Basic Game Production

using Scratch 1.4

By the end of this task I should know how to...

1) **import graphics (background and sprites) into Scratch**
2) **make sprites move around the stage**
3) **create a scoring system using a variable.**

Creating Computer Games

All computer games are created by programming PCs or Games Consoles with lots and lots of computer instructions. Instructions tell the computer system how to draw backgrounds, make characters move, keep scores etc.

Initially games were simple enough to be programmed by one person, often working in their own home. As the complexity of the games grew, companies began employing larger and larger teams of programmers and other specialists. To create a game today, often requires input from more than 100 people. Some of their jobs are listed below:

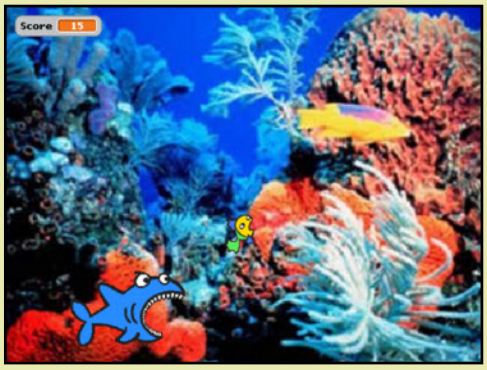
- Scripting
- Graphic Art
- Surround Sound Processing
- Artificial Intelligence
- Game Engine Coding
- Motion Capture
- Full Motion Video
- Marketing

In this unit you will be using Scratch. Scratch was developed (by the USA's famous MIT university) to teach school pupils how to program.

Your task will be to create a simple shark and fish chasing game in four stages.

1. Changing the background (or stage)
2. Inserting characters (or sprites)
3. Making the characters to move and interact
4. Creating a scoring system.

You will then be given the opportunity to develop the game further.

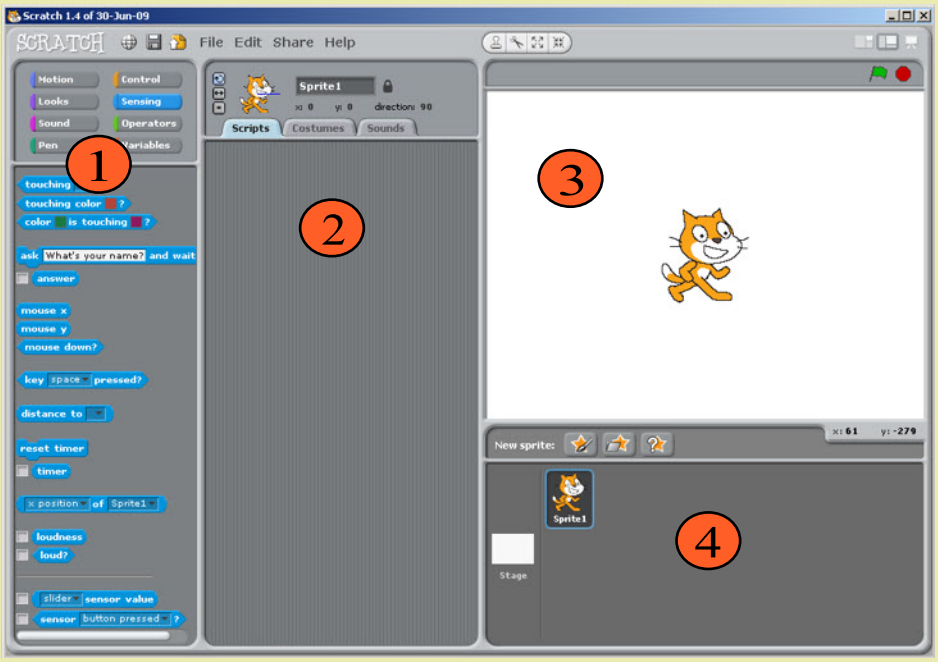


The Stage

Start up a new file in Scratch. If you can't find the program ask your teacher where it is.

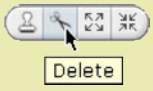
Scratch allows you to add sprites (small graphics like the cat below) to a stage and then control them using instructions called scripts.

The Scratch window is split into 4 sections:



1. **Blocks Palette**
   Blocks for programming your sprites.
2. **Scripts Area**
   Drag blocks from 1 and snap them together to write scripts.
3. **Stage**
   This is where what you produce will be shown.
4. **Sprite List**
   Shows thumbnails of all your sprites.

As we wish to start with a blank stage we start by deleting the cat sprite.

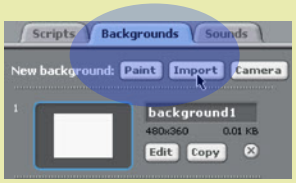
Click on 

and then click the scissors icon on the cat  .

The background colour of the stage is currently white. This can be altered using the Paint Editor.

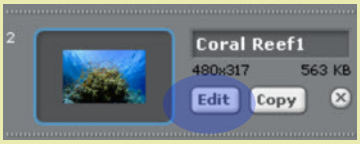
Click on the Stage icon 

and then select backgrounds and import.

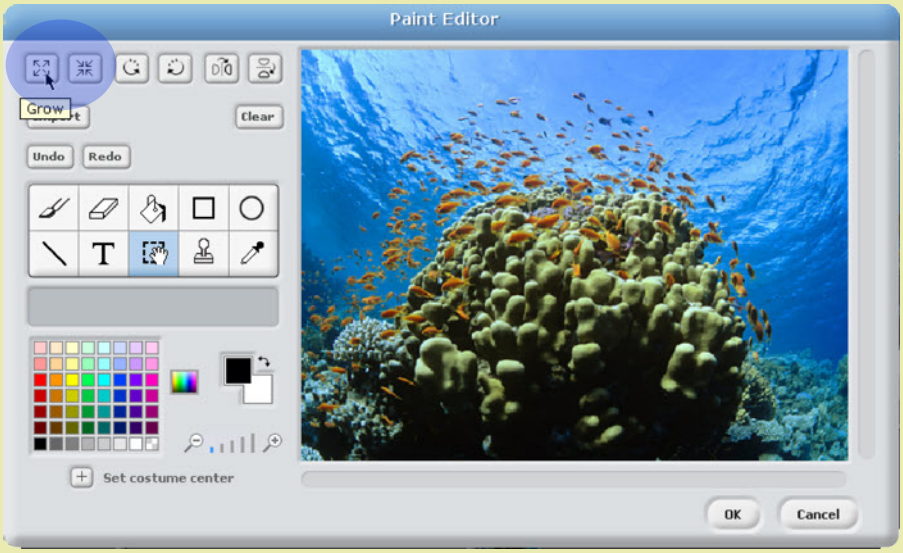


Use the window to navigate to the saved backgrounds and select one.

Now click on the edit button to open the Paint Editor window.



Use the grow and shrink buttons to make the background fill the stage.



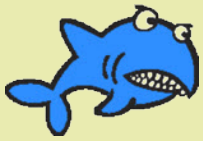
Try - Use the Paint Editor to add your name to the stage.

Try - The game will involve a shark chasing a fish around the stage. Add a suitable name for the game to the top of the stage.

Sprites

A sprite is a small graphic that is part of a bigger picture. This means that all characters and objects in games are called sprites.

Our game is going to have two sprites -

Shark 

Fish 

Later you will learn how to make the sprites move around the stage and react when they come in contact with each other.

Inserting Sprites

To add the shark sprite click on the 'Choose new sprite from file' button. 

Open the 'Animals' folder and select shark1-a.

Repeat the above steps to add either :


fish2


fish3


or fish4

Use the Shrink Sprite button to make the fish smaller than the shark.

You should now see two sprites in the sprite list (1 shark and 1 fish).

Each time you want to change how a sprite looks, moves or behaves you must select it from the sprite list.

Later on, when our shark catches the fish we will want it to open its mouth. To do this we have to add another costume to the shark sprite.

Select the shark sprite 

and click on costumes.

Click on the Import button and select shark1-b.

You should now see a second costume below the first.

Try - Click on the Edit button for each costume and use the Paint Editor to change the colour of your shark to one of your choice. You'll have to do it for both costumes.

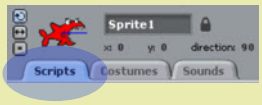
Try - Use the Edit button to colour your fish as well.

Making Sprites Move (Shark)

The next part of creating our game is the fun bit. To create a game we have to give our sprites instructions that tell them how to behave when the game is being played.

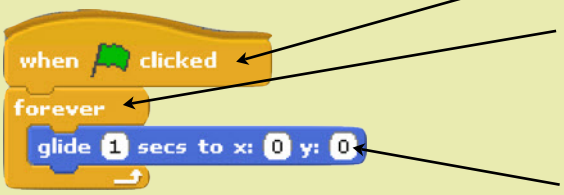
Instructions in Scratch are called scripts and are created using blocks.

Let's start with the shark. Select the shark from the sprite list. 

Click on the Scripts tab in the Scripts Area. 

Scratch blocks are grouped in 8 categories. To add blocks we simply select the appropriate category and drag across the blocks we want to use.

Using the Control and Motion categories to create the script shown below.



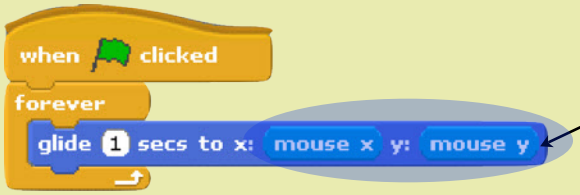
This starts the script when the Go flag is clicked.

This is called a loop. The *forever* loop will keep repeating the blocks inside it until the Stop button is clicked

The *glide* block will make the shark glide to a position(x,y) on the screen. The amount of time it takes to get there is set in seconds.

We can get the shark to glide to wherever the mouse is on the screen by adding the *mouse x* and *mouse y* blocks to the glide block.

Use the Sensing category to add mouse x and mouse y to the glide block.



This now reads as:

Take 1 second to glide to the x and y coordinates of where the mouse is pointing.

Click the Go flag  to start you game. Move the mouse around the stage and watch how your shark now follows the mouse.

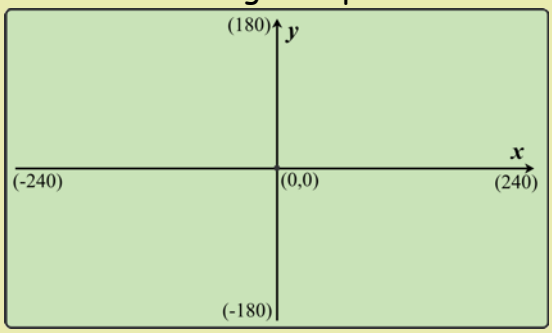
Try - Experiment by changing the glide time to values between 0.2 and 5 seconds.

Making Sprites Move (Fish)

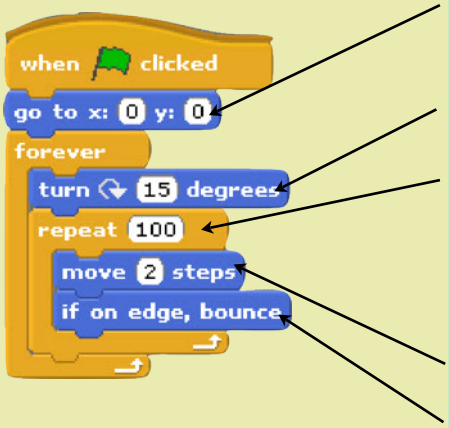
Each sprite has its own scripts so before we start creating a script for the fish we must select the fish from the sprite list.

The fish will start in the middle of the stage when the flag is clicked. Note from the graph on the right that the middle of the stage are coordinates (0,0).

Stage Graph



To make the fish move create the script below.



This moves the fish to the centre of the stage when the Go flag is clicked.

The fish turns 15 degrees before it starts to move.

The repeat loop can be set to repeat the blocks inside it (move & bounce) 100 times. Note that because we have one loop inside another, the repeat loop will have to finish before the script returns to the top of the forever loop

Move is used to make a sprite move across the stage.

Bounce makes the sprite change direction when it hits the edge of the stage.

Click on the Go flag and watch carefully how the fish moves. Try moving the shark to catch the fish and you will see that it is very easy because we can predict where the fish is going to be next.

We can use another block in the Operators category to improve how our fish behaves.

Add the *pick random* block to the turn block and enter 1 to 359.

The fish will now turn a random number of degrees between 1 and 359 each time the forever loop runs.

Try - Try changing the repeat and move blocks to different values. If the game is to work well, the fish should be difficult but not impossible to catch.

Keep trying different values until you are happy with how the fish is moving.

Interaction and Scoring

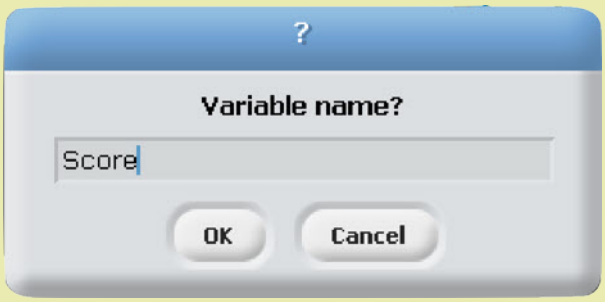
The last task is to keep a score of the number of times the shark catches (touches) the fish.

To keep score the game must be able to store a number. This is done using a variable.

We will create the variable on the stage so click on the stage icon next to the sprite list. 

Click on the Variables category and then Make a variable .

Create a variable called score.



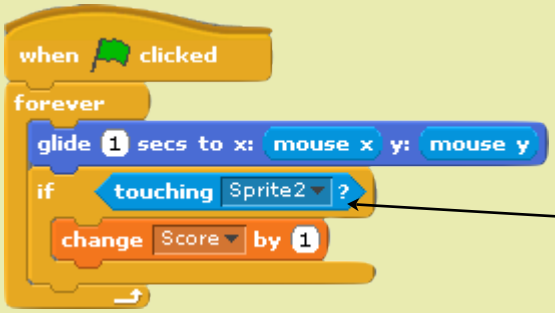
Now use the variable to create the script shown below.

When the flag the score will be set to 0.

Next we have to add something to the score when the shark catches the fish.

Select the shark from the sprite list. 

Use the Sensing and Control categories to add the following new blocks to the shark.

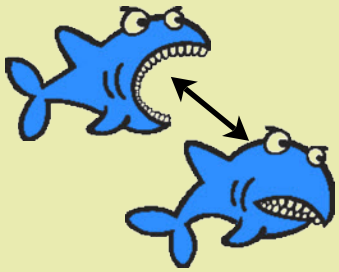


An *if* block is used to make decisions.

This block can be read as:

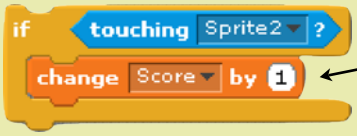
If the shark is *touching* Sprite 2 (the fish) then add 1 to the score variable.

Changing Costumes



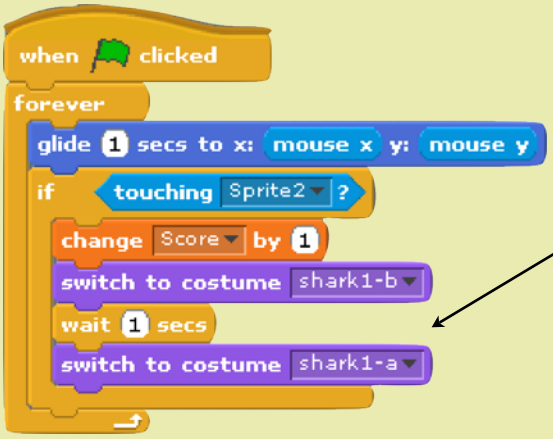
The last task is to animate our shark so that it looks like it trying to bite the fish when it catches it. To do this we need to use the second shark costume we added earlier (on page 4).

We already have an IF block that adds one to your score if the shark sprite touches the fish sprite.



If we want to make other things happen when the fish is caught we add more blocks inside IF block.

Use the Looks and Control categories to add three more blocks to the script.



These blocks change the shark costume, pause for 1 second and then change the costume back.

This makes it look like the shark is closing his mouth and opening it again.

You have now finished the basic game. **Try it!**

Intermediate Game Production

You can now choose which of the 7 Intermediate tasks you wish to attempt.

Bonus Points
Adding Speech
Two Players
Adding Sound
More Fish
Game Over
Setting Difficulty

Each task will describe how the game could be improved. You will be expected to think of how you would solve each problem and add more variables, blocks and scripts to the game.

A solution will be given on each sheet but **you should attempt to solve each problem yourself!**