

Module 1

Getting Started

Lesson 1 Computers Rule the World

Discussion about the importance of computer programming in our everyday lives. An answer is provided to the question “What is Scratch?”



Projector



CD Resource

Lesson 2 Say “Hello” to Scratch

Introduction to the look and feel of Scratch. Write your first script (computer program) using Scratch.



Computer



Scratch



Projector



Printable Resource

Lesson 3 Playing with Pictures

Change the appearance of Scratch characters and background images.



Computer



Scratch



Internet



Projector

Lesson 4 Tell me what to do

Computers need instructions to tell them what to do. This activity investigates how to provide effective instructions to solve everyday problems.



Projector



CD Resource



Printable Resource



Group Work

Lesson 5 Playing with Music

Now it's time to start interacting with the characters. A sprite will be programmed to move in different directions and play music using keyboard controls.



Computer



Scratch



Projector



Sound



Lesson 1 – Computers Rule the World

Resources:

How is software relevant to my life? (Resource 1), Mitchel Resnick (Director of MIT Media Lab) Scratch Video (Resource 2), Introduction to Scratch Video (Resource 3)

Key Vocabulary:

Programming, Scratch, Software

Description:

The PowerPoint Karaoke activity is used to stimulate a discussion about the impact of software on our everyday lives. Scratch will be introduced with the aid of video clips produced by the MIT Media Lab.

Learning Objectives:

1. To learn about and discuss the importance of software in modern society.
2. To introduce the Scratch software.

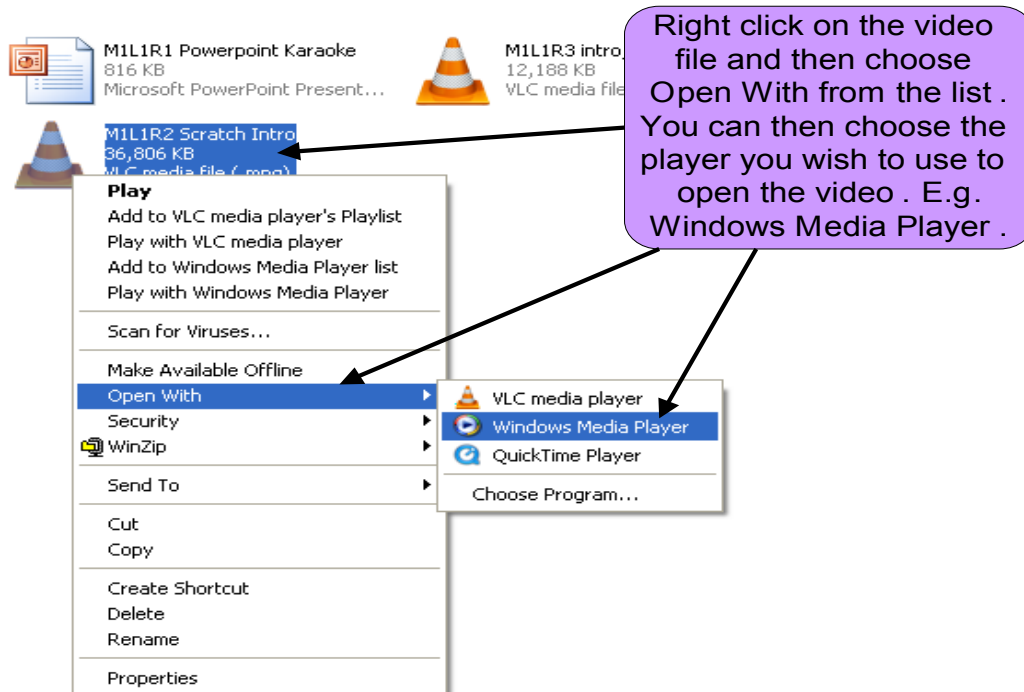
Lesson Introduction:

- The lesson will show how important software programs are in our everyday lives.
- Scratch will be introduced as a programming language that makes it easy to create interactive stories, games, animations, music and art.

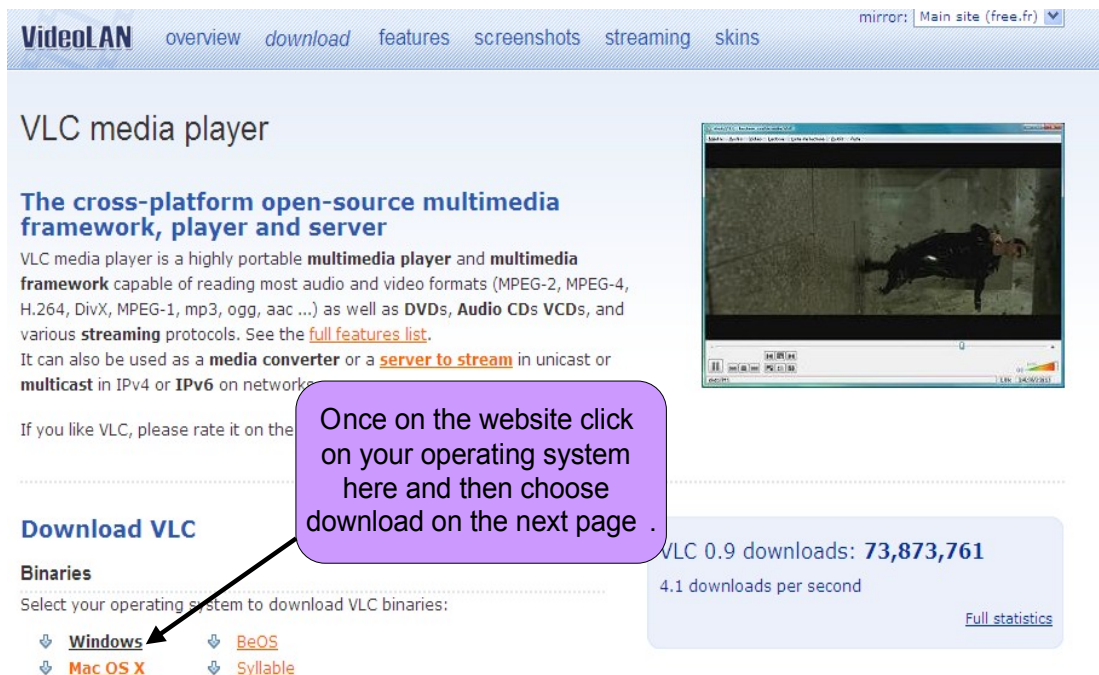
Lesson Breakdown:

1. Start the PowerPoint presentation (Resource 1). Students are asked to volunteer to speak for at least 30 seconds on a slide that contains 4 pictures. They can be helped and encouraged by the audience to talk about anything they feel is relevant to the pictures. There are 5 slides.
2. The title of the presentation “How is software relevant to my life?” is then revealed. Pupils are asked to give feedback on the pictures, taking the title of the slideshow into consideration. Some interesting facts about the importance of software in society today appear as the second run of the slideshow is presented.
3. Scratch is then introduced as the tool that students will use to learn software programming.
4. 2 Video clips, Mitchel Resnick Scratch Video (Resource 2) and Introduction to Scratch Video (Resource 3), produced by the MIT Media Lab, are shown to students to support the introduction of Scratch. These videos should be shown immediately after Slide 13 of the PowerPoint presentation (Resource 1).

Note: To open the videos, you can either double click on them or alternatively you can choose the player you wish to use to play the videos, as shown in the following image.



If you are having problems opening the videos in a particular player, it is possible to download a player called **VLC Media Player** for free from the following site: <http://www.videolan.org/vlc>. VLC player is a highly portable media player capable of reading most audio and video formats. You must choose your operating system before downloading the VLC player, as shown in the following image.



Resource 1

How is software relevant to my life?

A PowerPoint slideshow guides the entire lesson



CD Resource

"M1L1R1 PowerPoint Karaoke.ppt"

Slide 1



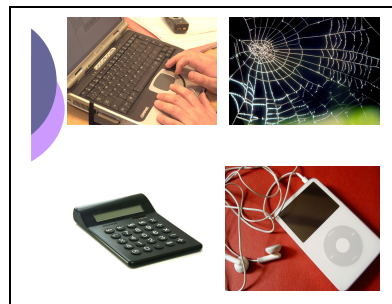
PowerPoint Karaoke

Talk for 1 minute about each slide.

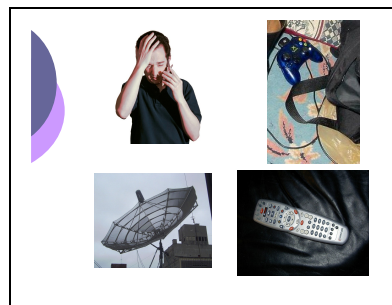
Module 1: Getting Started

5 Students are asked to volunteer to speak for at least 30 seconds on a slide that contains 4 pictures. They can be helped and encouraged by the audience to talk about anything they feel is linked to the pictures.

Slide 2



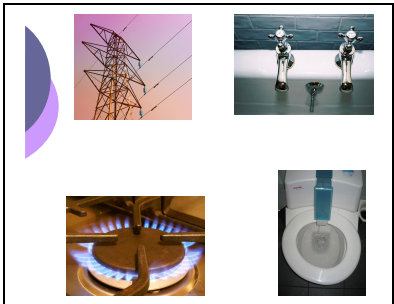
Slide 3



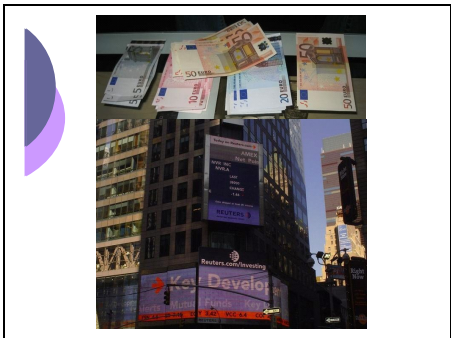
Slide 4



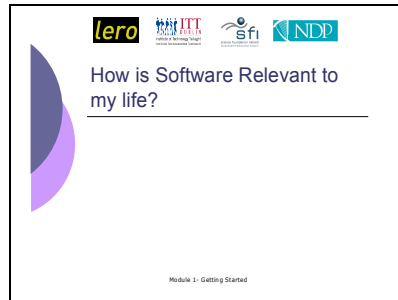
Slide 5



Slide 6



Slide 7



This is the title of the slideshow that has been viewed. Let's review the slides again taking the title of the show into account.

Slide 8



Computers, Internet websites, calculators and mp3 players simply would not function without software.

The first computers were huge machines. For example, the ENIAC computer weighed nearly thirty tons and occupied 680 square feet of floor space. ENIAC contained more than nineteen thousand vacuum tubes and performed about five thousand addition or subtraction operations every second. That is super slow for today but it was super fast back in those days. Today computers are tiny compared to what they were in 1949.

(<http://en.wikipedia.org/wiki/ENIAC>)

To understand the scale of software-driven social networking sites, consider MySpace. By 2008 MySpace had more than 110 million registered users; if it were a country it would be the 10th biggest in the world.

Slide 9



Thousands of lines of code are required for your modern mobile phone or games console to operate.

In June 2008, the Irish telecom regulator ComReg, reported that 25 million text messages are sent in Ireland every day

The technology contained in a single game boy unit in 2000 exceeds all the computing power that was used to put the first man on the moon in 1969.

Slide 10



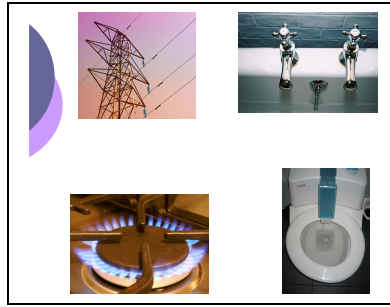
Larger scale machines such as cars or planes now use vast amounts of software to operate.

Even the systems for air traffic control and traffic lights depend on complex software programs to keep them running efficiently.

On July 28, 1962 a bug in the flight software for the Mariner 1 caused the rocket to divert from its intended path on launch. Mission control destroyed the rocket over the Atlantic Ocean. The investigation into the accident discovered that a formula written on paper in pencil was improperly transcribed into computer code, causing the computer to miscalculate the rocket's trajectory.

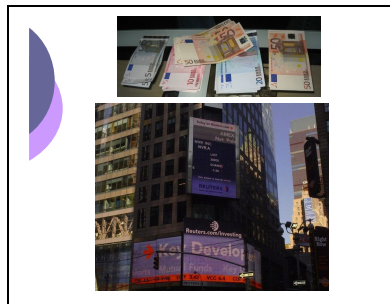
The cost of this mistake was \$18.5 million.

Slide 11




If the systems used at ESB monitoring stations were not working correctly, you might find yourself without your TV, lights, heaters and all home comforts you are used to.

Slide 12



It is often said money makes the world go round. \$34.2 billion of merchandise was sold on eBay in 2004. This is equivalent to the GDP of Cuba. Dublin native Brian Fallon set up Daft.ie in 1996. This was his transition year mini-company in St. Mary's College, Rathmines. Daft is now Ireland's biggest property website and is estimated to be worth millions.

Slide 13



Scratch is designed to help develop 21st century learning skills.

The Good News

You learn important mathematical and computational ideas, while also gaining a deeper understanding of the process of design.

The Better News

Scratch is a programming language that makes it easy to create your own interactive stories, animations, games, music, and art -- and share your creations on the web.

Visit the Scratch Website at: <http://scratch.mit.edu>

We will learn problem solving and programming skills using Scratch. You can visit the Scratch website at <http://scratch.mit.edu>.

We will introduce Scratch using 2 video clips.

In the next lesson we will be using Scratch on our computers.

Resource 2

Mitchel Resnick (Director of MIT Media Lab) Scratch Video

The director of the research group at Massachusetts Institute of Technology involved in the creation of Scratch talks about its exciting possibilities.



CD Resource

"M1L1R2 intro_mitchelresnick.mpg"

Resource 3

Introduction to Scratch Video

A short introductory video on the features of Scratch.



CD Resource

“M1L1R3 Scratch Intro.mpg”

Lesson 2 – Say “Hello” to Scratch

Resources:

Instructions to Download and Install Scratch (Resource 1), Quiz (Resource 2)

Key Vocabulary:

Blocks Palette, Execute, Green Flag, Interface, Scratch, Script, Sprite, Stage, Tabs, Toolbar

Description:

Students are introduced to Scratch and the Scratch interface, including the stage and (x,y) coordinates. Students write their first program in Scratch and are introduced to movement of sprites. Students learn how to add comments to their scripts and how to add notes to describe their projects in Scratch. Students complete a quiz on interface terms at the end of class.

Learning Objectives:

1. To introduce the students to the Scratch interface.
2. To allow students to build and save their first scripts.

Lesson Introduction:

- Reiterate the idea of how a programming language controls the behaviour of a machine using a set of instructions.
- Tell students that this lesson introduces them to the look and feel of Scratch.
- Tell them they will write their very first computer program (script) using Scratch.

Lesson Breakdown:

1. Students open Scratch via the desktop icon or Start → Programs method.

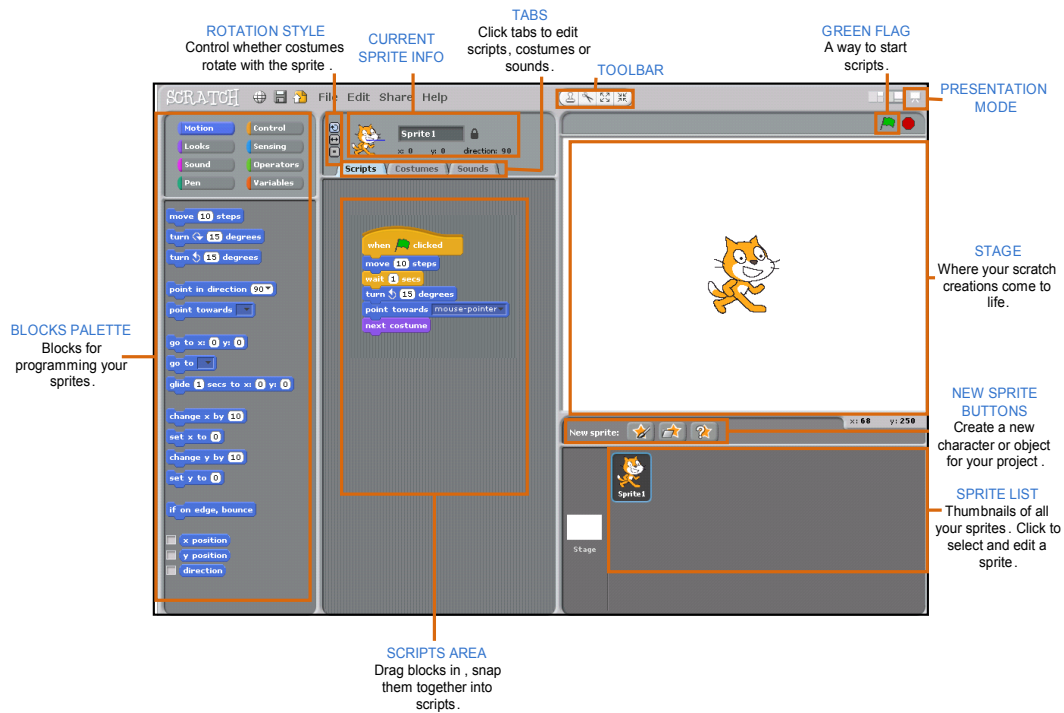
Open Scratch using the Start Menu



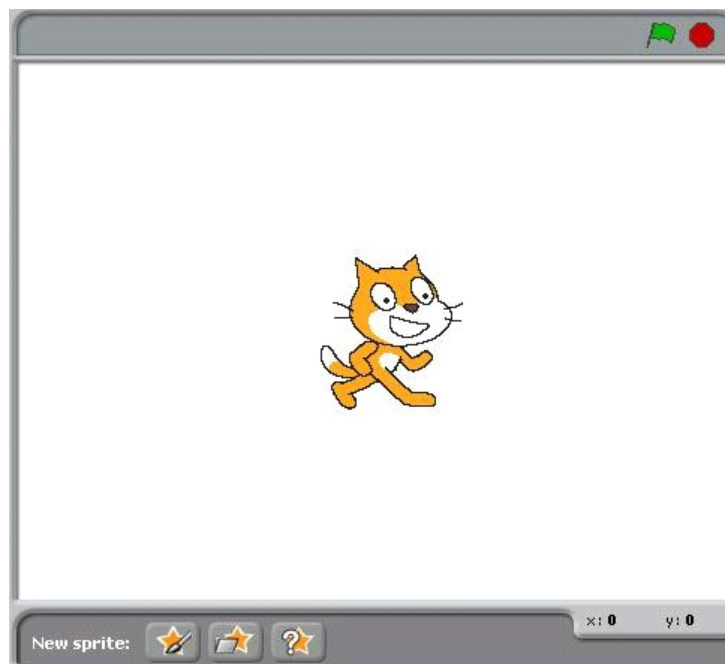
You can also use the Desktop Icon



2. Introduce students to the main features of the Scratch interface as outlined in the diagram.

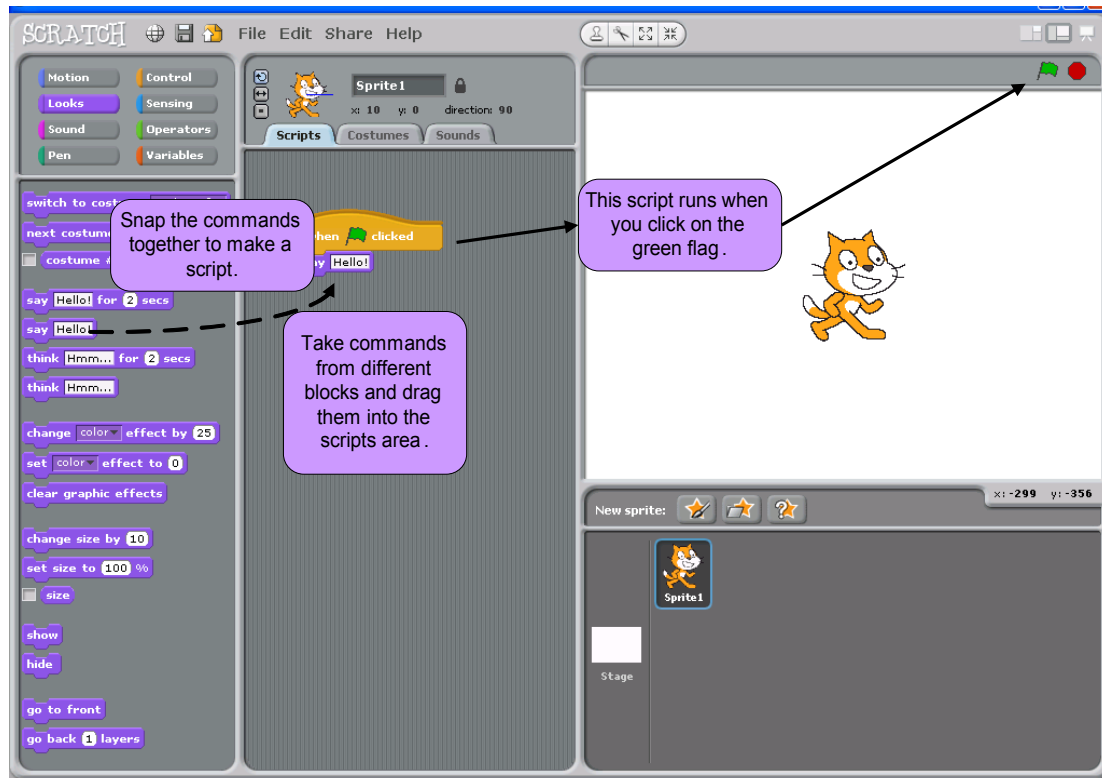


3. Ask students to find $x=0$, $y=0$ on the stage, using the mouse.

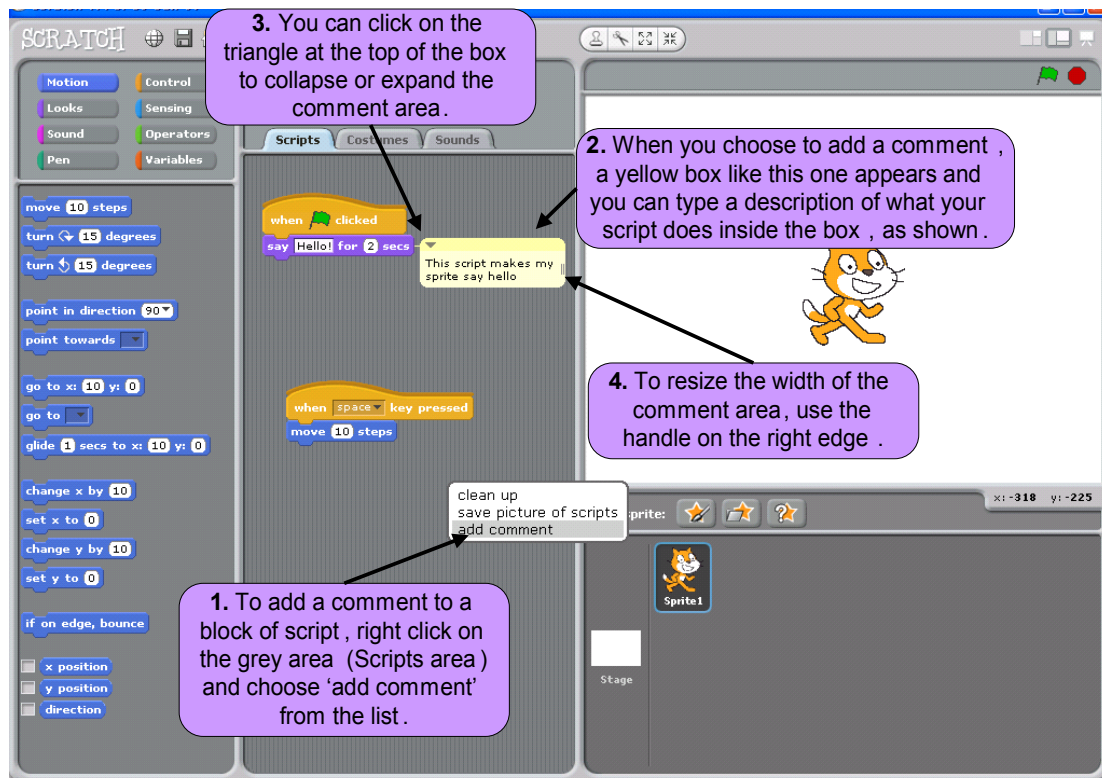


The x and y positions of the mouse on the stage are displayed here. Position the mouse on $x:0$ $y:0$.

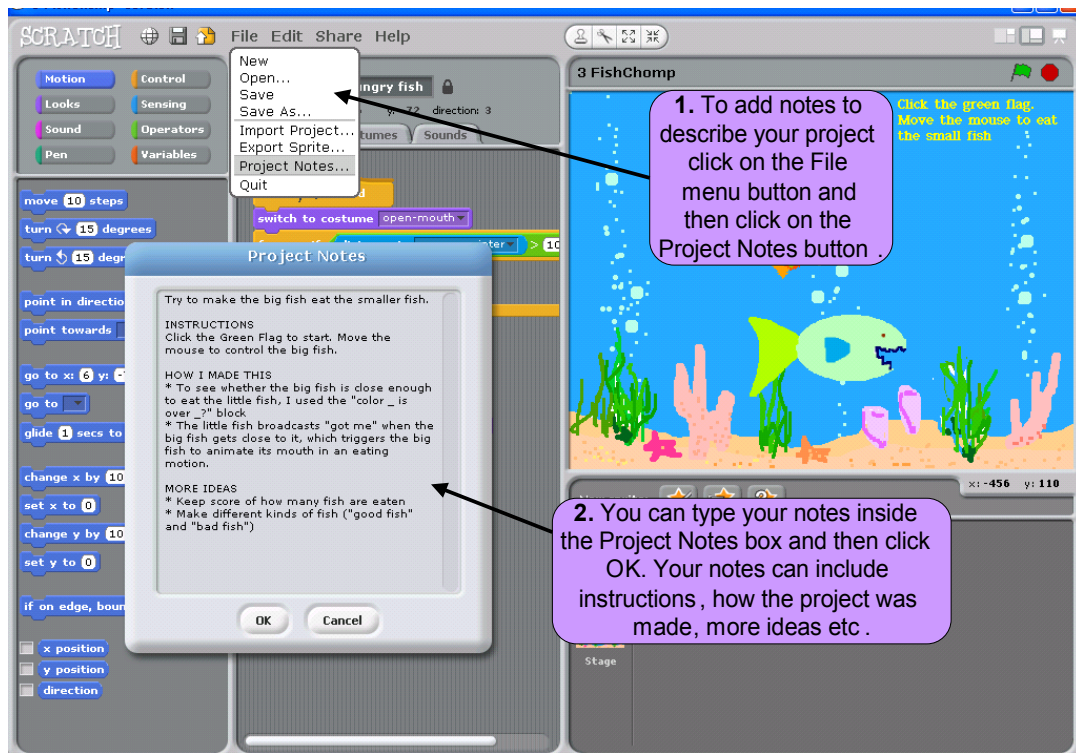
4. Ask students to complete a quiz (Resource 2) on the Scratch interface features that have been outlined.
5. Demonstrate how to write a script to make a sprite say hello when the green flag is clicked. See instructions below.



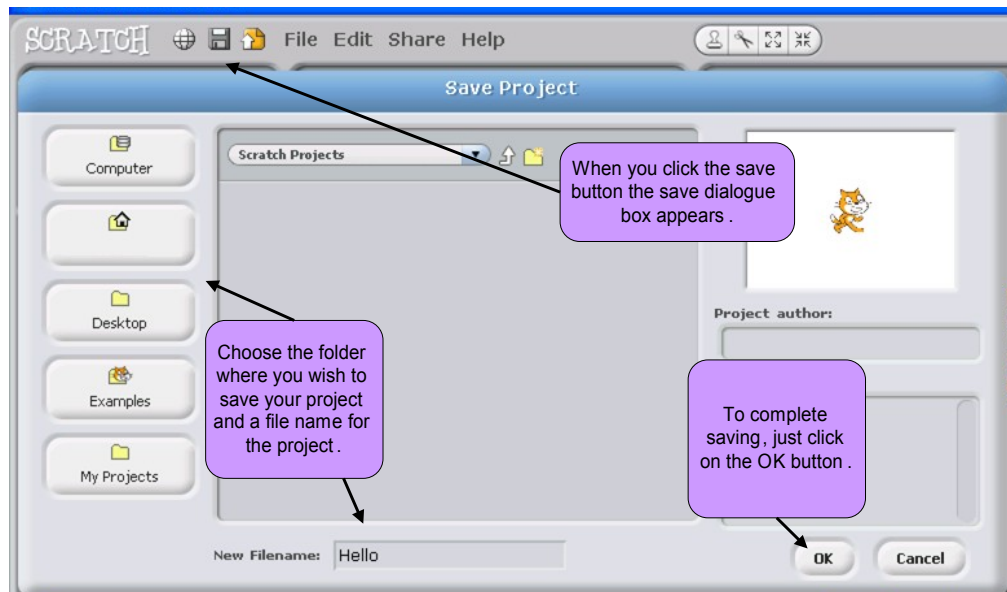
6. Explain to students that they can add comments to their scripts to describe what the scripts do. Real programmers also comment their code. It is a very useful for understanding what your script does, but it is also useful for others who may want to open your project and change your scripts to make their own projects. Sharing of projects so others can view them and change them is a very important part of the Scratch idea. The Scratch website <http://scratch.mit.edu> has many projects that can be downloaded so that scripts can be modified to create new projects. The following image shows how to add comments to a block of script.



7. Explain to students that as they build projects in Scratch they can use the Project Notes section to explain what their project is about. The Project Notes section can be accessed as shown in the following image and is very useful to explain how a project works.

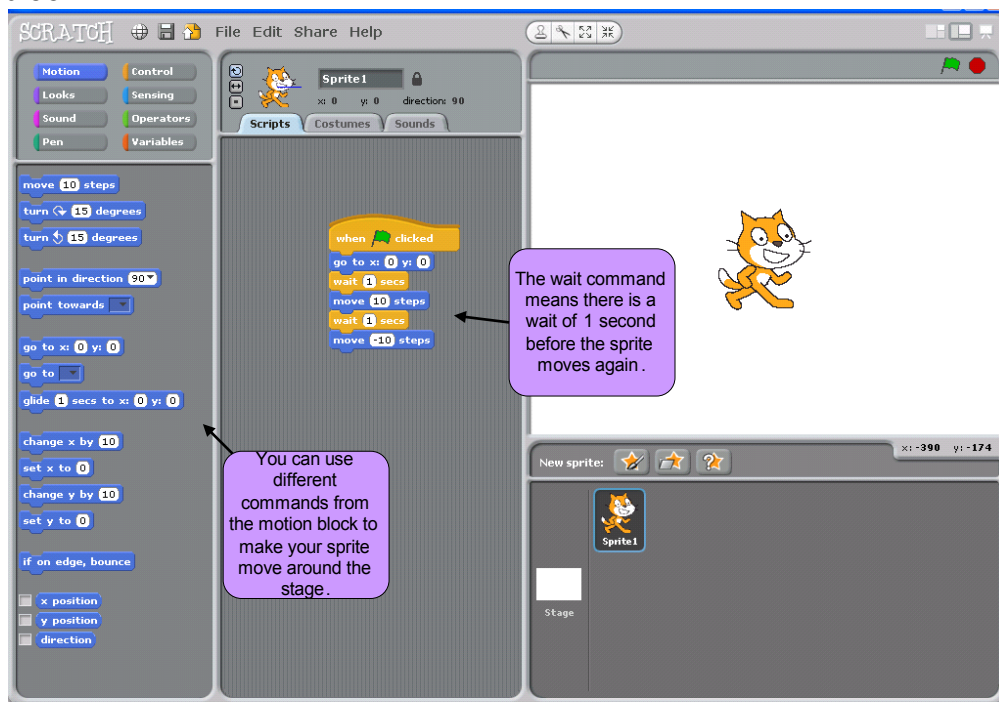


8. Demonstrate how to save the program.



➡ Extension activity

Build and execute the program in the following image to demonstrate how to make a sprite move. Students can then experiment with the Motion block.



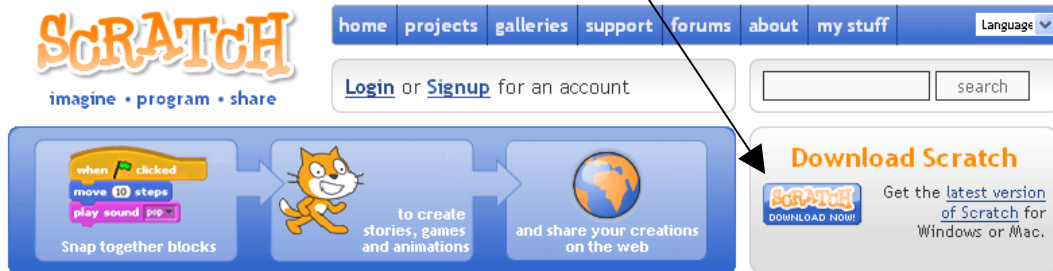
Resource 1

Instructions to download and install Scratch

Scratch Installation Instructions

Download the Scratch Installer

1. Go to the Scratch Homepage at <http://scratch.mit.edu>
2. Click on the link to Download Scratch



3. The next screen contains an optional form. You can fill it in to get email updates about Scratch or proceed directly to the download page.

A screenshot of the 'Download Scratch' form. The form has a title 'Download Scratch' and a subtitle 'Fill out the optional form below to get updates about Scratch. To get to the download page, click the button at the bottom of the form.' The form fields include: 'Email Address:' (with a 'Skip if under age 13' link), 'School or Organization:', 'City:', 'State/Province:', 'Country:' (a dropdown menu), 'Role:' (checkboxes for Student, Educator, Parent, Researcher, and an 'Other:' text field), 'Where did you hear about Scratch?', 'What interests you about Scratch?', and 'Anything else you want to share about your background or related experience?'. A 'Continue to Scratch download' button is at the bottom right. Below the form is a link: 'If you are having difficulties with this form, please go directly to [Scratch download](#)'.

4. On the next screen you will need to choose the appropriate file depending on your operating system (Windows or Mac). Click on the correct link to download the installation file to your computer.



Install Scratch from your Computer

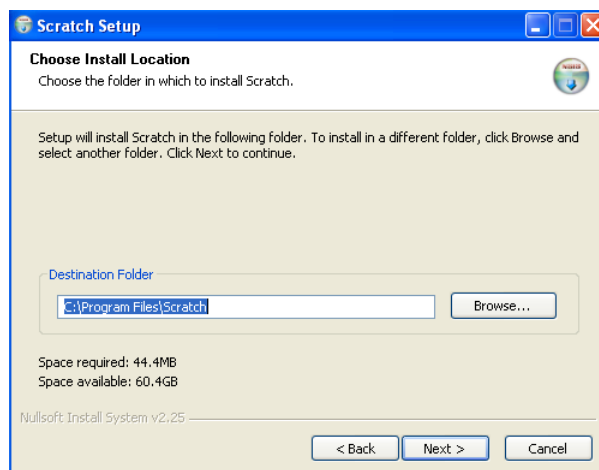
1. Double click on the downloaded installation file <ScratchInstaller.exe> to run the installation



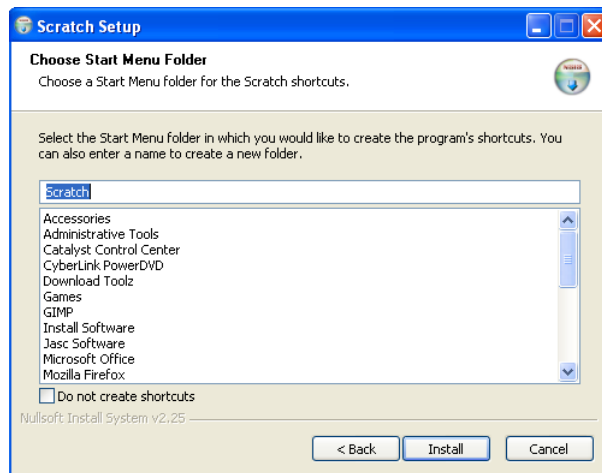
2. Follow the prompts on the installation wizard as shown below and you will be ready to use Scratch.



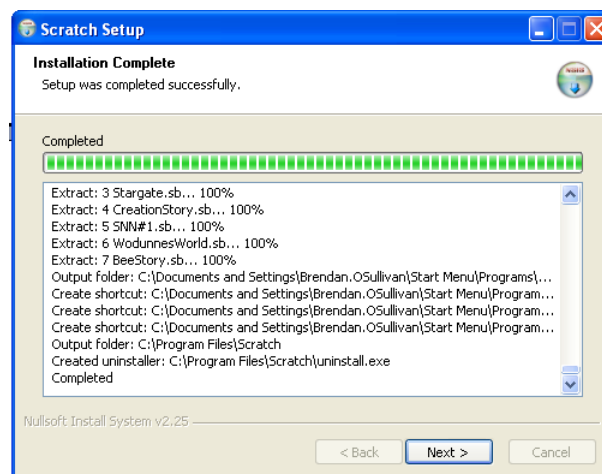
Click next



Click next



Click install



Click next



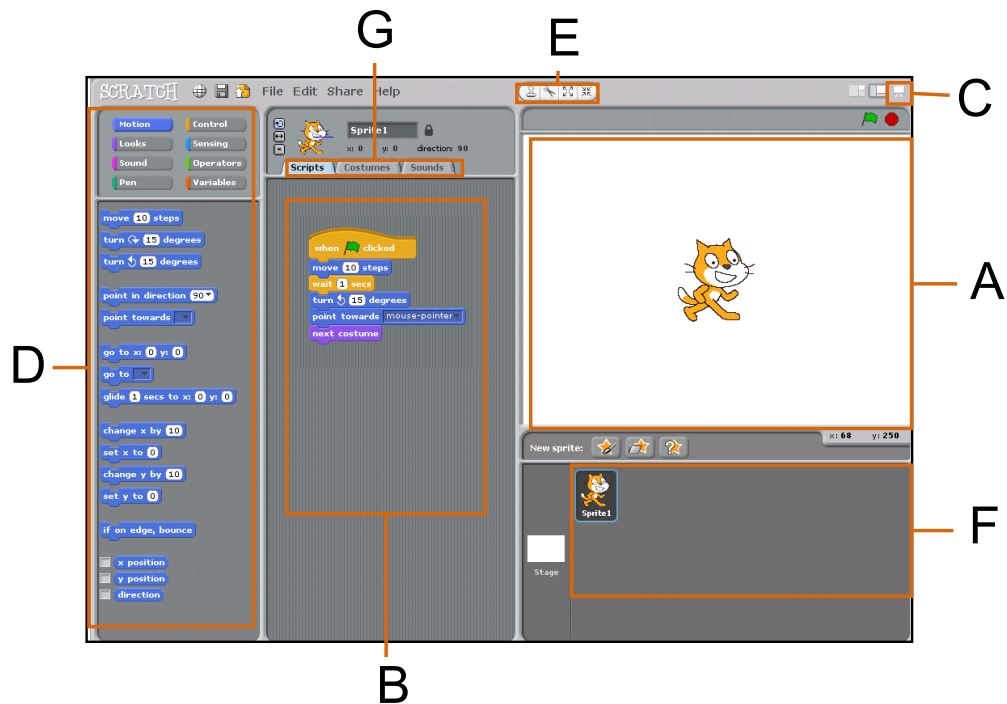
Click finish

Resource 2

Quiz

A worksheet for testing student knowledge of the Scratch Interface.

The Scratch Interface Quiz



Task:

Look at the Scratch Interface above.

In the table below see if you can match the letters to the different features of the Scratch Interface:

A
B
C
D
E
F
G

Blocks palette
Stage
Sprite list
Presentation Mode Button
Tabs
Scripts Area
Tool Bar

Lesson 3 – Playing with Pictures

Resources:

No CD or printable resources required.

Key Vocabulary:

Sprite, Library, Costume, Background, Import

Description:

Students learn how to import a sprite from a library and from an external source. They also learn how to edit a sprite including how to resize it. Students write a program to change the costume of a sprite and learn how to change the background in a project.

Learning Objectives:

1. To introduce students to graphics in Scratch and to allow them to import and edit sprites and to modify backgrounds.
2. To allow students to write a Costume Change program.

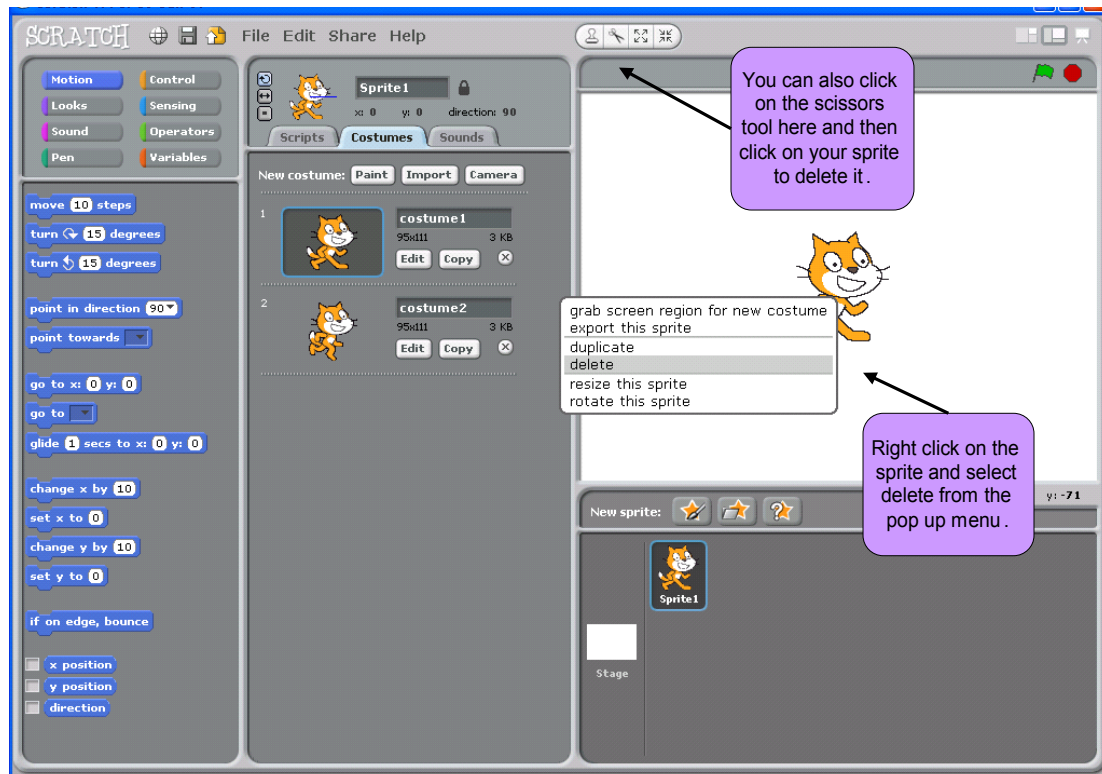
Lesson Introduction:

- Revise the Scratch interface and the idea of sprites as characters.
- Tell students that today they will import sprites from within Scratch and from other sources e.g. Internet.
- Tell them they will have fun changing the look of these characters and the backgrounds.
- Tell them they will write a Costume Change program for their chosen sprites.

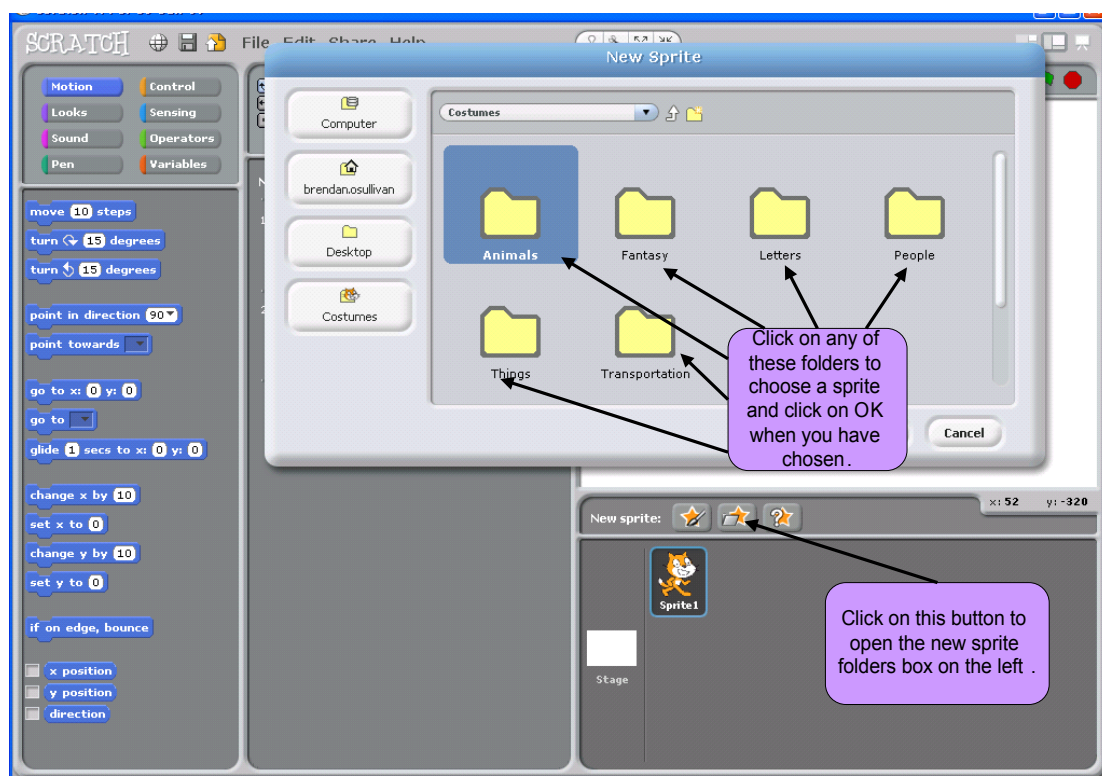
Lesson Breakdown:

1. Students open Scratch as explained in Lesson 2.
2. Revise the features of the Scratch environment, particularly the Blocks Palette, Scripts Area, Tabs and Sprite Buttons and Stage Area.

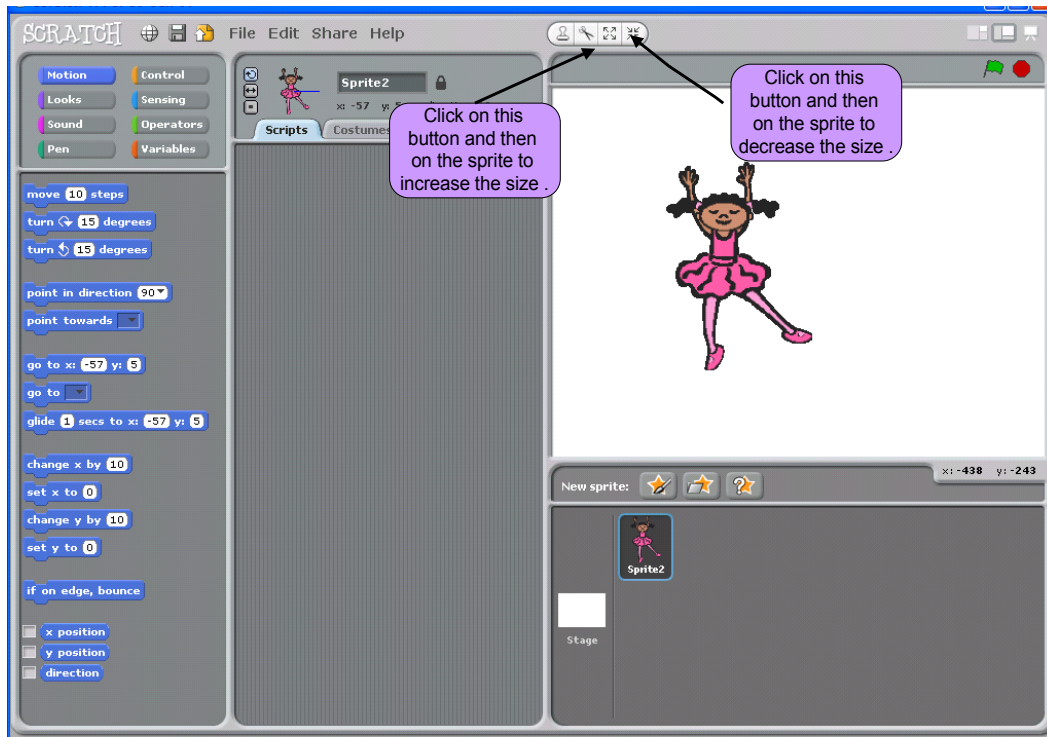
3. Show students how to delete a sprite using either the toolbar, or by right-clicking the sprite and deleting and ask them to delete the Scratch cat sprite now, as shown below:



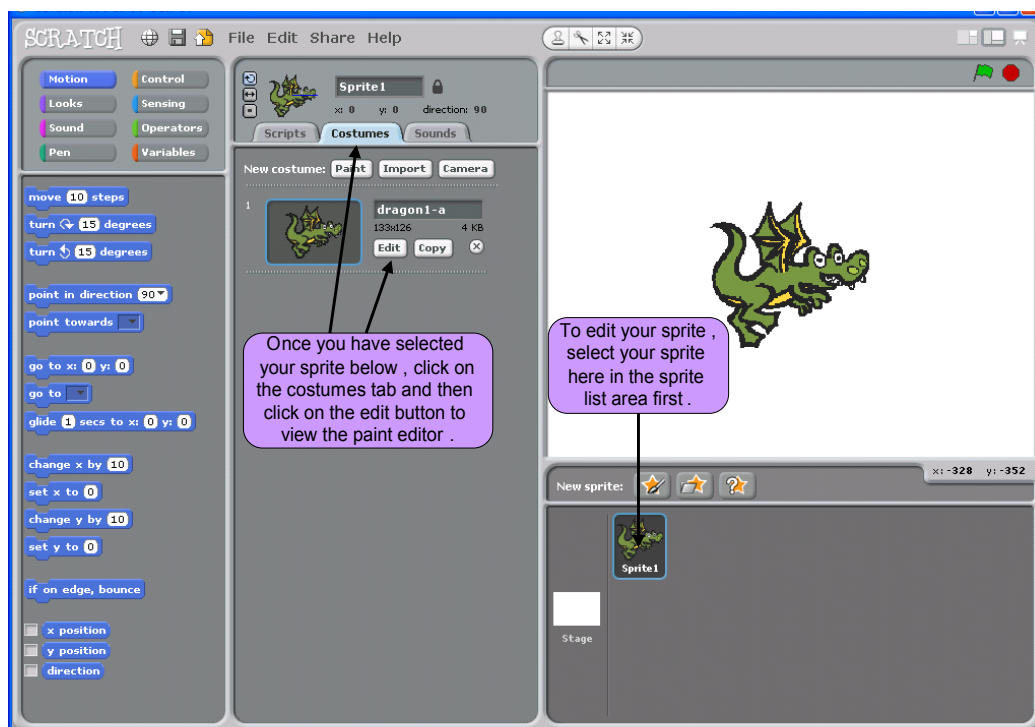
4. Show students how to import a sprite from a library in Scratch, as illustrated below:



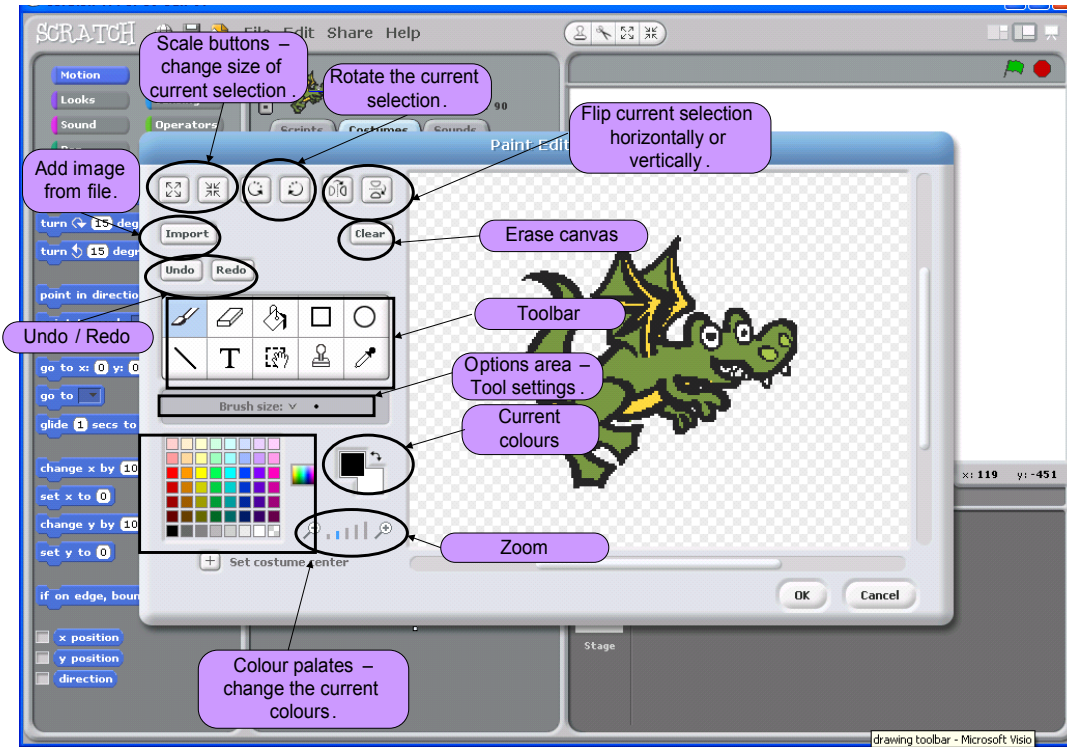
5. Ask students to import a sprite of their choice from the library.
6. Show students how to resize the sprite using the toolbar as shown below:



7. Show students how to edit a sprite by selecting the sprite in the sprite list area of the screen, clicking on the costumes tab, and clicking on edit, as shown below:



8. Ask students to select a sprite and to click on the edit button so that the paint editor appears.
9. Explain the various features of the paint editor as explained in the diagram below:

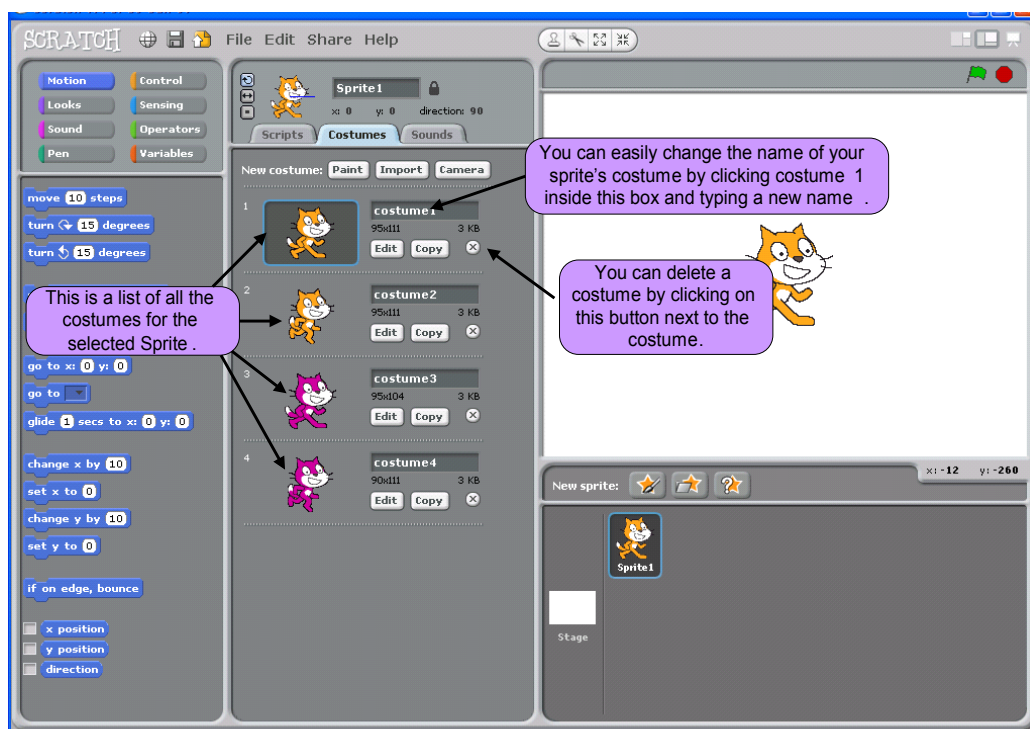


10. Ask students to experiment with their sprite by changing size and colour etc.
11. When they are finished they should click OK.
12. Ask the students to open a new project, by clicking the new button and tell them they are going to write a Costume Change program to animate a character.
13. Ask the students to leave the Scratch cat sprite on screen and to click the costumes tab.

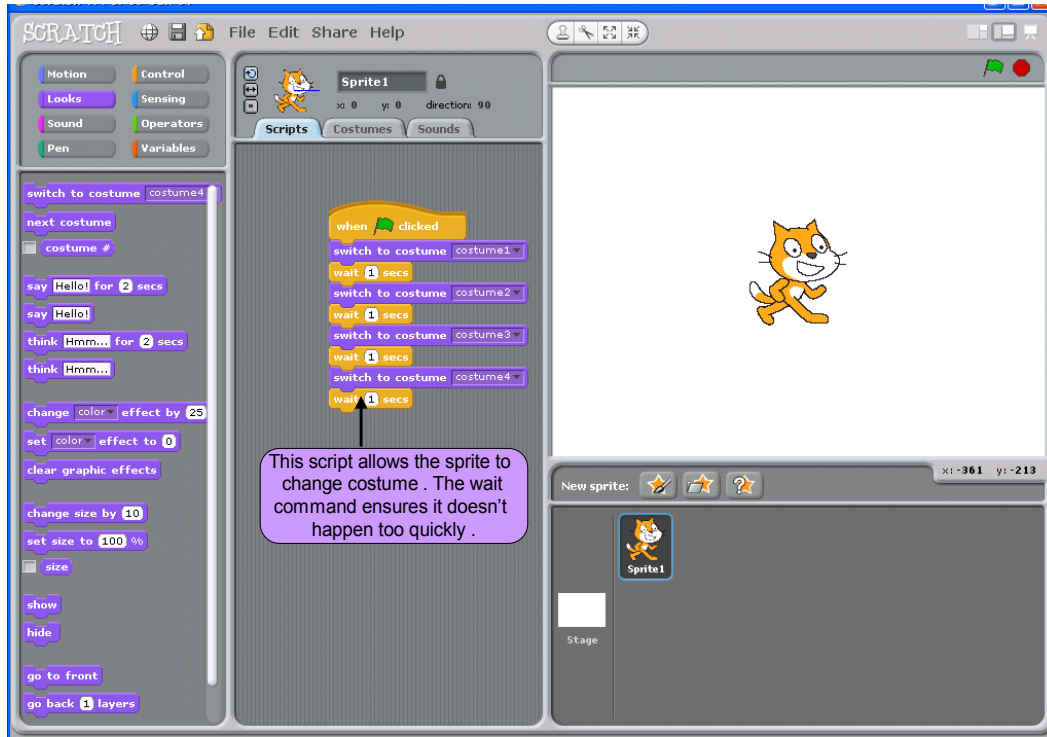
14. Show the students how to paint a new costume by clicking the new costume paint button as shown below and then import the first cat sprite from the animals directory as shown below:



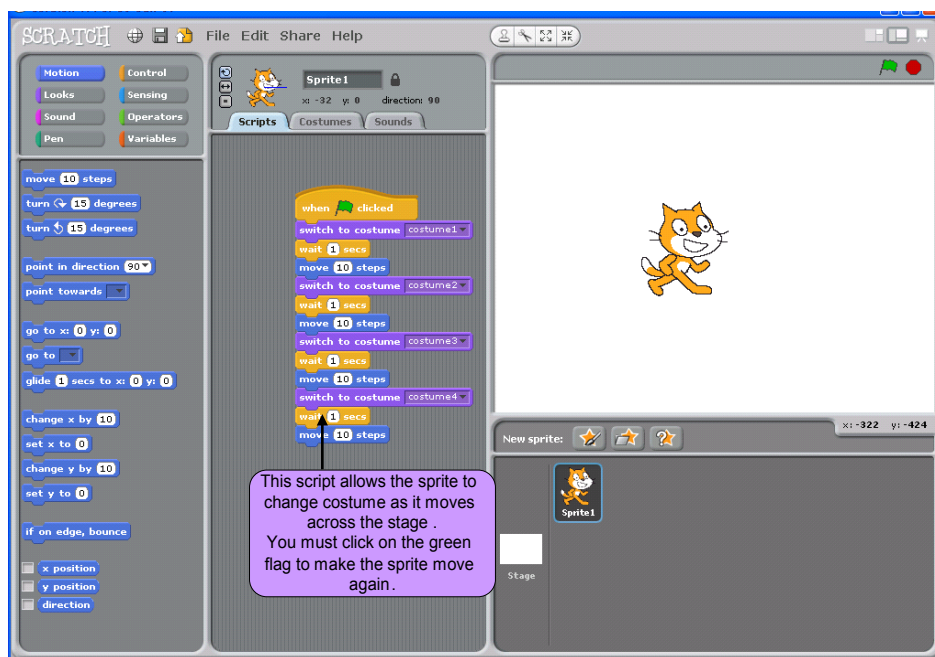
15. Get the students to change his colour, using the Fill Tool. When they click on OK, he should appear as costume3 on the list.
16. Repeat step 15 for another cat sprite from the library. This cat should appear as costume4 on the list as shown below. Show students how they can rename their costumes by clicking on the name e.g. 'costume1' and typing the new name over it. They can also delete a sprite using the x as shown in the following diagram:



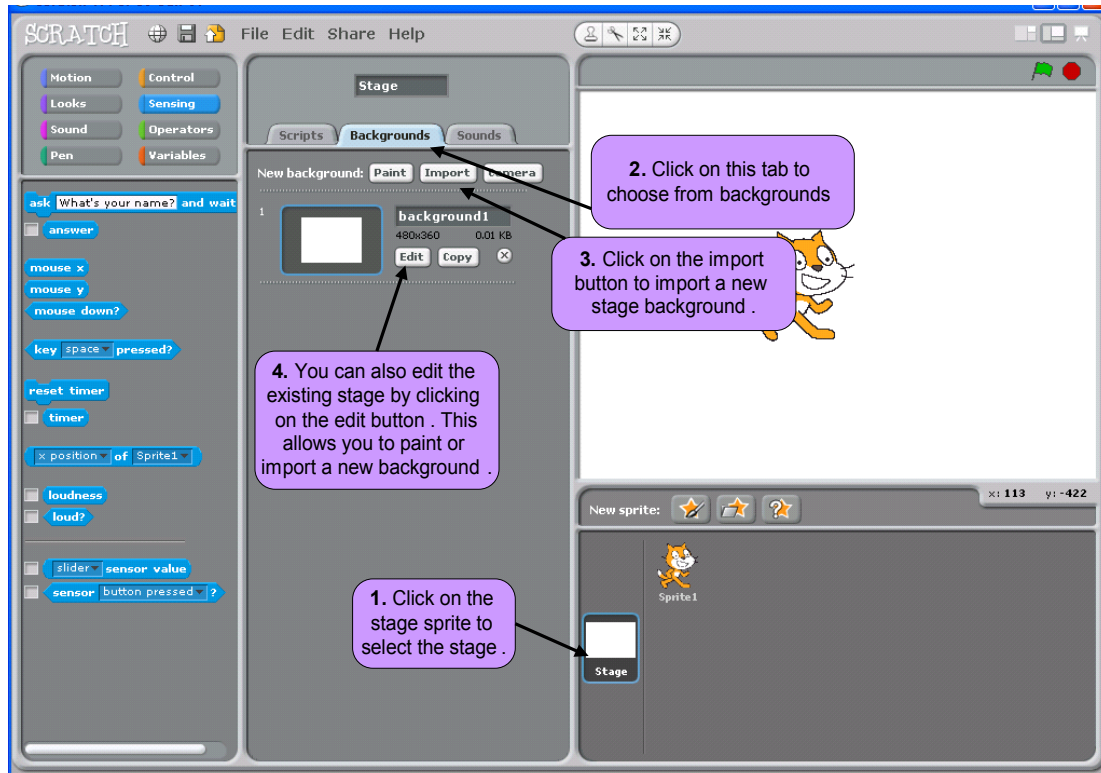
17. Click on costume1 from the list and click on the scripts tab again.
18. Build and execute the program below to change the costumes. The wait command ensures the change doesn't happen too quickly.



19. Ask the students to build and execute the script themselves.
20. Ask students to modify the script to make the cat move across the stage as he changes costume. See one solution below:



21. Ask students to click on the stage sprite in the sprite list area and ask them to click on the backgrounds tab as shown below. Ask them to click on the import button and they can choose a background from one of the folders of their choice:



Extension activity

Ask students to create a new project, using a new background of their choice, either created in paint editor or imported from the Scratch library. Ask them to make 2 sprites move around the stage. Sprites should wear two different costumes. The sprites can be edited using the paint editor and students can be creative! Students can also try to save an image from the Internet and import it into Scratch.

Lesson 4 – Tell me what to do

Resources:

Tell me what to do (Resource 1), Handout with 3 Algorithms (Resource 2)

Key Vocabulary:

Algorithm

Description:

Students explore what an algorithm is, using everyday examples. They will write an algorithm to solve a basic problem given by the teacher. They will then write their own algorithm to top up a mobile phone.

Learning Objectives:

1. To introduce students to algorithms and to help them understand what an algorithm is and how it fits into everyday life.
2. To allow students to write their own algorithms to solve basic problems.

Lesson Introduction:

- Introduce students to the idea that computers need instructions to tell them what to do.
- Tell them they will discover how to provide effective instructions to solve everyday problems.

Lesson Breakdown:

1. Divide students into groups of 2 or 3 at the start of class.
2. Begin PowerPoint presentation (Resource 1).
3. Present students with the 3 sets of instructions on the Handout with 3 Algorithms sheet (Resource 2): Baking a Cake, Directions to Mary's House and Buying a Packet of Crisps and a Drink.
4. Ask students to study and consider each set of instructions.
5. Ask the students what the 3 sets of instructions have in common i.e. they are all sets of instructions to solve a problem.
6. Using the 3 algorithms as an example, ask the students to define the characteristics of a good algorithm and write them on the board.
7. Show the students the definition and characteristics of a good algorithm on the PowerPoint presentation. Refer to the 3 examples on the resource sheet used earlier.
8. Encourage students to use the simple ordered English approach for writing algorithms using the Buying a Packet of Crisps and a Drink example.
9. Show the students the problem of the dog, the goat and the cabbage on PowerPoint. Ask them to write an algorithm, using the simple ordered English approach to solve the problem. The solution can be shown after students have tried the problem.

10. Ask the students to write an algorithm to top up the credit on their mobile phone. See sample solution on slide 9 of PowerPoint presentation (Resource 1).

Resource 1

Tell me what to do


This PowerPoint presentation guides the lesson activities. It outlines the definition and characteristics of a good algorithm.



CD Resource

"M1L1R1 Tell me what to do.ppt"

Slide 1



lero **IRIT** **SFI** **NDP**


Tell me what to do

Using sets of instructions...




Module 1 - Getting Started

Introduce the topic.

Slide 2



Consider these 3 situations....


- Baking a Cake 
- Giving Directions 
- Buying items in a shop 

Introduce the 3 scenarios and ask students to look at the 3 sets of instructions on the sheet in front of them.

Ask students to study each set of instructions and to make a list of the things they have in common and what is different about their structure or the way in which they are written.

i.e. All three examples are sets of instructions to solve a problem (algorithms). The first example 'Bake a Cake' is a list, but there are too many instructions that one must read through. The second example of 'Directions' is not very easy to understand and seems very 'wordy'. The final example of buying items in a shop is the clearest, ordered set of instructions and can be followed easily.

Slide 3




What do all 3 have in common?

- They are all sets of instructions.
- They all solve a problem.

These are called Algorithms.

Pose the question “what do all 3 have in common” and look for suggestions from students. Reveal the bullet points and discuss the term algorithm.

Slide 4



A good algorithm


1. Leads to completion.
2. Is accurate and unambiguous.
3. Has the appropriate level of detail.
4. Has well ordered instructions.
5. Covers all possible outcomes with practical solutions.

Reveal the slide title and ask the students to look at the 3 sets of instructions again and to decide what the characteristics of a good algorithm are, using these examples.

Ask students to decide what characteristics a poor algorithm would have.

Discuss the students' suggestions before revealing the list.

Slide 5



Human vs Computer Algorithms

- With human algorithms we can rely on
 - Intuition
 - Initiative
 - Prior knowledge
 - Independent decision-making
 - Learning from experience
- Computers do not have these things and therefore we must take this into account when we write the algorithm for a computer program.

Remind students that a human algorithm is a sequence of actions meant to be carried out by people. Many of our everyday activities are algorithmic (the recipe to bake a cake, using electrical devices, driving etc.).

Introduce the first bullet point by explaining the 5 things human algorithms rely on.


Compare the first bullet point with the second by explaining that a computer cannot rely on the same things as humans.

Instructions for computers need to be very specific and clear.


Suggest the simple ordered English approach used in the “Buying a Packet of Crisps and

a Drink” example as the clearest way of writing our algorithms in today’s lesson.



Slide 6



Write an algorithm...
Solve the farmer's problem




- A farmer has to take three items, a dog, a goat and a cabbage, across a river. He has a boat but it can only take two objects at once, in other words, the farmer and one other item.
- Write an algorithm to tell the farmer how to get the three items across the river.
- **Note:** If the dog and the goat are left together the dog will eat the goat.
If the goat and the cabbage are left together the goat will eat the cabbage.



Present the problem to students.


Ask students to write an algorithm using the step by step simple English approach to solve the problem.

Slide 7



The Farmer's Solution...


1. Begin on side A
2. Take goat across to side B
3. Return with empty boat to side A
4. Take dog across river to side B
5. Return with goat to side A
6. Take cabbage to side B
7. Return with empty boat to side A
8. Take goat to side B
9. END



Present the solution once students have tried to solve the problem for themselves.


Discuss the structure of the solution.

Slide 8




Top up your mobile phone credit!

How?



Ask students to write an algorithm to top up their mobile phone credit using the simple ordered English approach.

Slide 9



Possible solution

1. Buy a top up voucher in a shop or at a vending machine.
2. Dial 1741 on your phone.
3. Listen to the instructions.
4. Enter the code on the voucher and press the # key on your phone.
5. Listen for your new balance to make sure it's correct.
6. Hang up.

Present this as a possible solution, once students have presented their own possible solutions for topping up their mobile phones.

Resource 2

Handout with 3 Algorithms

The 3 algorithms on the handout are compared to identify features of a good algorithm.

What do the following 3 things have in common?

1. Baking a cake

1. Mix milk and vinegar. Melt the chocolate. Sieve flour, cocoa and soda into a bowl. Add margarine, sugar, eggs and half the milk mixture and beat well.
2. Add the melted chocolate and remaining milk. Beat until smooth. Transfer to 8" round cake tin which has been greased & lined. Bake in a preheated oven at 325°F, 160°C, Gas Mark 3 for 1 ½ hours approx. When cake is cold, split in half & use half the filling to sandwich cake. With remaining filling, completely cover cake & decorate to your choice.
3. Put sugar & evaporated milk into a saucepan. Stir and bring to the boil, then simmer for 5 minutes. Remove from heat. Add the broken chocolate & stir until melted. Repeat with margarine. Pour into a bowl & leave in fridge until it thickens.

2. Directions to Mary's House

From the town hall, follow Orchard Road for 2 kilometres until you reach a set of traffic lights. Turn left onto Western Road. Continue driving along Western Road until you reach a T-junction. Turn Right onto O'Connell Avenue and continue forward for 1 kilometre until you pass the Post Office on your left. Just after the Post Office turn left onto Meadow Drive. Mary's house is the second house on the right.

3. Buying a packet of crisps and a drink

1. Choose a packet of crisps and a drink from shelf.
2. Go to checkout.
3. Wait in queue.
4. Hand items to cashier.
5. Wait for cashier to say total price.
6. Take out money.
7. Hand money to cashier.
8. Wait for change.
9. Take items.

Lesson 5 – Playing With Music

Resources:

No CD or printable resources required.

Key Vocabulary:

Event handler, Sequence

Description:

The concept of Event Handlers is introduced by programming a sprite to move in different directions when the arrow keys are pressed. Students will also use the Sound block to map keys to notes and play musical scales. By end of the lesson new instruments and sprites will be introduced to make a band.

Learning Objectives:

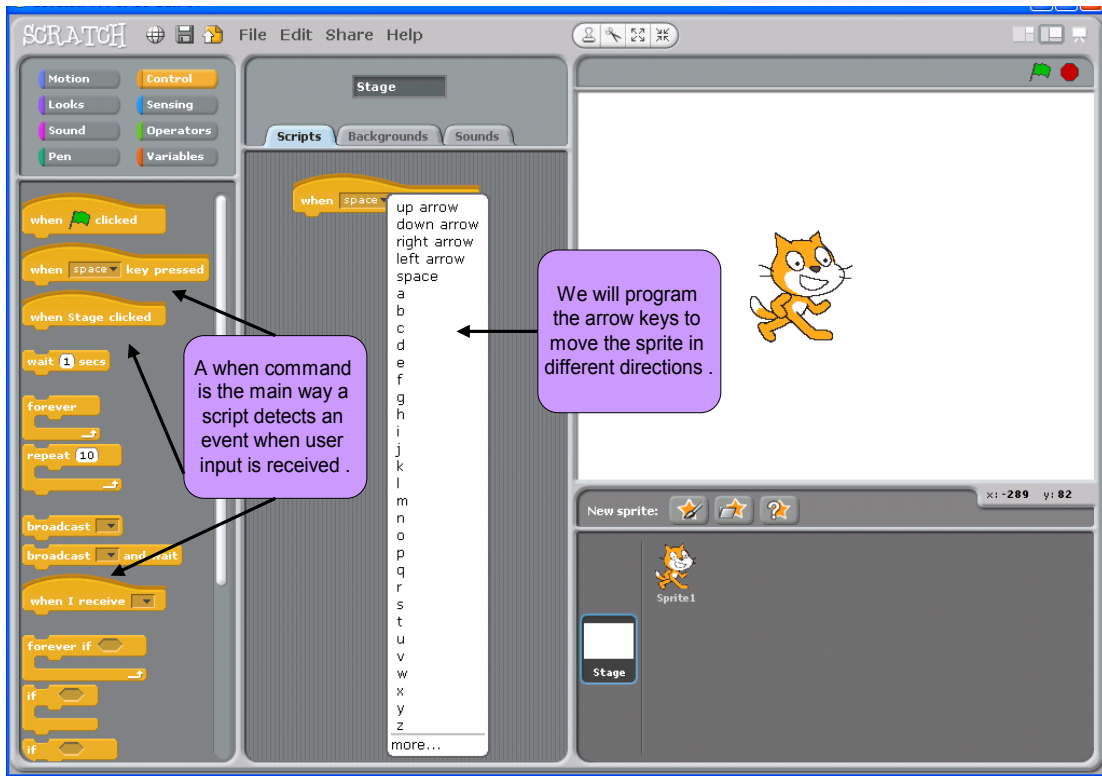
1. To learn about Event Handlers by programming scripts that are triggered by input from the computer keyboard.
2. To allow students to create a sequence of commands to play a musical scale.

Lesson Introduction:

- Reiterate that it is important that computer programs are written as a set of instructions in a sequence.
- Introduce the idea that computer programs need to respond to input from users.

Lesson Breakdown:

1. Explain that a when command can be used to respond to an event triggered by the user. For example, if a user presses the up arrow on the keyboard.



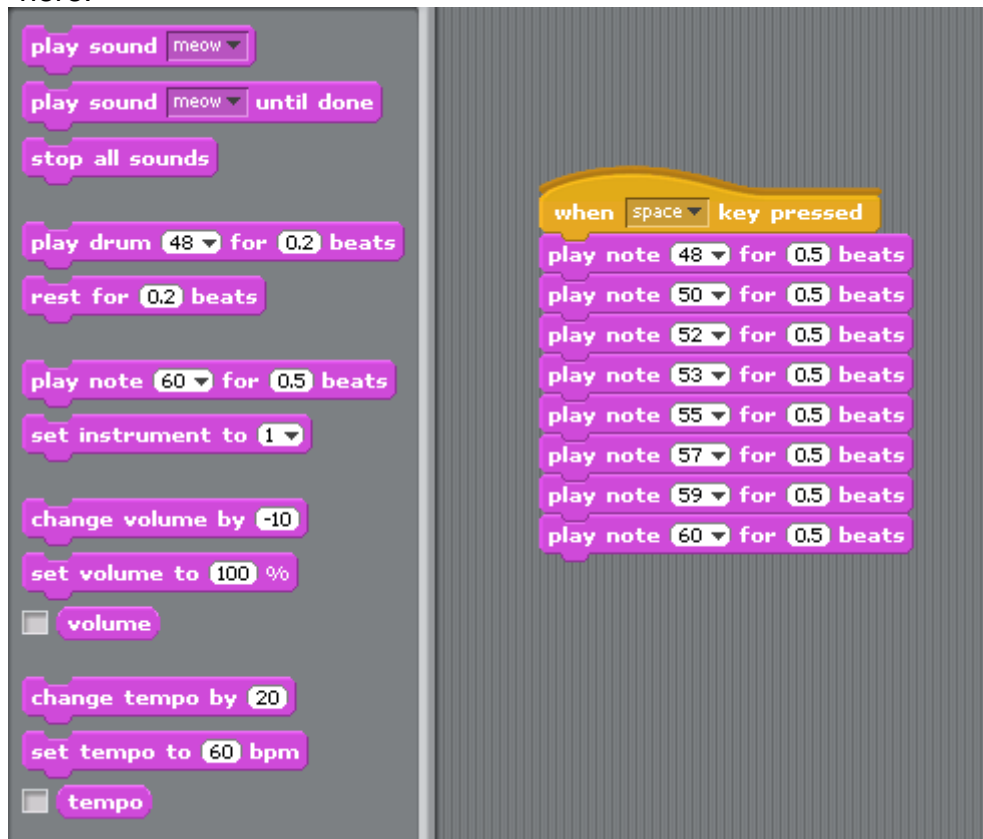
2. Show how to program the right arrow to move the sprite forward.



3. Pupils can try on their own to figure out how to program all of the arrow keys to move the sprite up, down, left and right.



4. The next challenge is to use sounds to get your sprite to play a musical scale when the space key is pressed. A possible solution is shown here.



5. Finally students can introduce new sprites as shown in Lesson 3 of this module. Using the commands of the sounds blocks palette an entire band can be created and controlled by users. Students should be encouraged to share their ideas.



Extension activity

Spend a lesson having a battle of the bands competition to see who can get Scratch to put on the best Stage Show.

References

Lesson 1

Resource 1

Slides 2-12 Photographs

<http://www.morguefile.com/> and <http://www.freefoto.com>

Slide 8 - Eniac Fact

<http://en.wikipedia.org/wiki/ENIAC>

Goldstine, Herman H. (1972). The Computer: from Pascal to von Neumann. Princeton, New Jersey: Princeton University Press. ISBN 0-691-02367-0

Slide 8 - MySpace Fact

MySpace cranks up heat in Facebook turf war by John Schwartz USA today available at http://www.usatoday.com/tech/webguide/2007-12-20-myspace_N.htm

Slide 9 - Text messages Fact

<http://www.rte.ie/news/2008/0618/broadband.html?rss>

Slide 10 - Mariner 1 Fact

"For Want of Hyphen Venus Rocket Is Lost;" By GLADWIN HILL Special to The New York Times. July 28, 1962, Saturday

Slide 12 - Ebay Fact

<http://www.geeze.us/news-archives/jul-09-2005.htm>

Slide 12 Daft Fact

<http://www.skooool.ie/skooool/senior.asp?id=4419>

Slide 13

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

Michael Resnick Video

<http://youtube.com/v/knFykmLlj0s>

Scratch Intro Video

<http://ilk.media.mit.edu/projects/scratch/videos/ScratchIntro.wmv>

Lesson 2

Scratch Interface Diagram

Scratch Reference Guide Page 2

http://scratch.wik.is/@api/deki/files/185/=ReferenceGuide_1.2.pdf

Lesson 3

Scratch Paint Editor Diagram

Scratch Reference Guide Page 8

http://scratch.wik.is/@api/deki/files/185/=ReferenceGuide_1.2.pdf

Lesson 4

Resource 1

Slides 2 and 6

All photographs were taken from <http://www.fotosearch.com/>

Slide 6 Farmer's Algorithm

http://aten.tau.ac.il/Basic_Programming/AlgoExercise.html