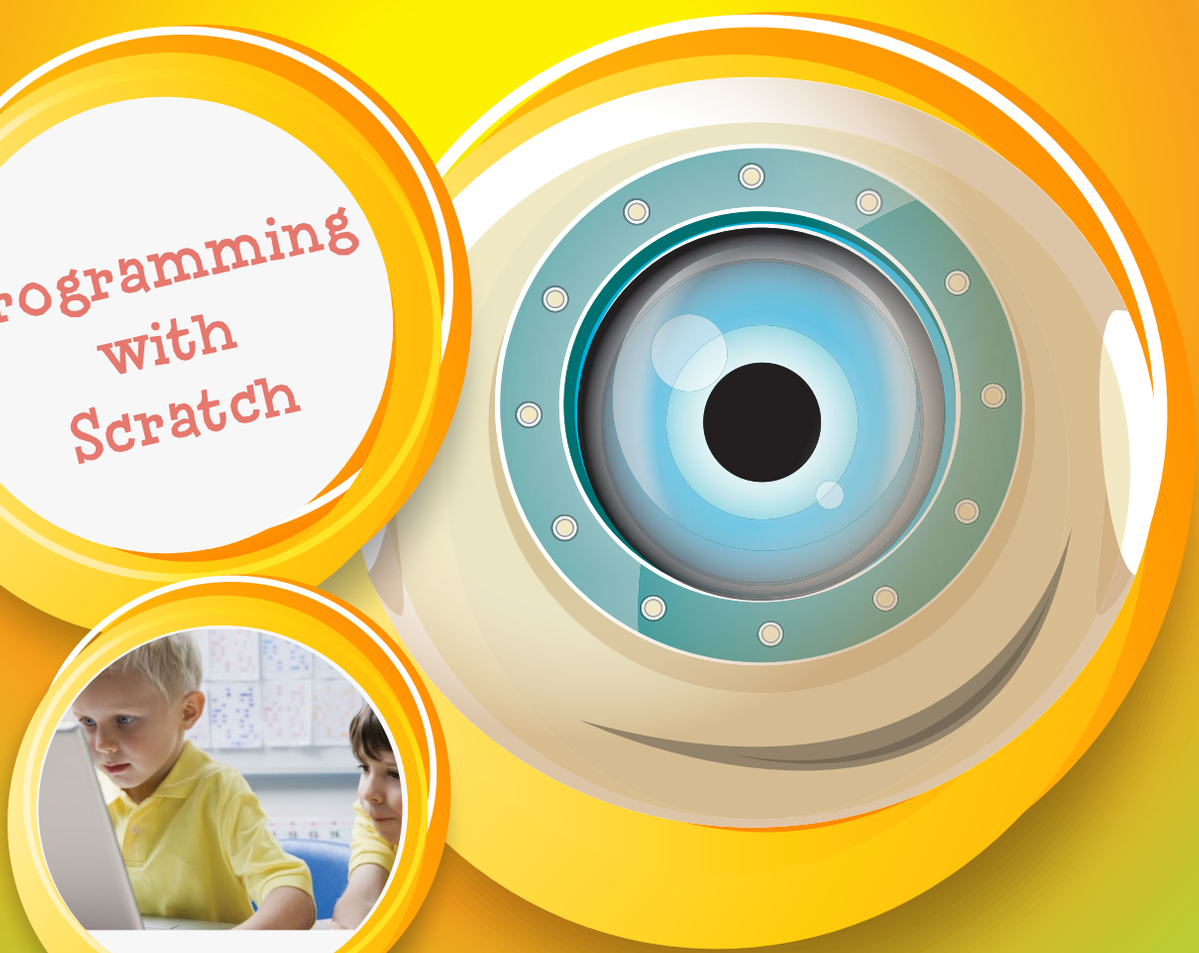
Year 2

iCompute

iProgram

Programming with Scratch



iCompute
www.icompute-uk.com

Programming

iProgram

Year 2

Overview

This unit of work introduces the children to a visual programming language: Scratch. Using the context of art and drawing, the children will be engaged in creatively developing simple animations.

Objectives

- See p2 for a detailed breakdown of lesson assessment focuses and associated success criteria.

National Curriculum

- understand what algorithms are, how they are implemented as programs on digital devices, and that programs execute by following precise and unambiguous instructions
- create and debug simple programs
- use logical reasoning to predict the behaviour of simple programs
- use technology purposefully to create, organise, store, manipulate and retrieve digital content
- recognise common uses of information technology beyond school
- use technology safely and respectfully, keeping personal information private; identify where to go for help and support when they have concerns about content or contact on the internet or other online technologies

Assessment

P15 contains a record of progress pro-forma

Software

See 'Preparation'

Hardware

Computers/tablets

Curriculum Links

- Mathematics
- English
- Art
- Design & Technology

Objectives

Lesson	Title	NC Links	Objectives	Vocabulary	Cross-Curricular Links	Success Criteria
2.2.1	iSequence	✶ To understand what algorithms are; how they are how implemented as programs on digital devices	✶ To understand that an algorithm is a process that consists of a series of steps that achieves a specific goal ✶ To understand algorithms can describe everyday activities and can be followed by humans and computers	Algorithm; instructions; sequence; input; output	✶ D&T ✶ English	✶ The children can sequence a set of instructions for making a sandwich
2.2.2	iInstruct	✶ To understand what algorithms are; how they are how implemented as programs on digital devices ✶ To create and debug simple programs	✶ To understand that algorithms are made up of steps ✶ To know that steps can be repeated ✶ To understand that computers need more precise instructions than humans do	Instructions; steps; order; repeat; forward; back; left; right	English	✶ The children create a sequence of instructions for a dance routine ✶ The children add repeat instructions for their routines
2.2.3	iMonster	✶ To understand what algorithms are; how they are how implemented as programs on digital devices	✶ To use digital drawing tools (Scratch) to create images	Cut; paste; undo; redo; copy; sprite	Art/Design	✶ The children draw their own monster sprite using Scratch software
2.2.4	iMove	✶ To understand what algorithms are; how they are how implemented as programs on digital devices ✶ To create and debug simple programs ✶ To use technology purposefully to create, organise, store, manipulate and retrieve digital content	✶ To program a simple animation involving movement	Input; output; statement; move; negative; steps; sprite	Maths	✶ The children create a simple animation where their sprite moves on the stage
2.2.5	iSpeak	✶ As Above	✶ To write a simple program that produces an output (text)	Duplicate; wait	English	✶ The children program two sprites to talk to each other using 'say' blocks
2.2.6	iCreate	✶ As Above	✶ To combine images and text to create a simple animation	Edit; undo; redo	Art/Design	✶ The children make a simple animation which has a background and sprites moving and talking

Preparation

- Read the lesson plans
- Spend an hour or so familarising yourself with the software you will be using and the Scratch interface

Resources

- Computers
- Ensure that any software you need is installed properly on each computer
- Ensure that any web access you need is available (eg. sites are not blocked)
- Worksheets for each lesson – entitled:
- Worksheet<year.unit.lesson> (eg.Worksheet2.2.1)
- Support materials for each lesson – entitled Resource <year.unit.lesson> (eg. Resource2.2.1)

Links

Before you start, you may find these weblinks useful.

Scratch can be accessed online here:
http://icomp.site/scratch

Guidance on teaching this unit with tablets or touchscreen devices:
https://icomp.site/scratch-tablets

iCompute's pre-written Scratch programs
(for all Resources.sb files)
https://icomp.site/scratch-studio-year-2

Updates

If any links are not working, obtain the latest version of this plan using your iCompute login

Resources

Resources: Resource2.2.1 folder; Worksheet2.2.1

Objectives

- To understand that an algorithm is a process that consists of a series of steps that achieves a specific goal
- To understand algorithms can describe everyday activities and can be followed by humans and computers

Success Criteria

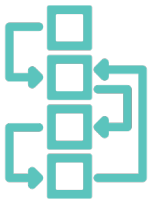
- The children can create an ordered set of instructions for making a sandwich

Vocabulary

Algorithm; instructions; sequence; input; output; process; list

1

- Explain that this unit of work is going to help the children to learn that some of the things we do every day are a 'routine' that involve doing things in order.
- Tell the children to imagine that a robot has visited class and it doesn't know how to do lots of things. Today we will be giving the robot the instructions it needs to be able to make us a sandwich!



2

- Invite the children to discuss all of the things you need to do to brush your teeth.
- Record their ideas on the IWB – do not put them into an ordered list
- Invite some of the children to begin to put the list into order - leave in mistakes or make some deliberate errors.
- When the list is complete - but incorrect. Invite the children follow the instructions on the board and act out the process (as described) for brushing teeth.
- When any errors are found, discuss and move the instruction to the right place in the sequence on the board.
- When the list of instructions is complete, read aloud. Pretend that you're out of breath with all of those words.
- Ask the children to suggest ways the instructions could have less words and make appropriate changes.
- Point out that the list is beginning to look like a 'code' that we could give to the robot. Explain that robots and computers need much clearer instructions than humans – demonstrate by saying to one child 'upstairs to do your…' mime brushing teeth and ask – do you know what I am trying to say? Do you think a robot would?

3

- Show images of the stages of making a sandwich on the IWB (Resources). Ask the children what items we need to make a sandwich. Draw a large circle on the board and label it 'Make a sandwich'.
- Ask the children to imagine that the 'Make a sandwich' circle is a room where the sandwich is put together. What would we need to put into the room?
- Work as a class to arrange the ingredients and equipment to the left of the circle – explain that what goes in to the room is the **input** to the sandwich making process, draw an arrow from the circle to the sandwich and label it **input**.
- Ask what will come out of the room. Establish that it will be a sandwich and move the sandwich to the right of the circle with an arrow from the circle to the sandwich. Label the arrow **output.**
- Invite one or two children to begin to order the instructional images inside the 'Make a sandwich' circle into the correct **steps** for the sandwich making **process**

4

Core

- Give out copies of the Worksheet2.2.1
- The children then cut the images for making a sandwich and stick them into the correct numbered box.
- The children then write a few words that describe the process under each image

Harder

- Some children could abbreviate their sentences into the minimum number of words that still convey the instruction.

Easier

- Some children could stick the images only and not write the instructions

nextsteps

Extension/Enrichment

- Children could work in pairs - one acting as a robot the other giving instructions - to guide the 'robot' around a table using only a combination of:
  - Move forward
  - Turn right
  - Stop
- Invite the children to program each other to make a real sandwich!
- Make sure the children know that a robot would do exactly what they are instructed to do…

Resources

Resources: Resource2.2.2 Folder; Resource2.2.2 (robot instructions cut up); Worksheet2.2.2; Music

Objectives

- To understand that algorithms are made up of steps
- To know that steps can be repeated
- To understand that computers need more precise instructions than humans do

Success Criteria

- The children create a sequence of instructions for a dance routine
- The children add repeat instructions for their routines

Vocabulary

Instructions; steps; order; repeat; forward; back; left; right

1

- Ask the children to identify any robots from films or TV they have seen.
- Show examples on the IWB of real robots and explain that robots need very clear instructions
- Tell the children that today they will be pretending to be robots by giving each other very clear instructions on how to dance

2

- Show instructions on the IWB for some dance moves (Resources), but not the **repeat** instruction
- Ask a child to stand up and select four of the moves and order them on the board
- Play some music and read out the dance moves in order using a robot voice
- Ask if they would like to see it again? Explain that we've only instructed the robot to perform each of the moves once
- What instruction could we add if we wanted the robot to do the dance routine again? Establish that a **repeat** instruction at the end could be added.

3

- Play the music again and with the child including the repeat
- Allow the child to continue repeating the routine and point out that he/she is beginning to look very tired
- What would happen if you carried on – would the robot dance forever? Why?
- What instruction we could add to the repeat instruction if we only wanted the robot to **repeat** the dance moves a few times? Establish that we could add repeat 3 or 4 etc.
- Change the instruction to 'repeat 4 times' and invite another child to act as the robot, this time repeating only four times

4

Core

- Hand out Worksheet2.2.2 (resources)
- The children work in pairs thinking up simple dance moves (no more than 4) for a robot to perform that is repeated 4 times.
- The children can draw/write the instructions on the worksheet
- They then take turns with one child reading out the instructions and the other acting as the robot and following them.

Harder

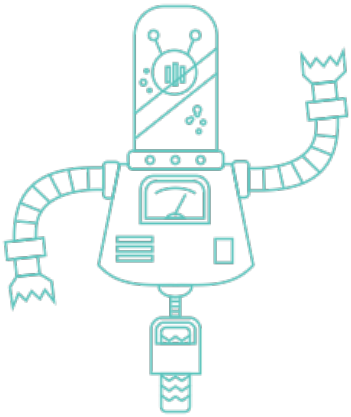
- Some children could incorporate more instructions and more loops

Easier

- Some children could use less instructions and draw rather than write them

Extension/Enrichment

- Children could swap partners and perform/read out the instructions of another pair then evaluate the other's routine using two stars and a wish.
- Children could extend their dance routines to add/repeat more moves making sure that their written/drawn instructions are amended to reflect the changes.



Resources

Computers; Scratch (Links)

Objectives

- To use digital drawing tools (Scratch) to create images

Success Criteria

- The children create their own monster sprite using Scratch

Vocabulary

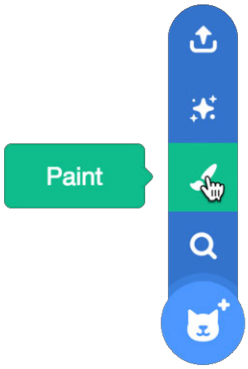
Cut; paste; undo; redo; copy; sprite

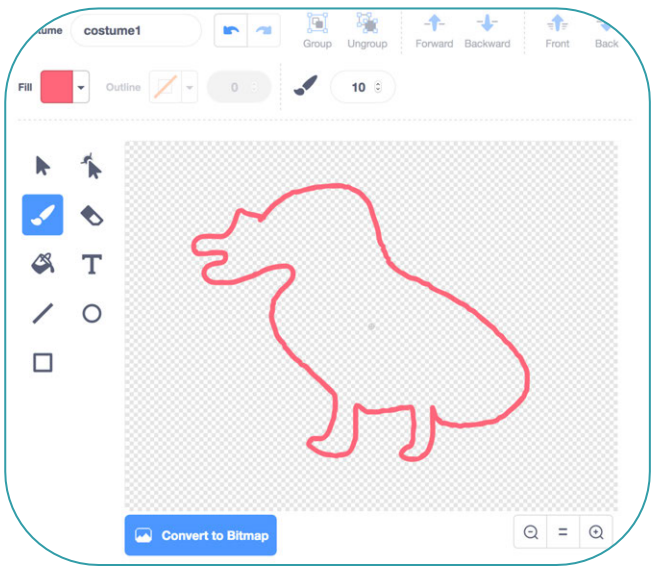
1

- Explain that, in this session the children will be drawing their own monster character that they will learn to animate in later sessions
- Load Scratch and model how to delete the default cat sprite

2

- Choose 'paint new sprite' and demonstrate how to draw a basic monster outline using Scratch's paint editor
- Show how to use the fill tool to colour the monster in
- Then use the shape tool and choose 'not filled' shape and draw an eye
- Change the colour and fill the eye with a different colour
- Show the children how to erase mistakes using the erase tool and undo
- Model a few other features and click OK to return to the main Scratch interface
- Show the children that we can now move this sprite with the mouse around the stage





3

Core

- The children use the drawing tools in Scratch to create their own monster sprite
- They use paintbrush and fill tools to colour in their sprite

Easier

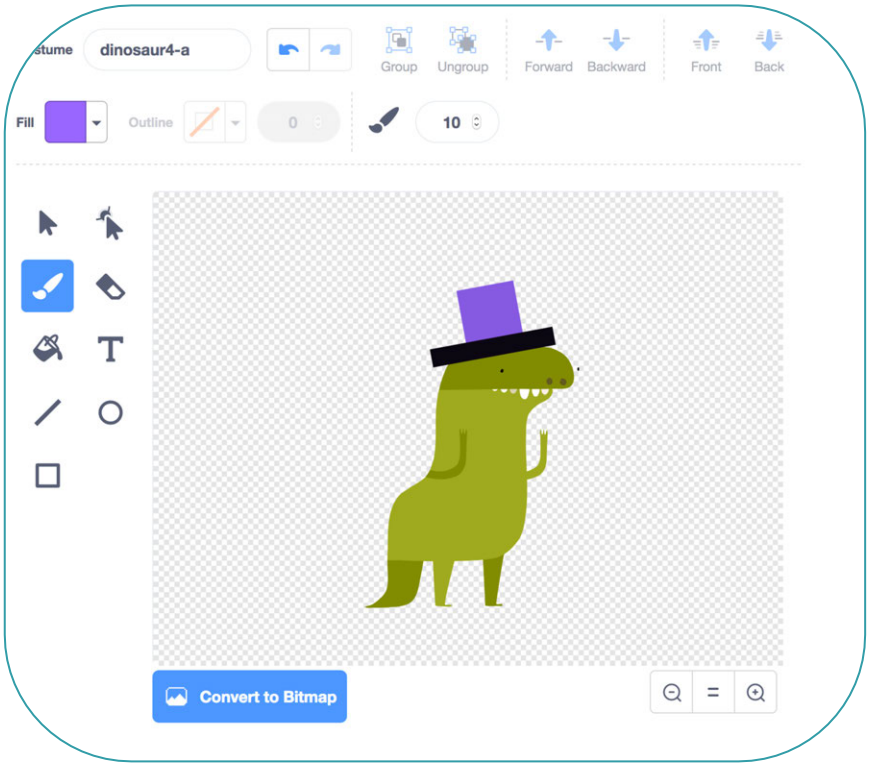
- Children could use imported pictures to edit with support

Harder

- Children could explore grouping the sprite and changing its size and direction in the paint editor

Extension/Enrichment

- Children could edit existing sprites by choosing a new sprite and then using the edit costume function
- They could add to the existing sprite (e.g. add a hat)
- They could also change colours
- They could delete part of the sprite using the eraser tool



Resources

Resources: Scratch Software; Resource2.2.4 (support materials)

Objectives

- To program a simple animation involving movement

Success Criteria

- The children create a simple animation where their sprite moves on the stage

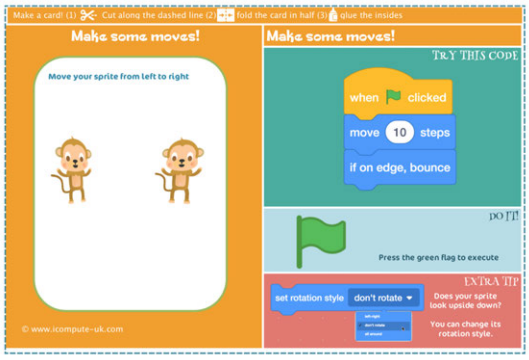
Vocabulary

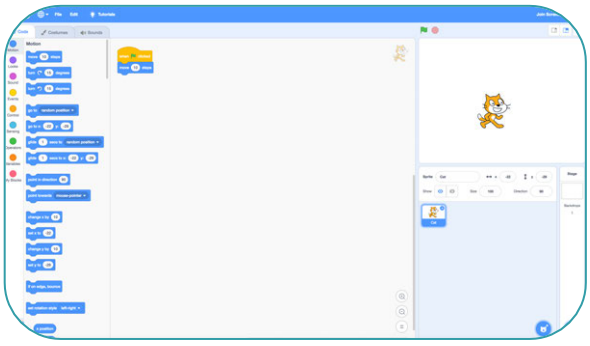
statement; move; negative; steps; sprite; execute

1

- Explain that, in this session the children will be learning how to program their sprite to move on the stage
- Load scratch and open a new sprite 
- Show the children the palettes on the left side of the scratch interface and click on the blue motion palette
- Explain that this is where the statements are that will make our monster move
- Drag out a 

block and drop in the script area





2

- Ask the children why they think the sprite hasn't moved
- Click on the block and show the children that the block makes the sprite move
- Explain that we need to tell the program to start – we need to **execute** the program
- Show the control palette and drag out 

to the script area and show that it snaps on top of the move block like LEGO construction bricks
- Click the green flag to demonstrate the code executing

3

Core

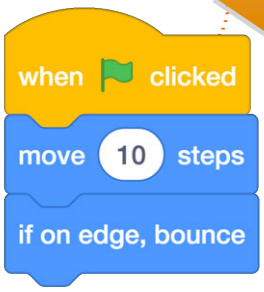
- Hand out Resource2.2.4 (resources)
- The children program their monster sprites to move on the stage using motion blocks
- They add a control block to their programs to make the animation execute
- They **test** and **debug** their programs

Harder

- Ask the children to add a statement that will make their monster sprite repeatedly move, when executed, using control blocks
- Can they add any statements that will stop the sprite disappearing off stage by bouncing off it?

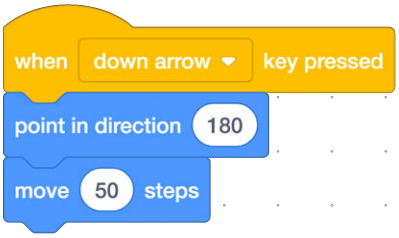
Easier

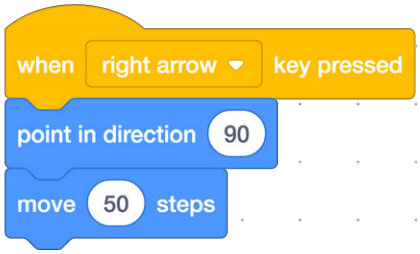
- Some children may benefit from pair work or adult support



Extension/Enrichment

- Challenge the children to experiment with different motion blocks to get their sprite to move in different ways
- Can they get them to move backwards?
- What about moving up and down?





Resources

Resources: Scratch; Children's scratch project files from session 2.2.4; Resource2.2.5 (support materials)

Objectives

- To write a simple program that produces an output (text)

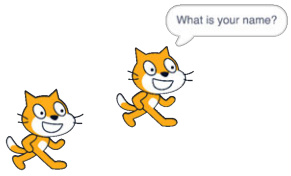
Success Criteria

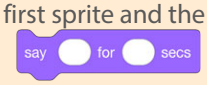
- The children's sprites say something when executed – in a speech bubble or as a sound

Vocabulary

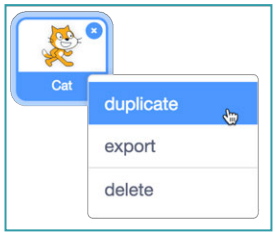
Input; output; sprite

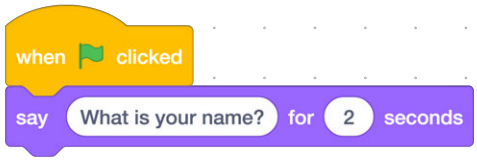
1

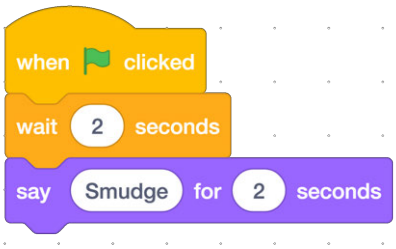


- Tell the children that today they will be learning how to program two sprites to talk to one another
- Load Scratch and model how to duplicate a sprite by right-clicking or tapping and holding (for tablets); or opening a new sprite from file
- Select the first sprite and the purple 'looks' category
- Drag out a 

block and drop in the script area
- Type 'Hello" in the first white field and add 2 seconds
- Click/tap the block to demonstrate that the sprite says 'hello' for 2 seconds
- Model how to edit the text, by double clicking on 'Hello' and changing it to a question, also increase the wait time to 5 seconds
- Click/tap the block to show the changes execute

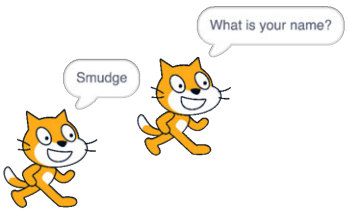






2

- Add a flag control for Sprite1 to instruct the code to execute when the flag is pressed
- Select the second sprite and add code to make this sprite reply to the question
- Execute the code and observe that both sprites are talking at the same time
- Show the control category and ask what block could we use to make Sprite2 wait a little while to give Sprite1 time to ask the question
- Establish that we could use a 'wait' block, drag one out and snap on top of the 'say' block
- Execute the code again to demonstrate that, this time, Sprite2 waits for Sprite1 to ask the question before replying



3

Core

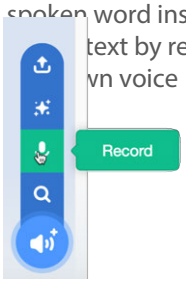
- The children amend their Scratch project files from the last session (Resources) to have two sprites appearing on stage (either by duplicating their own monster sprite or importing a new one)
- They then add blocks of code to make their sprites talk to one another
- They could use Resource2.2.5 if necessary, which contains some ideas for conversation

Harder

- The children could program longer conversations and use a combination of 'say' and 'think' blocks to make a short animated story

Easier

- The children could use spoken word instead of text by recording own voice



Extension/Enrichment

- Encourage the children to experiment with the sound, motion and colour categories to add interest to their animations
- Can the children program one sprite to move towards the other?
- Play a sound instead of using a Say block?
- Change the costume of the Sprite after they have spoken?
- As an alternative to using **Say** and **Wait** blocks, challenge the children to use **broadcast** blocks which is a better way of programming sprite interaction
- Broadcast blocks provide a means of triggering blocks of code to execute only when an internal 'message' is received
- If necessary model how this works with the children by writing on the board that they are to perform three star jumps when they receive a **broadcast** command of 'Jump'
- Pass out notes to the children some that are blank, along with some bearing the command 'Jump'

Resources

Resources: Scratch; Resource2.2.6.sb3 (Links); Children's Scratch projects from earlier sessions

Objectives

- To combine images and text to create a simple animation

Success Criteria

- The children make a simple animation which has a background and sprites moving and talking

Vocabulary

Edit; undo; redo

1

- Load Scratch on the IWB and open a scratch project which has two sprites moving and talking (Resource2.2.6)
- Draw attention to the lack of background and demonstrate how to draw a new background using the paint editor
- Open the Paint Editor by choosing 'Stage', then 'choose a backdrop' and selecting 'Paint'
- Demonstrate drawing a simple background by changing the paintbrush colour and brush size, drawing some grass and using the fill tool to fill the area just drawn
- Show the children how to use the shape tools to draw squares and circles
- Fill them with colour using the fill tool
- Make some mistakes and show the children how to correct them using undo/redo and the eraser tool

2

Core

- Children amend their monster scratch project files (Resources) to draw a background scene for their monster animation
- They use paint editor tools to make their backgrounds as artistic as possible

Harder

- The children could create more than one background and switch between them during their animation. NB: To add code to a background choose the stage, then the 'code' tab

Easier

- The children could edit imported background images with support

Extension/Enrichment

- Encourage the children to experiment with the sound, motion and colour categories to add interest to their animations
- Can the children program one sprite to move towards the other?
- Play a sound instead of using a Say block?
- Change the costume of the Sprite after they've spoken?

Assessment

Record of progress Write names in the appropriate box, with jottings on children on children whose attainment differs markedly from their group.	Expectations What children know, understand and can do
Some children will have not made as much progress and will: 	✶ Know that Scratch can be given commands to produce specific effects on screen ✶ Produce a command that achieves a simple effect (E.g. movement)
Most children will: 	✶ Execute short a sequence of commands that results in an effect ✶ Move a sprite in one direction on screen using steps ✶ Program and test a simple program
Some children will have progressed further and will: 	✶ Program a sequence of commands that results in a number of planned effects ✶ Test and correct simple programs ✶ Evaluate their own work and comment on improvements
Computing Guide - Working towards Yellow	