Progression of knowledge and skills: Y6 Computing

Topic	Knowledge (Pupils will learn…)	Skills (Pupils will learn how to…)
Computing systems and networks	• Explore how data is transferred over the internet. • Focus on addressing, before they move on to the makeup and structure of data packets. • Understand how the internet facilitates online communication and collaboration; they complete shared projects online and evaluate different methods of communication. • Understand how to communicate responsibly by considering what should and should not be shared	• Explore what is necessary for effective communication and the importance of agreed protocols. • Apply this understanding to IP addresses and the protocols computers have for communication • Use a Domain Name Server (DNS) to translate web addresses into IP addresses. • Understand the concept of packets and gain an understanding of the key parts of a packet: the header and the data payload. • Consider how people can work together when they are not in the same location. • Consider other approaches to online work: reusing and modifying work done by someone else. • Explore different methods of digital communication • Choose which method of communication would be appropriate for a scenario based on prior learning.
Creating media - website design	• Be introduced to creating websites for a chosen purpose. • Identify what makes a good web page and use this information to design and evaluate their own website using Google Sites. • Pay specific attention to copyright and fair use of media, the aesthetics of the site, and navigation paths.	• Explore and review existing websites and evaluate their content. • Understand that websites are created by using HTML code. • Look at the different layout features available in Google Sites and plan their own web page on paper. • Become familiar with the terms 'fair use' and 'copyright' and gain an understanding of why they should only use copyright-free images. • Revise how to create their own web page in Google Sites. Using their plan from previous lessons, they will create their own web page/home page. Preview their web page as it will appear on different devices and suggest or make edits to improve the user experience on each device. • Pupils plan their website, paying attention to the navigation paths. They will then create multiple web pages for their site and use hyperlinks to link them together as detailed in their planning. • Consider the implications of linking to content owned by other people and create hyperlinks on their own websites that link to other people's work.
Programming A - variables in games	• Find out what variables are and relate them to real-world examples of values that can be set and changed. • Use variables to create a simulation of a scoreboard. • Experiment with variables in an existing project, then modify them, before they create their own project. • Apply their knowledge of variables and design to improve their games in Scratch.	• To see examples of real-world variables (score and time in a football match) before they explore them in a Scratch project. • Design and make their own project that includes variables. • Identify that variables are named and that they can be letters (strings) as well as numbers. • Understand that variables are used in programs, and that they can only hold a single value at a time. • Explore why it is important to name variables and apply their learning in a Scratch project in which they make, name, and update variables. • Apply the concept of variables to enhance an existing game in Scratch. • Predict the outcome of changing the same change score block in different parts of a program, then they test their predictions in Scratch. • Experiment with using different values in variables, and with using a variable elsewhere in a program. • Design sprites and backgrounds for their project, then they design their algorithms to create their program flow. • Implement algorithms created and/or add more variables to enhance their project.

Data and information	• Introduce learners to spreadsheets. • Be supported in organising data into columns and rows to create their own data set. • Understand the importance of formatting data to support calculations, while also being introduced to formulas and will begin to understand how they can be used to produce calculated data. • Apply formulas that include a range of cells and apply formulas to multiple cells by duplicating them. • Use spreadsheets to plan an event and answer questions. • Create charts and evaluate their results in comparison to questions asked.	• Collect and organise data in a format of their choice and then decide how it can be structured in a table. • Input data into a spreadsheet. • Develop their understanding of the structure of a spreadsheet. To begin to understand cell references, data items and the concept of formatting cells and see data items formatted in different ways. • Choose formats for data items before applying formats in their own spreadsheet. • Begin to use formulas to produce calculated data. • Understand that the type of data in a cell is important (e.g. numbers can be used in calculations whereas words cannot). • Create formulas to use in a spreadsheet using cell references and identify that changing inputs will change the output of the calculation. • Calculate data using the operations of multiplication, subtraction, division, and addition and use these operations to create formulas in a spreadsheet. • Plan and calculate the cost of an event using a spreadsheet. • Gain skills to create charts in Google Sheets
Creating media - 3D modelling	• Develop their knowledge and understanding of using a computer to produce 3D models. • Familiarise themselves with working in a 3D space, moving, resizing, and duplicating objects. • Create hollow objects using placeholders and combine multiple objects to create a model of a desk tidy. • Examine the benefits of grouping and ungrouping 3D objects, then go on to plan, develop, and evaluate their own 3D model of a building.	• To be introduced to the concept of 3D modelling by creating a range of 3D shapes that they select and move. • To manipulate 3D objects digitally. • To resize objects in one, two, and three dimensions, to lift and lower 3D objects relative to the workplane, and finally to combine two 3D objects to make a new shape and l recolour 3D objects. • To develop their understanding of manipulating digital 3D objects • To be introduced to the dimensions of shapes in Tinkercad which will enable them to accurately resize and move shapes • To explode 3D models of buildings to see what shapes they comprise of and then plan their own 3D building. • To create a computer 3D model based on their design.
Programming B - sensing movement	• Bring together elements of all the four programming constructs: sequence, repetition, selection, and variables. • Use a physical device - micro:bit • Begin with a simple program to build in and test within the new programming environment, before transferring it to their micro:bit.	• Be introduced to the micro:bit as an input, process, output device that can be programmed. • To familiarise themselves with the device itself and the programming environment, before creating their own programs • Explore how if, then, else statements are used to direct the flow of a program. • Apply their knowledge of if, then, else statements to create a program that features selection influenced by a random number to create a micro:bit fortune teller project. • Use the buttons to change the value of a variable using selection; then develop their programs to update the variable by moving their micro:bit using the accelerometer to sense motion. • Apply their understanding of the importance of order in programs. • Use operands in selection to determine the flow of a program. • Modify a program which will enable the micro:bit to be used as a navigational device. • Pick out features of a step counter, a piece of technology with which they are likely to be familiar. They will relate those features to the sensors on a micro:bit. In the main activity, pupils will design the algorithm and program flow for their step counter project. • Make a micro:bit step counter.