

# USING SCRATCH TO DEVELOP NUMERACY (PDST/LERO)

---

SCRATCH



## Contents

Course Introduction .....	3
Module 1 .....	4
1. Introduction to Scratch .....	5
2. Scratch 2.0 Offline Editor .....	5
3. Getting Started with Scratch .....	5
4. Discussion Time.....	5
Module 2 .....	16
1. Exploring numeracy in Scratch.....	17
2. Communicating in Scratch .....	17
3. Gaming in Scratch .....	17
4. Drawing with Scratch .....	17
5. Discussion Time.....	17
Module 3 .....	32
1. Exploring numeracy in Scratch.....	33
2. Problem Solving .....	33
3. Images and Video.....	33
4. Discussion Time.....	33
Module 4 .....	44
1. Internet Safety and Cyberbullying .....	45
2. Scoilnet, Scoilnet Maps and Encyclopaedia Britannica.....	45
3. Quizzes in Scratch .....	45
4. Discussion Time.....	45
Module 5 .....	54
1. Create a Scratch Project.....	55
2. Support for Teaching Scratch.....	55
3. Other ICT Resources from PDST Technology in Education .....	55
4. e-Learning Action Plan .....	55

## Course Introduction

Scratch is a free application, developed by the MIT Media Lab, which allows users to create and share their own interactive stories, animations and games.

It is easier to use than traditional programming languages as it consists of graphical blocks which snap together.

This course enables teachers to learn how to use Scratch and introduce it to their pupils to help them explore aspects of the curriculum in an exciting and engaging manner. The course focuses on using Scratch to create projects which support the concepts, content and skills of the mathematics curriculum.

Participants completing this course will be enabled to:

- Use Scratch programming to support teaching and learning in the primary curriculum.
- Develop projects using Scratch.
- Plan effective lesson strategies for using Scratch in the classroom.
- Develop an e-Learning action plan to outline how it will be used in your classroom.

## Module 1

---

SCRATCH



## **Module 1**

### **Suggested Duration of Module**

4 Hours

### **Objectives**

Content of Day/Module 1: Introduction to Scratch

- 1. Introduction to Scratch**
- 2. Scratch 2.0 Offline Editor**
- 3. Getting Started with Scratch**
- 4. Discussion Time**

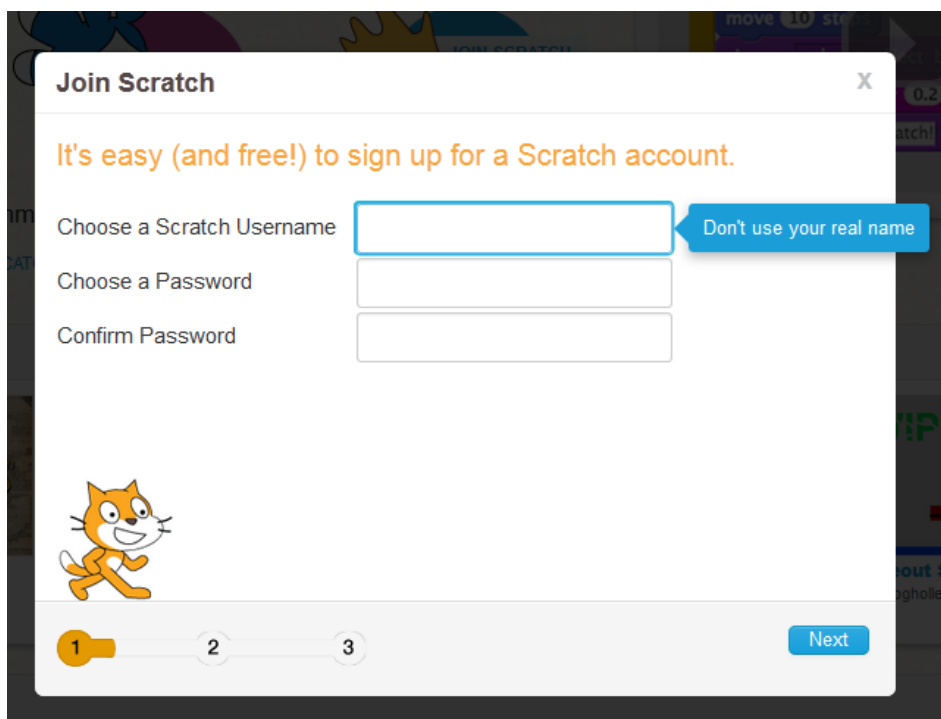
## 1. Introduction to Scratch

There are two versions of Scratch, Scratch 1.4 and Scratch 2.0. Scratch 2.0 offline editor will be used for this course. But first let's check out Scratch 2.0 online. We would recommend working with FireFox or Chrome when working with Scratch online. Internet Explorer can be problematic during Sign In.

Navigate to Scratch 2.0 at [scratch.mit.edu](http://scratch.mit.edu). Play the Introduction to Scratch 2.0 video linked to from the Scratch home page (or at <http://vimeo.com/65583694>)

The Scratch community put a strong emphasis on collaboration and sharing work. The [scratch.mit.edu](http://scratch.mit.edu) website contains millions of projects created by users around the world. You can join this community by creating a Scratch account. Some teachers ask children to create individual accounts. However, others create a single class account.

Click on "Join Scratch" to sign up for a Scratch account if you don't have one already.

The image shows a 'Join Scratch' sign-up form. At the top, it says 'Join Scratch' with a close button (X). Below that, a message reads 'It's easy (and free!) to sign up for a Scratch account.' The form has three input fields: 'Choose a Scratch Username', 'Choose a Password', and 'Confirm Password'. A blue tooltip points to the username field with the text 'Don't use your real name'. At the bottom left is the Scratch cat logo. At the bottom right is a 'Next' button. A progress bar at the bottom shows three steps, with the first step (1) highlighted in orange.

If you have an account already click on “Sign In”.

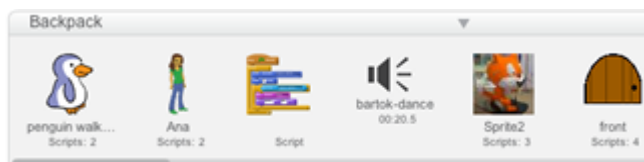


Click “Create” to navigate to the Scratch programming environment. This is how to access the Scratch 2.0 online programming environment. Take some time to explore the [scratch.mit.edu](http://scratch.mit.edu) website (Click on Scratch, Create, Explore, Discuss and Help menu options)

### New Features in Scratch 2.0

Here are some of the new features in Scratch 2.0. This information is taken from [www.scratch.mit.edu](http://www.scratch.mit.edu) ([scratch.mit.edu/overview](http://scratch.mit.edu/overview); 2013) The new features are being highlighted for participants that are familiar with Scratch 1.4.

- When you create a project, you are the only one who can see it. After you share it, anyone can see and remix it.
- Your **Backpack** lets you copy and move sprites, costumes, backdrops, and scripts from any project to another. If you're logged in, you can open your Backpack inside any project (it's at the bottom of the screen). (Not available in offline editor yet)



- You can use the webcam on your computer to interact with projects by moving your hands or body.
- You can now make your own programming blocks.

More Blocks



- Use clone blocks in your scripts to make copies of sprites.

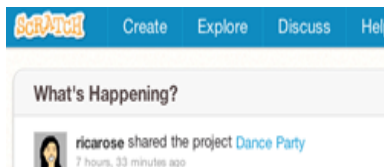




- Store numbers in cloud variables to create surveys and other projects.




## Community Features

1. On the home page, you can see what others have shared recently and scroll to see many more projects.



2. When viewing a project, click  to see how it works and experiment with the code.
3. Inside any project, click  to save your own version and make changes. After you share it, the project page will highlight the original creators and links to their projects.
4. Discover interesting projects by clicking on the Related Projects bar at the bottom of any project page.



5. Click your username or icon to go to your **Profile** page, where you can feature one of your projects and tell others what you're working on.
6. People can post comments on your Profile page and  your Scratch account to see updates.
7. Galleries are now called **Studios**, and can be curated by you and others you invite.
8. The Search features lets you find and preview projects more easily.

We will now prepare to use the Scratch 2.0 offline editor.



## 2. Scratch 2.0 Offline Editor

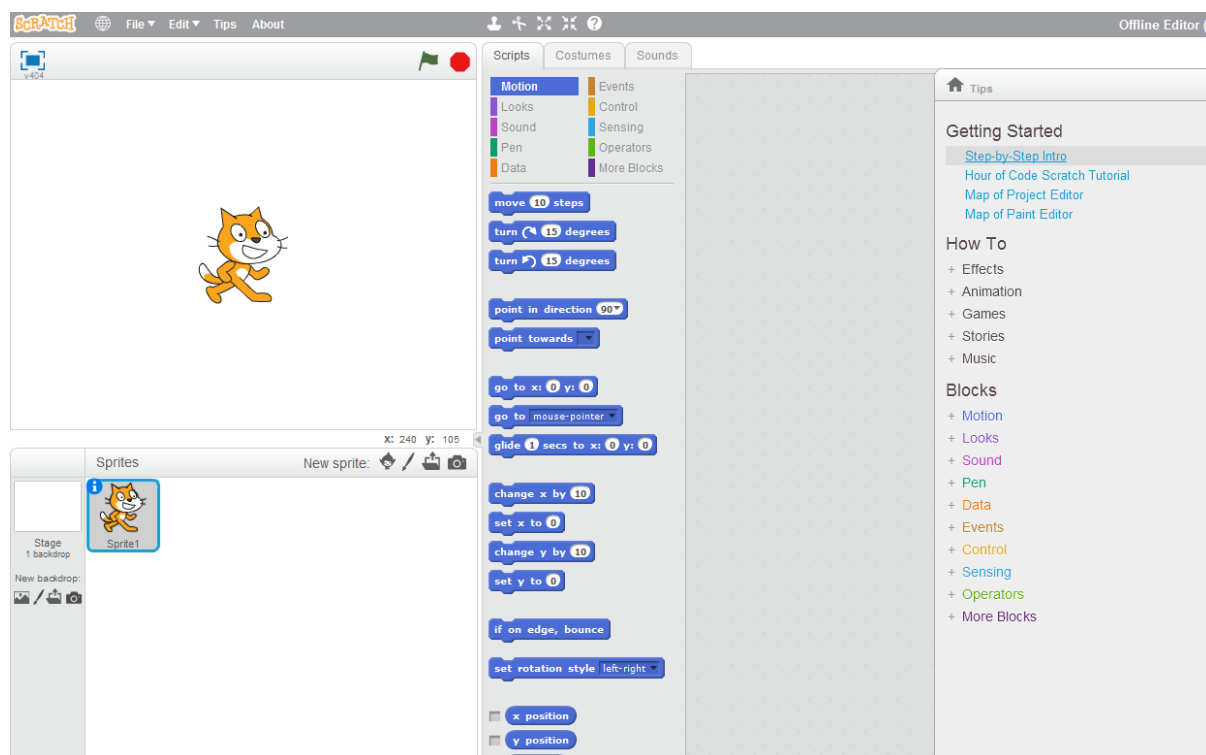
This course will use the Scratch 2.0 offline editor. Education Centres will pre install Scratch. If the offline editor is not installed, follow the instructions on <http://scratch.mit.edu/scratch2download/> and install the offline editor. The Scratch 2.0 offline editor is offline which means it is not dependent on a fast broadband connection.

## 3. Getting Started with Scratch

Once the offline editor is installed, start it and click on Tips, Getting Started. The Scratch Project Editor is described under Tips, Getting Started, Map of Project Editor. Click on this to view the details of the Project Editor.

Next, start the Step-by-Step introduction.

Note: If the Tips window disappears, click on the ? at the side of the screen to make it reappear.



Step through each of the 13 steps in the Step-by-Step introduction. This Step-by-Step guide introduces fundamental programming concepts and allows participants become familiar with the Scratch programming environment.

## Participant Activity

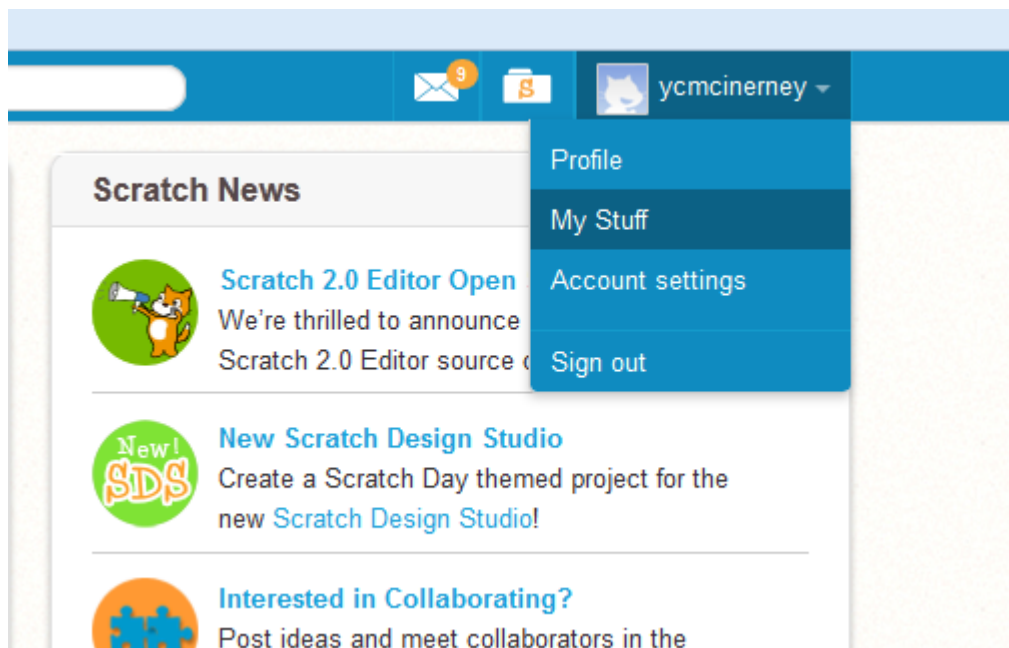
Participants are now ready to complete some project work using resources on [scratch.ie](http://scratch.ie). This website is the home of Scratch in Ireland. It provides regular updates about training and the National Scratch Competition. It also contains a large bank of lesson plans for both primary and secondary schools. To avail of the lesson plans it is necessary to register.

1. Go to [www.scratch.ie](http://www.scratch.ie).
2. Click on “Resources”, “Primary Resources”, “Lesson Plans”.
3. Click on “Register Here” and fill in your details.
4. A password will be emailed to you that will enable you to access all lesson plans.
5. When you register you will receive updates on Scratch in Ireland. If you wish, you may opt out of these email updates.

Click on <http://scratch.ie/primary/resources>. Choose Lesson 1 – Under the Sea. Complete Challenge Time 1. If you feel confident in this task, and if time allows, continue onto Challenge Time 2, and the Ultimate Challenge.

To save your project, click on “File” and “Save” or “Save as” and give the project a meaningful name. This will save the project within the selected folder.

If you are ready to share your project online, click on “Share to website”. To view your project online, go to [scratch.mit.edu](http://scratch.mit.edu), sign in, and under your username, click on “My Stuff”.

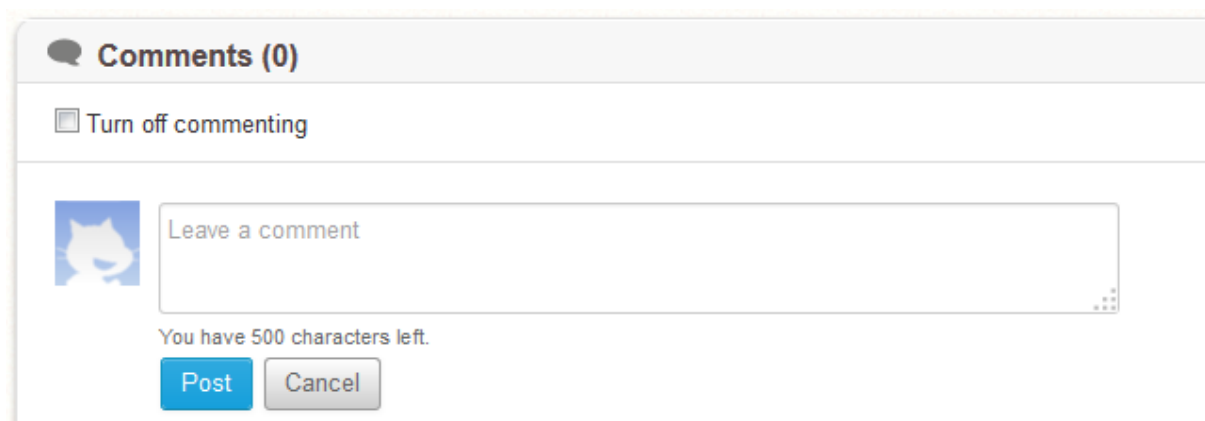


## Shared projects

When you share your project from the offline editor, it is visible by everyone on scratch.mit.edu. When you save your project on scratch.mit.edu, you need to click Share to share your project.

## Comments

Once a project is shared, another user can add comments to your project. You can turn off the functionality by click “Turn Off Commenting” if you don not want users to comment on your project.

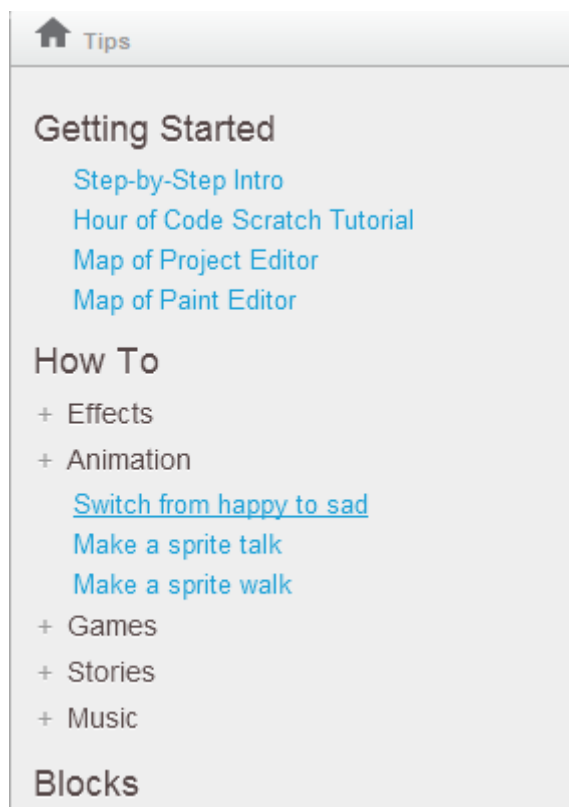


## Studios

You can add projects to studios by clicking “+ New Studio”, decide if you want to “allow anyone to add projects” and then click on “Add Projects” to add projects to your studio.

## Costume Changes

Costume enables users to create very effective animations quite easily. Click on Tips, “How To”, “Animation” and complete the three examples “Switch from Happy to sad”, “Make a sprite talk” and “Make a sprite walk”.

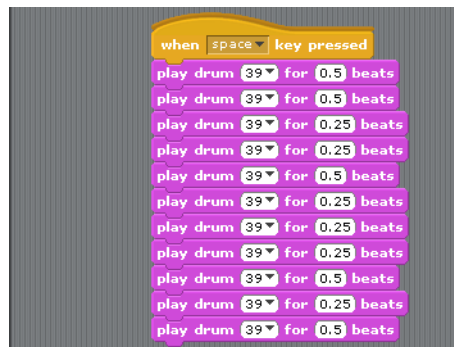


## Sound in Scratch

To use sound/music in Scratch you will need a set of speakers or headphones. Experiment for a few minutes with the Sound Palette

### Drum Beat

- Drag out the “play drum 48 for 0.2 seconds” block.
- Use the drop down arrow to select different percussion instruments.
- Duplicate this block and change the length of the beats to create a rhythm.



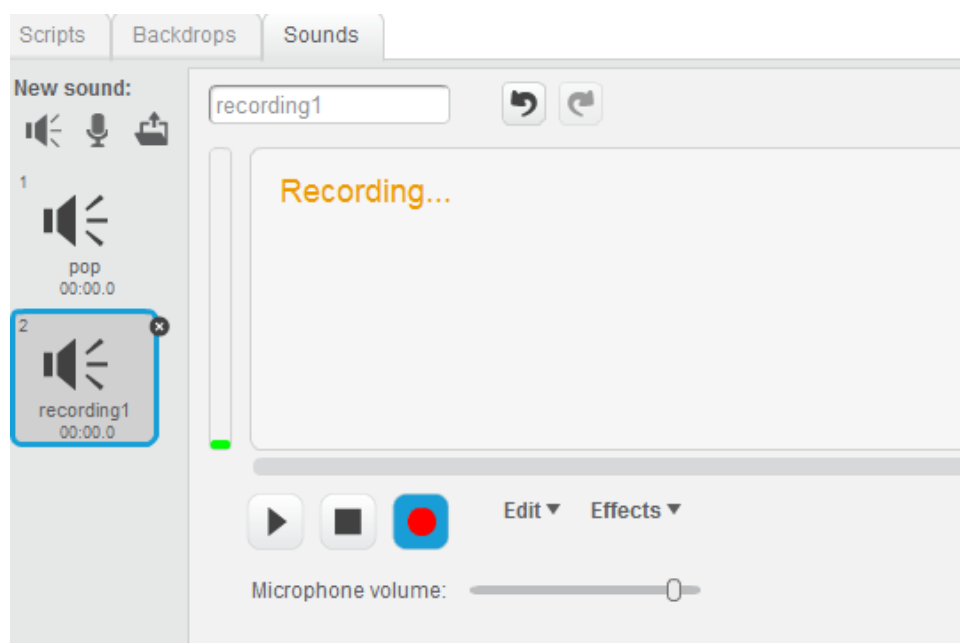
## Musical Notes

Input the following code. Do you recognise the music? Can you set up a round for this tune? Would you like to record someone singing this tune and play the sound clips in sync with the instrument? Try varying the instruments.



## Sound Files

- Click on the sounds tab. Click the loudspeaker to “Choose sound from library” and select some sounds from the folder. Once selected these will appear as a list in the sound tab.
- You may also import sounds that have been saved to desktop from websites such as [www.pacdv.com/sounds](http://www.pacdv.com/sounds).
- Click back into the scripts tab.
- Use the “Play Sound” block to incorporate the sounds you have chosen.
- In the sounds tab you can also record your own sounds. Most laptops have an inbuilt microphone. If this is not present you must attach an external microphone to record.



- Sounds may also be added to the background.

## **Participant Activity**

1. Go to <http://scratch.ie/primary/resources>. Choose Lesson 3 – Battle of the Bands. You will need to enter the password to access this lesson.
2. Complete Challenge Time 2. If you feel confident in this task, and if time allows, continue to the Ultimate Challenge.

#### **4. Discussion Time**

Today's objectives were to cover:

1. Introduction to Scratch
2. Installation of Scratch 2.0 Offline Editor
3. Getting Started with Scratch
4. Discussion Time

Discuss the day with your colleagues and reflect on what you have learned.

Discuss the cross curricular use of Scratch in particular for music. Discuss aspects of Scratch that are relevant to numeracy. What could you use in your classroom if you were teaching tomorrow?

## Module 2

---

SCRATCH





## Module 2

### **Suggested Duration of Module**

4 Hours

### **Objectives**

Content of Module 2:

- 1. Exploring numeracy in Scratch**
- 2. Communicating in Scratch**
- 3. Gaming in Scratch**
- 4. Drawing with Scratch**
- 5. Discussion Time**

Participants are asked to bring a digital camera for Module 3 of this course (tomorrow). If your computer does not have an SD card reader please bring the USB connector for your camera.

In Module 2, we will consider Scratch in the context of the primary mathematics curriculum. We will further enhance Scratch capabilities of participants, with particular emphasis on concepts related to numeracy.

## 1. Exploring Numeracy in Scratch

The mathematics curriculum for primary comprises five strands:

Number
Algebra
Shape and Space
Measures
Data

The skills of the Maths Curriculum should be emphasised. Explain to participants that Scratch can be used in lots of different subjects while still incorporating these skills.

Skill	Examples of its use in Scratch
Applying and problem-solving	<ul style="list-style-type: none"><li>• Breaking down a problem into individual tasks</li><li>• Creating solutions to solve problems</li></ul>
Communicating and expressing	<ul style="list-style-type: none"><li>• Pair work or Group work</li><li>• Class discussion at end of Scratch lesson- ask children what went well, what didn't work, ask children to solve Scratch errors for each other</li></ul>
Integrating and connecting	<ul style="list-style-type: none"><li>• Connect informally acquired mathematical ideas with formal mathematical concepts</li><li>• Carry out mathematical activities which involve other areas of the curriculum</li></ul>
Reasoning	<ul style="list-style-type: none"><li>• Sequential and logical thinking</li><li>• Reading through pieces of code</li></ul>
Implementing	<ul style="list-style-type: none"><li>• Use appropriate Scratch tools to complete tasks</li></ul>
Understanding and recalling	<ul style="list-style-type: none"><li>• Recalling the function of each block</li><li>• Understanding which blocks will be most helpful in a specific project or a specific piece of code</li></ul>

For more details on this see:

[http://www.curriculumonline.ie/getmedia/9df5f3c5-257b-471e-8d0f-f2cf059af941/PSEC02\\_Mathematics\\_Curriculum.pdf](http://www.curriculumonline.ie/getmedia/9df5f3c5-257b-471e-8d0f-f2cf059af941/PSEC02_Mathematics_Curriculum.pdf)

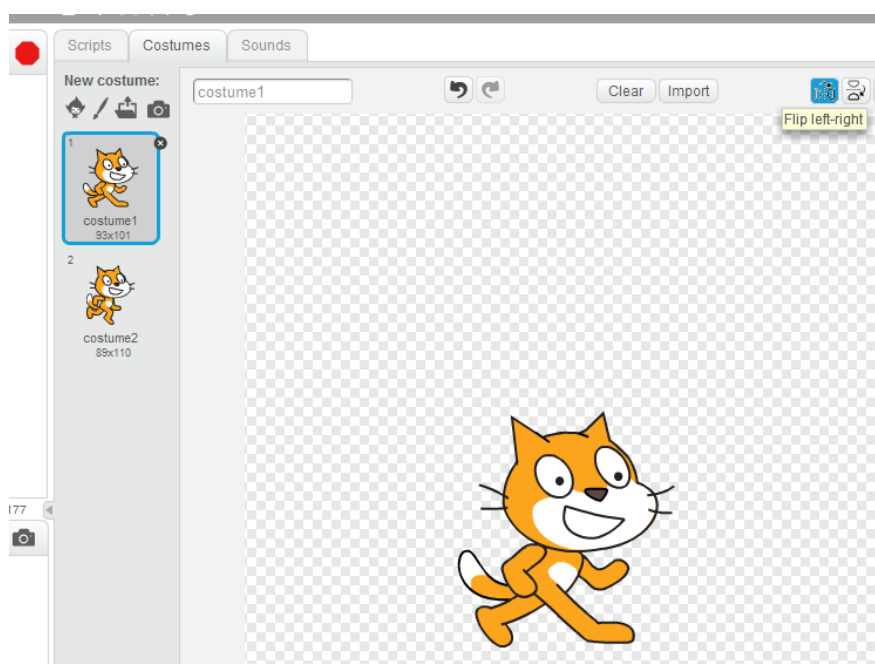
## 2. Communicating in Scratch

In Scratch we use broadcasts to communicate between sprites and between different parts of the program. Broadcasting is a very useful tool to control the flow of a program. We will cover this important concept now.

A broadcast is used to send a message from one sprite to another or from one part of the program to another. A broadcast is like a radio signal. It is sent out from one sprite or background. It is received by all the sprites and the stage. Once a sprite receives a broadcast it can be programmed to react in a particular way.

### Participant Activity

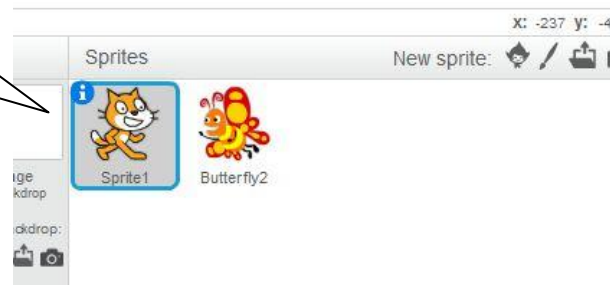
1. Select two Sprites from the folders or draw two sprites.
2. We are going to code a conversation between these sprites so have them face each other.
  - a. If a sprite is turned the wrong way, click on the costumes tab. This brings you into the Paint Editor. Click “Flip Left-Right”.



3. Give your sprites meaningful names e.g. Clever Cat
4. We will code the first sprite's first words. Ensure the sprite is highlighted in blue in the Sprite List. Click on the Scripts tab.



When you want to create code for a sprite, ensure that the correct sprite is highlighted in the Sprite List.



5. Use the purple Looks Palette. Pull out “Say Hello for 2 secs”. Click on Hello to edit the speech. Add a control block.



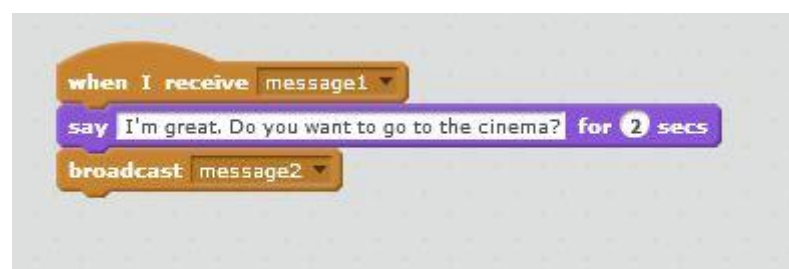
6. To get the second sprite to speak we must first send out a signal that the first sprite is finished speaking. This is what a broadcast is for.
7. Click on the Events Palette. Pull out the “broadcast” block. Click on the black dropdown arrow. Select message1 or create a different name of your choice to name the message.
8. Broadcasts work in the code but do not appear on the stage.



9. If you run your code now this broadcast is sent out by the sprite and received by every sprite and the stage. We must now program our second sprite to react once it receives the broadcast.
10. Click the second sprite in the Sprites Area.
11. Pull out the “When I receive message 1” block from the Control Palette or use the dropdown arrow to select the name of your broadcast message. You can then reply and ask something else.



12. You can then add a second broadcast to this piece of code. This will send a broadcast out and the first sprite will receive it



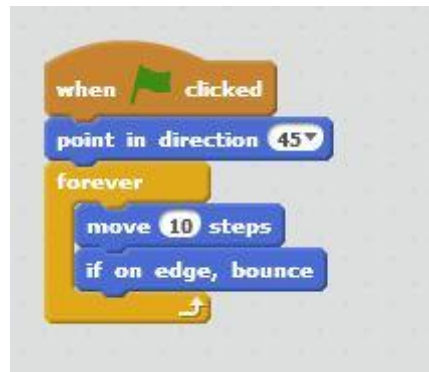
13. Continue in this way until the conversation is complete.

Broadcasts can be used to initiate any action within Scratch. The following activity uses broadcasts while also incorporating the Sensing Palette.

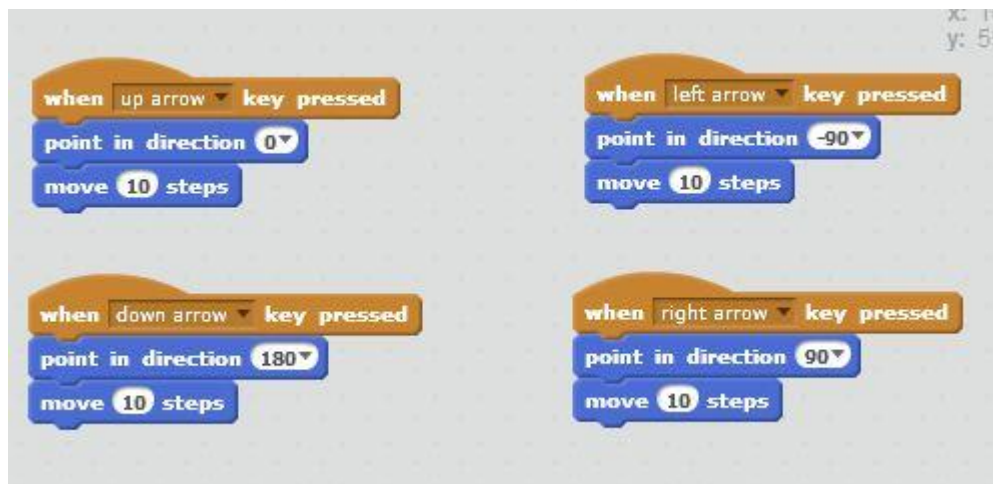
### 3. Gaming in Scratch

This is a simple chase game. One sprite will move randomly around the screen. A second sprite, controlled by the user, will move around the screen trying to avoid the first sprite.

1. Choose a sprite. We will call it The Chaser. Create a script that allows it to move randomly around the screen.

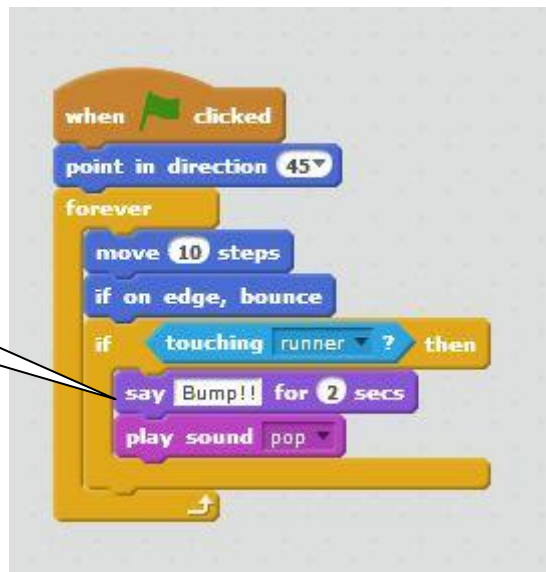


2. Choose a second sprite. We will call it The Runner. Create scripts that allow it to move up, down, left and right.

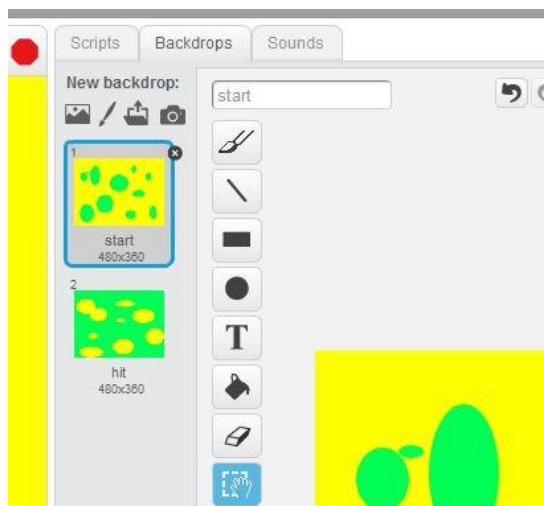


3. We want a reaction when The Chaser catches The Runner. This requires the "if" loop and the Sensing palette.

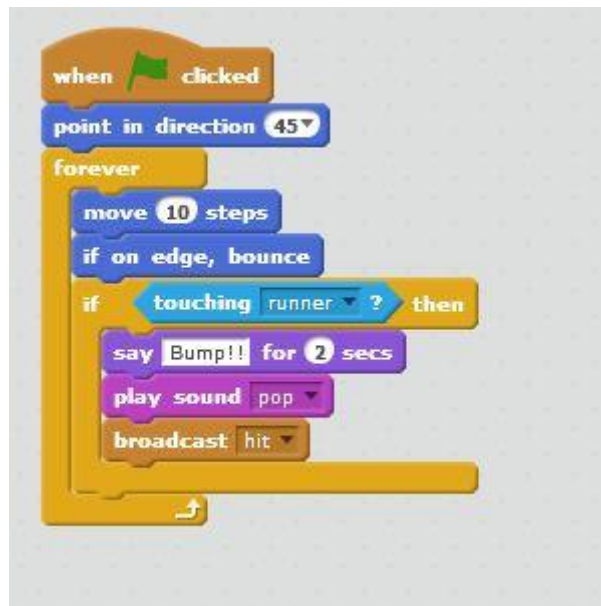
Choose any reaction e.g.  
movement/ sound etc.



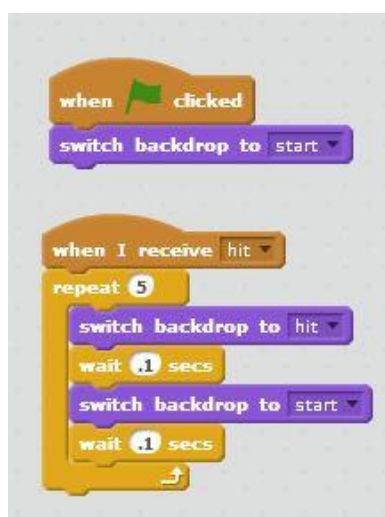
4. Create or import two backdrops for your game. One will show while the game is running. One will show when the sprites are touching and the game ends.



5. The Chaser will send out a broadcast when it touches The Runner.



6. When the stage receives this broadcast it will switch backgrounds.



7. Save this game with a meaningful name.

## Variables

A variable is a value that is not constant, it changes. While variables are only mentioned explicitly in the sixth class Maths Curriculum, they can be seen at younger levels too:

Area = Length\*width ( $a = l \times w$ )

Diameter = 2\*radius ( $d = 2 \times r$ )

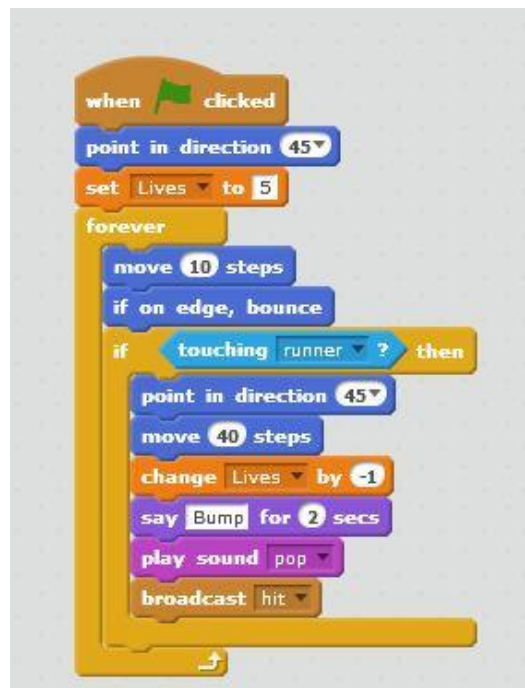
Speed = Distance/Time ( $S = D/T$ )



In Scratch, the children can use simple variables to enhance their games.

### Participant Activity

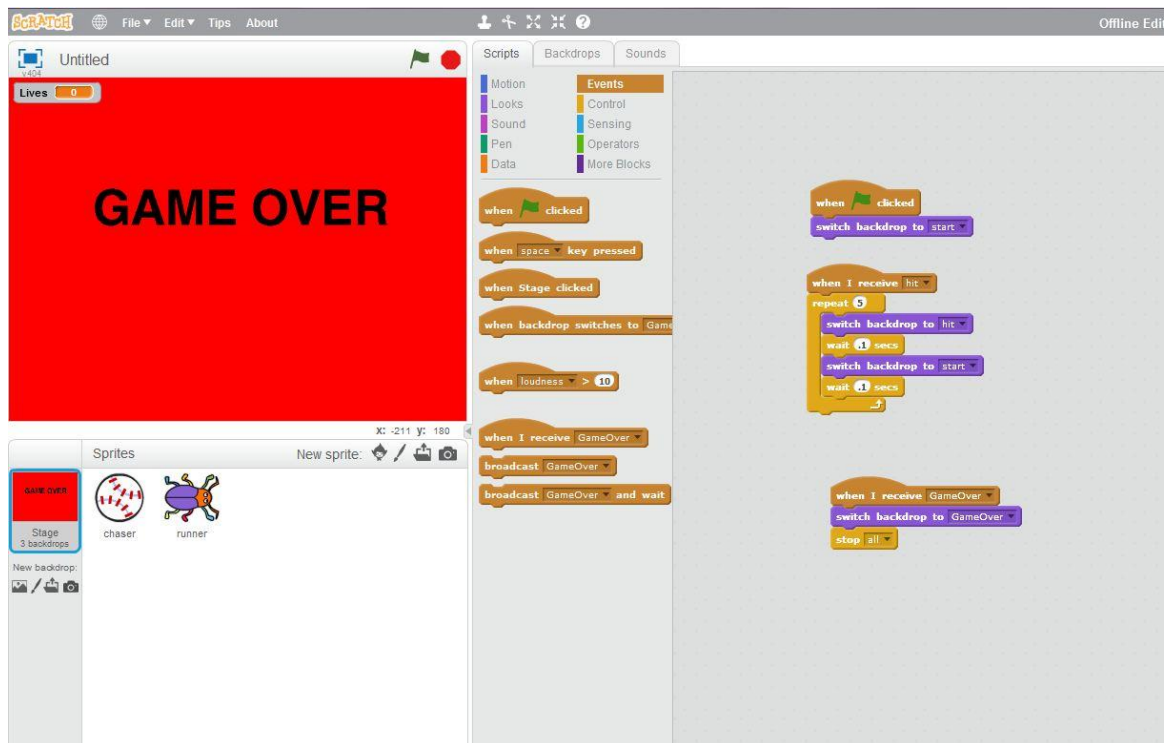
1. Reopen the chase game you created earlier.
2. Click on the Data Palette and select “Make a variable”.
3. Create a variable called “Lives”.
4. We want to start with five lives. Each time the sprites touch we will lose a life.



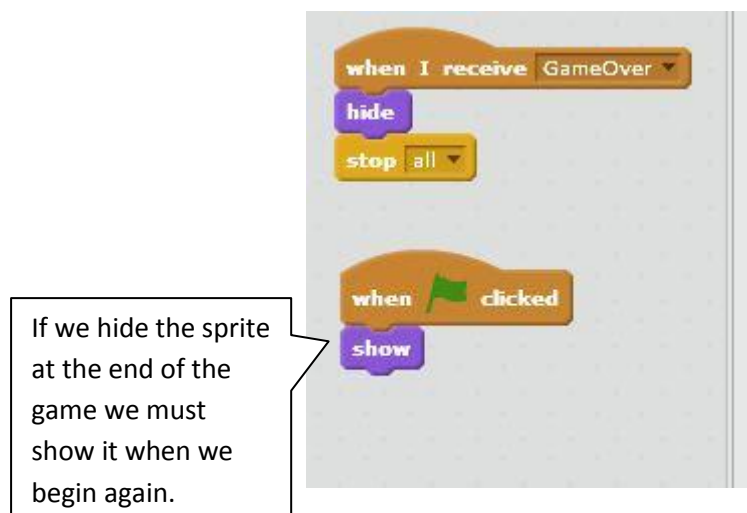
5. What happens when the lives reach 0? We need to create a program to tell the sprites what to do. This will require a broadcast.
6. The piece of code will constantly check the lives. If the lives are 0, it will send out a broadcast. We can call this broadcast Game Over.



7. Create another background that will appear when the game ends.



8. We can hide the sprites at the end of the game.



## Cloning

You may need multiple instances of the same sprite in a Scratch game. It is possible to duplicate a sprite in Scratch using the new Scratch 2.0 feature cloning. You can view the use of the clone function in the Bubbles Video Sensing project <http://scratch.mit.edu/projects/10128119/>. The following describes the cloning blocks.

Tips

Creates a clone (temporary duplicate) of the specified sprite

Choose which sprite to clone from the drop-down menu.

Use to tell the clone what to do once it's created.

Sample Script:

```

when space key pressed
  create clone of myself
  play sound meow
  
```

**Troubleshooting:**  
If you can't see the clone, move it so the original sprite doesn't cover it. The clone appears in the same location as the sprite initially.

Make sure you've chosen the sprite you want to clone from the menu in the block.

Note: The clone only lasts while the project is running.

Tips

Tells a clone what to do once it is created

The script is triggered as soon as the clone is created.

```

when I start as a clone
  move pick random 1 to 10 steps
  say Hello! for 2 secs
  wait 5 secs
  delete this clone
  
```

Clones also respond to all "when..." blocks triggered after they are created.

Deletes the current clone

Place this block in a script after the clone is done its actions.

All clones are automatically deleted when the program is stopped.

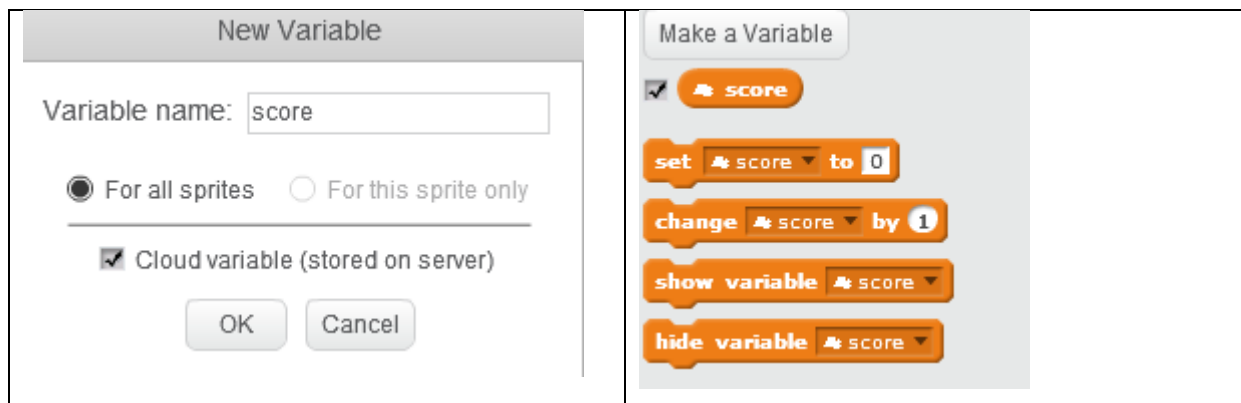
Sample Script:

```

when I start as a clone
  move pick random 1 to 10 steps
  say Hello! for 2 secs
  wait 5 secs
  delete this clone
  
```

## Scores in the Cloud

When working online in Scratch, you have the option to create a variable in the cloud. (denoted by the cloud!) For example, a score cloud variable in your game is shared and seen by everyone running your Scratch project. You could use this to keep a high score for your Scratch game.



## 4. Drawing with Scratch

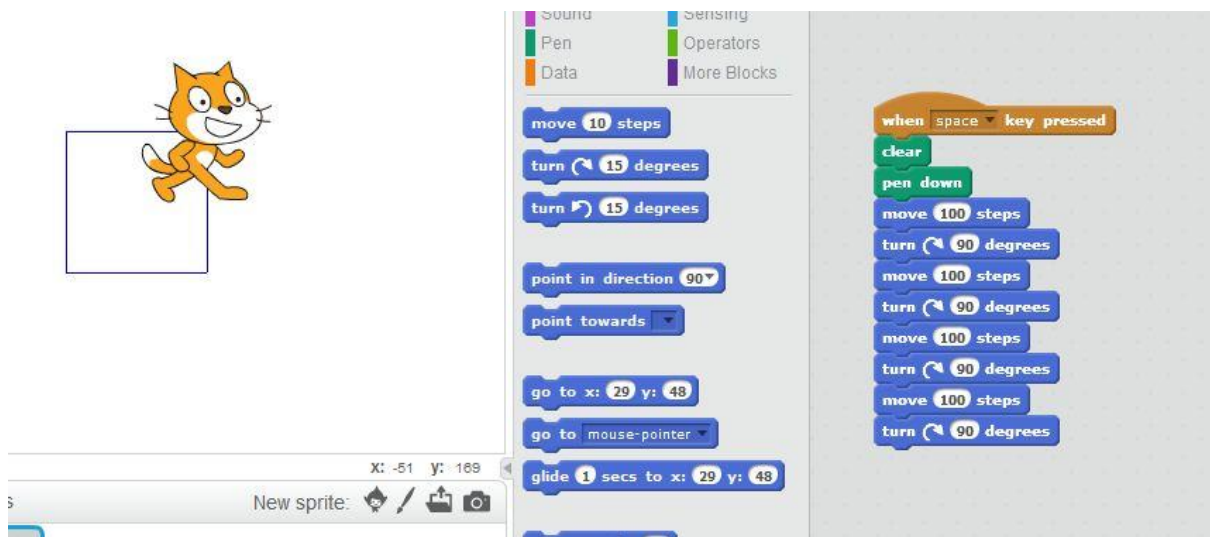
Drawing in Scratch relies on very specific instructions. Scratch drawing works in a similar way to the programming software, Logo.

It is as if your sprite is holding a pencil. Whenever the sprite moves it leaves a line behind it. To draw a shape we must “walk” that shape.

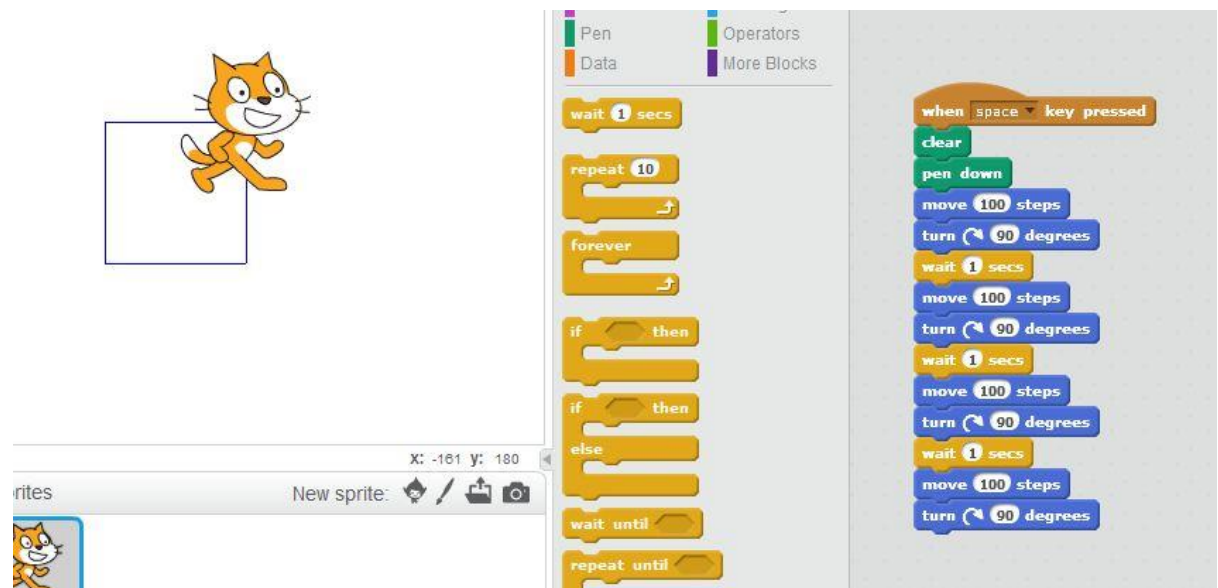
The Pen Palette contains the blocks we will use.

### Participant Activity - Draw a square

1. Use and Events block.
2. Add the “pen down” block.
3. Using the Motion Palette, walk the sprite in a square.



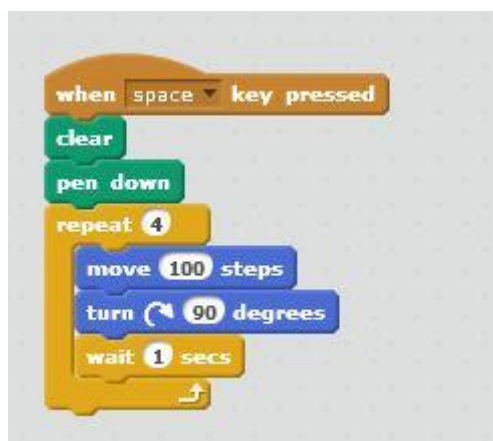
4. Add a wait block to slow down the script.



5. Save this project as “Square”.

6. Create a new project that draws a triangle. Continue with a pentagon and a hexagon.

Did you notice this code is repetitive (Because the movements and turns are all equal)? In Scratch we can use a repeat loop to create a shorter piece of code.



Use the same basic controls to draw a rectangle, triangle, pentagon and other shapes. Can anyone draw a circle? Remember there are 360 degrees in a full circle rotation.

### Participant Activity – 2D shapes

1. Go to [www.scratch.ie](http://www.scratch.ie) or to where you have saved the downloaded lessons.
2. Open Lesson Plan 5 – In the Garden.
3. Complete Challenge Time 1

When the basic shapes are mastered, you can make patterns by using repetitions.

### Participant Activity – Repetition

1. Go to [www.scratch.ie](http://www.scratch.ie) or to where you have saved the downloaded lessons.
2. Open Lesson Plan 5 – In the Garden.
3. Complete Challenge Time 2. If time allows continue to the Ultimate Challenge.

## 5. Discussion Time

Today's objectives were to cover:

1. Exploring numeracy in Scratch
2. Communication in Scratch
3. Gaming in Scratch
4. Drawing with Scratch
5. Discussion Time

Discuss the day with your colleagues and reflect on what you have learned.

Discuss aspects of Scratch that are relevant to numeracy. What could you use in your classroom if you were teaching tomorrow?

## Module 3

---

SCRATCH





## Module 3

### **Suggested Duration of Module**

4 Hours

### **Objectives**

Module 3 will cover:

- 1. Exploring numeracy in Scratch**
- 2. Problem Solving**
- 3. Images and Video**
- 4. Discussion Time**

## 1. Exploring Numeracy in Scratch

In this section, we will cover more features in Scratch that can be used to create Scratch numeracy related projects. These include variables and operators, timers and maths functions.

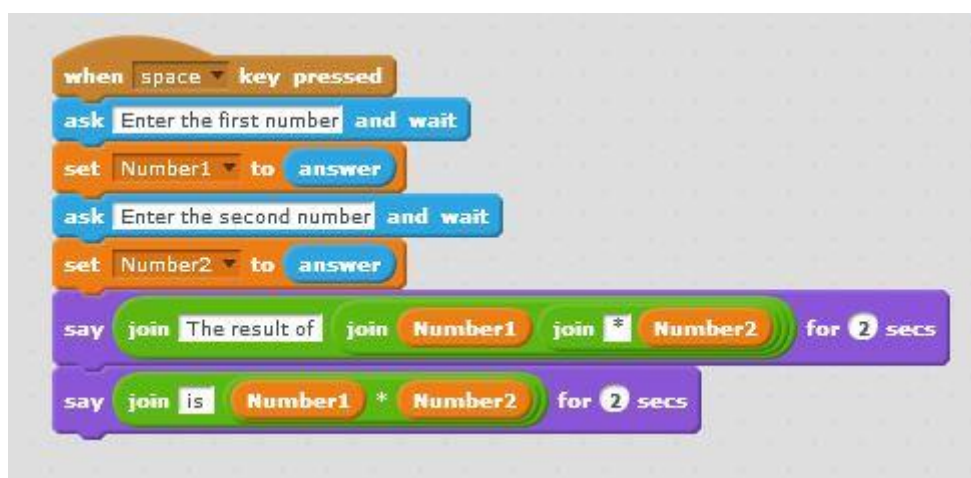
### Variables and the Operators

Variables can be used in conjunction with the Operator's Palette for specific maths tasks.

The Operator's palette in Scratch is very useful and we can immediately see applications for Mathematics tasks. The various blocks will instruct the program to carry out addition, subtraction, multiplication, division and rounding.

### Participant Activity – Find the Product of two numbers

1. Create two variables where we will store numbers inputted by the user. Here we have named them "Number 1" and "Number 2".
2. Use the Sensing Palette to ask the user to type a number. The blue question and blue answer enables people to add user interaction to their projects.
3. These numbers will be saved as variables.
4. The green Operator's Palette will allow us to multiply the two numbers together.



5. Alter the above code to make addition, subtraction and division tasks.

6. Modify the code to allow the user to choose what type of operation is performed.
7. Take note of the nested join statements and experiment with them.
8. Save your work.

### Participant Activity – Find the Area of a Shape

This lesson uses Scratch to calculate the area of a square or rectangle.

1. Create 2 variables, one called Length, the other Width.
2. Allow the user to enter the values for length and width. You will use the blue “ask” and “answer” block from the sensing palette.
3. Use the Operator’s palette to find the area.
4. Output the answer onto the screen.

The image shows a Scratch script for calculating the area of a rectangle. The script consists of the following blocks:

- when green flag clicked** (orange block)
- set Length to 0** (orange block)
- set Width to 0** (orange block)
- ask "What is the length of the rectangle?" and wait** (blue block)
- set Length to answer** (orange block)
- ask "What is the width of the rectangle?" and wait** (blue block)
- set Width to answer** (orange block)
- say join "Area = " Length \* Width for 5 secs** (purple block)

Four callout boxes provide instructions for each step:

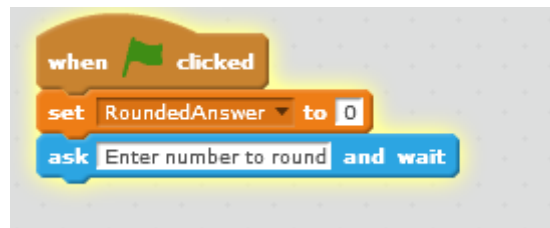
- 1. Initialise length and width variables to 0.** (points to the 'set Length to 0' and 'set Width to 0' blocks)
- 2. Use the sensing palette to ask the user for the length and width.** (points to the 'ask' blocks)
- 3. Set the length and width variable to these new values.** (points to the 'set Length to answer' and 'set Width to answer' blocks)
- 4. This piece of code calculates the area and shows it on the screen.** (points to the 'say' block)

### Participant Activity – Rounding Numbers

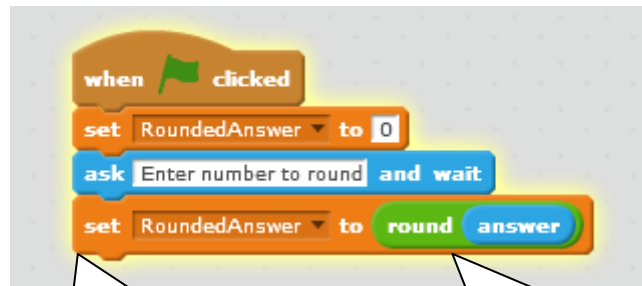
1. Click on stage and then click on backdrops.
2. Choose a backdrop from the library.
3. Select a sprite. You may delete the cat if you use another sprite. This sprite will ask the user to input a decimal and will give the rounded answer.
4. The script will use a variable and operators. We will call the variable “RoundedAnswer” as this is what we are looking for. The first block, “set

RoundedAnswer to 0” will ensure that, each time we run the script, the initial value will be 0.

5. The sprite must ask the user to input a decimal.



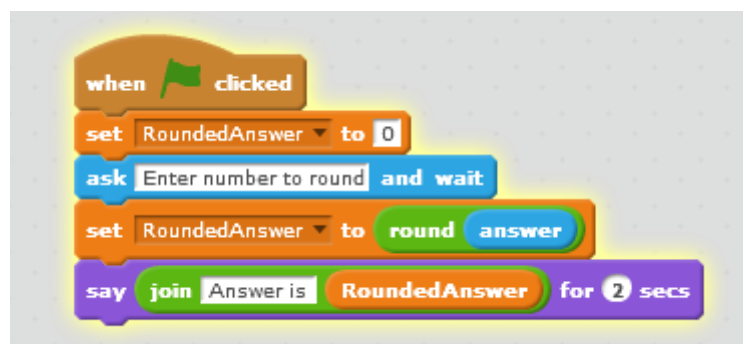
6. The blue block asks a question of the user and the response is saved in the blue “answer” block. This answer can then be used in various calculations. In this case we will round it to the nearest whole number and output the solution as the variable “RoundedAnswer”.



This block will change the variable from 0 to the solution.

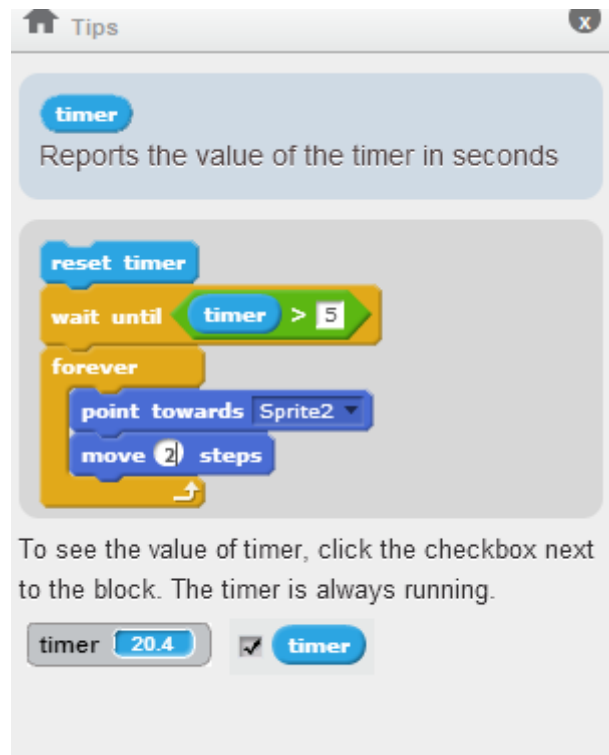
This operator will calculate the closest whole number to the inputted decimal.

7. We must now display the solution on the stage.



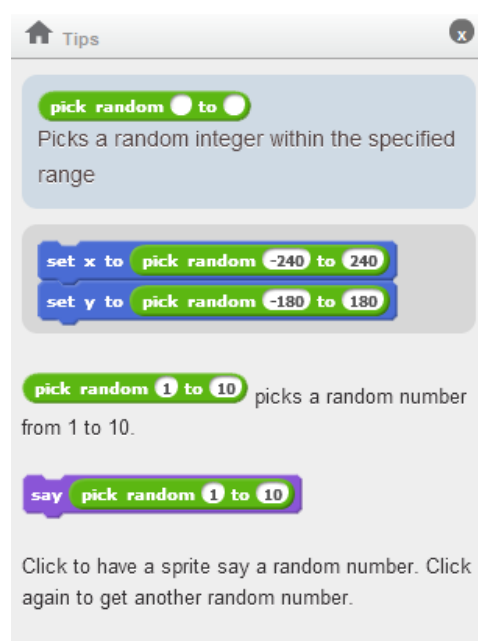
## Timers in Scratch

A timer can be a useful tool in Scratch allowing the user to complete a particular action within a specified time period for example in games or quizzes.



## Random numbers

Random numbers can be used in Scratch projects to reposition sprites in random locations, to assign random numbers of lives,



## **Participant Activity – Use Scratch to demonstrate the solution of a word problem**

In Microsoft Word or on a piece of paper, create a word problem which is suitable for your class level. Ask the person/group beside you to solve this word problem in Scratch. They may create an animation that shows the solution or a game that demonstrates understanding of the Maths concept.

### **2. Problem Solving**

Children will be problem solving from the moment they start using Scratch. They will plan, execute and debug their programs. You may also wish to use Scratch to support more formal mathematical problem solving.

Word problems feature frequently in Maths textbooks. They can be the most challenging aspect of any topic. However, by allowing children to use Scratch we can bring the problem to life. View the following examples.

Q1. There are 12 apples in one box. How many apples are there in 6 boxes?

A1. <http://scratch.mit.edu/projects/ballns/2524268>

Q2. The monkey has 16 bananas. He eats 7. How many bananas are left?

A2. <http://scratch.mit.edu/projects/ballns/2553431>

There are some great examples of Scratch projects created by the students using their problem solving skills and numeracy. See the following projects that progressed to the Scratch competition national finals.

“So you think you can count” <http://scratch.mit.edu/projects/19247368/>

“Lámh Lámh Eile” <http://scratch.mit.edu/projects/19325973/>

All projects that progressed to the finals of the national scratch competition are available here: <http://scratch.ie/competition>

### 3. Images and Video

Scratch projects may use images and video to increase the engagement factor and to allow students to personalise their projects.

#### The use of images in Scratch

In order to make Scratch projects personally meaningful to children, it may help to use a greater range of images. These may be downloaded from the Internet, uploaded from external devices such as digital cameras, or taken from files already saved on the computer.

A great source of images which are available to use free of charge and under creative commons licensing is [www.imagebank.ie](http://www.imagebank.ie).

#### [Imagebank.ie](http://www.imagebank.ie)

ImageBank is a photo library website where people can search for and share photos for educational use. ImageBank focuses on providing photos of Irish places and spaces but other subjects are also included, such as animals, sports and flowers.

ImageBank is intended as a supportive resource for students and teachers in Irish education. More importantly, it is a growing resource and users are encouraged to share any relevant photos. Creative Commons licensing is used to allow you share your photos and specify how others may use them.

**NB:** Searches in Scoilnet find images from Imagebank - just enter your search term in the Scoilnet search box and click on the Images tab in the results.

#### Participant Activity – Downloading images from the Internet

1. Create a folder on your desktop.
2. Label it “Scratch Resources”. This is where you will save all images and sounds that you intend to use in Scratch.
3. Find an image you wish to use on [www.imagebank.ie](http://www.imagebank.ie) or [www.scoilnet.ie](http://www.scoilnet.ie)
4. Save the image to your new folder, giving it a meaningful name.

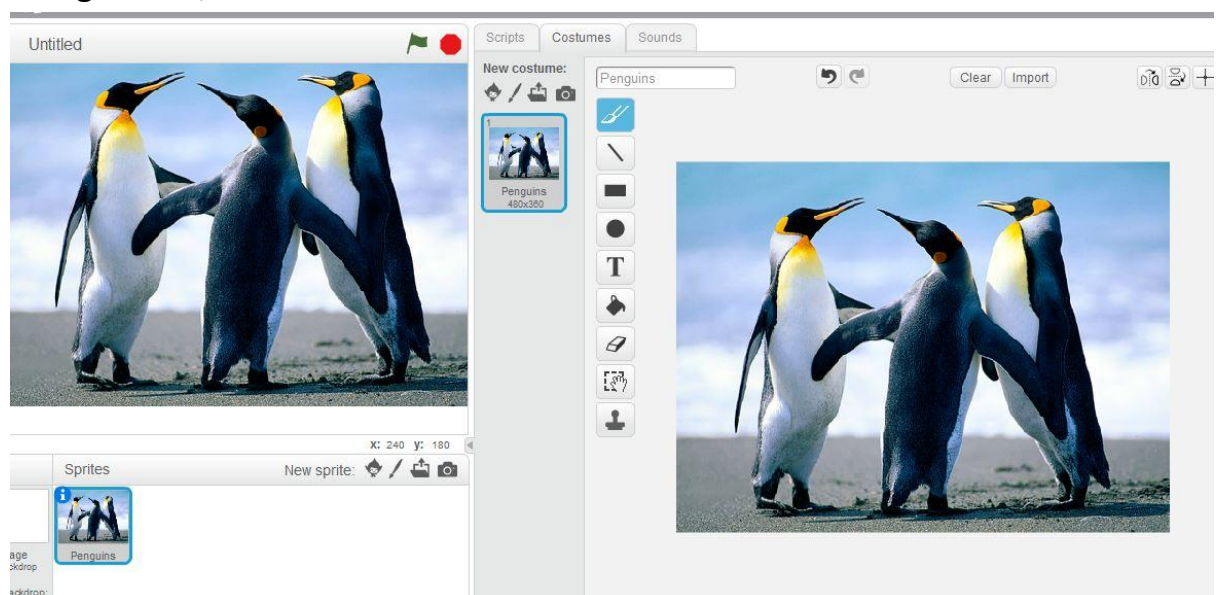
#### Participant Activity - Save an image from a digital camera

1. Find an image you wish to use on the digital camera.
2. Open the “Scratch Resources” folder.

3. Drag the image into the “Scratch Resources” folder.
4. Once the image is saved to the folder you can open it in Scratch.

### Participant Activity – Using Images in Scratch

1. Open Scratch interface.
2. Delete the cat sprite.
3. Click on “Upload sprite from file”.
4. Navigate to the desktop.
5. Choose the folder you created earlier, “Scratch Resources”.
6. Select the photo you wish to use.
7. When you import a photograph it may contain other images that you do not want, e.g., a white background that has to be deleted.
8. Click on the photograph in the sprites area.
9. Click on the costumes tab
10. Click on edit. This will open the paint editor where you can delete backgrounds, alter colours and size and add text.



11. The paint editor in Scratch 2.0 uses vector graphics and bitmap graphics. For full details on the Scratch Paint Editor see:  
[http://wiki.scratch.mit.edu/wiki/Paint\\_Editor](http://wiki.scratch.mit.edu/wiki/Paint_Editor)



### **Participant Activity – Create a Scratch Project that Incorporates Images**

1. Working in pairs with the digital camera, spend approximately 15 or 20 minutes taking some images within the room or outside.
2. Incorporate sounds and images into a numeracy focused Scratch project.

### **Participant Activity – Upload an image to ImageBank**

ImageBank is a store of images for teachers, by teachers. Teachers are encouraged to upload any suitable images they may have.

1. Select one of the images you have on your digital camera.
2. Ensure it has a relevant and accurate title.
3. Go to [www.imagebank.ie](http://www.imagebank.ie).
4. Click on “Add a photo”.
5. You will be asked to create a password.
6. The terms and conditions outline that you must be the copyright holder of any images you add.
7. Click on “Browse” to find a photo on your computer.
8. Add photo.

### **Participant Activity - Images in Scratch**

1. Go to [www.scratch.ie](http://www.scratch.ie) or to where you have saved the downloaded lessons.
2. Open Lesson Plan 4 – A Cartoon About Me!
3. Complete the Ultimate Challenge.

### **Video in Scratch**

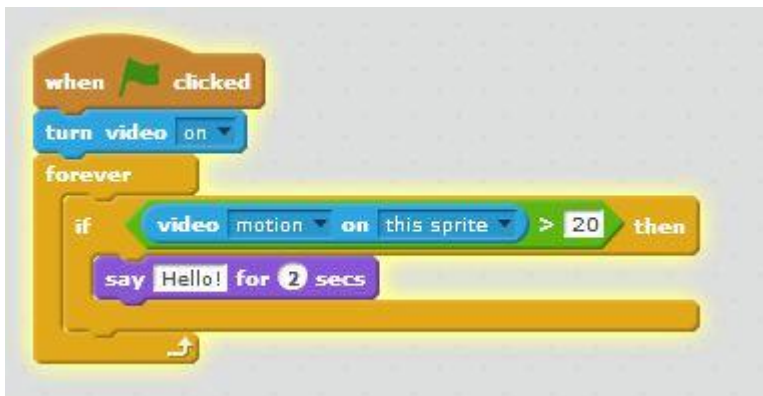
The new Video feature in Scratch 2.0 is definitely worth checking out and introducing to your students. Browse the Video Sensing projects on [http://scratch.mit.edu/starter\\_projects/](http://scratch.mit.edu/starter_projects/) to experience the video capabilities. You will need a webcam on your machine to explore video.

To get started with video, get a sprite on the Stage and turn on the video.

Once you turn on video, you will be asked to Allow Camera and Microphone Access. Click “Allow”.



Check the video motion on the sprite and perform an action if motion is detected on the sprite. (In this case we say Hello)



You can check the value of the video motion by checking the “video motion on this sprite” box so it displays on the screen.

### **Participant Activity – Explore Video Functionality**

Working in groups, build a Scratch project using video blocks.

### **4. Discussion Time**

Today’s objectives were to cover:

1. Exploring numeracy in Scratch
2. Problem Solving
3. Images and Video
4. Discussion Time

Discuss the day with your colleagues and reflect on what you have learned. Discuss aspects of Scratch that are relevant to numeracy. What could you use in your classroom if you were teaching tomorrow? Discuss how devices such

as Lego WeDo kits can be connected to Scratch. Does anyone in the class have experiences to share? Does anyone have anything to demonstrate to the group?

## Module 4

---

SCRATCH



## Module 4

### **Suggested Duration of Module**

4 Hours

### **Objectives**

Module 4 will cover:

- 1. Internet Safety and Cyberbullying**
- 2. Scoilnet, Scoilnet Maps and Encyclopaedia Britannica**
- 3. Quizzes in Scratch**
- 4. Discussion Time**

## **1. Internet Safety and Cyberbullying**

The purpose of this section is to provide an overview of Internet safety and its importance in the classroom context. It also looks at the issue of cyber bullying.

### **Webwise**

- The Webwise Primary School Programme has been developed for primary school teachers who wish to introduce internet safety into their teaching of the Social Personal and Health Education (SPHE) curriculum.
- The first part of the resource focuses on skills needed for surfing the web such as effective and safe searching, downloading images and determining what online content can be trusted.
- The second section deals with the skills required to safely and effectively communicate online or by text message. It deals with issues relating to sharing personal information online, treating others with respect, cyberbullying, responding to media, in particular digital media and dealing with spam.
- The Webwise programme utilises a range of teaching methodologies with particular emphasis on active learning, the principal learning and teaching approach recommended for SPHE.
- It provides opportunities for cross-curricula integration in particular with Drama, Language and Visual Arts.
- It is made up of paper-based classroom activities and digital interactive lessons. The interactive cartoons are designed to be used as a whole-class activity using a whiteboard or digital projector. The activities may also be set up for a pair, an individual, or a small group to use at a classroom computer.

### **Useful Links**

- Webwise  
<http://www.webwise.ie>
- Internet Safety Advice Sheet  
<http://www.webwise.ie/InternetSafety.pdf>
- Facebook Tips for Teachers

<http://www.webwise.ie/FacebookTipsForTeachers.pdf>

- Common Sense Media  
<http://www.commonsensemedia.org/educators/cyberbullying-toolkit>
- Stop Cyberbullying  
<http://www.stopcyberbullying.org>
- Stamp out Cyber bullying  
[http://www.stpaulsbessbrook.org/students/stamp-out-cyber-bullying/?utm\\_source=twitterfeed&utm\\_medium=twitter](http://www.stpaulsbessbrook.org/students/stamp-out-cyber-bullying/?utm_source=twitterfeed&utm_medium=twitter)
- Archdiocese of Dublin Education Secretariat  
<http://education.dublindiocese.ie/safer-internet-day/>
- Beat Bullying  
<http://www.beatbullying.org/>
- Advice and Guidance on Cyberbullying  
<http://www.cybersmile.org/advice-help>
- Basic Internet Safety  
<http://www.netsmartz.org/InternetSafety>
- Get with it – A Guide to Cyberbullying  
<http://www.hotline.ie/documents/Cyberbullying.pdf>
- Cyberbullying  
<http://childwatch.ie/CyberBullying.php>

The Scratch Community Guidelines for the scratch.mit.edu website and community can be viewed here:

[http://scratch.mit.edu/community\\_guidelines/](http://scratch.mit.edu/community_guidelines/)

## **2.Scoilnet, Scoilnet Maps and Encyclopaedia Britannica**

Tutors will introduce some of free ICT in education resources available, which may be useful in the context of Scratch and are developed or supported by PDST Technology in Education.

Scoilnet <https://www.scoilnet.ie/>

Scoilnet Maps <http://maps.scoilnet.ie/>

Encyclopaedia Britannica <http://www.britannica.com/>

### 3. Quizzes in Scratch

Quizzes are a useful way of integrating Scratch across the curriculum. For example, when pupils finish a task in History ask them to create a quiz for their peers. While developing the children's coding skills, this task also demonstrates their knowledge of the History topic. Please check out this example quiz at <http://scratch.mit.edu/projects/18742762>.

#### Participant Activity

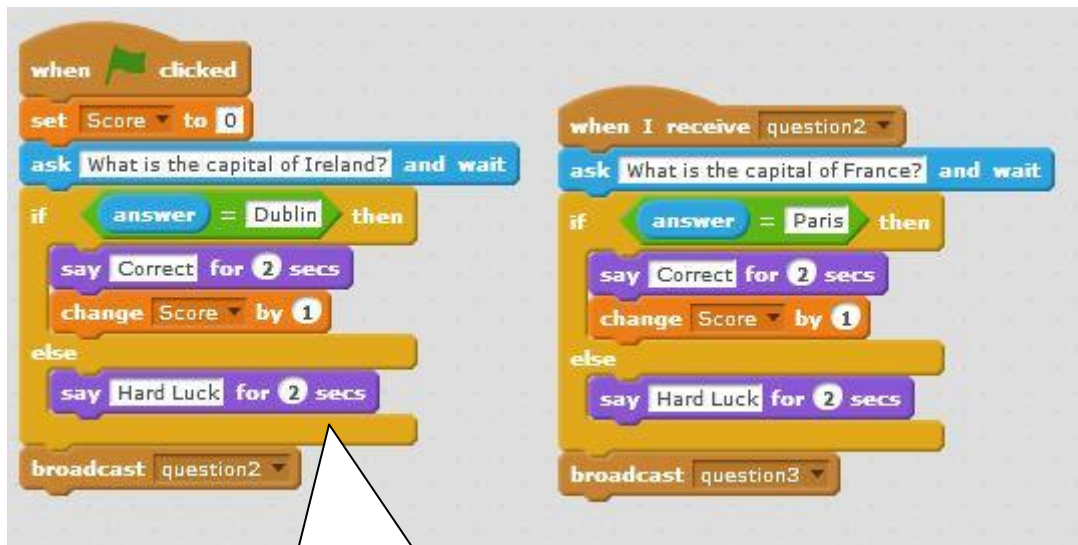
1. Decide on a topic for your quiz. This example is based on capital cities.
2. Select a sprite as your quizmaster.
3. Choose a background.
4. Because the quiz requires user input, we must use the blue "ask" block and the blue "answer" block. These are in the Sensing Palette.
5. When the code runs it should ask the question and wait for the answer. It should then react to signify if the answer is correct or incorrect.

Variable can be omitted for younger children.



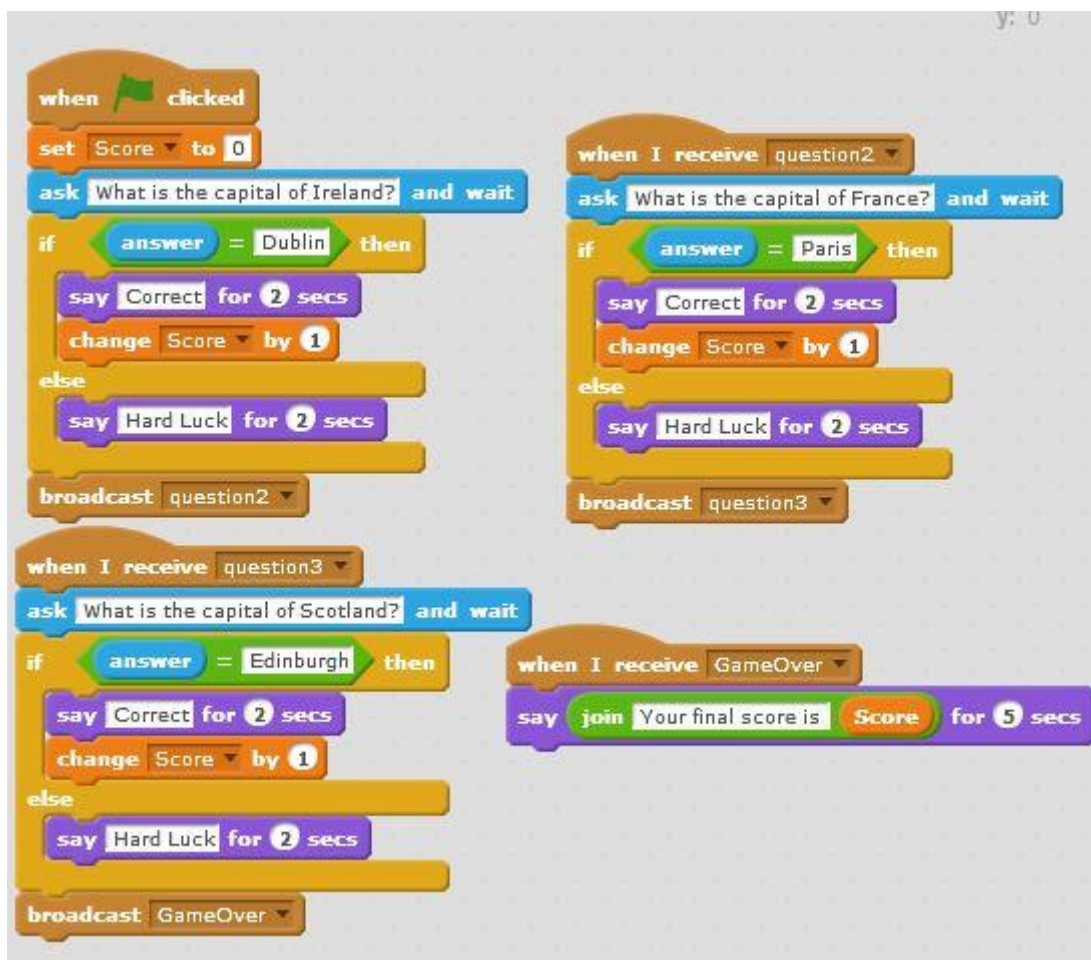
6. The code for any further questions follows the same pattern.





Right click on the code and select duplicate. You can then edit the sections that are different in each question.

7. The quiz can have any amount of questions. To end the game, you can display the user's score or you could change backdrop.



8. Increase the challenge by repeating questions that are answered incorrectly. Add in sound effects when questions are answered. Change backdrops for each question.

## Lists in Scratch

A list or array is a way of storing several variables. It can be used in many ways but here we will use it to create a quiz. Lists can be quite challenging and therefore more suited to older classes.

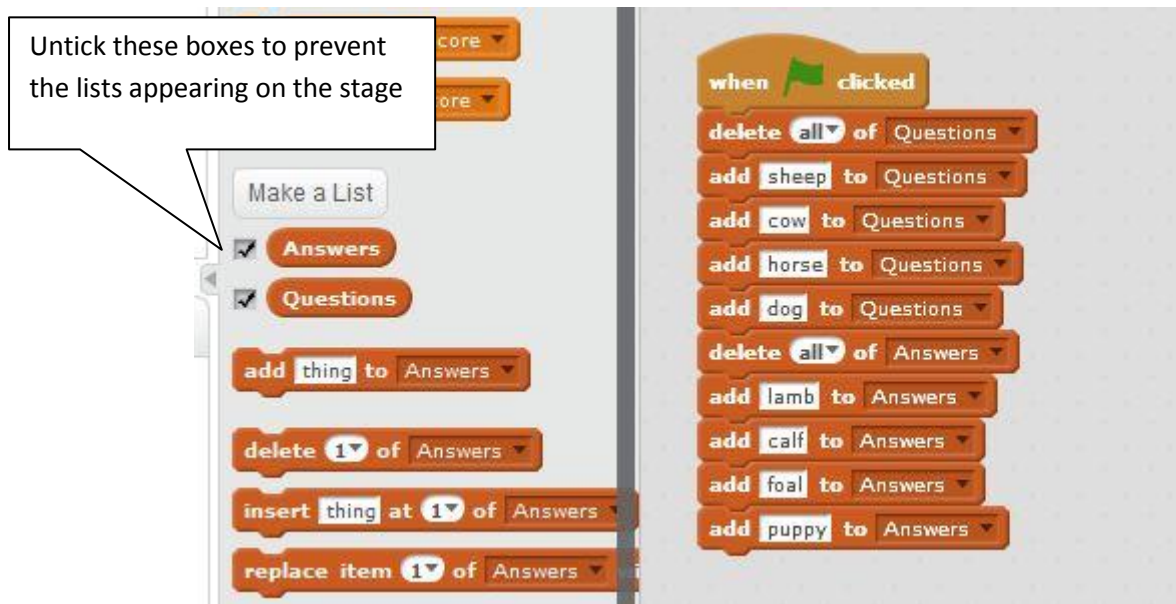
### Participant Activity

1. Select a sprite
2. Select a topic for your quiz e.g. young animals. Decide on your questions. It is better if your question has a one word answer as users must get it exactly correct.
3. Create a list. Give it a meaningful name e.g. Questions.
4. For the list just enter the part of the sentence that changes. For example, instead of entering “What is a young sheep called?”, “What is a young cow called?” simply enter “sheep”, “cow”.

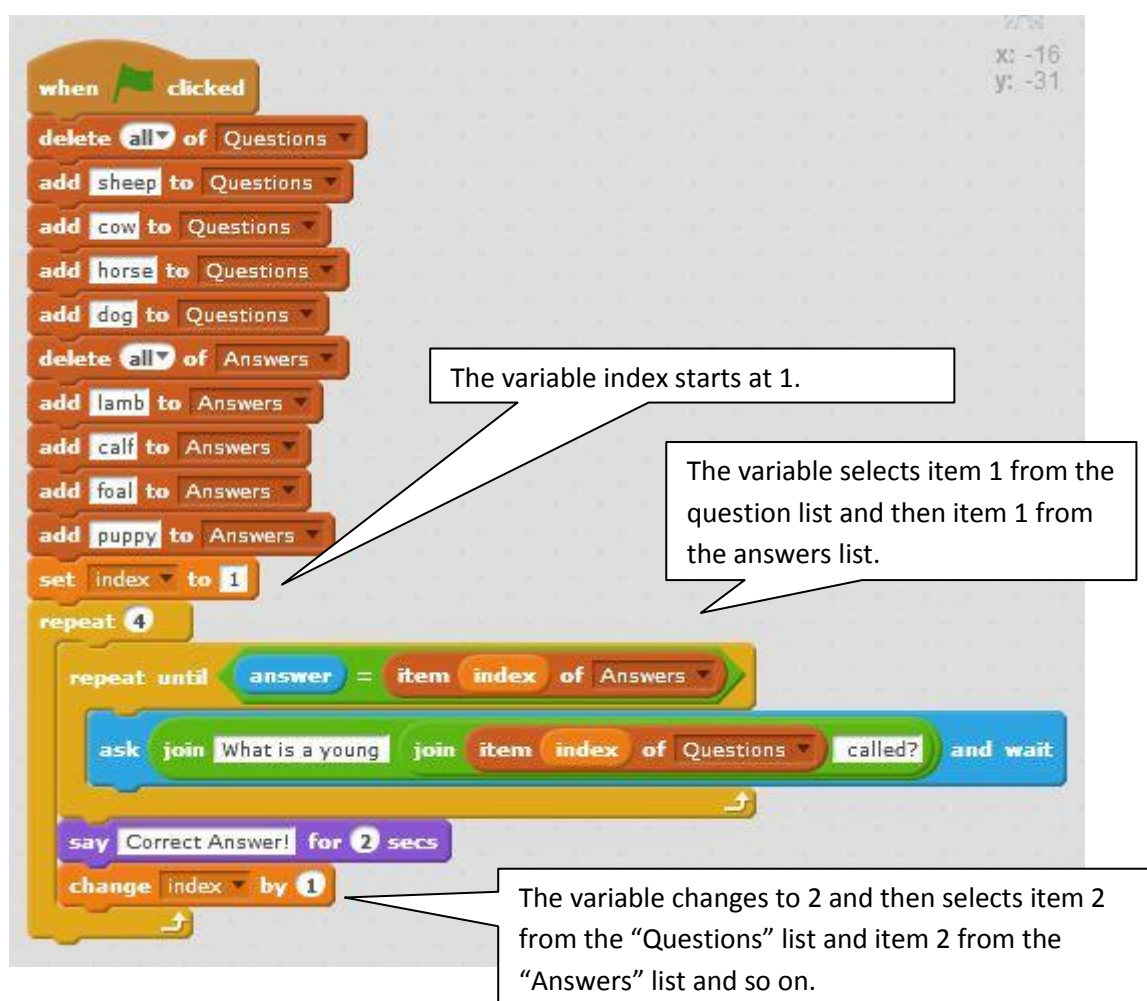
Each time you start the program it will add all the items to your list. This block deletes previous questions so that you don't have duplicate



5. Create a second list called Answers. Enter the solutions to your questions. It is very important that the order of the two lists matches e.g. if sheep is your first question, lamb must be your first answer.



6. Now we must code to ask the question and to ensure the program associates the question with its correct answer. We will create a variable index used to access particular elements in the list.



## Build your own programming blocks


In Scratch 2.0 it is possible to build your own programming blocks. You can create a stand alone block or a block that takes inputs.

### Participant Activity

Time permitting create a simple block or view one of your existing projects and see if it is possible to make your code more efficient by defining your own block.


**Make a Block**  
Creates a custom block

Click "Make a Block" to get this dialog box:




Click inside the block to edit its name.


When you click OK, the new block will appear in **More Blocks**:




A **define** block will appear in Scripts. Use **define** to tell the custom block what to do:



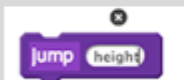
You can create a custom block with inputs, for example:




Click "Make a Block" to see this dialog box:



Click **Options** to add inputs. For example, add a number input:



In the **define** block, inputs will appear as rounded, light purple blocks. To use them, drag copies of the input blocks into other blocks within the define script.



Note: The input blocks can **only** be used in the define script.

#### **4.Discussion time**

Today's objective was to cover:

1. Internet Safety and Cyberbullying
2. Scoilnet, Scoilnet Maps and Encyclopaedia Britannica
3. Quizzes in Scratch
4. Discussion Time

## Module 5

---

SCRATCH



## Module 5

### **Suggested Duration of Module**

4 Hours

### **Objectives**

Content of Module 5:

- 1. Create a Scratch Project**
- 2. Support for Teaching Scratch**
- 3. Other ICT Resources from PDST Technology in Education**
- 4. e-Learning Action Plan**



## **1.Create a Scratch Project**

1. Break into groups of 2 or 3.
2. Plan out a project.
3. Build it in Scratch.
4. Be as creative as you can. Use the resources you have seen throughout the week.
5. Incorporate the following into your project:
  - i. At least two sprites
  - ii. Costume or Background Change
  - iii. Broadcast
  - iv. Variable
  - v. Images and sounds
  - vi. Numeracy aspect
6. Upload your project to [www.scratch.mit.edu](http://www.scratch.mit.edu).

## **Present your project to the Group**

Present your project to the group.

Get feedback on your project.

## **Implement a new and improved version of your project**

Create a version 2 of your Scratch project by adding some new features or functionality.

## **2.Support for Teaching Scratch**

### **ScratchEd**

<http://scratched.media.mit.edu/>

Launched in July 2009, ScratchEd is an online community where Scratch educators share stories, exchange resources, ask questions, and find people. Since its launch, more than 7,500 educators from all around the world have joined the community, sharing hundreds of resources and engaging in thousands of discussions.



### **Participant Activity**

In pairs or groups go to <http://scratched.media.mit.edu/> and find some resources suitable for your class. Discuss any other Scratch resources you find useful.

### **Discuss Scratch**

<http://scratch.mit.edu/discuss/>

Discuss Scratch is a forum section within [www.scratch.mit.edu](http://www.scratch.mit.edu). There are hundreds of discussions where you can find solutions to your Scratch questions or you can also start your own discussion. The discussions are categorised into “Welcome to Scratch”, “Making Scratch Projects”, “About Scratch”, “Interests Beyond Scratch” and “Scratch in Other Languages”.

### **Participant Activity**

1. Consider an aspect of Scratch you found particularly challenging this week.
2. Search <http://scratch.mit.edu/discuss/> for help with this topic.
3. Create a small project that reinforces your learning on this topic e.g. if you found costume changes difficult create a project with one sprite who changes costume.

## **3.Other ICT Resources from PDST Technology in Education**

Tutors will introduce some additional free ICT in education resources available, which may be useful in the context of Scratch. These include the PDST website which also has resources related to the next topic in this module, i.e. e-Learning Planning. Participant materials in support of these topics are available at <http://www.pdsttechnologyineducation.ie/en/Training/Training-Materials/>

## **4.e-Learning Action Plan**

The e-Learning Plan area of the PDST Technology in Education website contains a number of useful resources to assist schools in developing their e-Learning Plan, from the NCTE’s e-Learning Handbook and Roadmap to case studies and

video exemplars highlighting how teachers are integrating ICT in their classrooms. <http://www.pdsttechnologyineducation.ie/en/Planning/>

The purpose of this section is to provide an overview of the benefits of planning for e-Learning in the primary classroom. It also includes links to a number of useful resources to assist schools in developing their e-Learning plan.

### **Benefits**

The benefits of using a school e-Learning plan include:

- Assessing where the school is in terms of ICT development and where it would like to be
- Identifying the ICT equipment, software and skills available in the school
- Ensuring an agreed vision for ICT within the school aligned with the Whole School plan
- Allowing the school to integrate ICT more effectively into their teaching and learning activities based on the particular circumstances of the school
- Identifying SMART (Specific, Measurable, Achievable, Relevant, Timebound) tasks to integrate ICT more effectively into teaching and learning

### **Participant Activity**

- Using the template provided, develop an e-Learning plan for your class for the first term of the school year.

### **Tips**

- Aim to ensure that the e-Learning team includes a range of ICT skills and proficiency reflective of the staff as a whole.
- Be realistic. Look at what task will be attainable by the staff as a whole. Once this has been achieved, you can start on the next goal. The journey of a thousand miles starts with a single step!

### **Useful Links**

- PDST e-Learning Plan  
[www.pdsttechnologyineducation.ie/en/Planning/](http://www.pdsttechnologyineducation.ie/en/Planning/)
- PDST e-Learning handbook  
<http://www.pdsttechnologyineducation.ie/en/Planning/e-Learning-Handbook/>
- PDST e-Learning Roadmap

<http://www.pdsttechnologyineducation.ie/en/Planning/e-Learning-Roadmap>

- PDST eLearning Plan templates  
<http://www.pdsttechnologyineducation.ie/en/Planning/e-Learning-Plan-Templates/>
- PDST e-Learning Case Studies  
<http://www.pdsttechnologyineducation.ie/en/Planning/e-Learning-Case-Studies/>
- PDST CPD information  
<http://www.pdsttechnologyineducation.ie/en/Training/>

Today's objective was to cover:

1. Create a Scratch Project
2. Support for Teaching Scratch
3. Other ICT Resources from PDST Technology in Education
4. e-Learning Action Plan