INTRODUCTION TO SCRATCH
Scratch is an amazingly simple way of getting into coding. It has a user friendly interface using a lot of visual cues. Because it is open source you can adapt existing projects to your own needs, or share your projects to get feedback. Scratch provides opportunities to integrate cross curricular content, for example Maths, Music or Humanities.

Task 1 - opening up scratch
- Type in www.scratch.mit.edu (you can chose to login or be a guest).
- Click on Create.

Task 2 - becoming familiar with the scratch interface.

As Scratch is open source, it is not immune to abuse. Try not to cut and paste code or blocks from an unknown source, try to reproduce it for yourself to ensure you understand what it's for.
1.0 CREATING IMAGES IN SCRATCH

There are many ways of adding graphics to your project. Import a file, a photo or draw one in Scratch.

EXERCISE 1
Make your sprite your own!

Task 1 - change an existing sprite
- Click ‘choose sprite from library’ in the ‘sprites display’ area.
- Delete existing sprite by using scissor tool.
- Choose a Sprite by double clicking.
- Click on Costume in the tasks bar (you should see the above display).
- In vector mode, change your Sprite’s appearance by playing with the colours, sizing and maybe add some shading! If the sprite is grouped not allowing changes, click on the sprite and then click ungroup at the bottom of the Drawing tools.
- Add personality to your Sprite by adding accessories or new features.
1.0 CREATING IMAGES IN SCRATCH CONTINUED...

Task 2 - add a costume
- Duplicate the Sprite’s costume by right clicking and then select *duplicate*.
- Change the Sprite’s appearance again, for example add graphical movement - like its whizzing through the air.

You can always hit *Undo!* at the top of the screen. And if you have been drawing and it disappears, check to see if it is buried in a layer behind.
EXERCISE 2
Animate your Sprite!

Scratch uses the x,y co-ordinates to determine where the sprite should be.

Task 1 - make it move
• Click on the Scripts Tasks section.
• Add event ‘when flag clicked’.
• Add a motion block telling the Sprite where they need to be: x=0, y=0
• Using motion blocks rotate the Sprite back and forth 15 degrees, 5 times. Add a wait control block after each motion.
• Add more move blocks, arrange them so that the Sprite moves slowly up, across fast, back down and then returns to the start, forming a precise rectangle.

Task 2 - change costume and make some noise
• Using a look block add a costume change as your sprite moves through the air.
• Add a sound block as it’s buzzing along.

Task 3 - add some interactivity
• Add a 2nd code script that changes the colours of the sprite when you press the space bar.

Our first block is an event block and it sets up when the code should run, you can have multiple sets of code running at the same time.

When code doesn’t have a time element it may run but it won’t be visible to the eye. Try adding a Wait block to slow down the run time.

We have now started to program our Sprite by using simple move blocks, these blocks can be fitted together in an infinite amount of ways.
3.0 STORY TELLING
Characters, plot and good dialogue what could be more thrilling!

Excercise 3

Task 1 - tell a story
Start by getting your Sprites to speak to each other.
• Start a new event/project, delete any previous events.
• Add a second Sprite; you may need to mirror or change your Sprite’s size to get the right layout.
• Add a background, this will set the scene of your story.

Your scene should look something like this

• Add **event** ‘when flag clicked’ to start new program
• To add dialogue, use the ‘say for 2 secs’ in **looks** blocks.
• To get the another Sprite to respond, add a ‘broadcast’ in **event** blocks. Click on the arrow to add a title for each message.
• Switch Sprites and, use ‘when I recieve message’ **event** to cue the next turn in the conversation. Repeat steps: ‘say’, ‘broadcast’, & switch sprites for ‘when I receive message’.

This is what each speech bubble should look like after the first turn
4.0 EXTERNAL INPUTS

*Using external inputs, allows scratch to interact with outputs such as makey makey.*

Excercise 4

Task 1 - draw on the screen
Use the keyboard to control a pen on the screen.
• Open new project.
• Add a pencil Sprite.
• Change the pencil’s centre point to the end of the tip (under costumes tab).
• Add the following blocks:

![Code blocks for drawing](image)

• Add code blocks for the down, left and right arrows.

CHALLENGE:
Using the ‘If and Then’ block can you make the pencil draw with just the mouse?
5.0 RESOURCES

Scratch for educators:
http://scratched.gse.harvard.edu/

Scratch Projects
https://scratch.mit.edu/explore/?date=this_month

Scratch and MakeyMakey projects
http://makeymakey.com/gallery/?tag=scratch