Curriculum Map Year 8 COMPUTING

Topic Name	Term	**Skills** developed with link to NC Subject content	Reflection on previous link in the curriculum	Progress to future link in the curriculum
Understanding computers	*Autumn HT1*	● 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	*Year 7: Networks* *Year 7: Coding in Scratch part 1*	*Year 9: Graphics* *Year 9: Sound editing in Audacity* *Year 9: Computational thinking and logic*
Programming in Scratch part 2	*Autumn HT2*	● Understand several key algorithms that reflect computational thinking; 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; 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 ● Create, reuse, revise and repurpose digital artefacts for a given audience, with attention to trustworthiness, design and usability	*Year 7: Impact of technology* *Year 7: Coding in Scratch part 1* *Year 7: Introduction to coding with Kodu*	*Year 8: Introduction to Python programming* *Year 9: Python programming with sequences of data* *Year 9: Computational thinking and logic*
Mobile app development	*Spring HT1*	● Design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems ● Create, reuse, revise and repurpose digital artefacts for a given audience, with attention to trustworthiness, design and usability	*Year 7: Using media*	*Year 8: Developing on the web: HTML* *Year 9: Computational thinking and logic*
Introduction to Python programming	*Spring HT2 and Summer HT 1*	● 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	*Year 7: Introduction to coding with Kodu* *Year 7: Programming in Scratch part 1* *Year 7: Modelling data - Spreadsheets* *Year 8: Programming in Scratch part 2*	*Year 9: Python programming with sequences of data* *Year 9: Computational thinking* *GCSE: Algorithms and programming*

Modelling data - Spreadsheets	*Summer HT2*	● 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	*Year 6: Introduction to spreadsheets* *Year 7: Scratch part 1*	*Year 7: Scratch part 2*