

We are software Developers

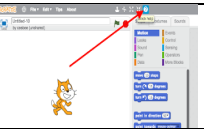
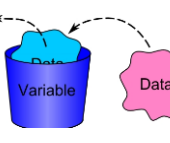
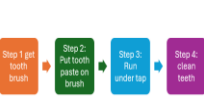
Prior Knowledge

You programmed using BeeBots in **Unit 1.1: We are treasure hunters.**

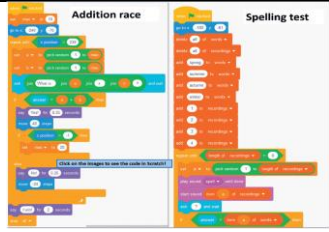
You programmed on screen in **Unit 2.1: We are astronauts.**

You will build on learning from **Unit 3.1: We are programmers.**

Key Vocabulary:

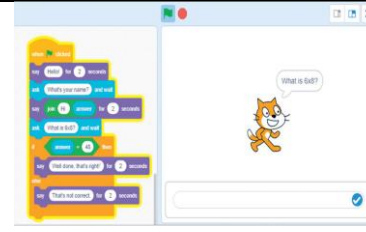
algorithm:		a sequence of precise instructions or steps (sometimes a set of rules) to achieve an objective	repetition		programming construct which allows a group of instructions to be repeated a number of times, or until a certain condition is met
bug:		an error or mistake in a program or algorithm, causing the computer or robot to behave in a manner that was not originally intended	scratch		simple, block-based programming language in which programs for characters are built by snapping together code blocks
debug:		correct mistakes in a computer program or algorithm	sequence		placing programming instructions in order, so that each happens one after the other
input		data supplied to a computer, in this case, pressing buttons on the robot	sprite		a graphical character in a program that can be given its own sequence of instructions
output		information produced by a computer – in this case, an animation	variable		let's computer programs store, retrieve or change simple data. Typically thought of as a particular location in the computer's memory that holds a specific item of data
program		a sequence of instructions (or sometimes a set of rules) that can be followed by a computer			

Key learning: What is debugging and why is it important?



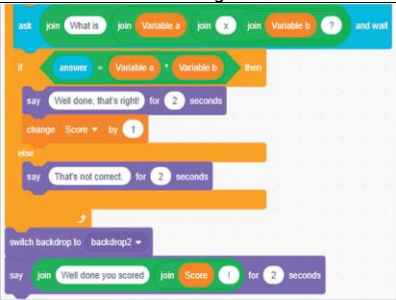
Session 1: Analysing an educational Scratch game

How do **if** else statements work?



Session 2: Creating a simple program for a game

How do **if else** statements work?



Session 3: Adding repetition and messages

Why is repetition useful in coding?

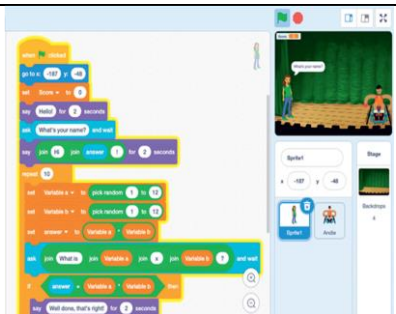
Does it matter the order in which the numbers in a multiplication sum are typed?



Session 4: Improving the interface of the game

What sign represents multiplication in coding?

Why does a game need an interesting interface?



Session 5: Building in progression

Why does a game need an interesting interface?

Why is it harder to code a division than multiplication game?



Session 6: Testing and refining the game

How can I add levels to my game?

Key Applications (apps)



Next Steps

You will further develop their programming skills in **Unit 4.2: We are makers** and **Unit 5.1: We are game developers**.