Curriculum Knowledge Map

Computing/CS

Year 8	Rotation		
	Digital Literacy	Introduction to Python Programming	Computer Systems
	This unit takes learners from having very little knowledge of spreadsheets to being able to confidently model data with a spreadsheet. The unit uses engaging activities to progress learners from using basic formulas to writing their own COUNTIF statements. This unit will give learners a good set of skills that they can use in computing lessons and in other subject areas.	This unit introduces learners to text-based programming with Python. The lessons form a journey that starts with simple programs involving input and output, and gradually moves on through arithmetic operations, randomness, selection, and iteration. Emphasis is placed on tackling common misconceptions and elucidating the mechanics of program execution. A range of pedagogical tools is employed throughout the unit, with the most prominent being pair programming, live coding, and worked examples. The Year 7 Programming units (scratch) are prerequisite for this unit.	This unit looks at how networking hardware is used for communication including the Internet before identifying how users and organisations can protect themselves from cyberattacks. Learners will develop an understanding of the terms 'internet' and 'World Wide Web', and of the key services and protocols used. Practical exercises are included throughout to help strengthen understanding. .

Declarative
What should they know?
	Data Modelling
	• The concept of spreadsheets and why they are useful.
	• How to navigate a spreadsheet via its rows and columns
	• How to identify columns, rows, cells and cell referencing
	• Formulas always begin with an = symbol
	• The name given to each of the +,-,*,/ symbols.
	• What each of the above symbols carry out
	• How a range of cells is defined/represented
	• How errors look like (#REF!)
	• What basic functions do:
	  ○ SUM
	  ○ MIN
	  ○ MAX
	  ○ AVERAGE
	• The name given to cells in a horizontal line.
	• The name given to cells in a vertical line.
	• Workbooks can be made up of separate 'sheets'.
	• You can reference cells in numerous different sheets.
	• The meaning of data
	• The meaning of information
		• Algorithms are step by step instructions that computers can follow.
		• The concept of IPO, that computers take input, process and output.
		• Variables are locations in memory that computers store data.
		• What constitutes a program and how they are made?
		• The difference between algorithms and programs
		• That machines need translators for executing programs.
		• The basic characteristics and offerings that IDE's provide.
		• Binary selection can be used to control the flow of a program.
		• The symbols for arithmetic operators and comparison operators and what each does.
		• The meaning of the term 'Syntax' and 'Syntax Error'
			Networks & Security
			• What a computer network is and how they are made up
			• Basic network hardware that is used to connect devices and networks.
			• The term LAN and WAN and the characteristics/definition of both
			• Basic understanding of the term protocol
			• How data is transmitted between computers across networks
			• The term 'bandwidth' and how bandwidth affects network performance.
			• The different types of connections (wired/wireless) alongside advantages and disadvantages of both.
			• What the internet is
			• How data travels between computers across the internet (submarine cables resource)
			• The difference between the internet and the world wide web
			Security
			• The difference between data and information
			• The need for the Data Protection Act
			• That human errors pose security risks to data.
			• The term 'Hacking' in the context of cyber security
			• The need for the Computer Misuse Act
			• The term Malware as 'Malicious Software'

Curriculum Knowledge Map

			• The most common Malware Threats • Effective preventative measures for Cyber Attacks
Procedural *What should they be able to do?*	Excel Data Modelling • Identify columns, rows, cells, and cell references in spreadsheet software. • Use formatting techniques in a spreadsheet. • Use basic formulas with cell references for a calculation in a spreadsheet. • Use formatting techniques in a spreadsheet. • Use the autofill tool to replicate cell data. • Use the functions SUM, COUNTA, MAX, MIN and average in a spreadsheet. • Create appropriate charts in a spreadsheet. • Use a spreadsheet to sort and filter data. • Use the functions COUNTIF and IF in a spreadsheet. • Use conditional formatting in a spreadsheet. • Collect data. • Analyse data • Explain the difference between data and information. • Explain the difference between primary and secondary sources of data	• Use an IDE to write an execute a python program. • Locate and correct common syntax errors. • Arrange a program statement in a sequence (parson problems) • Walk through a sequences, branches and loops and sketch the state and output. • Use binary selection to control the flow of program execution (if,Else) • Use Selection to control the flow of program execution. • *Use iteration to control the flow of program execution.* • Use variables as counters in iterative programs. • Combine iteration and selection within a program.	• Define what a computer network is and explain how data is transmitted between computers across a network. • Use diagram software to create network diagrams representing LANS/WANS • List examples of hardware necessary for connecting devices to networks • Compare wired to wireless connections and list examples of specific technologies currently used to implement such connections. • Explain the difference between the internet, its services, and the World Wide Web. • Identify what happens to data entered online. • Critique online services in relation to data privacy • Identify and recommend strategies to minimise the risk of data being compromised through human error. • Question how malicious bots can have an impact on societal issues. • Examine how different types of malware cause problems for computer systems. • Explain how networks can be protected from common security threats
Disciplinary Literacy (Tier 3 Vocab)	• Active cell • Autofill • AutoSum • Cell • Cell Address • Cell Grid • Column • Drag • Formula • Function • Row • Sheet Tab • Workbook • Conditional Formatting	• Output print • Assignment = • Input input() • Arithmetic Operators and expressions +,-,*,/,// • Input (numerical) int,float • Binary Selection if, else • Comparison/relational operators ==,!=,>,<,>=,<= • Selection if, elif, else • Iteration while • Logical Operators and, or, not • Variable • Data Types • Integer • Float • String	• Networks • the internet • hub • server • router • protocol • personal computer • wired • wireless • WiFi • Bandwidth • Broadband • Packet • World Wide Web • WWW

Curriculum Knowledge Map

		• Boolean • IDE (integrated development environment) • Program • Syntax Error • Logic Error • Loop • Sequence • Selection • Iteration	• Spam • Privacy • Security • web browser • Profiling • Data Protection Act • Computer Misuse Act • Hacking • Malware • Protection methods such as firewalls, anti-malware, and password authentication
Assessment Opportunities	1. End of unit summative assessment - The assessment tasks involve working with a spreadsheet containing UK forestry data, and can be self-assessed by learners themselves using the assessment WAGOLL answer sheet.	1. Formative Assessment – MCQ based on programming techniques. 2. Rubric for Program end lesson - learners will apply and consolidate what they've learnt by extending and modifying a number guessing game into an iterative version that allows them multiple guesses.	1. End of unit summative assessment - The summative assessment for this unit will be in the form of a set of multiple choice questions MSF based on networks and security.
National Curriculum Links	• Design, use, and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems. • 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.	• Can understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation. • Can analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems. • Use two or more programming languages, at least one of which is textual, to solve a variety of computational problems. • Understand several key algorithms that reflect computational thinking; use logical reasoning to compare the utility of alternative algorithms for the same problem. • Understand how instructions are stored and executed within a computer system. • Design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems.	• Understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems. • Understand a range of ways to use technology safely, respectfully, responsibly, and securely, including protecting their online identity and privacy; recognise inappropriate content, contact, and conduct, and know how to report concerns.