Prior Knowledge

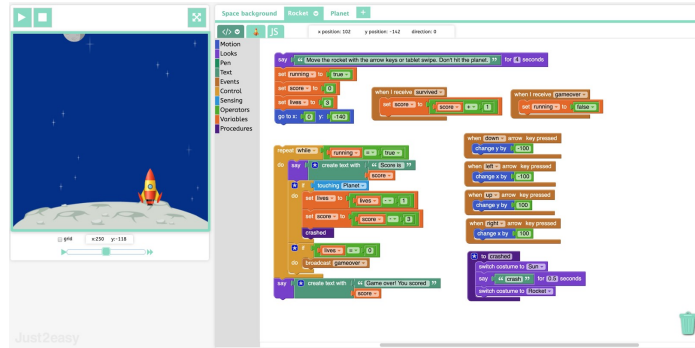**Sequencing with block coding**
Repetition
Selection
Events: a programming construct in which the flow of the program is determined by events such as user actions (mouse clicks, key presses), sensor outputs, or message passing from other programs or threads.)

I can create a simple set of instructions (sequence, code) which includes loops (repetitions) and Selection to program (control) an on screen object (sprite) or floor robot and to be able to refine and edit the process (de-bug) to create a game simulation or model. (j2code-ks2-y5 lessons 1 and 2)

I can design solutions (algorithms) that use repetition and two way selection i.e. if, then, else

I can use computational thinking to edit (de-bug) deconstruct and evaluate a program to solve a problem or make the program more challenging or refined

I can name different types of inputs and outputs (Events) and use them

NC Learning Objectives

To design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts

To use sequence, selection, and repetition in programs; work with variables and various forms of input and output

To use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs

Evidence, Assessment and Advice

Teachers to set work using Google classroom in order for children to easily access resources.

Children should save their Busy Things project on chromebooks as well as their work on J2code.

For code.org, log ins to be created in order for data to be saved. Do not create personal log ins. Create picture log-ins for GDPR reasons. See Computing lead for support in creating log ins

Scheme of Lessons

In this set of lessons, we are looking at variables.
A variable stores a piece of information in a computer's memory while a pro gram is running, that can be retrieved when needed.
A variable is an example of a data structure. As pupils move on to secondary school, they will learn about other data structures such as arrays.
A variable can be a number or text or perhaps true/false.

Please note that lesson plans are built into these PowerPoints. They are visible under the slides when not in slideshow mode.
Lesson 1–: Unplugged– Pupils will begin to understand the concept of variables through practical examples.
Lesson 2: Busy Code Section 5 Variables– Tutorial and Challenge dance steps 2. (Also contains a recap of Looping)
Lesson 3: Busy Code section 5: Variables– Project: Call and Response. **The children need to save their 'projects' Busy things in order to be accessed later by children, teachers and key leaders.**
Lesson 4: J2code– Main Task 1 and 2 on powerpoint. Task 1. Add in code to decrease the score by 3 when the rocket crashes. Task 2. Add in a "lives" variable so that the game ends after 3 crashes.
Lesson 5: J2code- Challenge section- Children can "clone" the planet sprite and change it to a different planet. They can then experiment with the code so the object of the game is to hit the new planet. Look .**There is a chance to share the game with a younger year group here.**
Lesson 6: Code.org- Conditionals (selection):
Course F:
Course F (2023) - Code.org
Variables:
Lessons 6 - 12

Vocabulary

Variables: A variable is a simple way of storing one piece of information somewhere in the computer's memory whilst a program is running, and of getting that information back later.
Algorithm – a process or set of rules to be followed
Debug – identify and remove errors from (computer hardware or software).
Flowchart – a diagram of the sequence of movements or actions of people or things involved in a complex system or activity.
Output – produce, deliver, or supply (data).
Symbol – a thing that represents or stands for something else
Script – a program or sequence of instructions that is interpreted or carried out by another program
Commentary – explanation or annotation in the source code of a computer program.
Palette – a table in which selected colours
Trace– read through code

What will I know by the end of this unit?

I can create a simple set of instructions (sequence, code) which includes Variables to program an on screen object (sprite) or floor robot and to be able to refine and edit the process (de-bug) to create a game simulation or model.
I can combine a variable with relational operators (<=>) to determine when a program changes e.g. if score >5, say "well done"
I can use computational thinking to edit (de-bug) deconstruct and evaluate a program with smaller parts to solve a problem or make the program more challenging or refined
I can name different types of inputs and outputs (Events) and use them within simple games, simulations or models
I am able to use post tested loops e.g. until, and a sequence of selection statements in programs, including if, then, else
I can predict what will happen in a program or algorithm (e.g. change of output) when the input changes (e.g. sensor, data or event)