Before you get started

The program that you are about to use is a free, visual programming language called Scratch, from the folks at MIT.

You can download it from the Scratch web page:

http://scratch.mit.edu

You can upload your creations here, and view the many creations already there. It’s a great way to learn new tricks and see some cool stuff!

How does Scratch work?

It’s very simple! First, start Scratch. On the left-hand side you have lots of colourful tiles. These tiles can be dragged into the middle and snapped together to form scripts. Think of a script as a series of steps, where each tile represents one step.

On the upper right-hand side you have the stage. This is where your scripts come to life! And below that, you have your sprite window. For example, when you start Scratch, you will see a picture of a cat sprite. Each sprite can have its own script.

By making them each do something interesting, you can write a program!

How can I use these instructions?

These instructions are broken down into three levels (columns). The first level gives you a general description of the task. The second step, just in case you get stuck, gives you a bit more information. The third step, will give you even more detailed instructions, just in case you get really, really stuck!

We suggest trying this exercise. Once you’re done, you should upload your miniature ocean to the Scratch website. That way you can download it at home.

Then, feel free to experiment and try some new things! We’ve provided some suggestions for you at the end.
Step 1: Create an aquarium

Click on the picture of the folder below the canvas and find an appropriate aquarium background.

Notice that it has appeared in the sprite window. Now you can write a script to go with it!

To start a new script, select the start tile from the Control menu.

Let's get started by adding a starfish to our aquarium.

We're going to animate him so that he moves back and forth continuously.

Now we simply need to “switch costumes” back and forth!

At the top you will see a drop-down menu. Select the arrow to its right and choose Media > Backgrounds.

Again click on the folder below your canvas (aquarium).

To animate we need to simply switch back and forth between two costumes (or images).

Under the Looks menu, find “next costume”

Find the one that looks like the aquarium! (Media > Backgrounds > Nature)

Drag the flag-start tile to your workspace. Now, whenever you click the green flag, everything attached to this tile will run!

Click the arrow until you find the “Costumes” then “Animals” folder. Find the purple starfish and click OK. Drag him to where you want him to be.

We can load another costume by selecting the Costumes tab in the middle column and selecting Import. Then find the other starfish and click OK.

Select this and drag it to your workspace. (Don't connect it to anything just yet!)
Step 1: Create an aquarium, con’t

We want the starfish to remain animated as long as the program is running, so we’ll need to have it loop.

Go ahead, and give it a try!

What’s wrong?

Let’s try again!

To have something loop, you’ll need to exercise some control. Click the control menu on the left-hand side. Find the tile that says “forever” and drag it to your workspace.

Click the green flag.

How could we exercise some more control to slow things down?

Notice how there is a gap in the middle of forever. Everything that is connected within this gap will play forever, or until you stop the program. Drag your “next costume” piece into the middle. Then connect the loop to your start flag tile.

Click the stop sign to stop.

Try “wait ___ secs” from the Control menu. 0.5 is a good number. Connect this piece inside your loop.

Hit and test it out!
Step 2: Add some fish

Click on the picture of the folder below your aquarium.

Like the starfish we just did, we need to write a script for our fish. This script will be connected to your fish sprite only (completely separate from the starfish script).

Now we need to make sure that the fish appears at the start of the program.

We’re also going to make sure that the fish always appears in the same spot.

Let’s get this fish moving across the screen. Like the starfish, this fish will keep moving at all times.

Finally, we don’t want the fish to disappear off the side of the screen, so we need to fix that!

Last, ensure the fish stays upright when it “bounces” off the wall.

At the top you will see a drop-down menu. Select the arrow to its right and choose Media > Costumes.

Under Looks select “show”. This will mean we can always see our fish at start of our program (later we’ll make the little guy disappear!).

Look under the Motion menu on the left.

Click the Control menu on the left-hand side.

Click the Motion menu on the left-hand side.

Check out the Motion menu again.

Select the sprite and click the double arrow beside it in the workspace.

Select the “go to x:__ y: __” and connect it to “show”. Type in the x and y coordinates. Note that the current x-y location of the mouse appears below your aquarium.

Find the tile that says “forever” and drag it to your workspace so that it connects to the flag-start tile.

Select “move 10 steps” and drag it to the middle of “forever”.

Select “if on edge, bounce” and add it to your “forever” loop.

Find one that looks like a fish! You’ll also want to resize the fish: click on shrink, then click on your fish until it gets to the size that you want.

Drag the flag-start tile to your workspace. Now, whenever you click the green flag, everything attached to this tile will run!

Drag “show” to the workspace and connect it to your program.

Select the “go to x:__ y: __” and connect it to “show”. Type in the x and y coordinates. Note that the current x-y location of the mouse appears below your aquarium.

Find the tile that says “forever” and drag it to your workspace so that it connects to the flag-start tile.

Select “move 10 steps” and drag it to the middle of “forever”.

Select “if on edge, bounce” and add it to your “forever” loop.

Hit and test it out!
Step 3: Add a shark

Now let’s add a predator to our tank!

Let’s copy our fish script into our shark script. Since their basic behaviour will be the same, this will save us some work.

This shark is hungry-- let’s make it eat the fish if it gets too close.

What happens when the shark is touching the fish? It should open its mouth and the fish should disappear.

Return to the sprite selection (see Step 2 for help)

Click on the fish sprite and drag its program over to the shark sprite and let go.

First we need to check to see if the shark and the fish sprites are touching. Use an if statement block to ask.

Opening its mouth is as simple as switching to a different costume that has an open mouth. First we need to import the costume.

Check out the Looks menu and add this to your program.

Find one that looks like a shark! You’ll want to resize the shark smaller, too.

If you do it correctly the program will re-appear on the fish. And when you click the shark, a copy of the program will be there.

Under the Control menu drag the if-structure to your workspace.

Notice the shape of the missing part. Select the Sensing menu and drag over “touching __ ?”

Then use the drop down menu on these new piece to select the fish sprite.

Select the shark and then click the Costumes tab. Click Import and select the desired sprite.

Choose “switch to costume” from the Looks menu. Use the pull-down menu to select the new shark costume. Drag this into your if statement block.

Now add your if-statement to your script.
Step 3: Add a shark, continued

To make the fish disappear, modify your fish script so that if it is touching the shark, it hides.

To help out the fish, we should have him wait long enough for the shark to react to him (i.e. eat him). One second should be enough.

Unfortunately, our shark has no manners! After eating a fish his mouth never closes again. We need to fix this by changing him back to his original costume.

Let’s have the shark think something as he eats the fish.

When a sprite is hidden, it doesn’t go away. We need to make sure that our shark isn’t touching our hidden fish.

Let’s go back to our shark and his costume. Add another “switch to costume” block from the Looks menu and add it after the if-statement.

Check out the Looks menu.

From the Control menu, pull over an “if” statement. From the Sensing menu, add “touching ___ ?” and select the shark sprite. Inside the if statement, drag over “hide” from the Looks menu. After hiding, move the fish away by telling it to go some place “off screen”.

Under Control menu pull over “wait 1 second” and attach it above “hide”.

Now add your if-statement to the loop your fish script.

This time choose the other costume from the list.

Drag over the “think Hmm... for 2 secs”. Change the time to 0.25 secs, and have it say what you want Add it to the if-statement (that’s when the eating happens, after all).

Hit and test it out!
Step 4: Congratulations!

You now have a working digital ecosystem! You’ve got some time now to spruce up your aquarium, or try something new.

Here are a couple ideas to improve your fish tank:

1. Add some more fish!

2. Design your own artwork for your fish.

3. **Challenging:** Once the shark eats all the fish, let’s modify the program so that our fish regenerate after some time has past. This will make our aquarium more interesting.

Check out step one if you forgot how.

Creating your own artwork is easy. Just select the “Paint new sprite” icon and design away!

After a certain amount of time has past, change the fish back to visible. You’re going to need a way to tell the shark whether or not the fish is currently visible or hidden. It wouldn’t look right to eat an invisible fish!

You can ask for more than one condition to be true in an if-statement by using the “___ and ___” under the **Numbers** menu.

What might be the other question you want to ask?

You can create your own variables under the **Variables** menu. Variables allow you to store information, such as whether or not something is true (you can think of 1 for true and 0 for false, for example).

Think about how you might use a variable to help you out here. Do some exploring to see what tools you have available to you, and don’t be afraid to ask for some help!

Hit and test it out!