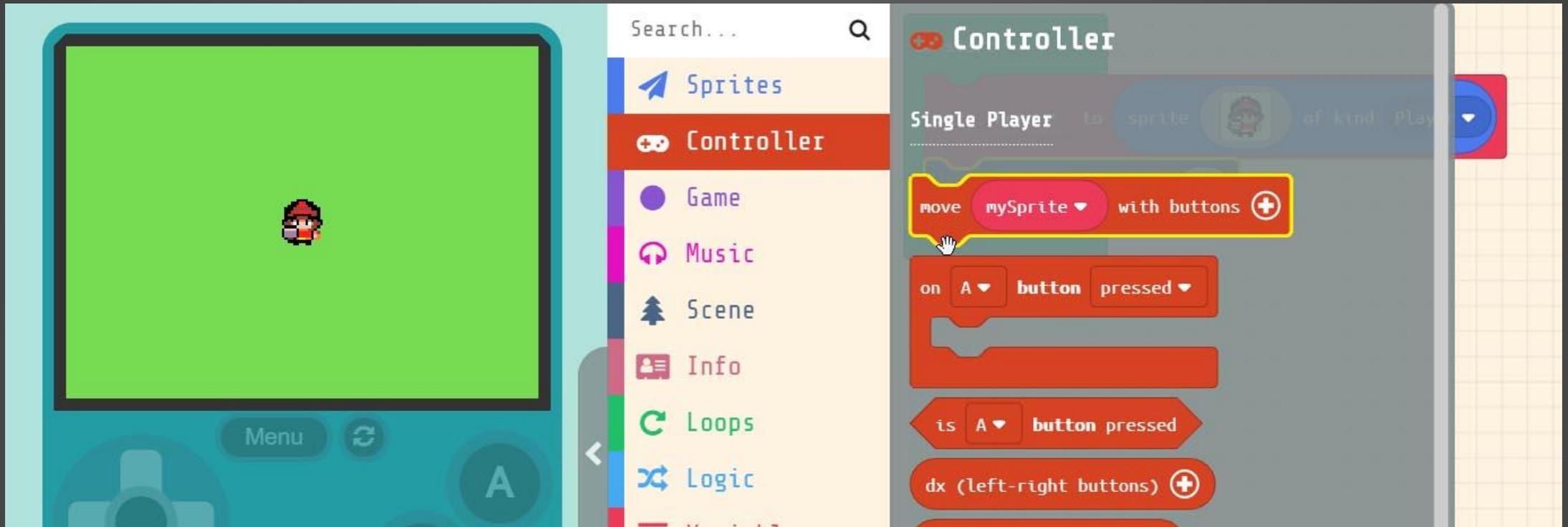


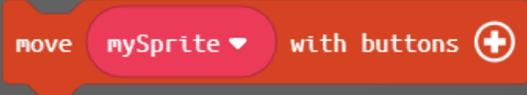
Grab the  Block

and drag it into the  block just below our 'Hero'

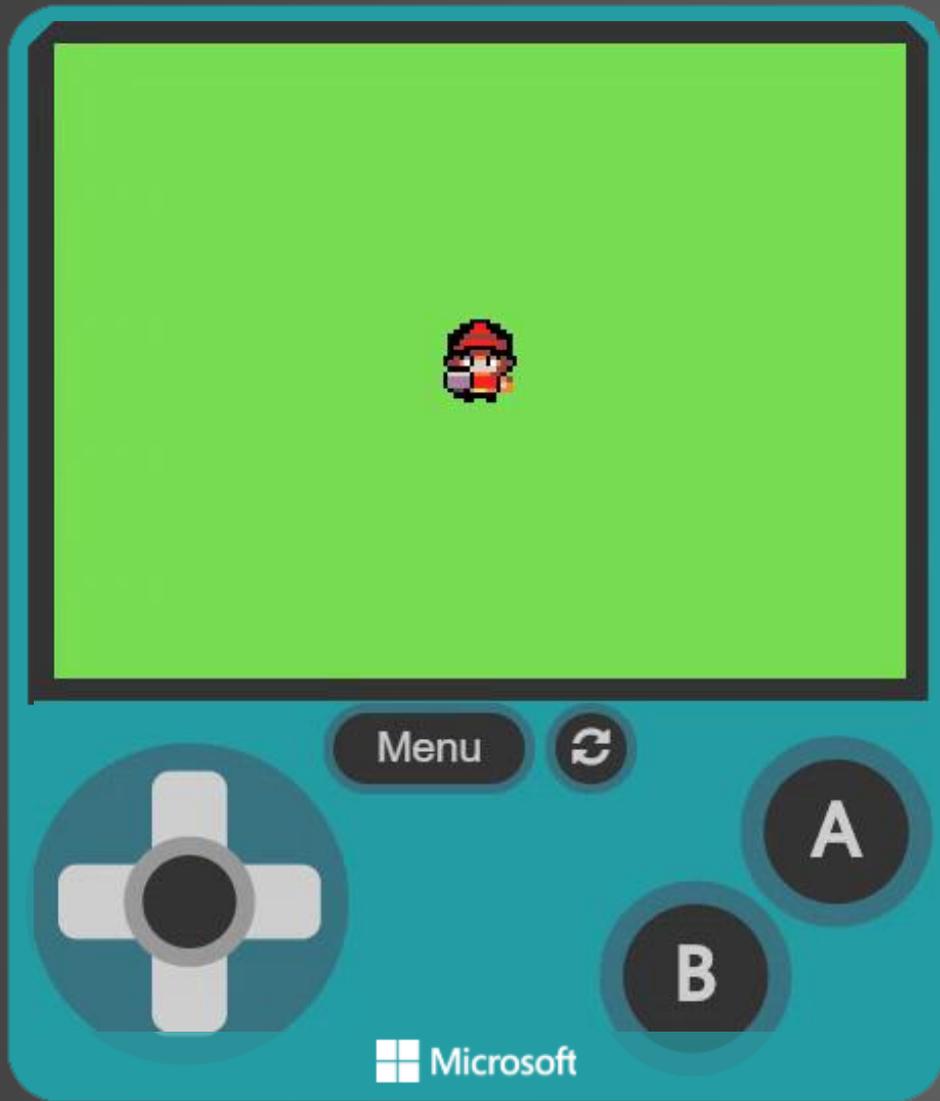


Click inside the GREY circle
and select the color GREEN



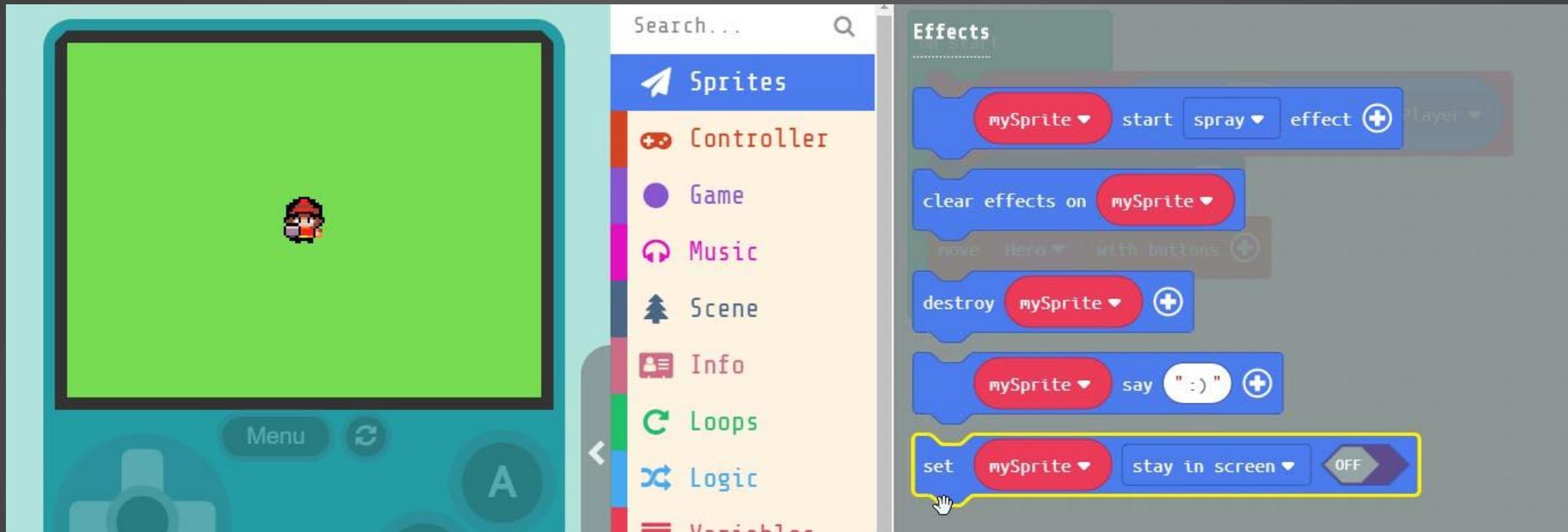
Grab the  block

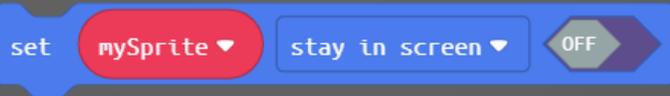
Drag it into the  block just below the other blocks

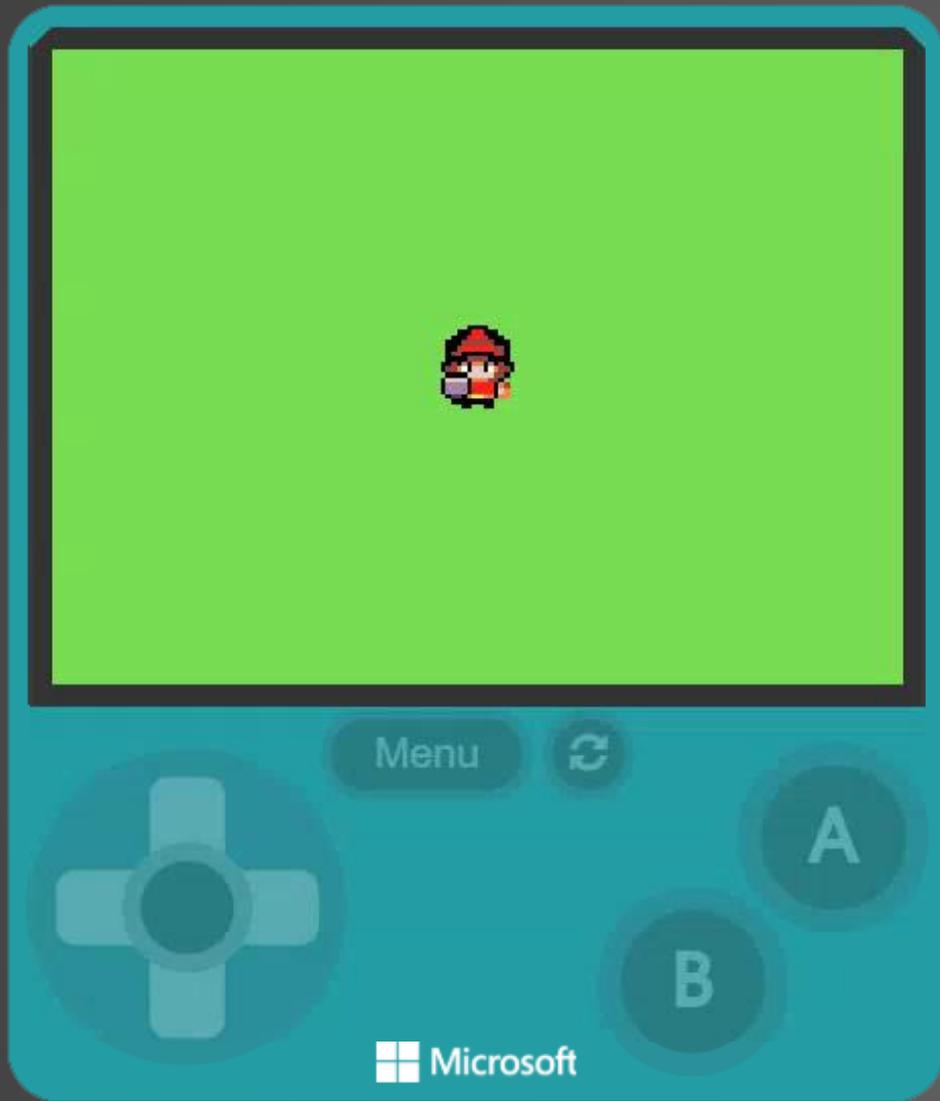


```
on start
  set Hero ▼ to sprite  of kind Player ▼
  set background color to 
  move Hero ▼ with buttons 
```

Change the name to 'Hero'
Now we can move our Sprite
around the screen
You'll notice the 'Hero' leaves
the screen

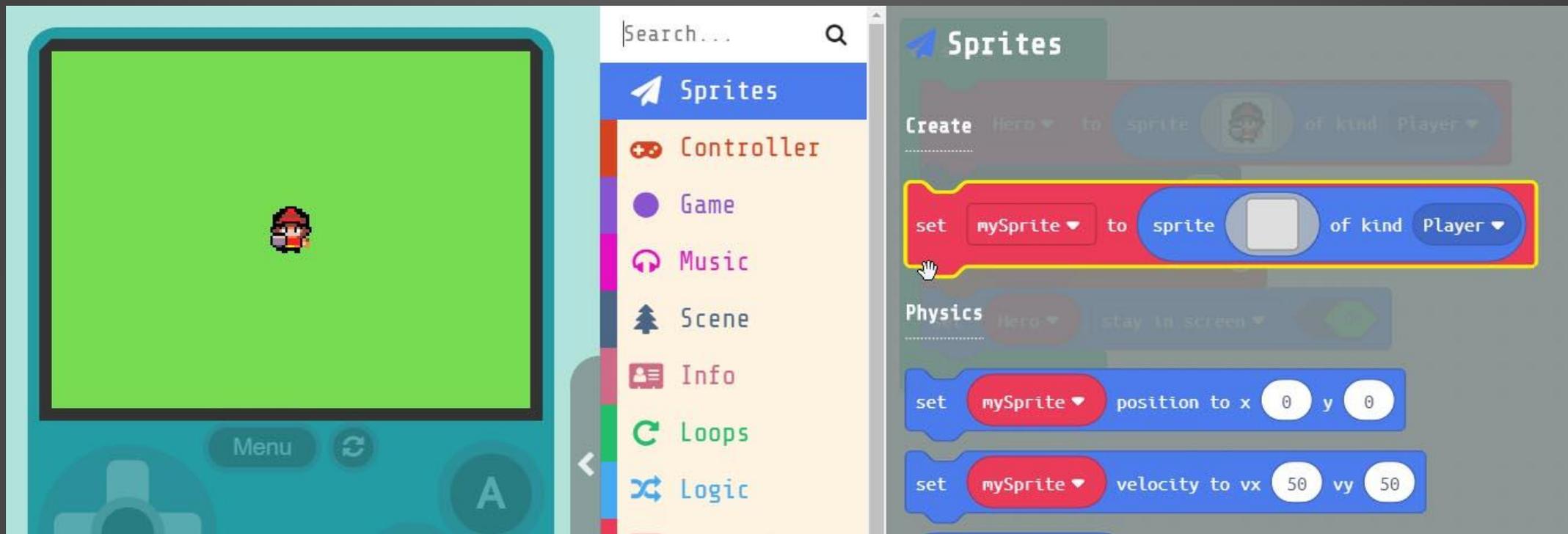


Grab the  Block
drag it into the  block just below the other blocks



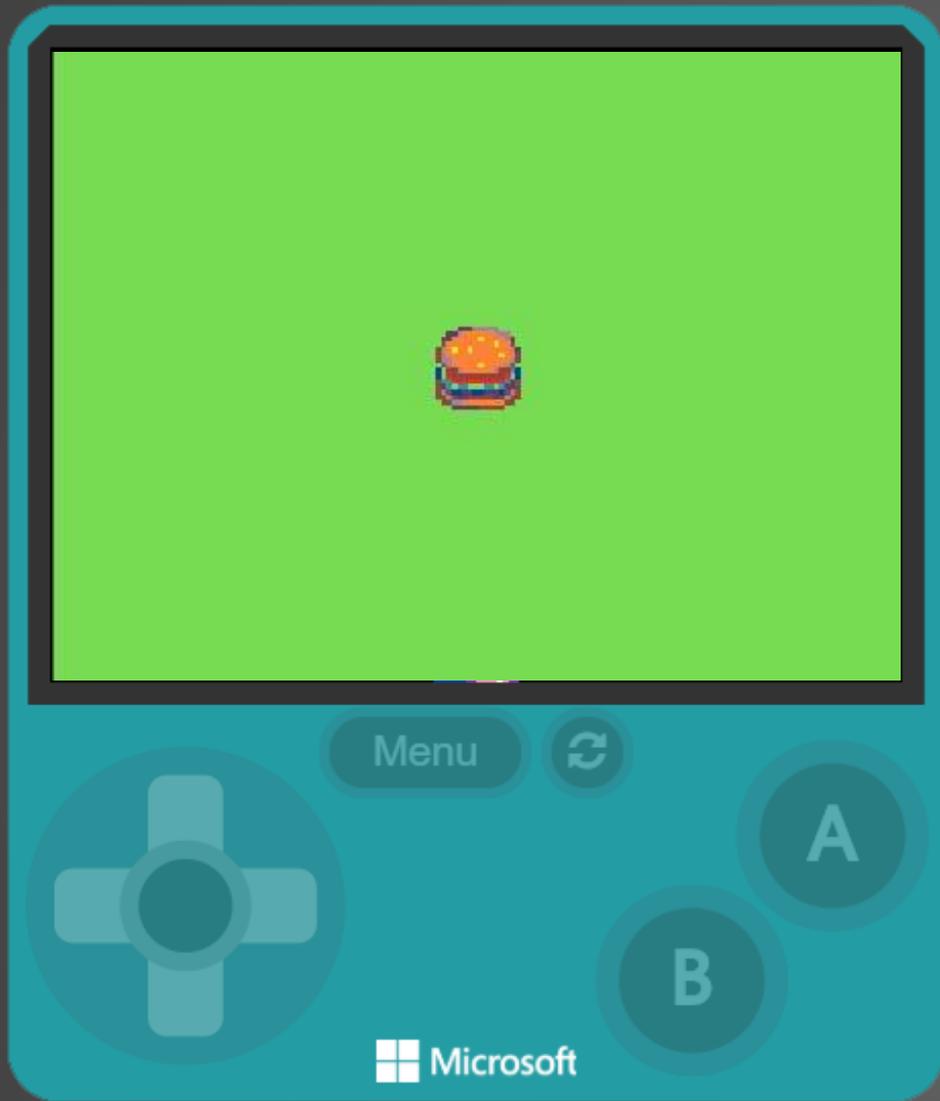
```
on start
  set Hero to sprite [Mario] of kind Player
  set background color to green
  move Hero with buttons
  set Hero stay in screen ON
```

Change the name to 'Hero'
and set the block to 'ON'
Now our 'Hero' stays on screen



Grab the  Block

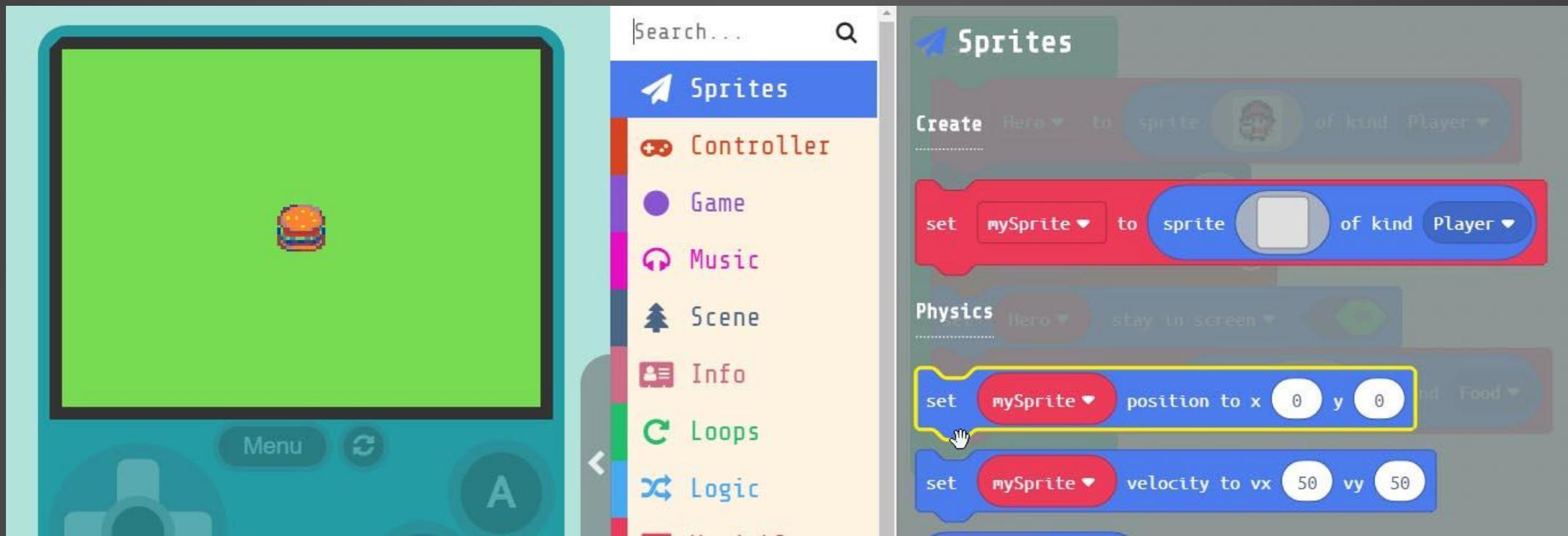
drag it into the  block just below the other blocks

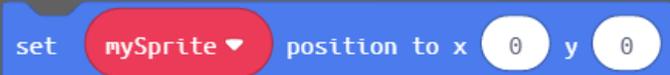


```
on start
  set Hero to sprite [Mario] of kind Player
  set background color to green
  move Hero with buttons
  set Hero stay in screen ON
  set Hamburger to sprite [Hamburger] of kind Food
```

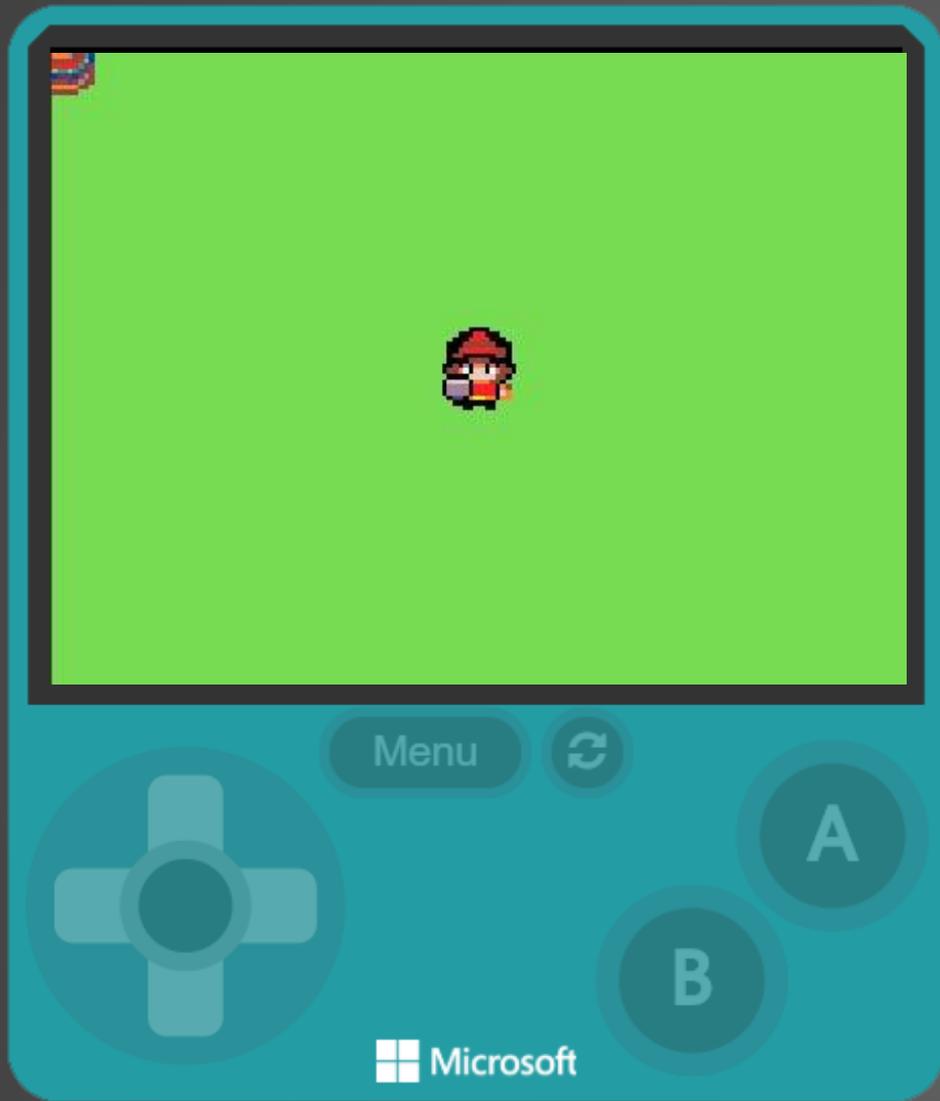
Change the name to 'Hamburger'
change the 'kind' to 'Food'

By default our Hamburger Sprite
appears in the center of the screen



Grab the  block

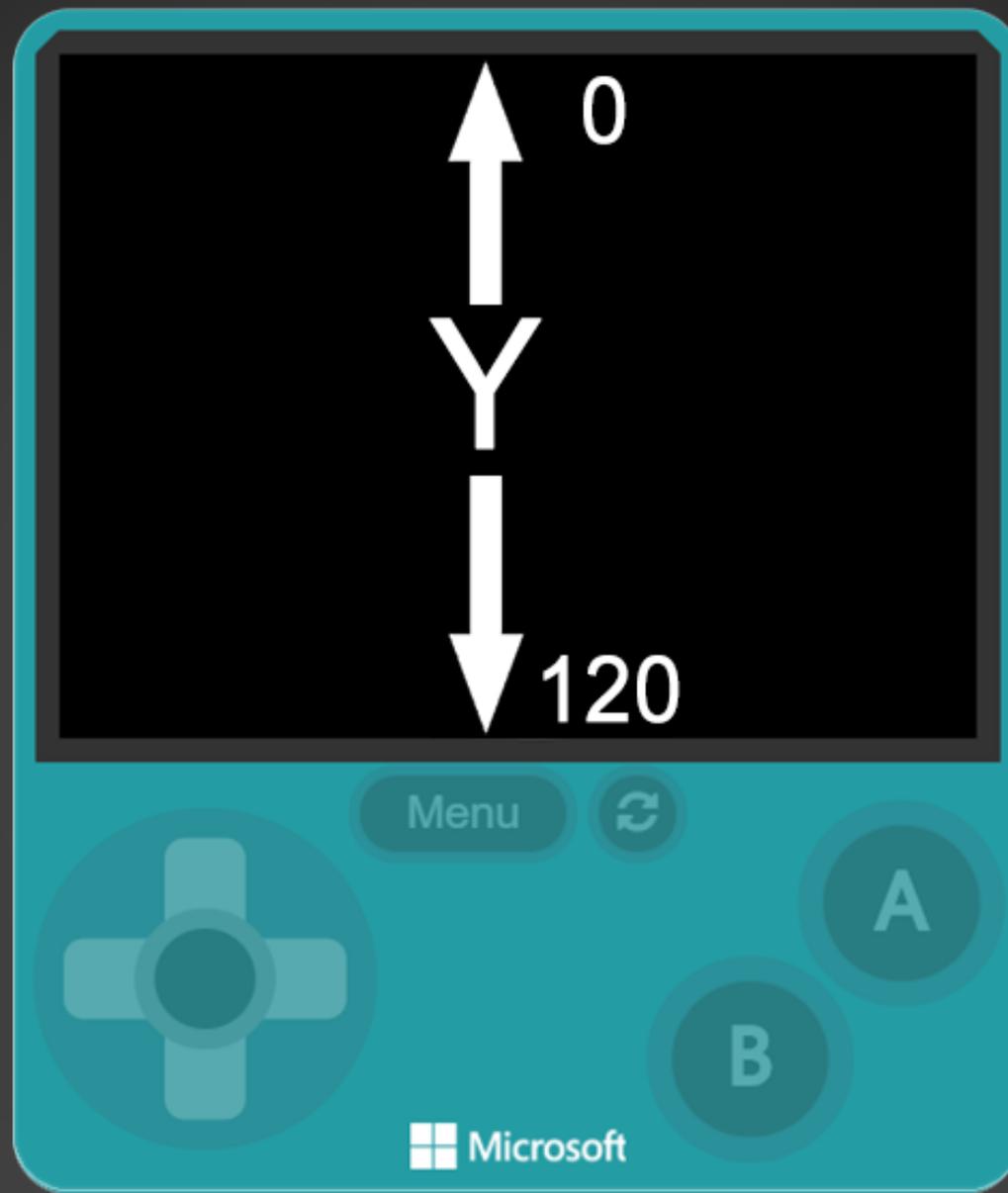
drag it into the  block just below the other blocks



```
on start
  set Hero to sprite [Mario] of kind Player
  set background color to [green]
  move Hero with buttons
  set Hero stay in screen ON
  set Hamburger to sprite [Hamburger] of kind Food
  set Hamburger position to x 0 y 0
```

Change the name to 'Hamburger'
change the 'kind' to 'Food'
By default our Hamburger Sprite
now appears in the top corner

The x & y parameters represents pixels on the screen





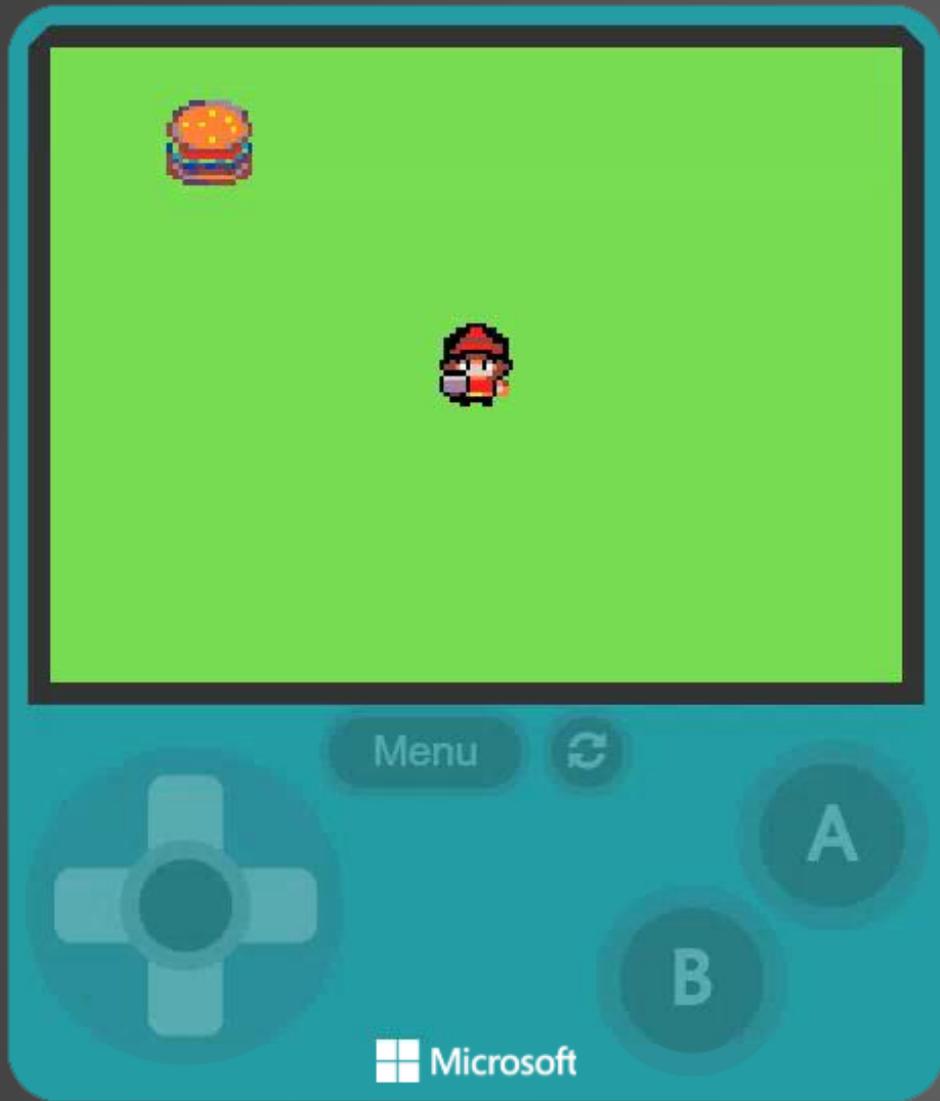
Computer Term:

PARAMETER

A parameter is a value that we add inside a block. This number is passed into the block. In the example block below '0' would be the parameter. Parameters can also be referred to as 'ARGUMENTS'

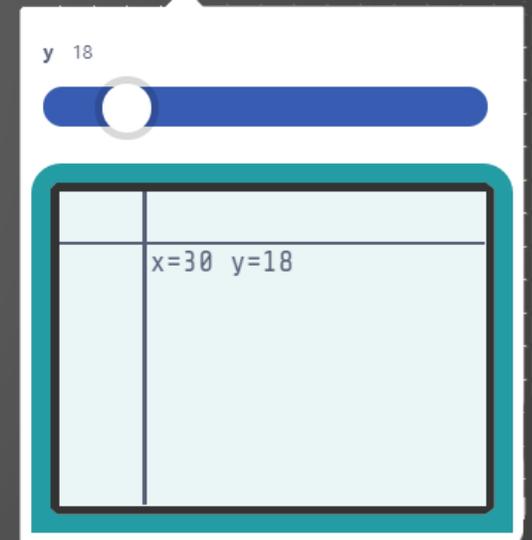
EXAMPLE:





```
on start
  set Hero to sprite [Mario] of kind Player
  set background color to green
  move Hero with buttons
  set Hero stay in screen ON
  set Hamburger to sprite [Hamburger] of kind Food
  set Hamburger position to x 30 y 18
```

Click on either the x or y parameter and set the position of the 'Hamburger'





Computer Term:

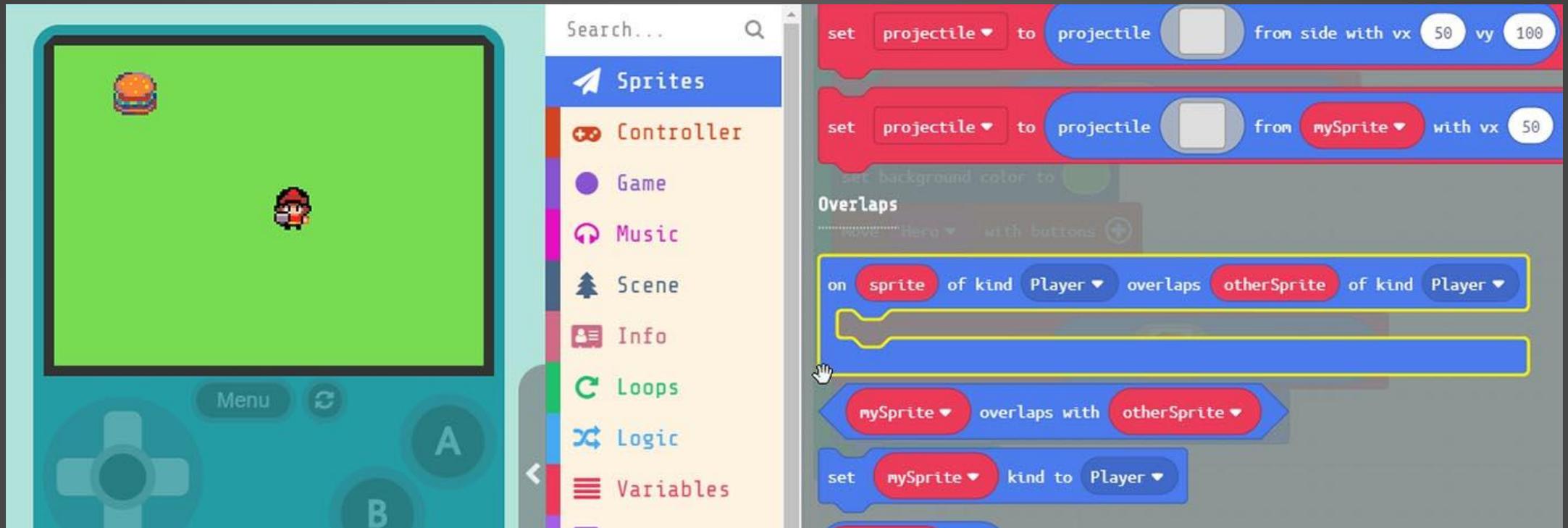
EVENT

An event is an action or occurrence detected by a program. Events can be a user action like clicking a button or when Sprites overlap.

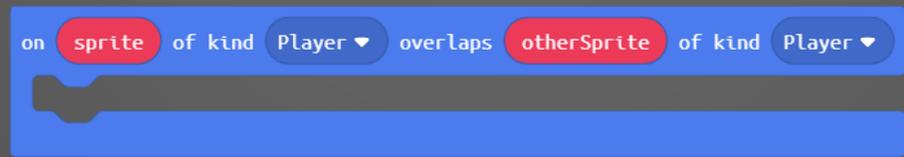
EXAMPLES:

on **sprite** of kind **Player** overlaps **otherSprite** of kind **Player**

on start

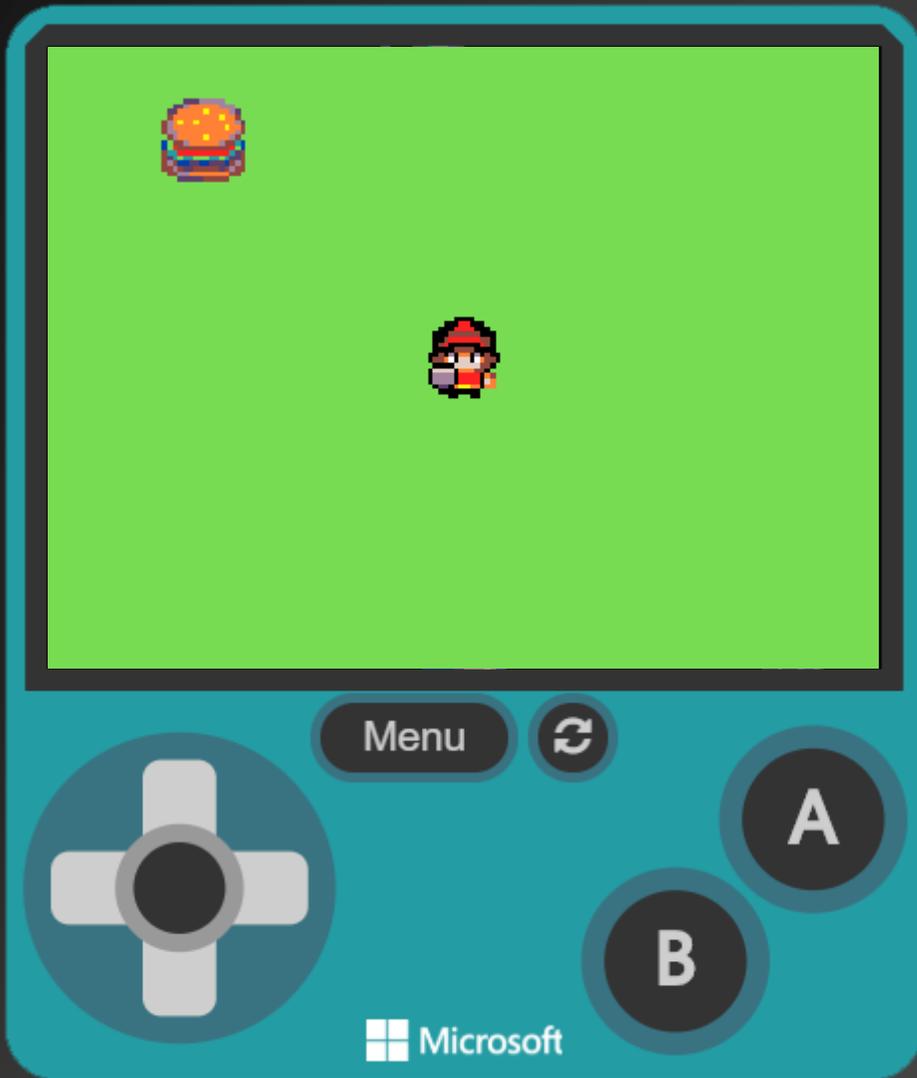


Grab the



Event block

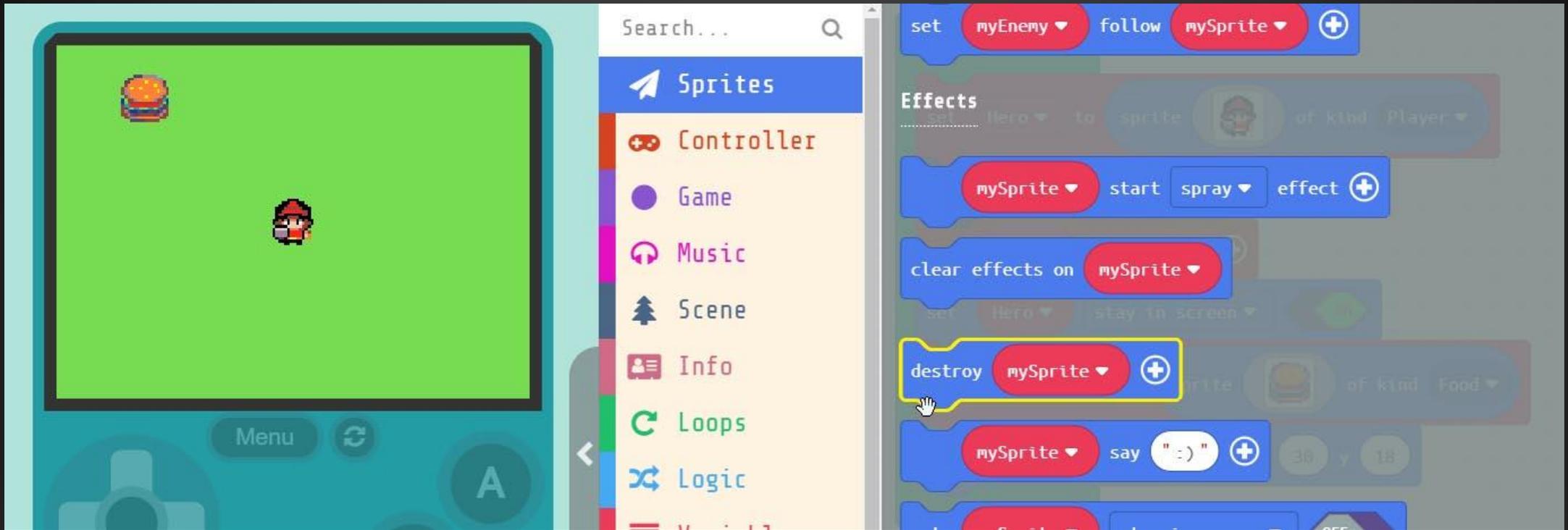
drag it into workspace.



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

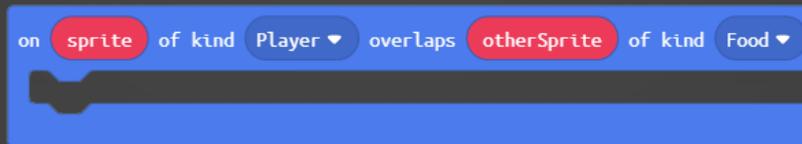
Now let's change the parameters inside our 'overlap' block.

We can leave the first kind to 'Player' but change the last kind to 'Food'



Grab the  block.

Put inside our



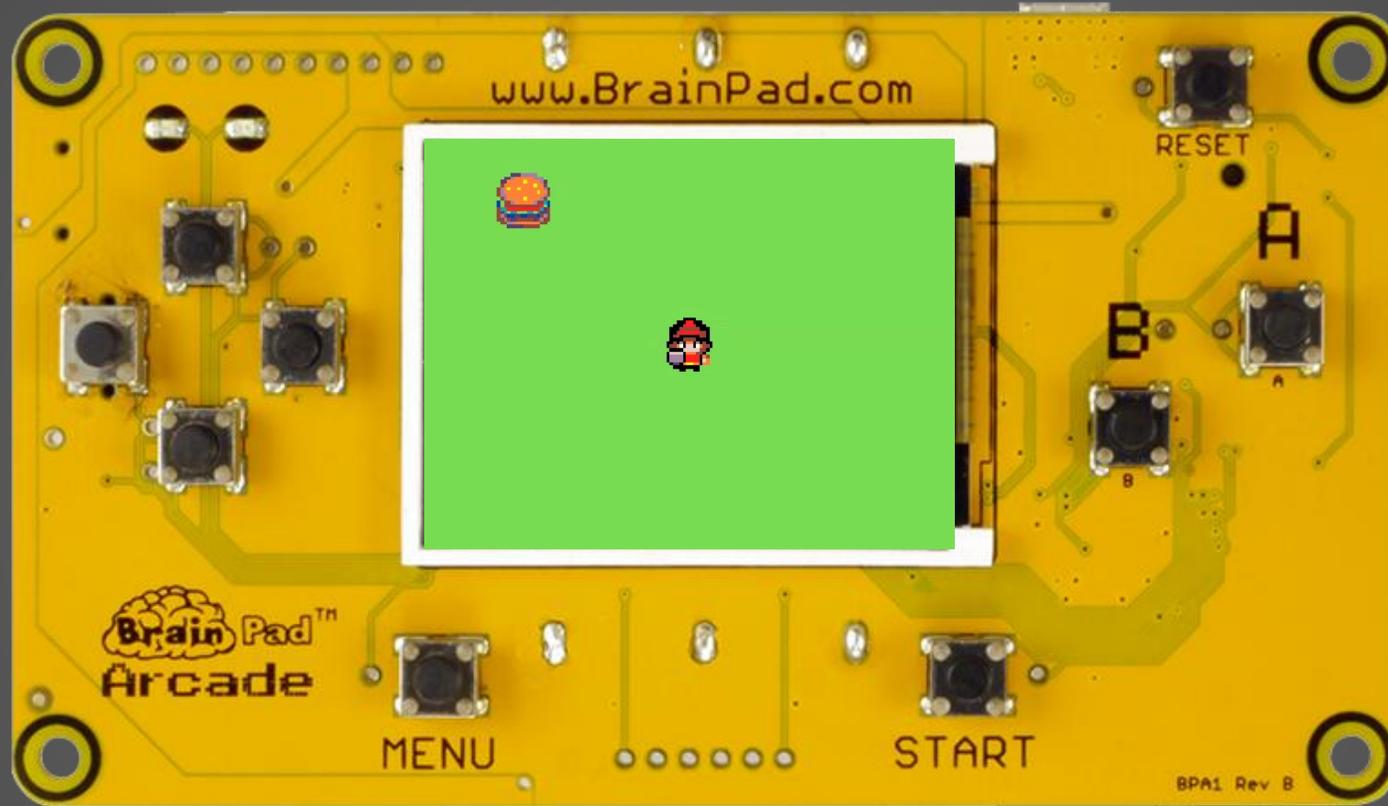
Event block.



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

We need to create a 'NEW' variable called 'otherSprite'
This ensures that every time our 'Hero' overlaps the 'otherSprite' that is of kind 'Food' we eat it.

Hardware Break



Now let's load what we have on to the BrainPad.

EXTRA CREDIT





Code to Blocks:

JavaScript:

```
let Hero = sprites.create(img`...`  
    , SpriteKind.Player)
```

Block:

The image shows a Scratch block diagram. It features a green 'on start' block containing a red 'set' block. The 'set' block is configured to set the variable 'Hero' to a 'sprite' of kind 'Player'. The 'sprite' field is populated with a pixel art character icon, and the 'kind' dropdown menu is set to 'Player'.



Code to Blocks:

JavaScript:

```
controller.moveSprite(Hero)
```

Block:





Code to Blocks:

JavaScript:

```
sprites.onOverlap(SpriteKind.Player, SpriteKind.Food, function (sprite, otherSprite) {  
    otherSprite.destroy()  
})
```

Block:

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

destroy **otherSprite**

The block is a blue horizontal bar. The top part contains the text 'on' followed by a red pill-shaped button with 'sprite', 'of kind', a blue pill-shaped button with 'Player' and a dropdown arrow, 'overlaps', another red pill-shaped button with 'otherSprite', 'of kind', and a final blue pill-shaped button with 'Food' and a dropdown arrow. Below this, there is a blue block with a notch on its top edge, containing the text 'destroy' followed by a red pill-shaped button with 'otherSprite' and a dropdown arrow, and a white plus sign in a blue circle. The bottom part of the block is a solid blue bar.