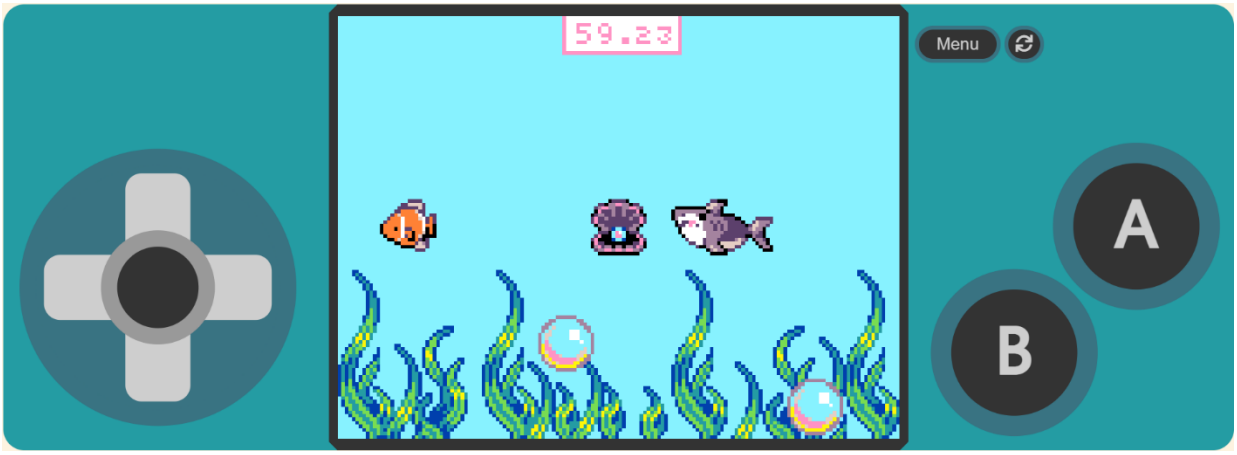
Second exercise Sprites



Description

In this project we work the variable concept, the creation of an opponent, image change, and the sprite´s animation.

A sprite is a graphic element that has been designed in a bit map. We work with that in that task. Different attributes will be assigned to this element such as position, velocity, acceleration...

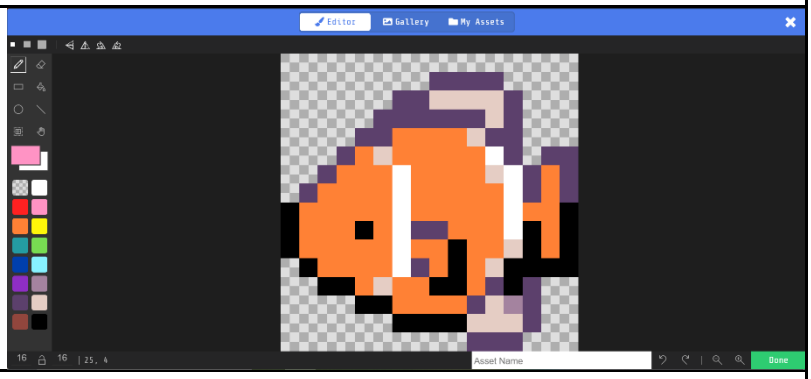
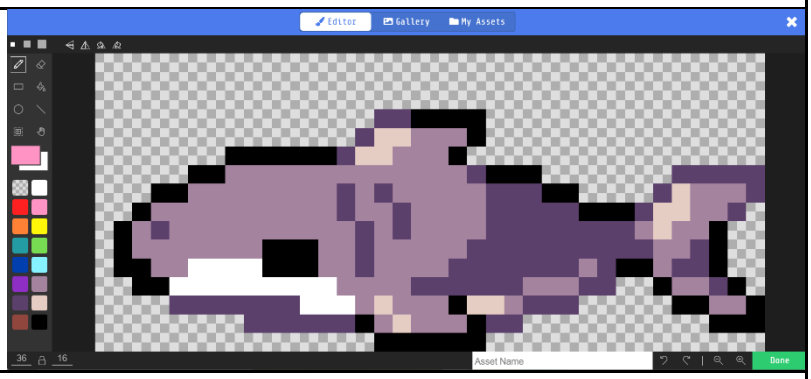
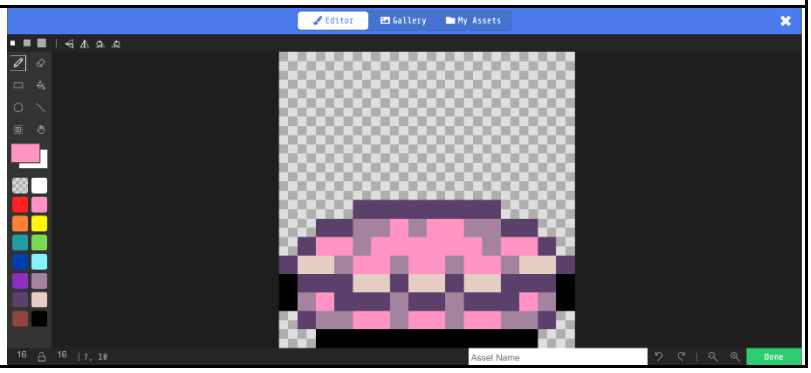
The game is about a fish which tries to get some pearls that appear inside the oyster, but the shark will not let the fish take it that easy because he will follow him.

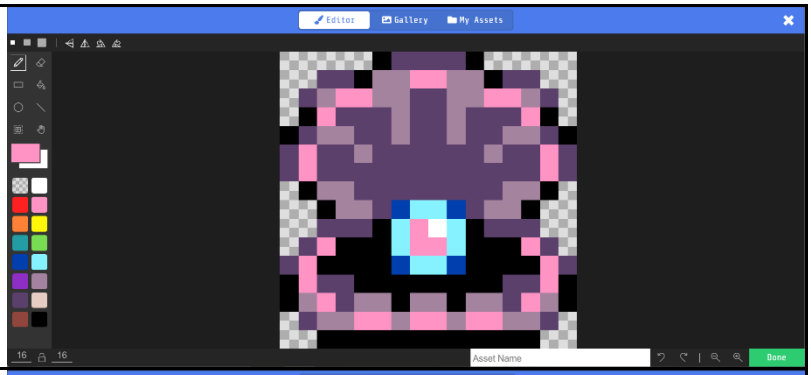
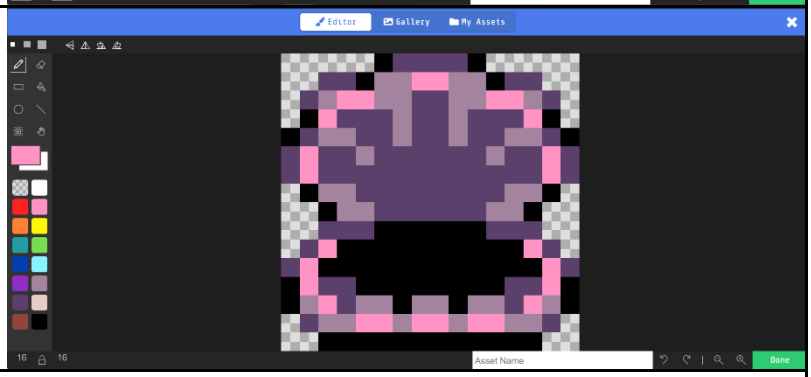
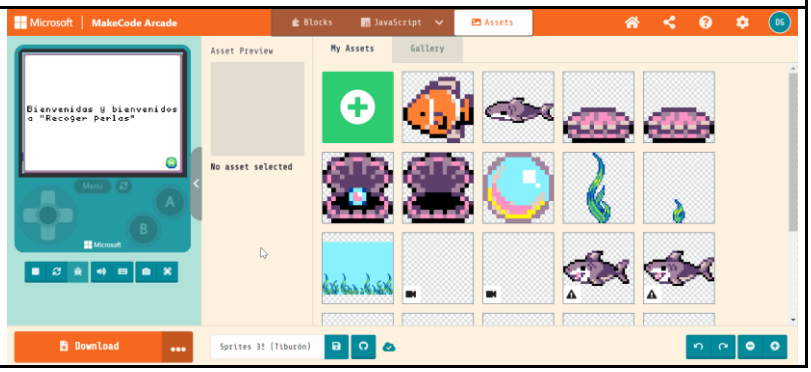
We will use **MakeCode Arcade** to create the game.

Programming and designing goals.

- Create a ¨player¨ sprite and control its movement.
- Generate a ¨food¨ sprite that allow us to increase the score.
- Program the interaction between the sprites.

Programming the game.

ASSETS CREATION	
MAIN CHARACTER SPRITE CREATION	
We highly recommend the creation of a 16 x 16 matrix to set the Sprite of kind Protagonist.	
ENEMY SPRITE CREATION	
We highly recommend the creation of a 36 x 16 matrix to set the Sprite of kind Shark	
ADDITIONAL SPRITE CREATION	
We highly recommend the creation of a 16 x 16 matrix to set the Sprite of kind Oyster.	

We highly recommend the creation of a 16 x 16 matrix to set the Sprite of kind Oyster.	
We highly recommend the creation of a 16 x 16 matrix to set the Sprite of kind Oyster.	
ANIMATION CREATION	
In assets we click on ¨+¨. Here we press gallery and choose our shark´s animation. Lastly, we name it and click on ¨Done¨	

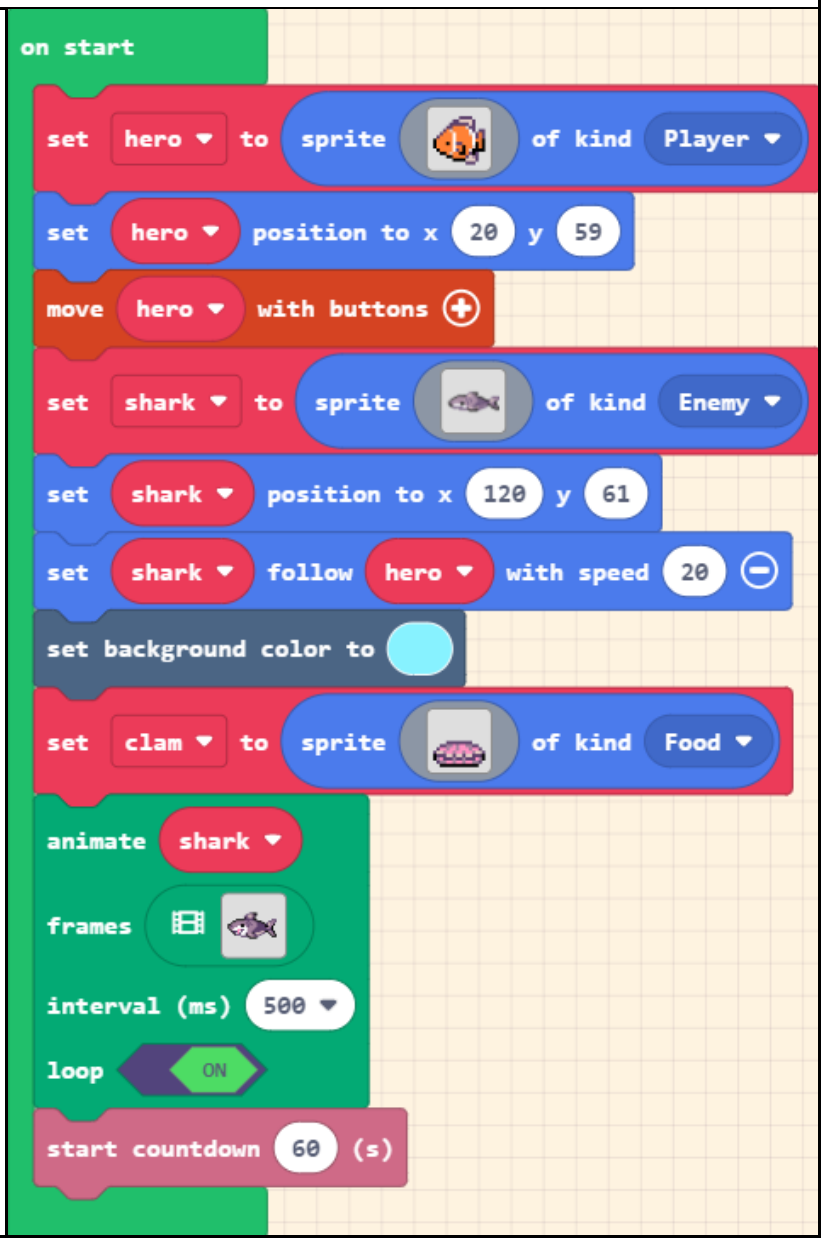
MAIN PROGRAMMING

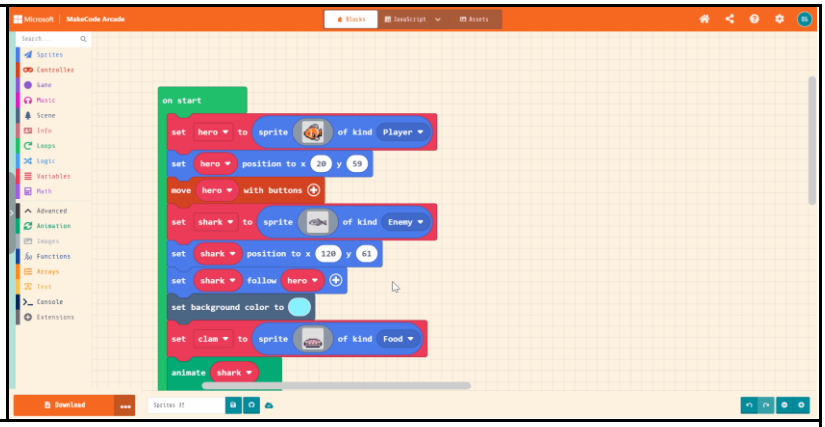
ON START GAME CREATION

We start with different sprites'creation that appear in the task and its position in the scene. We give control to our ¨player¨ sprite, a fish in this case. The ¨enemy¨ sprite will follow the fish adding the ¨ my Enemy follow my sprite ¨. We can click on ¨+¨ to decrease velocity, for example to 20 so the game is not impossible for the player. Besides we create the sprite of kind ¨Food¨

We add colour to the background

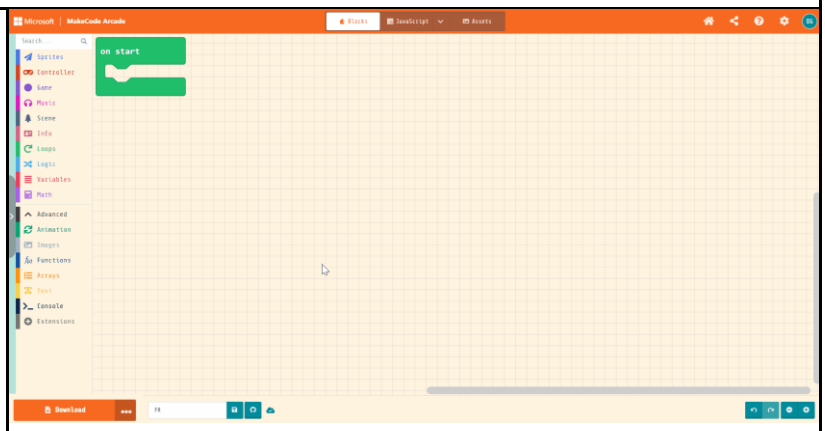
We will use the ¨ animate ¨ block from ¨ Animation ¨ to create an animation. It is important to leave activated the ¨loop ¨ button because once we finish the animation it repeats the movement again.





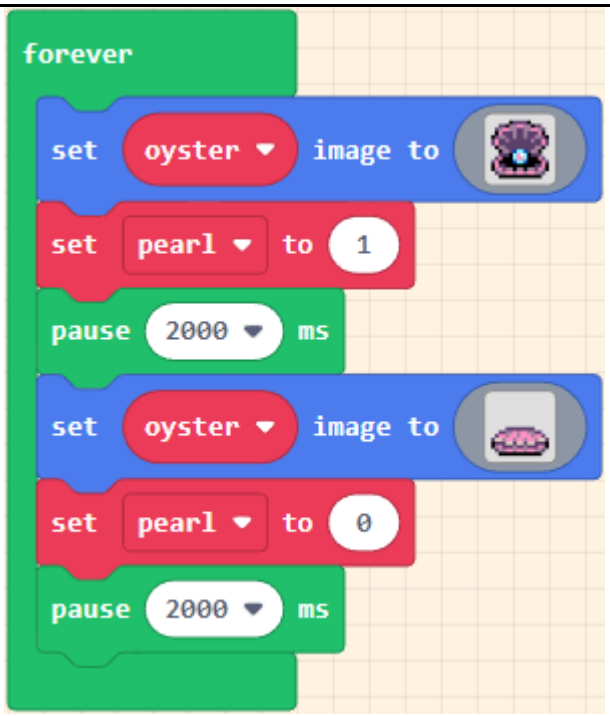
OPENING THE OYSTER MECHANIC CREATION

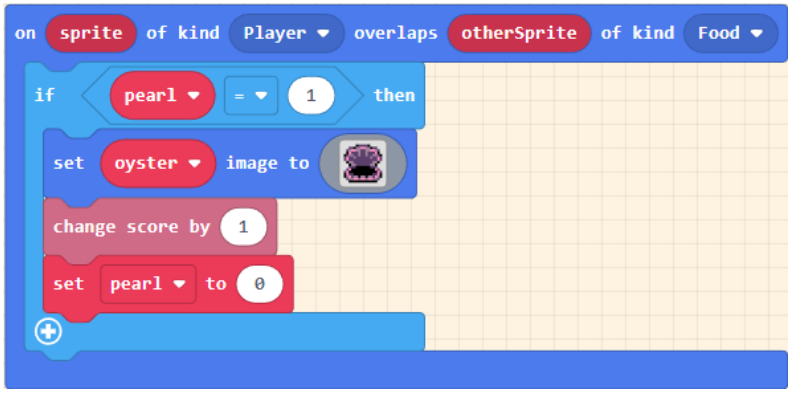
Firstly, we create a **variable** named ¨**pearl**”. In order to do so we click on ¨variable¨ and then we click again on make a variable.



Inside a ¨ **forever** ¨ block, we will introduce the blocks to change the image, we stablish the value of the variable ¨ **pearl**¨ to 1 when it appears and to 0 when it disappears. It is like an abstraction when having the pearl or not.

It is important placing the pause after the image change and the variable´s set so the player has time to get points.



So, in this group of blocks we clearly see when there is a pearl, we indicate it in the game as well, it has a 2 second difference in each status.	**pearl = 1 -> La perla está y se puede atrapar.** **pearl = 0 -> La perla no está y no se puede atrapar.**
GETTING PEARL AND POINTS MECHANIC CREATION	
When the fish touches the oyster, if there is a pearl inside, it takes it, changing the image and increasing the score setting that there is no pearl. If there is no pearl when touching the oyster, nothing happens.	
FINAL GAME MECHANIC CREATION	
When the shark overlaps the fish, the game is over adding the block ¨game over¨	

With that programming we use the sprite of kind ¨player¨ sprite to get points when touching the sprite of kind ¨food¨ while these spawn on the screen with a 1-minute timer. Now it is your time to give it a personal touch. Here you have the link to check our game https://makecode.com/_AHWED8PpjPFs.

Glossary

Sprite: It is a graphic element designed in a bit map. We can apply different attributes such as position, velocity, acceleration...

Bit map: It is a pixel grid we use to design sprites.

Scene: Space where the videogame takes place.

Player: Participant in the game.

Randomness: Generation of numbers with the same probability of appearance.

Count down: Time we set in order to create a situation when it runs out.

Variable: It is a store associated to an Id, inside the store there is a value that can be modified.

Animation: The animation of the sprite is the sensation of movement due to the different frames.

Loop: sequence of instruction s that is continually repeated in a determinate or indeterminate way.

Example: Forever, while, for.

Conditionals: Sequence of instructions that execute depending on the value of a condition.

Example: If, If…Else

If: Conditional statement that depending on the result of a logical operation it executes a sequence of instructions, or it omits.