

Learning Programme

Python – 2nd Year

Topic/Content	Objectives/Skills	Homework	Assessment	Success Criteria (for E/S/D at KS3)	Stretch & Challenge (Thirst for Learning)
Introduction to Python	Create simple code including the input and print scripts.		<p>The homework will be used when forming a judgement/grade for the end of unit mark</p> <p>Students will complete a programming test on the computer. This will test all of the programming techniques that the student has learnt so far.</p>		<p>Students can download python free and use it at home to practise programming.</p> <p>Plenty of resources on the shared area for students to make use of both in and out of school.</p> <p>Students can use websites such as the following to complete tutorials to learn new techniques:</p> <p>https://www.tutorialspoint.com/python</p> <p>https://www.learnpython.org</p>
Print Statements	Create variables and declare the data types for variables.				
Using Inputs in Python	Create variables. Allow users to answer questions and store the response in a variable and output elsewhere	Complete section 1 of the python section on the code academy website			
Using Numbers in Python	Be able to declare the data types for variables. Use integers and operators to complete calculations				
Inputs and Integers	Combine strings and integers to create a small program	Complete section 2 of the python section on the code academy website			
Introduction to IF statements	Understand the term Selection Know what an IF statement does Make use of IF statements in your programs				
Selection Recap	Be able to make use of IF statements in complex scenarios	Complete section 3 of the python section on			

		the code academy website			https://www.codecademy.com/learn/learn-python
Loops	Understand what a loop is used for when programming Be able to make effective use of the WHILE statement	Complete section 4 of the python section on the code academy website			www.thepythonguru.com/
Programming Challenges	Make effective use of a number of different programming techniques	Revise for the end of programming assessment			
Python Assessment	Complete the end of unit programming assessment making use of the programming techniques previously used				

Excellent	Secure	Developing
<ul style="list-style-type: none"> Recognise pieces of code that can be used to solve two or more similar problems Can appropriately select pieces of pre-prepared code, combining them with my own code, to solve a problem Can translate a specification into the code required to produce a solution 	<ul style="list-style-type: none"> Can use and manipulate numeric variables and string variables in my own programs. Can use and manipulate lists and/or tuples in my programs. Can use Python code conventions (including importing modules) accurately, and vary the rules within the programs. Can write a short program and then use Python to code and execute the program, knowing the 	<ul style="list-style-type: none"> Can recognise different types of data: Integer, Float, Long and Complex Can code using syntax and typography, carefully and precisely. (i.e. text, language and symbols the computer will understand) Can code using selection and iteration, such as for and while loops Can describe the difference between numeric and string variables

<ul style="list-style-type: none">• Can rationalise if Python is an appropriate piece of software to solve a given problem• Can justify when and why I would Python as a solution, and outline the benefits and limitations of the software• Can explain the meaning of text in an Python program, without necessarily using the Python interface• Be able to write programs in a text-based language and be able to create your own data structures.	<p>limitations/suitability of the software.</p> <ul style="list-style-type: none">• Can write procedures with parameters and functions that returns a value• Can justify when and why I would Python as a solution• Correctly use variables, lists and simple procedures in your programs.• Use selection and repetition correctly in your programs.	<ul style="list-style-type: none">• Can describe what a simple program given to me will do, and be able to describe some of the coding that I will need in my programs• Can run and test a simple program given to me, and then make a similar one of my own using the same ideas• Be able to plan a sequence of instructions for something that you want to happen.
--	---	--