

# Coral & Seawater Acidity Model

A model of reactions to changes in the acidity of the sea

[The numbers and calculations involved are not representative of the real sea!!! Or of how the PH scale works]

Find the Starter Code here: <https://scratch.mit.edu/projects/365818132/editor/>

## STEP 0

We start this project with three sprites:

Paddle, Line, Fish

These will help us model a relationship between the acidity of the sea and the wellbeing of creatures who live in it.

## STEP 1

**Draw** a new Coral sprite.

It's important that you draw the coral using **just one colour**.

Coral is mostly a medium brown colour in real life.

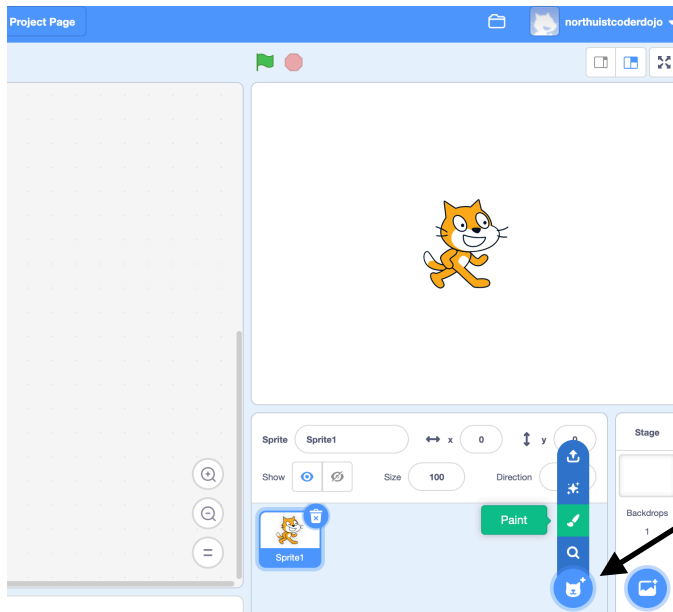
Position your Coral sprite on the seabed so that it looks like it lives there.

If you need help with how to draw a sprite there are instructions on the next page.

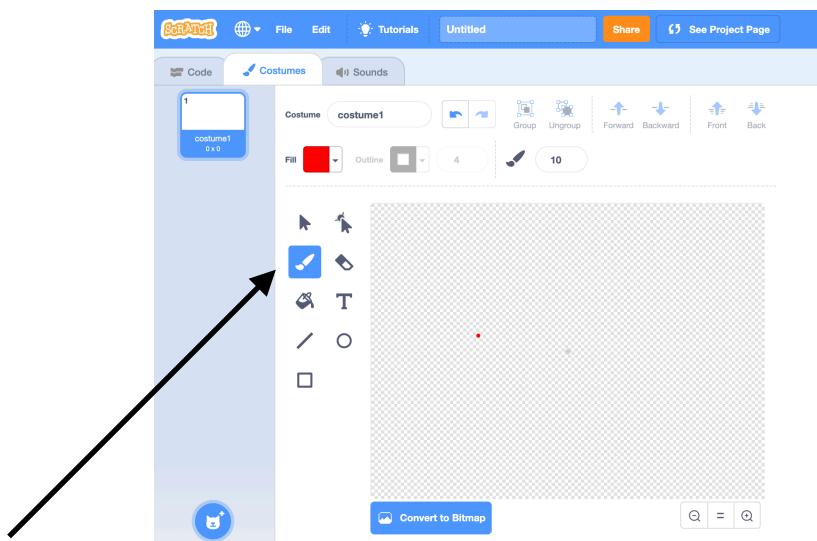
Otherwise, skip to **STEP 2**

How to draw a sprite in Scratch:

Choose the 'Paint' option



Use the brush tool (highlighted blue below):



## Step 2: Adding Code to the Coral Sprite

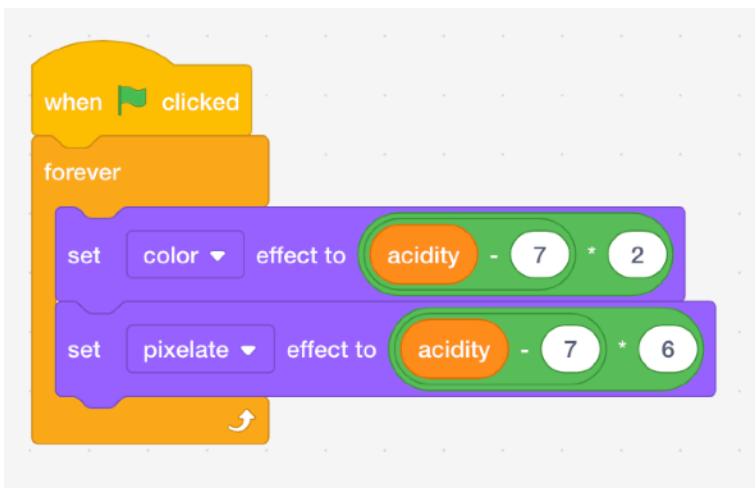
The acidity variable goes in the left side of a green 'minus' block. And that goes inside the left side of a \* or 'multiplier' block.



### Operator Blocks:

In coding we use these symbols for maths:

- + for adding
- for subtracting
- \* for multiplying
- / for dividing



The purple '**set colour**' block will change the colour of the Coral sprite according to whatever value is passed in.

The purple '**set pixelate**' block will change how broken the coral looks. We'll see what that means later on.

You can see we're using a variable called '**acidity**' here. We'll learn more about that that is doing later in the program.

For now we need to know that the value stored in '**acidity**' will change the colour and pixellation of the sprite.

The green blocks are doing some maths to the acidity value so it makes things look better.

### STEP 3 - The Fish

Look in the Fish Sprite. You will see this code:

```
when green flag clicked
  forever loop
    if costume name = right then
      glide pick random 1 to 3 secs to x: pick random 0 to 220 y: pick random -100 to 100
      wait pick random 1 to 3 seconds
      switch costume to left
    if costume name = left then
      glide pick random 1 to 3 secs to x: pick random -220 to 0 y: pick random -100 to 100
      wait pick random 1 to 3 seconds
      switch costume to right
```

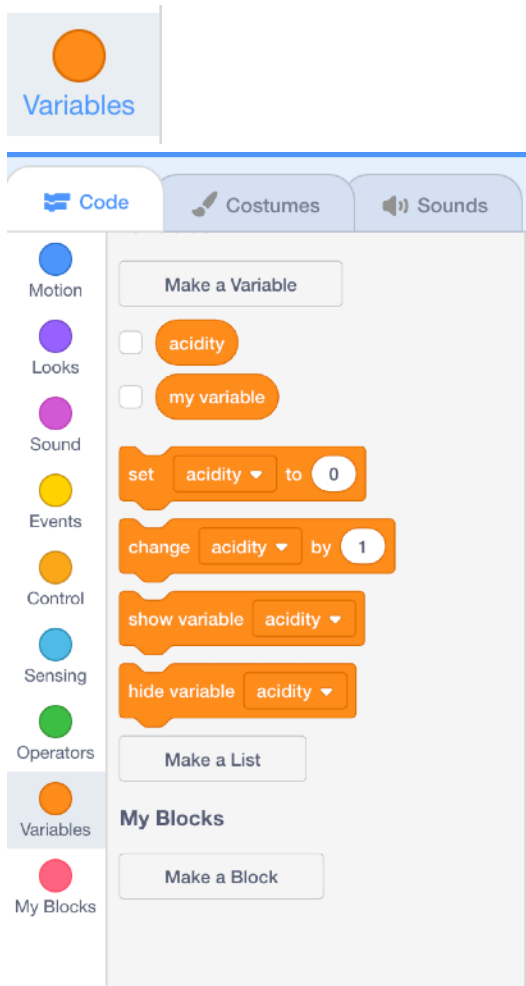
You don't need to do anything with it! Just read and see if you can figure out what it does.

Click the green flag to run it.

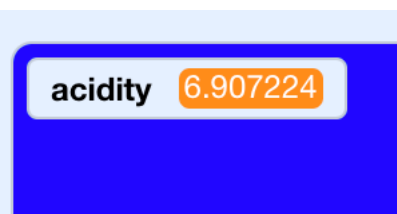
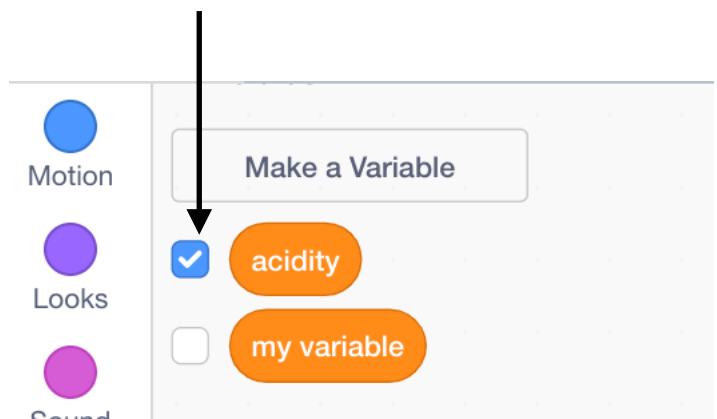
## STEP 4 - Using the acidity Variable

The slider sprites on the right side of the screen control a variable called 'acidity'

Go to the Variables tab in the menu



Click the little box next to 'acidity' to make the variable show on screen



Now you can see the value stored in 'acidity'.

When the program starts it's set to 7.00056  
This is because of the maths going on in the background.

The program calculates the acidity value based on the **Y position** of the slider. If you slide it up the acidity goes up (to a maximum of 14) and if you slide it down, the acidity goes down to 0.

## OPTIONAL EXTRA

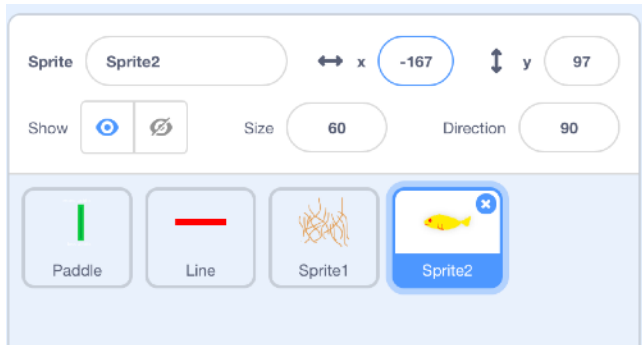


If you want to round the number down to a whole number or **integer** we can add this code to the red slider sprite:

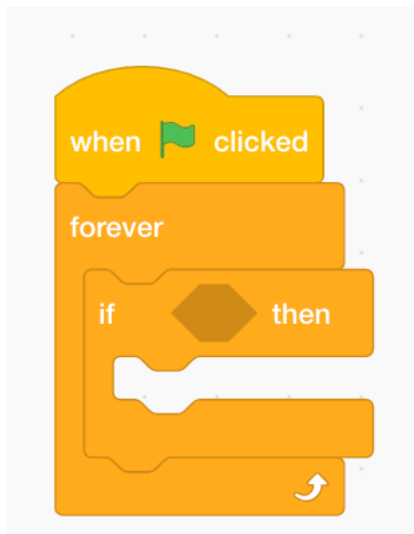
## STEP 5 - Adding Code to the Fish

We're going to add code to the fish so it responds to the acidity value.

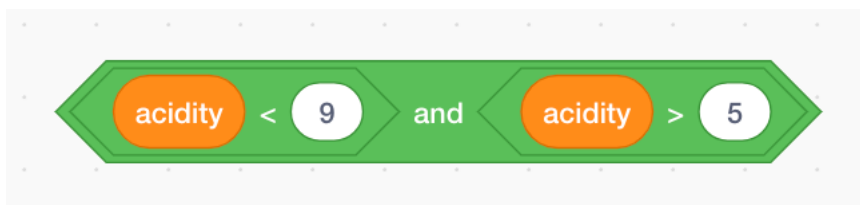
In the Fish Sprite (You have to select the Fish sprite so it shows blue)



First, add these blocks to the fish



Next create a set of blocks that will check a **condition**.



There are 3 green blocks here.

An **'and'** block

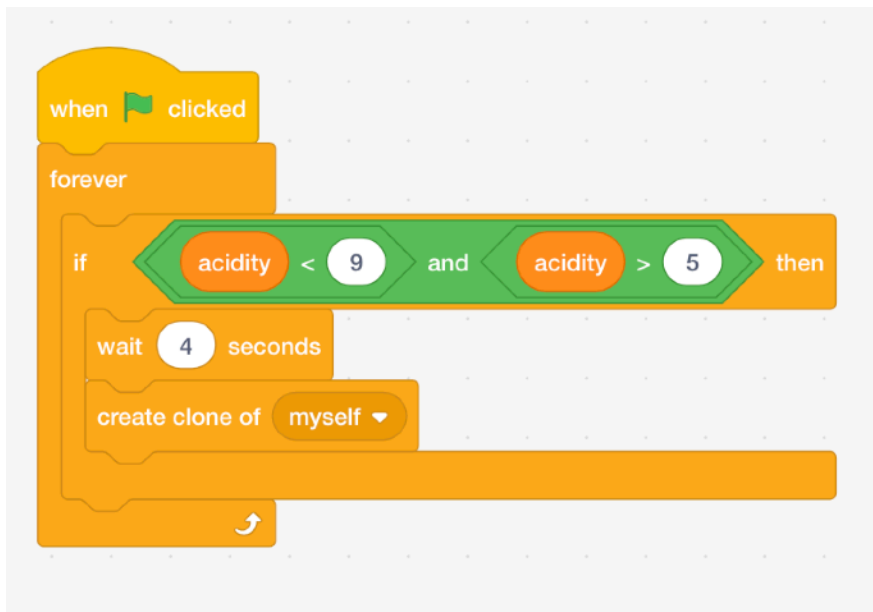
A < block

A > block

This combination of blocks has 2 possible outputs or 'effects' : True or False.

While **acidity** is **greater than 5 but less than 9**

Add the green operator blocks to the if block.



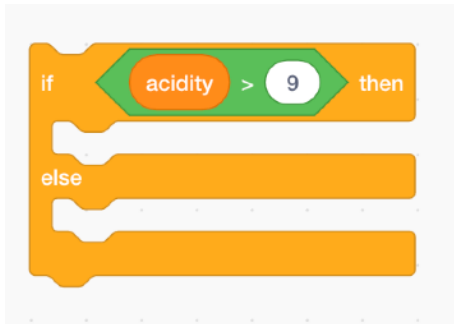
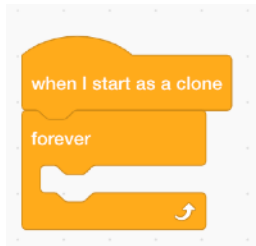
Inside the if block, add a **'wait 4 seconds'** block  
And a **'create clone of myself'** block.

This means:

If the acidity is not too high or too low; the fish species will be happy and will produce more of itself.

Next we need to write some code for what happens when a new fish appears.

## STEP 6 - Adding code to the clone



THINK:

We know that if acidity is greater than 5 **and** less than 9, the fish is happy.

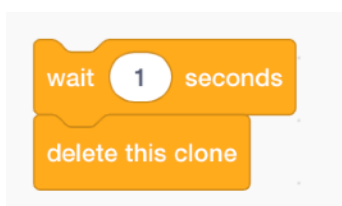
This block means:

If acidity level is greater than 9, do something

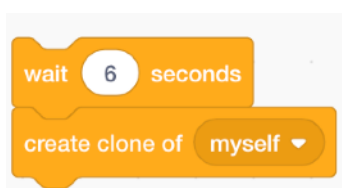
And if acidity level is **not** greater than 9, do something else.



Put the 'if else' block inside the 'forever' block.



This block will delete a clone, as if it has died from too much acidity..



This block will produce another fish as though the fish are still happy.

**Work out which one goes in which part of the 'if. Else' block before going to the next the page to check.**



The sequence of blocks looks like this. **Check yours is the same** before carrying on.



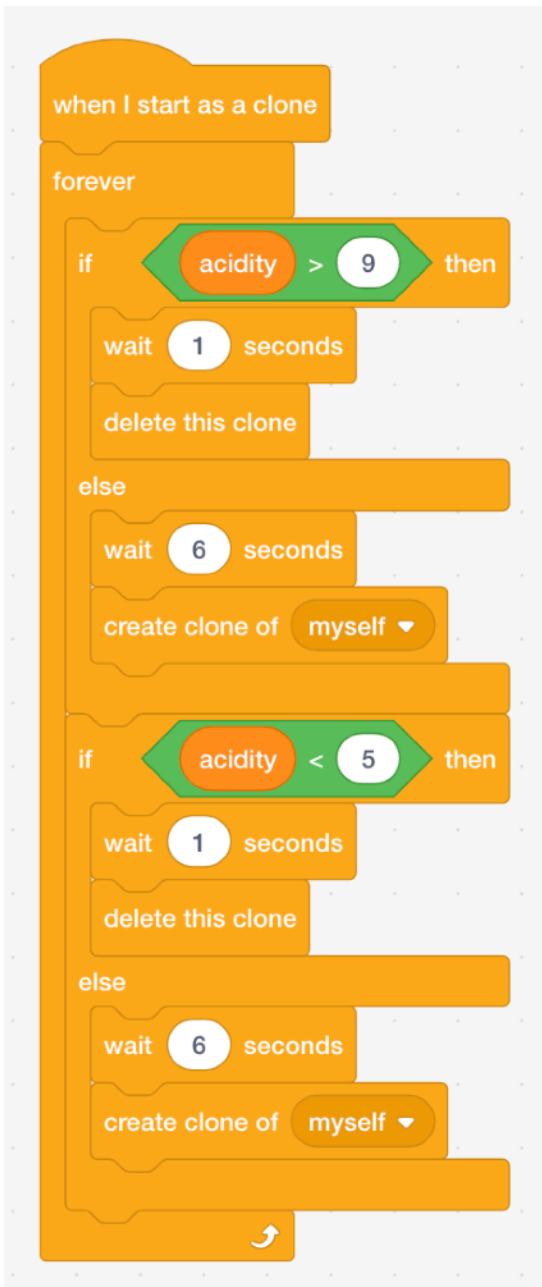
We now have code that checks if the acidity is greater than 9. Remember the other side of the condition?

We need to check when it goes below 5.



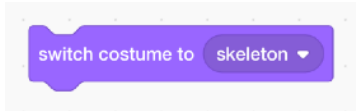
The finished sequence looks like this.

Make sure the 'if else' blocks are stacked one above the other and not nested inside each other.



That's it. Run the program and try out different acidity levels to see how the fish and the coral respond.

## Extensions



1. Add a couple died

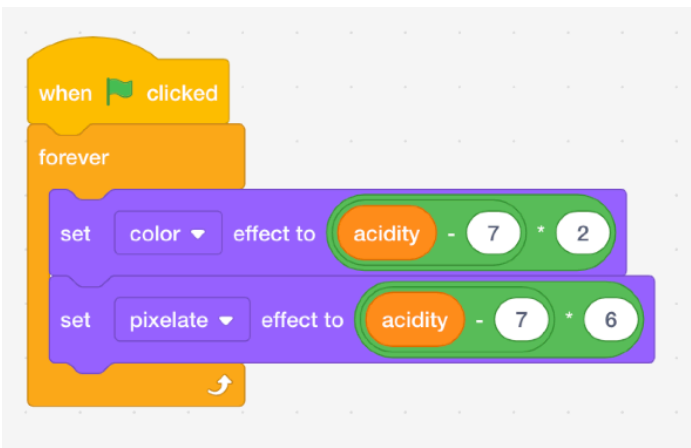
of blocks to the code to show when a fish has

2. Can you add code to keep a count of the number of clones / fish?

(hint : look for the code block that creates a new fish)

3. Experiment with the values in the code blocks to change the way the fish moves. Can you figure out a way to use the **acidity** variable to tell the fish how fast to move, depending on the acidity?

4. Play around with the operator blocks in the coral sprite



You might be able to find values that create more dramatic changes to the colour or pixelation of your sprite!

////////////////////////////////////