Computing KS3 Department: Curriculum Overview 2023-24

Curriculum Intent:

In Maths, IT & Computing, our aims are for all students;

- To have a passion for and resilience towards Maths, IT and Computing
- To develop strong problem solving, digital literacy and numeracy skills
- To be able to communicate their learning in Maths, IT and Computing effectively
- To be aware of E-Safety and how to report concerns and keep themselves safe & healthy online
- To gain qualifications to best prepare students for life after Fullbrook

Year 7	Term 1		Term 2		Term 3		End Points
	Half Term 1 [4 lessons]	Half Term 2 [4 lessons]	Half Term 3 [3 lessons]	Half Term 4 [3 lessons]	Half Term 5 [3 lessons]	Half Term 6 [4 lessons]	
Topic	E-Safety	MS Office	Cats & Dogs	Scratch	Fireworks	CODE.org	

Skill	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.	Be able to use word processing, spreadsheet and presentation software to solve a range of tasks.	Collect Data & Record Use basic formulas Formatting IF function	Be able to develop a range of games (Racing, Asteroid, Underwater)	Make use of simple shapes Combining Paths Advanced Techniques Photo Manipulation Use of Pen Tool Text Formatting Magazine Cover Animated Banner (GIF)	Be able to use block-based programming to solve programming problems.	Be able to create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability Understand the hardware and software components that make up computer systems, Be able to use block-based programming languages 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. Students will be able to use Fullbrook's network effectively including folder structure, emails, accessing the SLE and Class charts.
Content	<u>Lesson 1</u> Be able to use the software and hardware to access email, SLE, class charts. Create folder structures. Save work. <u>Lesson 2</u> To understand what **cyberbullying** is. To understand the **impacts of cyberbullying** on the individual To learn how to **protect yourself from cyberbullying.** To know how to **report cyberbullying** <u>Lesson 3</u> To understand the terms **sexting** and **grooming** To understand the **consequences of sexting** To learn how to **protect yourself** from sexting To be able to identify the **different types** of grooming To be able to explain the **impacts** of grooming and methods you can follow to **stay safe** <u>Lesson 4</u> To understand what a **phishing** email is and how they work To understand what they **warning signs** of a phishing email are. Be able to complete an assessment on what you have learnt so far. To learn about the potential **career** pathways in E-safety	<u>Lesson 1</u> Be able to use a range of formatting skills in MS **Word**. Know keyboard shortcuts. Be able to do screenshots. Develop understanding of folder structure. <u>Lesson 2</u> Understand what **spreadsheets** are used for. Be able to **format** a spreadsheet. Understand cell referencing. Be able to carry out basic **operations** (+, -, *, /) <u>Lesson 3</u> Be able to create and **format** a power point. Be able to use **transitions**, **hyperlinks** and **animations** in power point. Be able to print multiple slides on one page. <u>Lesson 4</u> Be able to combine multiple applications to complete a task, based on a business scenario	<u>Lesson 1</u> Be able to collect information and record sources. Understand what makes a **reliable** source. **<u>Lesson 2</u>** Be able to use basic to advanced **formula** and **functions** in Excel. **<u>Lesson 3</u>** Be able to **model** different scenarios in spreadsheets. To use **Publishing** software to create a flyer.	<u>Lesson 1</u> Learn how to use basic script – move cat/sprite 10 paces, show dragging blocks, how to stop a script, how to save a script. Show: using a forever loop, Bounce and rotations. **<u>Lesson 2</u>** Learn how to use variables in scripts, as part of repetition constructs for games (E.g. for score/points counters). **<u>Lesson 3</u>** Learn how to use duplicate sprites and re-use pre-written scripts. Be able to extend scripts via the use of message banners. **<u>Lesson 4</u>** Learn how to use tools (E.g. Pen tool) to create different shapes and incorporating a range of colours to enhance the quality of program designs.	<u>Lesson 1</u> Be able to develop Simple **shapes** and combine **paths** to develop new designs. <u>Lesson 2</u> Be able to use advanced techniques to **manipulate** photographs using **selection**. <u>Lesson 3</u> Be able to design and create magazine cover and an **animated** GIF banner.	<u>Lesson 1</u> Identify and locate bugs in a program. Translate movements into a series of commands using **sequencing**. <u>Lesson 2</u> Create **sprites** and objects and assign them costumes and behaviours. <u>Lesson 3</u> Create an interactive animation using **events**. Develop programs that respond to timed events. Develop programs that respond to user input. <u>Lesson 4</u> Break down a long sequence of instructions into the largest repeatable sequence. Use a combination of sequential and looped commands to reach the end of a maze. Identify the benefits of using a loop structure instead of manual repetition.	
Prior Knowledge Required	Any KS2 E-Safety SoW Use of desktop PCs @ home or schools.	Use of Email, attachments, E-Safety, MS Office experience.	MS Office SoW (Spreadsheets) KS2 Spreadsheet	Scratch / Other block based programming languages.			

Feedback Points	Peer assessment of posters Verbal feedback	Verbal feedback. Self and peer assessment.	Peer assessment Self-assessment Verbal Feedback	Peer assessment Verbal feed-back Self-assessment	Self Assessment, Peer Assessment Verbal Feedback	Self Assessment, Peer Assessment Verbal Feedback Summative assessments.	
Key Questions	Why is folder structure important? What are some dangers online? What can you do to ensure your identity is safe? Who do you report concerns to? How can you report concerns?	Why should we use formatting in documents? How can spreadsheets save time? Why are some presentations less effective than others?	What are the benefits of using spreadsheet software? Where are spreadsheets used in the real world? How can I design a product to appeal to a target audience?	What are the benefits of using block-based programming languages over text-based programming languages? What steps can I take to help with the design of computer programs?	What jobs make use of digital graphics editing?		
Direct Vocab Instruction	E-Safety, Online, Report, Phishing, Danger, meeting, email, attachment, link, virus, concern, logging, network, password, login, folders, attachment, Internet.	Document, save location, save, formatting, indent, bold, italic, alignment, header, footer, text, link, picture, images, graphics, ruler, page numbers, table, rows, columns, relative, cell, reference, formula, functions, SUM, Borders, Shading, data types, dragging, tabs, worksheet, One-drive, Save-as.	Cell Reference, Biased, Reliable, Formulas, Functions, Columns, Rows.	Sprite, programming, coding, text, loop, constructs, repetition, selection, sequence, variable, parameter, background. Costume, image, library, flow, movement, direction, degrees, path, values, execute, run, evaluate, reverse	Combine, Unite, Punch, Selection, Magic Wand,		
Standardised Homework	Research Tasks Quizzes Problem Solving Independent Learning	Research Tasks Quizzes Independent Learning	Research Tasks Quizzes Problem Solving	Problem Solving Research Tasks Quizzes	Research Tasks Quizzes Problem Solving	Research Tasks Quizzes Problem Solving	

Year 8	**Term 1**		**Term 2**		**Term 3**		**End Points**
	Half Term 1 [3 core & 1 bonus lessons]	**Half Term 2 [3 lessons & 1 bonus]**	**Half Term 3 [2 lessons & bonus]]**	**Half Term 4 [3 lessons]**	**Half Term 5 [3 lessons]**	**Half Term 6 [3 lessons & 1 bonus]**	
Topic	**Python (Text Based)**	**Python (Text Based)**	**MS Word, PowerPoint (& Excel) Use of applications**	**Web Design (Rocket Cake)**	**Game Making (Click Team Fusion)**	**Game Making (Click Team Fusion)**	
Skill	Be able to solve problems by developing programs that make use of variables, calculate numerical values, selection, iteration, boolean variables and logical operators.		Be able to use word processing, spreadsheet and presentation software to solve a range of tasks.	Identify pros and cons of different devices and internet connection methods. Be able to plan and develop a multipage website using web development software.	Be able to develop understanding of a range of video games from multiple genres and export them for use at home. ~Shoot Em Up style game. ~Choco-break (pong style) ~2 Player Maze ~ Platform Game Be able to Plan, Develop & Test an original own game Be able to review games.		Be able to 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 Be able to create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability
Content	**<u>Lesson 1 – First Steps</u>** Learning Objectives: 1. Describe what algorithms and programs are and how they differ 2. Recall that a program written in a programming language needs to be translated in order to be executed by a machine 3. Write simple Python programs that display messages, assign values to variables, and receive keyboard input 4. Locate and correct common syntax errors <u>Homework:</u> Research 3 Apps made using Python. Explain what do they do? How much are they worth?	**<u>Lesson 4 – More Branches</u>** **Some classes will cross into 2nd term due to Inset & Early Closure* - Learning Objectives: 3. Use multi-branch selection (`if`, `elif`, `else` statements) to control the flow of program execution 4. Describe how iteration (`while` statements) controls the flow of program execution Homework: Q & A (PG Book) **<u>Lesson 5 - Quiz</u>** Students will put together the skills learned to create a general knowledge quiz (first in Scratch) that will keep the players score and allocate lives. IF the lives = 0 then a game over screen is displayed. Homework: Plan the same quiz but for Python. **<u>LESSON 6 – PUTTING IT ALL TOGETHER</u>** Use the homework planning to develop the quiz in Python. Additional features can be added such as timers, power ups and extra lives.	**<u>Lesson 1</u>** Be able to use a range of formatting skills in MS **Word**. Know keyboard shortcuts. Be able to do screenshots. Develop understanding of folder structure. <u>Lesson 2</u> Understand what **spreadsheets** are used for. Be able to **format** a spreadsheet. Understand cell referencing. Be able to carry out basic **operations** (+, -, *, /) <u>Lesson 3 (Extra)</u> Be able to create and **format** a power point. Be able to use **transitions**, **hyperlinks** and **animations** in power point. Be able to print multiple slides on one page.	**<u>Lesson 1: Research</u>** **The purpose** and **features** of websites. **Devices** that **connect** to the internet. **Methods** to **connect** to the internet. <u>LESSON 2: **PLANNING**</u> Write text that will appear on your website. Collect **assets** that will be used on the website. -File Formats -Copyright Designs & Patents Act. -Pixabay Building of site using the help sheet on the SLE. Homework: Complete the text for the site; All About Me, Hobbies, The Future.	**<u>Lesson 1</u>** ~Be able to make a Shoot Em Up style game. Folder set up What is a games engine? What is ClickTeam Fusion? <u>Intro to Workspace</u> -Events Editor -Frame Editor -Storyboard Editor *Show end game result. Creating **assets** Adding **movement** **Interaction** -Position (bounce on walls) -Collisions (other assets) **How to Export and save to OneDrive** <u>Homework:</u> Complete testing table of the game you have developed. **<u>LESSON 2:</u>** **<u>Be able to make a pong style game.</u>** ~Choco-break (pong style)	**<u>Lesson 4</u>** ~ Platform Game -Examples of platform games. -App Size (to add unseen level content) -Frame Focus (to follow P1) - Backdrops (platforms & ladders) -Event Editor Export game to OneDrive and share with a partner. **<u>Homework</u>**: Review each other's games. **<u>LESSON 5:</u>** **<u>Planning</u>** booklet for game ideas using the **'Code a Game Booklet'** **Planning (part 1)**	

	LESSON 2 – CRUNCHING NUMBERS Learning Objectives: 1. Describe the semantics of assignment statements 2. Use simple arithmetic expressions in assignment statements to calculate values 3. Receive input from the keyboard and convert it to a numerical value Homework: Answer questions in booklets. TAsks 1 – 3. **LESSON 3 –** AT A CROSSROADS Learning Objectives: 1. Use relational operators to form logical expressions 2. Use selection (`if`, `else` statements) to control the flow of program execution 3. Generate and use random integers Homework: Create an RPG game in Trinket using selections **LESSON 4 - MORE BRANCHES** **Some classes will cross into 2nd term due to Inset & Early Closure* - Learning Objectives:	Homework: Revision for the formal assessment. **LESSON 7 – FORMAL ASSESSMENT & MAGIC 8 BALL** Learning Objectives: 1. Complete an MS Forms based assessment. 2. Understand and use sequence in an algorithm 3. Understand and use selection in an algorithm (IF, Else and Else if) 4. Understand the importance of comments in code Ask the 8-ball a yes or no question and the 8-ball will reply with answers such as "Yes", "No", "Without a doubt" etc. (Seemingly being able to predict the future) The Magic 8 Ball is made up of 20 responses – 10 positive, 5 negative and 5 neutral. Possible answers and instructions are located here: https://teachwithict.weebly.com/magic-8-ball1.html EXT Lesson **LESSON 8 - THE SORTING HAT** **Some classes will not get this far due to Inset & Early Closures.* Learning Objectives • Understand and use sequence in an algorithm • Understand and use iteration in an algorithm (FOR and WHILE loops) • Understand and use selection in an algorithm (IF, Else and Else if) Students will be creating their very own 'Harry Potter' style sorting hat that will select a house at random when a button is pressed. Link to resources and tasks are here:		200 words min per section. LESSON 3: **BUILDING** Building of site using the help sheet on the SLE and modelling by the teacher. Be able to apply the following: -Templates -Containers -Navigation Menus -Formatting -Styles Buttons -Web Form Buttons -Scrolling Galleries -Embedding Video -Embedding Audio -Floating Text -Floating Images	**Export to OneDrive** **Homework**: Complete testing table of the game you have developed. **LESSON 3:** ~2 Player Maze Additional players Movement Controls Timers Obstacles **Export to OneDrive** **Homework**: Complete testing table of the game you have developed.	Choose genre Aim of game Choose Environment Story / Overview Original Game Artwork (Player, Enemy, Obstacles, Collectables.) **Homework**: Complete designs in booklet of your artwork. **LESSON 6:** **Planning** booklet for game ideas using the **'Code a Game Booklet'** **Planning (part 2)** Movements Collisions -Leave frame -Obstacles -Collectables Score / Lives Special Events (1up or reduce score etc) Adding Sounds (file formats) **Develop** Develop & test original game **Homework** Create your own testing table based on your gameplay. Lesson 7 & 8 Develop &Test How to **review** a game. Export game to OneDrive and share with a partner. **Homework**:	

	1. Use multi-branch selection (`if`, `elif`, `else` statements) to control the flow of program execution 2. Describe how iteration (`while` statements) controls the flow of program execution Homework: Q & A (PG Book)	https://teachwithict.weebly.com/sorting-hat.html				Review each other's games.	
Prior Knowledge Required	Block Based Programming in Scratch	Block Based Programming in Scratch or other block-based programming languages	Basic understanding of layout of menu systems for MS Office applications Understanding the key capabilities of each of the different applications (MS Word, Ecel & Power point)	KS2 work on networks. (Limited)	Scratch Python Fireworks		
Feedback Points	1 to 1 verbal feedback. Peer feedback. Kahoot results from Plenary.	Kahoot quiz feedback.	Teacher feedback (Verbal) based on tasks End of topic quiz/progress based on techniques	Verbal Feedback Peer Assessment Self Assessment	Verbal Feedback Peer Assessment Self Assessment		
Key Questions	What is an algorithm? What is a variable? How can you assign different data types for string and integers? How can I import a random integer? What is an example of selection?		Why should we use formatting in documents? How can spreadsheets save time? Why are some presentations less effective than others?	What is the purpose of websites? What devices can connect to the internet? What are the different features? How do devices connect to the internet? What is the Copyright Designs and Patents Act?	What are different genres of games? What is a games engine? What are features of various genres of video games? Can you explain the purpose of a testing table? What is the purpose / objective of the game?		
Direct Vocab Instruction	Algorithm, program, programming language, syntax, data types, input, output, variables, assign, conditions, selection.		Document, save location, save, formatting, indent, bold, italic, alignment, header, footer, text, link, picture, images, graphics, ruler, page numbers, table, rows, columns, relative, cell, reference, formula, functions, SUM, Borders, Shading, data types, dragging, tabs, worksheet, One-drive, Save-as, mail merge, effective, paragraphs, default, auto-complete, export, insert, hyperlink, transitions, animations, images, links, reflections, custom, paths	Wi-Fi, Ethernet, Satellites, Copyright Designs and Patents Act, Desktop, Laptop, Tablets, Smart Phones, File Formats.	Games Engine, Shoot-Em-Up, Platform, Assets, Obstacles, Events, Frames, Storyboards		

Standardised Homework	Research Tasks. Booklet based tasks. Problem solving Python programming tasks.	Research Tasks. Booklet based tasks. Problem solving Python programming tasks.	Research tasks Document completion tasks Simple problem-solving tasks,	Extended writing, Research Tasks,	Research Tasks. Booklet based tasks. Problem solving Design based	

Year 9	**Term 1**		**Term 2**		**Term 3**		**End Points**
	Half Term 1 [4 lessons]	**Half Term 2 [4 lessons]**	**Half Term 3 [3 lessons]**	**Half Term 4 [3 lessons]**	**Half Term 5 [3 lessons]**	**Half Term 6 [4 lessons]**	
Topic	**MS Word, PowerPoint (& Excel) Use of applications**	**Photoshop / Photopea**	**VB**	**VB**	**Theory**	**MIT APP Inventor**	
Skill	Mail Merging, Format formal Letters (formal layout), Writing specific requirements,	Be able to use Photoshop to solve a range of design problems. Develop understanding of the use of digital graphics. Know the suitability of different image file formats.	Learn how to use a Text based Integrated Development environment (IDE) and all associated tools	Extend understanding of the IDE in relation to editing, error correction and troubleshooting computer programs.	Learn how to apply basic CS concepts to real-world scenarios Understand the importance and relationship number systems have within computer systems (Eg Use of Binary) Understand how logical operations are incorporated into computer systems	Learn how to use external devices (Eg. Tablets) to test online applications simulating website access	Be able to 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. Be able to create, re-use, revise and re-purpose digital artefacts for a given audience, with attention to trustworthiness, design and usability 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 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 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
Content	Lesson 1 Be able to use a range of formatting skills in MS **Word**. Know keyboard shortcuts. Understand how to use advanced techniques to enable the creation of documents for different target audiences (e.g. Mail Merging) Lesson 2 Understand what **spreadsheets**	**Lesson 1 – Getting Started** Be able to use a range of techniques to create an original image. Be able to understand the suitability of different image file types. Be able to export images in a range of different formats.	**Lesson 1** Understand the difference between form based and console-based applications To learn about the three programming constructs Understand what variables are and how to use them Understand what a 'Sequence of instructions' comprise **Lesson 2**	**Lesson 1** Learn how to pass variables onto forms via controls. Understand the importance of planning computer programs using graphical notation **Lesson 2** Learn how to combine the 3 programming constructs (Sequence, selection and	**Lesson 1** Learn how to convert numbers between different number systems (Binary, Decimal and Hexadecimal) Understand how computers perform simple arithmetic in binary (such as addition and multiplication) **Lesson 2**	**Lesson 1** Understand what is an app? Understand why we use apps. Understand how blocks are used and linked together when creating apps. Learn how to create and test a simple App on an external device (Eg. A tablet) **Lesson 2**	

	are used for. Be able to **format** a spreadsheet. Understand cell referencing. (Relative). Learn how to create graphs/charts to emphasize results after data analysis. Lesson 3 Be able to create and **format** a power point slides for professional use. Learn how to incorporate link to internal and external resources within documents. Lesson 4 Be able to combine multiple applications to complete a task, based on a business scenarios	**Lesson 2 – Digital Graphics & Purpose.** Understand why digital graphics are used, including: to entertain, to inform, to advertise, to promote and to educate Understand how digital graphics are used, including: magazine covers, CD/DVD covers, adverts, web images and graphics, multimedia products, games. **Lesson 3 – Making a MOVIE poster.** Understand features of graphic products such as: ○ Rule of thirds ○ Margins Create a new document with correct properties Compile an image using basic techniques such as: ○ Crop and move ○ Text ○ Eraser ○ Layers **Lesson 4 – Designing a CD cover** Understand how to make effective choices and uses of images Be able to change the brightness and contrast of an image Be able to add shapes and change their stroke and fill Make use of the paint brush tool and change brushes Apply filters to a layer	To learn about repetition (Iteration) and apply it within simple programming scenarios. **Lesson 3** Learn how to incorporate variables into simple procedures using images Learn how to use simple animation and graphics To understand how to modify control properties	Iteration) to make simple programs **Lesson 3** Learn how to use different data structures such as arrays and lists in structured programs, that also include animation	Understand the key components of computer systems. Understand how computers represent sound and images in digital format. Learn how to maximise storage capacity using compression algorithms (Lossy / Lossless), and RLE (Run length Encoding) when applied to dictionaries. **Lesson 3** Understand the importance of Boolean Logic when processing instructions on computer systems. Learn about the different logic operations (AND/OR/NOT/EXOR) and their corresponding truth tables Learn how to draw simple logic circuits using logic diagram notation	Learn the role of a storyboard when designing apps. Understand the layout if MIT app inventor (Incl. Palette, designer and components). Learn how to create a simple app from a design and then test it using a tablet. **Lesson 3** Understand the different components in the Tool bar (within app inventor) Learn how to use the canvas to design apps. Learn how to use simple logic operations in apps. Learn how to use variables, and other components such as the timer, scores, noise, speed, direction. **Lesson 4** Learn how to display app code using general purpose applications. Understand the benefits of commenting app (program) code Understand the difference between iterative and final testing (of apps) Learn how to write, edit and test simple block-based apps.	
Prior Knowledge Required	Understanding of simple text editors, understand basic formatting features used in MS Office	Fireworks File Formats	Understanding of block-based programming languages	Understanding of block-based programming languages	Understanding of text and block-based programming languages Understand how to use a tablet	Understanding of text and block-based programming languages	

						Understand how to use a tablet	
Feedback Points	Teacher feedback (Verbal) based on tasks End of topic quiz/progress based on techniques	Verbal Feedback Peer Assessment Self-Assessment	Verbal Feedback (teacher) Peer Assessment Self-Assessment	Verbal Feedback (teacher) Peer Assessment Self-Assessment	Verbal Feedback (Teacher) Peer Assessment Self-Assessment	Verbal Feedback (teacher) Peer Assessment Self-Assessment	
Key Questions	Why should we use formatting in documents? How can spreadsheets save time? Why are some presentations less effective than others?	How are digital graphics used? Why are certain image file types suitable for one purpose but others are not? How can you make a digital graphic more appealing to a target audience?	Why do I need to translate program code into binary before it can be run?	What I the importance of testing computer programs before they are used?	What is hexadecimal? What are the main components of computer systems? How do I record sound on computers? How are images saved on computer systems? How are circuit diagrams used to create circuit boards?	How do I get mobile applications to run on different types of hardware? Why do I need to test phone applications on external devices?	
Direct Vocab Instruction	Document, save location, save, formatting, indent, bold, italic, alignment, header, footer, text, link, picture, images, graphics, ruler, page numbers, table, rows, columns, relative, cell, reference, formula, functions, SUM, Borders, Shading, data types, dragging, tabs, worksheet, One-drive, Save-as.	Extraction, Importing, Layers, Selection, Transforming, Typography, Brush, Rubber/Eraser, Magic wand tool, Cropping, Brightness & Contrast, Hue & Saturation, Black & White, Pen Tool, Clipping Mask, Exporting (file formats), PNG, JPEG, PSD.	programming, coding, text, loop, constructs, repetition, selection, sequence, variable, parameter, background, Foreground, image, library, flow, movement, direction, degrees, path, values, execute, run, evaluate, reverse, integer, data, string, print, read, output, display	Forms, controls, buttons, label, textbox, font, font size, colour, indent, alignment, centre, left, right, justify, top, character, data, data type, integer, Boolean, decimal, real, float, long, short, on, off, positive, true, false, syntax, logic, error, testing, incremental, iterative, final, team, logical, concept, planning, flowchart, digit, binary,	Binary, decimal, hexadecimal, conversion, addition, multiplication, number systems, bits, byte, megabyte, integer, tens, hundreds, thousands, power, columns, image, bit depth, audio, sample, resolution, modify, colour depth, sample rate, seconds, sampling, pixel, width, height, CPU, LAN, WAN, Processor, memory, RAM, ROM, Virtual, router, hub, switch, logic, gate, circuit, operation, function, AND, OR, NOT, EXOR, truth table, input, output, display, operation, combined,	Block, app, testing, execution, planning, programming, coding, text, loop, constructs, repetition, selection, sequence, variable, parameter, background, Foreground, image, library, flow, movement, direction, path, values, execute, run, evaluate, reverse, integer, data, string, print, read, output, display, tablet, testing, canvas, palette, designer, buttons, points, direction, noise, storyboard, template, internet, testing, iterative, final testing	
Standardised Homework	Research Tasks Quizzes Independent Learning	Reseach based task Exam style questions. Design Based Tasks	Research Tasks. Booklet based tasks. Problem solving Design based	Research Tasks. Booklet based tasks. Problem solving Design based	Research Tasks. Booklet based tasks. Problem solving Design based	Research Tasks. Booklet based tasks. Problem solving Design based	