

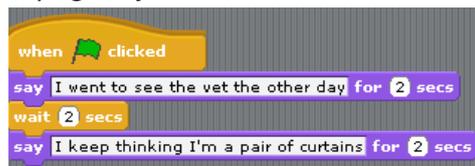
Scratch Lesson Plan

This lesson plan represents a summary of what I covered when I taught Scratch to a class of 11 and 12 year olds. The objective was to teach important programming concepts, as well as the product itself. Note that this was written AFTER the lessons but is still largely based on my original lesson plan.

Lesson One – Scratch Scripts and Sprites

Pre-lesson work – Ask the students to prepare a joke. It should involve 2 characters and at least 2 rounds of dialog.

- Objective
 - To familiarise the students with Scratch and Scratch terminology (Stage, Scripts, Instructions).
 - Create a joke animation to give the students an understanding of how to write in sequence.
- Introduction
 - Explain what we're going to be doing this week and why
 - Going to be learning how to use Scratch.
 - To teach us important lessons about how computer programs work.
 - Discuss 'what is software?' and how it differs from hardware.
 - Talk about a making some toast
 - Ask the students to explain the steps involved.
 - Discuss how things have to happen in order to get the desired outcome and how this relates to writing software.
 - Introduce Scratch
 - Explain the User Interface
 - The Stage
 - The Instructions
 - The Script
- Exercise - Students use Scratch to create a simple animation of their joke
 - Section 1
 - Go through creating my joke, only up to the point of having the characters saying the joke.



- Ask the students to create their joke. Don't worry too much about the small presentation details at the moment; the main objective is to get their characters talking in sequence by correctly setting the 'wait' times.

- Section 2
 - As a group, identify anything wrong with our jokes and how they may be improved.



- Explain how we can control the stage background .
 - Ask the students to make improvements to their joke.
- Section 3
 - Saving work
 - Ensure the students are familiar with the concept of x and y coordinates on a graph, and explain how this relates to the Scratch stage.
 - Demonstrate how to make the characters glide in from the edge of the screen and meet in the middle before starting the joke.

```

when clicked
go to x: -284 y: -127
glide 1 secs to x: -150 y: -92
say I went to see the vet the other day for 2 secs
wait 1 secs
say I keep thinking I'm a pair of curtains for 2 secs

```

Lesson 2 – Variables

- Objective
 - To familiarise everyone with variables as a general programming concept and within Scratch.
- Introduction
 - Recap what we covered in the previous lesson.
 - Finish off the motion from yesterday
 - EXTENSION OPPORTUNITY – for those who have finished, suggest animating their characters by changing the costumes as they move.
- Exercise (Variables)
 - Talk about and demonstrate the problem of adjusting the wait time i.e. it has to be done in several places. Explain how in a large program this could take a long time and lead to mistakes.
 - Introduce variables as a general programming concept and within Scratch.
 - Demonstrate how we can use variables to improve our program by replacing the wait time constant with a variable.
 - Ask the children to do the same to their jokes.

- EXTENSION OPPORTUNITY - They could add faster or slower buttons to reset the variable and dynamically alter the speed of their joke.

```

when clicked
go to x: -284 y: -127
glide 1 secs to x: -61 y: -92
say I went to see the vet the other day for Talk Time secs
wait Talk Time secs
say I keep thinking I'm a pair of curtains for Talk Time secs

```

Lesson 3 – Events

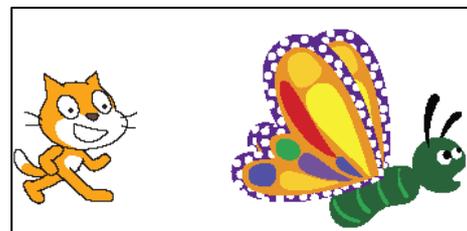
- Objective
 - To familiarise the children with the programming concepts of events and looping, and how these can be applied in Scratch.
- Introduction
 - Recap what we covered in the previous lesson.
 - Talk about the concept on an ‘event’, how performing one action can cause another action to occur, such as clicking ‘search’ of Google causes a search to happen.
 - Explain how events are achieved in Scratch by using ‘Broadcasts’, and how these ‘Broadcasts’ enable sprites to send messages to each other.
 - Create a demonstration where a character broadcasts a message that tells a second character to change colour.

```

when clicked
say Change colour for 2 secs
broadcast Change Colour

when I receive Change Colour
change color effect by 100

```

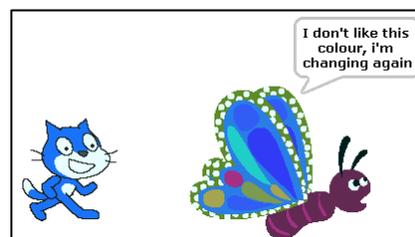


- Add in extra steps to show how we could create an infinite loop.

```

when I receive Change Colour
change color effect by 100
wait 2 secs
say I don't like this colour, i'm changing again for 2 secs
broadcast Change Colour

```



- Exercise
 - Ask the students to add a reset button to their joke animation and then use a broadcast to restart the joke whenever the button is clicked.

- EXTENSION OPPORTUNITY - Add a new sprite which listens to the joke and then laughs when it has finished. This is a good opportunity to see how much easier this is to do with a broadcast, rather than for example, telling the sprite to wait for n number of seconds until the joke has finished.
- EXTENSION OPPORTUNITY - Discuss how broadcasting the 'reset' message at the start of the joke means that we don't have to duplicate the code.

```

when clicked
  go to x: 268 y: -101
  glide 1 secs to x: 108 y: -84
  wait Talk Time secs
  say Why what's wrong? for Talk Time secs
  wait Talk Time secs
  say Oh pull yourself together for Talk Time secs

when I receive Restart Joke
  go to x: 268 y: -101
  glide 1 secs to x: 108 y: -84
  wait Talk Time secs
  say Why what's wrong? for Talk Time secs
  wait Talk Time secs
  say Oh pull yourself together for Talk Time secs

```

```

when clicked
  broadcast Restart Joke

when I receive Restart Joke
  go to x: 268 y: -101
  glide 1 secs to x: 108 y: -84
  wait Talk Time secs
  say Why what's wrong? for Talk Time secs
  wait Talk Time secs
  say Oh pull yourself together for Talk Time secs

```

Lesson 4 – Our first game

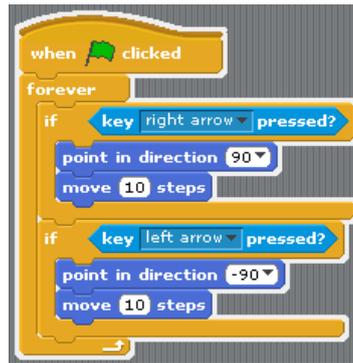
- Objective
 - To put into practice everything the students have learnt so far and re-enforce the learning.
- Introduction
 - Create a quick demonstration of how to make the Scratch Cat follow the mouse pointer around the stage.

```

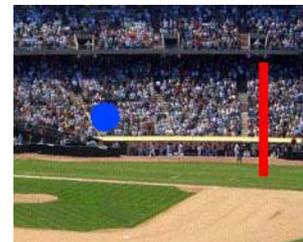
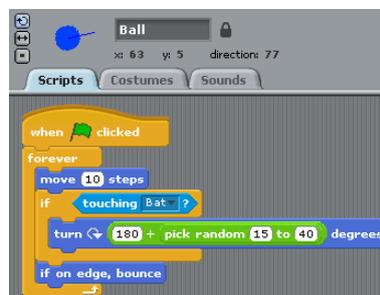
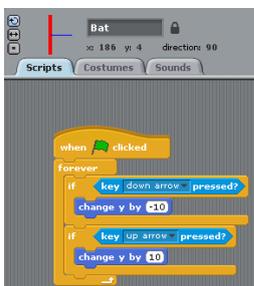
when clicked
  forever
    point towards mouse-pointer
    move 10 steps

```

- Alter the script so that the cat moves in response to pressing the keyboard arrow keys, rather than the mouse movement. Do this by introducing IF statements and a LOOP which keeps checking for which key is being pressed.



- Exercise
 - Demonstrate a simple bat and ball game and talk about the techniques involved.



- Ask the students to produce something similar, but not a copy. For example, they could make a simple cricket-type game where hitting certain parts of the screen scores more points than others. Or they could position their bat at the bottom of the screen rather than the right-hand side.

Lesson 5 – Completing the game

Depending on the length of the lessons and the abilities of the children, you may find (as I did) that the fifth lesson will simply be a continuation of the fourth.

- Introduction
 - Discuss with the children how we could improve our bat and ball game.
- Exercise
 - Add more functionality to the game:
 - Game over if the ball touches the back wall
 - Add a second bat for a two player game
 - Add a scoring system
 - Add an introduction screen where the player can choose the difficulty (e.g. speed, number of balls).
 - EXTENSION OPPORTUNITY – Tackle creating a more challenging game, such as this fish game (<http://scratch.redware.com/video/fish-game>).