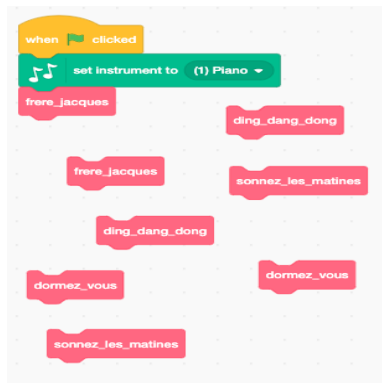
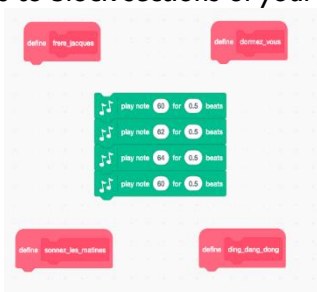


Introduction to programming and sequencing

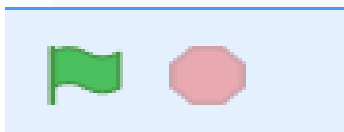
Using the programme Scratch, you will use blocks to provide instructions to perform a sequence.



You will use subroutines to block sections of your programme.



You will use the stop start buttons to your programme.



and play

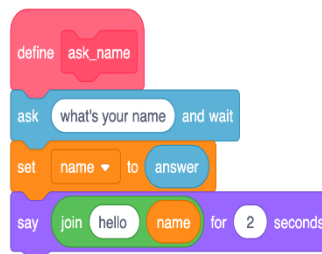
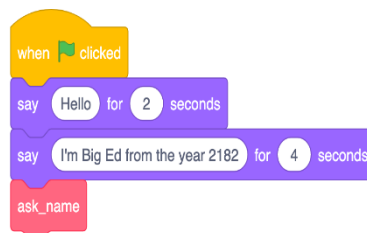
Sequences and Variables

Common misconceptions or mistakes that can be made with this are below – you need to watch out for these as you use Scratch.

- A variable can store multiple values – it may store the history of values assigned to it.
- A variable is just a pairing of a name with a value. It is not stored within the computer.
- Assignment statements such as $a = b$ work in both directions.

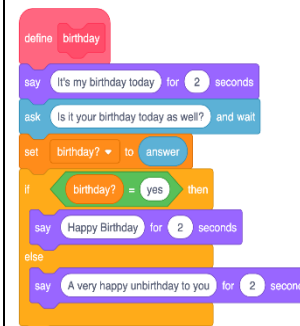
Meeting BigEd! - ncce.io/biged

Using the Scratch programme, you will predict what you think will happen from the programme – does it do what you thought? Why? If not, why not and what could you change or add?

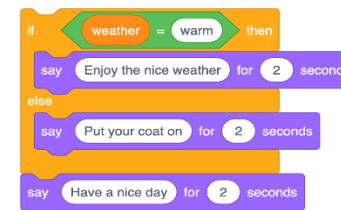


Selection

When setting a series of questions, you need to consider how the program will respond. What do you think will happen? Are you right? If so, why? If not, why? You will investigate and modify a series of questions and determine the required solution.



You will investigate IF statements, which will allow you to check a condition and carry out an operation to see if it is TRUE or FALSE.



Keywords and definitions

Sequencing – instructions performed in order, with each executed in turn.
 Subroutines – part of a programme.
 Instructions – how to complete an action.
 Execute – completing an action.

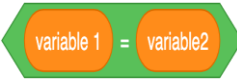






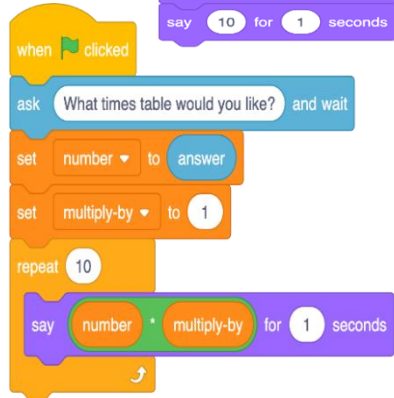
Keywords and definitions

Variables – things that can change.
 Commands – programming instructions.
 Input – process – output – flow of information.
 Storage – how data is stored and can be retrieved.

Sources of data

IF Statement – a block that will allow you to carry out and check to see if a condition of an operation is true or false.

Programming essentials in Scratch

Operators	Count Controlled Iteration	Problem Solving
<p>Comparison operators compare the values of expressions.</p> <p> That variable 1 is equal to variable 2</p> <p> That variable 1 is greater than 10</p> <p> That 200 is less than variable 1</p> <p> That 'number' is greater than 30 and 'city' is equal to Athens (when both are true)</p> <p> That 'number' is greater than 30 or 'city' is equal to Athens (when either are true)</p> <p> That 'age is greater than 18'</p>	<p>You will use this counting programme to code. You will need to look for patterns and repetition in the programme and how they can be modified.</p> <p></p> <p>Debugging is the process of looking for errors in your code and taking steps to fix them.</p> <p></p>	<p>Problem Solving – Moves Like Jim!</p> <p>Moves Like Jim is a dance game that requires the user to make Jim perform a dance move by pressing keys on the keyboard at the right time. Your end of unit task is to complete a partially built programme that allows the user to play the game and to make Jim perform some smooth moves.</p> <p>Before you start the task, spend some time looking at the code and running it to see what happens. (ncce.io/movejim)</p>
<p>Keywords and definitions</p> <p>Logic operators – perform logical operations. Comparison operators – operators that compare the value of expressions.</p>	<p>Keywords and definitions</p> <p>Iteration – repeatedly executing instructions. Count-controlled – where iteration is done through the method of counting. Condition-controlled – where the iteration is Debugging – where errors are identified and corrected.</p>	<p>Keywords and definitions</p> <p>You will use these words to check your understanding of the unit of work we have covered this term: Sequence Subroutines Selection If/Else Condition Comparison Operator Logic Operator Iteration</p>