

Year 3 Computing – Events and Actions

What I should already know

- **Programming** is when we make a set of instructions for computers to follow.
- We use **programs** in order to code our own stories and animations. Scratch Jr involves sprites (characters on the screen).
- We use **algorithms** (a set of instructions to perform a task) to program the sprite to do different things

What I will learn by the end of this unit

- **Programming** is when we make a set of instructions for computers to follow.
- **Scratch** is a program that we can use in order to code our own stories and animations. We can use **event and action command blocks** in order to make sprites carry out acts when certain prompts take place.
- We use **algorithms** to sequence movements, actions and sounds in order to program effective animations.

Event and Action Blocks

-Event Blocks: Event blocks are coloured yellow and are used to sense different events that happen, e.g. the green flag being clicked, when a key is pressed, or when a sprite is pressed. They are needed for every project.

	Putting this at the beginning of your sprites programming will mean the program will only start when the green flag is clicked.
	This will begin a sprites actions when the selected key from the drop down bar is pressed. The drop down bar allows you to select any key on the computer keyboard. Meaning you can control your sprites movements by setting up movements in connection with different keys.
	A sprites actions will start when you click on it. For example: If that picture was the correct answer to a question you could get it to move.
	This will start a sprites action when the backdrop changes. Using the drop down bar you can select any backdrop you have in the selected backdrop library for your program. For more info on backdrops see page on creating and selecting backdrops.

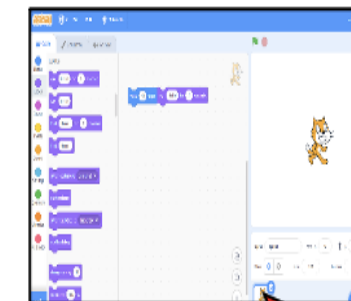
-Action Blocks: Action blocks include 'Motion' blocks (coloured blue), 'Sound' blocks (pink) and 'Looks' blocks (purple). They make the sprite move, make sounds and change appearance when the event is triggered.

<p>Motion</p>	<p>Looks</p>	<p>Sound</p>
----------------------	---------------------	---------------------

The Basics of Scratch

-What is Scratch? Scratch is a website/ app that lets us code our own stories, games and animations.

-Scratch helps us to learn how to use programming language, whilst also being creative and using problem-solving skills.

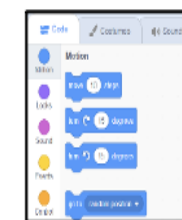


There are three main areas in Scratch:

-The Blocks Palette (on the left) contain all of the different blocks: puzzle piece commands which control the animation.

-Code Area (in the middle) is where the blocks are placed to create a program.

-Stage with Sprite (right) is where the output of the program is presented. The sprite is the character.



Adding/Removing Sprites: This can be done here, at the bottom of the stage. There are many sprites to choose from.

Attributes: There are three attributes of the sprite which we can change to make our animation: Code, Costumes, Sounds.

-Backdrops: Backdrops can be added by clicking on this icon (bottom right of the screen, below the stage).



Sequencing and Algorithms

-A sequence is a pattern or process in which one thing follows another. In Scratch, blocks can stack vertically on top of one another to create sequences.

-Designing an algorithm (set of instructions for performing a task) will help you to program the sequence that you require.



-Programming is when we move the blocks into the position (based on our algorithm design). Programming uses a code that the computer can understand.

Trialling and Debugging

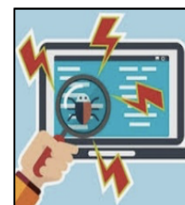
-Programmers do not put their computer programs straight to work. They **trial** them first to find any errors:

-Sequence errors: An instruction in the sequence is wrong or in the wrong place.

-Keying errors: Typing in the wrong code.

-Logical errors: Mistakes in plan/thinking.

-If your algorithm does not work correctly the first time, remember to **debug** it.



Disciplinary Skills

- Design, write and debug programs that accomplish specific goals.
- Solve problems by decomposing them into smaller parts.
- Use sequence, selection and repetition in programs.
- Detect and correct errors in algorithms and programs.

Key Vocabulary

motion	event	logic	move	resize
extension block	pen	action	errors	test