




Year 9 Spring Term
'An ambitious curriculum that meets the needs of all'

Medium Term Planning – Python Programming & Algorithms

Curriculum Intent	In addition to working further on objectives from Year __, pupils will be taught, following National Curriculum guidelines, the following this term:
Skills/National Curriculum Links	<p>Computing – KS3</p> <p>Key stage 3 Pupils should be taught to:</p> <ul style="list-style-type: none"> • design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems • understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem • use two or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions • understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming; understand how numbers can be represented in binary, and be able to carry out simple operations on binary numbers [for example, binary addition, and conversion between binary and decimal] • understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems • understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits • undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users • create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability • understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognize inappropriate content, contact and conduct and know how to report concerns.
Numeracy	Operators != == < <= > >= BIDMAS + / * - // %
Literacy	<p>Vocabulary Tier 2: program, code, BIDMAS</p> <p>Vocabulary Tier 3: Integrated development, IDLE, interactive mode, Script mode, variable, string, syntax, assignment statement, augmented assignment operator, data type, integer, float, round, selection, sequence, iteration, module, function, syntax error, logic error, debug, binary search</p> <p>Reading: Presentations, worksheets, and homework</p> <p>Writing: Complete worksheets and skill task</p> <p>Oracy: Learn how to pronounce difficult or new keywords</p> <p>SMSC: Understand identity theft, how the online world can be fun but dangerous if not used sensibly</p> <p>PSHE: Understand how programming is being used in everyday life</p> <p>Careers: Data analysis, software programmers, E-commerce, working in banks etc..</p> <p>Literacy: literacy slide will provide a definition of the keyword, antonym and synonym</p>
Adaptation	Throughout this topic, quality first teaching will provide differentiation:
QFT/SEND Provision	<p>By product: Learning will produce work on a variety of different levels, a mix of individual, think pair share, designing original maters, Q&A with teacher, teacher marking and self-marking.</p> <p>By resource: presentations, worksheets with extension tasks</p> <p>By Intervention: by providing different levels of supervision/support, seating plan, use of TA</p> <p>By Progressive Questioning: exploring pupils' understanding through interactive dialogue.</p> <p>By Grouping: according to prior attainment, gender, social preference, preferred learning style.</p> <p>By Task: Pupils should be involved in the identification of targets which are meaningful to them and in the selection of an appropriate task from the given range.</p> <p>By Offering Optional Activities: In class or as homework, to extend learning.</p> <p>This QFT/SEND provision will be explicit within the lesson-by-lesson schemes of work.</p>

Implementation Curriculum Delivery	To be able to: This SOW is currently being adapted and updated to incorporate searching and sorting algorithms.																						
Learning Outcomes (Knowledge)	<table border="1"> <tr> <td data-bbox="279 208 523 880" rowspan="10">Introduction to Python</td><td data-bbox="523 208 906 398">Strings print function</td><td data-bbox="906 208 1455 398"> Know what Python is and some of the applications it is used for Run a simple Python program in Interactive mode using the print function Write, save and run a program in Script mode Understand what a syntax error is </td></tr> <tr> <td data-bbox="523 398 906 589">Variables</td><td data-bbox="906 398 1455 589"> Know the rules for variable names Use variables in a program Use + concatenation Use # to make comments Use index [] </td></tr> <tr> <td data-bbox="523 589 906 689">Arithmetic</td><td data-bbox="906 589 1455 689"> Perform arithmetic using the BIDMAS rule Write a program involving input, calculation and output </td></tr> <tr> <td data-bbox="523 689 906 790">Data Types</td><td data-bbox="906 689 1455 790"> Understand the importance of using correct data types: string, integer or float Use the integer, float and round functions </td></tr> <tr> <td data-bbox="523 790 906 902">Selection Celebrity</td><td data-bbox="906 790 1455 902"> Use the print() and input() function Create sensible variable names to store a value Use selection </td></tr> <tr> <td data-bbox="523 902 906 1025">Selection Quiz ith score system</td><td data-bbox="906 902 1455 1025"> Use selection statements Learn how to create a quiz Learn how to use a score system </td></tr> <tr> <td data-bbox="523 1025 906 1160">Selection 8 Ball game with random randint</td><td data-bbox="906 1025 1455 1160"> Use random randint to generate a random number Create a magic 8 ball game Have 20 responses </td></tr> <tr> <td data-bbox="523 1160 906 1238">iteration for</td><td data-bbox="906 1160 1455 1238"> Create iterative code Use a for loop to repeat a section of code </td></tr> <tr> <td data-bbox="523 1238 906 1350">iteration while</td><td data-bbox="906 1238 1455 1350"> Use a while loop in a program Use an if statement within a while loop Use a function to generate a random number </td></tr> <tr> <td data-bbox="523 1350 906 1485">Writing Algorithms</td><td data-bbox="906 1350 1455 1485"> Learn to write algorithms in pseudocode Identify different types of program errors: syntax errors, run-time errors and logic errors and how to fix them </td></tr> </table>	Introduction to Python	Strings print function	Know what Python is and some of the applications it is used for Run a simple Python program in Interactive mode using the print function Write, save and run a program in Script mode Understand what a syntax error is	Variables	Know the rules for variable names Use variables in a program Use + concatenation Use # to make comments Use index []	Arithmetic	Perform arithmetic using the BIDMAS rule Write a program involving input, calculation and output	Data Types	Understand the importance of using correct data types: string, integer or float Use the integer, float and round functions	Selection Celebrity	Use the print() and input() function Create sensible variable names to store a value Use selection	Selection Quiz ith score system	Use selection statements Learn how to create a quiz Learn how to use a score system	Selection 8 Ball game with random randint	Use random randint to generate a random number Create a magic 8 ball game Have 20 responses	iteration for	Create iterative code Use a for loop to repeat a section of code	iteration while	Use a while loop in a program Use an if statement within a while loop Use a function to generate a random number	Writing Algorithms	Learn to write algorithms in pseudocode Identify different types of program errors: syntax errors, run-time errors and logic errors and how to fix them	
Introduction to Python	Strings print function		Know what Python is and some of the applications it is used for Run a simple Python program in Interactive mode using the print function Write, save and run a program in Script mode Understand what a syntax error is																				
	Variables		Know the rules for variable names Use variables in a program Use + concatenation Use # to make comments Use index []																				
	Arithmetic		Perform arithmetic using the BIDMAS rule Write a program involving input, calculation and output																				
	Data Types		Understand the importance of using correct data types: string, integer or float Use the integer, float and round functions																				
	Selection Celebrity		Use the print() and input() function Create sensible variable names to store a value Use selection																				
	Selection Quiz ith score system		Use selection statements Learn how to create a quiz Learn how to use a score system																				
	Selection 8 Ball game with random randint		Use random randint to generate a random number Create a magic 8 ball game Have 20 responses																				
	iteration for		Create iterative code Use a for loop to repeat a section of code																				
	iteration while		Use a while loop in a program Use an if statement within a while loop Use a function to generate a random number																				
	Writing Algorithms	Learn to write algorithms in pseudocode Identify different types of program errors: syntax errors, run-time errors and logic errors and how to fix them																					
Current learning to be developed in the future within:	This will support pupils to make an informed option choice with the programming elements of the GCSE specification. This is being built upon from learning sequence and iteration in year 7 using Microbits, and sequence using Python, and in year 8 sequence and selection using Python and Edison Robots. In year 9 sequence and selection are revisited and further developed using iteration count-controlled and iteration condition-controlled.																						
Assessment	<ul style="list-style-type: none"> Refer to assessment maps for formative and summative assessment opportunities. 																						
Impact	<ul style="list-style-type: none"> Learning will be tested during Summative Assessment 1 and 2 Assessment results will indicate pupils emerging, developing, securing or mastering. Data review documentation will indicate if pupils are underachieving, meeting or exceeding MEG grade. In line with the departmental marking policy, quality written feedback will be provided for the specified marked piece 																						