

Computing - Long term plans

Year 7

Term	Autumn Term		Spring Term		Summer Term	
Half term	HT1	HT2	HT3	HT4	HT5	HT6
Unit Titles lengths:	Digital Literacy & Information Technology 1: E-safety Website Review & Planning	Digital Literacy & Information Technology 1: E-safety Website Design	Computational Thinking 1: Flowol – Flow Control & Algorithmic Thinking	Computational Thinking 1: Advanced Flowol & Algorithm Design	Computational Thinking 1: Game Design – Concepts & Logic	Block Programming 1: Creating 2D Games (GameMaker / scratch)
Learning objectives Substantive and procedural knowledge covered in the unit	<p>Learning Objectives:</p> <ul style="list-style-type: none"> - Understand how to safely navigate and evaluate online content. - Identify key features of high-quality websites. - Analyse the structure, design and safety features of existing websites. - Begin developing critical thinking about website usability and E-safety messaging. <p>Substantive Knowledge:</p> <ul style="list-style-type: none"> - Features and types of multi page websites (e.g. landing, master, site pages). - Core principles of E-safety (cyberbullying, phishing, secure passwords, etc.). - Understanding of good versus poor website design (navigation, interactivity, clarity). - UX/UI concepts: layout, accessibility, audience targeting. <p>Procedural Knowledge:</p> <ul style="list-style-type: none"> - Log in and access resources securely. - Evaluate websites using a checklist (navigation, media use, clarity, interactivity). - Review and critique E-safety content. - Use teacher and peer feedback to reflect and plan improvements. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> - Design and create a multi-page website focused on E-safety. - Apply key web design principles (navigation, layout, interactivity). - Integrate media and content with awareness of audience and purpose. <p>Substantive Knowledge:</p> <ul style="list-style-type: none"> - Purpose and layout of web pages (header, footer, nav bar, hyperlinks). - Design consistency and accessibility standards. - Effective multimedia use (images, videos, interactivity). - Web design planning, structure, and testing phases. <p>Procedural Knowledge:</p> <ul style="list-style-type: none"> - Use website creation tools (e.g. PowerPoint, Google Sites, Wix). - Build a homepage and additional pages linked via nav bar. - Insert and organise assets (text, images, buttons). - Peer-assess and refine content based on feedback. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> - Understand what algorithms are and why they're used. - Begin creating basic flowcharts to solve real-world problems. - Explore the concepts of sequence, selection, and iteration. <p>Substantive Knowledge:</p> <ul style="list-style-type: none"> - What an algorithm is and its real-world applications. - Flowol mimic systems and their representations. - Core control structures: sequence, decision (IF), and repetition (loops). <p>Procedural Knowledge:</p> <ul style="list-style-type: none"> - Build and test simple flowcharts using Flowol. - Simulate real-life scenarios using MIMIC systems. - Apply selections and loops to control outcomes. - Debug flowchart logic and make improvements 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> - Refine algorithmic design using combinations of control structures. - Enhance Flowol diagrams with variables and subprocedures. - Develop debugging and testing skills. <p>Substantive Knowledge:</p> <ul style="list-style-type: none"> - How variables influence outcomes in a system. - Use of sub-procedures to break down complex tasks. - Optimising algorithms for efficiency and clarity. <p>Procedural Knowledge:</p> <ul style="list-style-type: none"> - Add and manipulate variables in Flowol. - Create sub-procedures to handle repeated tasks. - Test and optimise logic flow. - Respond to feedback and adapt solutions. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> - Understand core elements of a game (sprites, levels, scoring). - Plan a 2D game using block programming concepts. - Develop logic for basic interactions and challenges. <p>Substantive Knowledge:</p> <ul style="list-style-type: none"> - Game structure: start, play, win/lose conditions - Roles of sprites, backgrounds, and interactions - Common gameplay mechanics (collision, health, levels) <p>Procedural Knowledge:</p> <ul style="list-style-type: none"> - Design characters and level layout - Plan game flow with logic maps or storyboards - Begin using block code to set conditions (e.g. when sprite touches wall, lose a life) 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> - Build a playable 2D game using block-based coding. - Implement interactivity, scoring, and levels. - Debug and refine based on testing and peer review. <p>Substantive Knowledge:</p> <ul style="list-style-type: none"> - Drag-and-drop programming logic (e.g. movement, collisions) - Variables and events in games - Debugging and refining interactive systems <p>Procedural Knowledge:</p> <ul style="list-style-type: none"> - Create sprites, add objects to levels, and set game behaviours - Use conditional logic to create outcomes - Debug errors, test for bugs, and iterate on feedback
Key ideas/ Themes:	<ul style="list-style-type: none"> - Responsible digital citizenship - Evaluating digital content - Safety and ethics in online spaces - Building confidence in digital critique and communication 	<ul style="list-style-type: none"> - User-centred design - Multimedia communication - E-safety in action - Structure and clarity in presentation 	<ul style="list-style-type: none"> - Problem-solving through logic - Systems simulation and control - Understanding flow and dependencies 	<ul style="list-style-type: none"> - Refinement and optimisation - Breaking problems into reusable parts - Evaluating success and making improvements 	<ul style="list-style-type: none"> - Creative problem solving - Designing for enjoyment and engagement 	<ul style="list-style-type: none"> - Real-world applications of logic - Game mechanics and structure - Resilience through trial and improvement
Prerequisite knowledge:	KS2: Use technology safely, respectfully, responsibly; recognise acceptable/unacceptable behaviour; evaluate digital content; understand how websites present information.	KS2: Create content using presentation tools; evaluate digital products; understand multimedia and layout principles.	KS2: Design, write, and debug programs; use logical reasoning; understand device input/output.	KS2: Use repetition, selection, and variables in programs; debug and refine based on tests.	- KS2: Use block-based tools; understand conditionals, loops, and sequences; evaluate program effectiveness.	- KS2: Use inputs in programming; create interactive digital products; test and improve digital outcomes.
Outcomes (Stickable- output that the student produces to demonstrate their knowledge)	<ul style="list-style-type: none"> - Website review worksheet identifying strengths/weaknesses using criteria - Green Pen review sheet showing personal responses to feedback (DIRT) - Mind map comparing good vs bad websites - Annotated example of an evaluated site (print or screenshot) - Milestone assessed review document with teacher commentary 	<ul style="list-style-type: none"> - Full multi-page website draft with E-safety theme and interactive elements - planning document for each page (layout, purpose, assets) - Milestone assessment of interactivity/navigation features - Final evaluation reflection using sentence stems (Green Pen DIRT) 	<ul style="list-style-type: none"> - Completed Flowol MIMIC diagrams with control structures labelled - Screenshot portfolio of simulations including success/fail outcomes - Teacher-assessed milestone sheet with personalised targets - Annotated flowchart with explanation of how loops and decisions work - Self-review Green Pen reflection of test or simulation performance 	<ul style="list-style-type: none"> - Extended Flowol system simulation with sub-procedures - Debug MIMIC with identified errors and improvements - End-of-unit milestone task marked with WWW/EBI - Green Pen improvement plan after milestone review 	<ul style="list-style-type: none"> - Game review worksheet identifying strengths/weaknesses using criteria - Green Pen review sheet showing personal responses to feedback (DIRT) - Game storyboard showing character, levels, outcomes - Planning document identifying intended game mechanics 	<ul style="list-style-type: none"> - Completed, playable GameMaker project with levels, scores, and interactions - Teacher-assessed milestone with WWW/EBI on gameplay and structure - Green Pen peer review and reflection worksheet

Computing - Long term plans

Year 8

Term	Autumn Term		Spring Term		Summer Term	
Half term	HT1	HT2	HT3	HT4	HT5	HT6
Unit Titles lengths:	Digital Literacy 2: E-safety	Information Technology 2: Hardware / Software / Networks	Information Technology 2: Data Representation & Logic (Part 1)	Information Technology 2: Logic & Truth Tables (Part 2)	Block Programming 2: Scratch	Textual Programming 1: Basic Python Programming
Learning objectives Substantive and procedural knowledge covered in the unit	<p>Learning Objectives:</p> <ul style="list-style-type: none"> - Explore and evaluate digital conduct and its implications. - Understand risks associated with privacy, passwords, cyberbullying, and malware. - Practise digital citizenship and ethical online behaviour. <p>Substantive Knowledge:</p> <ul style="list-style-type: none"> - Safe password creation and management. - Understanding digital footprints and digital identity. - Cyberbullying types, reporting methods, and impact. - Digital etiquette and responsible online communication. - Digital well-being and screen time awareness. <p>Procedural Knowledge:</p> <ul style="list-style-type: none"> - Use strong password strategies and managers. - Manage digital profiles and privacy settings. - Recognise, report, and respond to cyberbullying. - Use reflection techniques (e.g. green pen) to improve understanding. - Create educational artefacts (e.g. e-safety posters). 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> - Identify and explain core components of a computer. - Understand how networks operate (LAN/WAN). - Differentiate hardware and software roles. <p>Substantive Knowledge:</p> <ul style="list-style-type: none"> - Fetch-execute cycle and CPU function. - Internal components: RAM, ROM, ALU, motherboard. - Definitions and relationships: hardware, software. - LANs, WANs, internet, routers, switches. - Input/output devices and their function. <p>Procedural Knowledge:</p> <ul style="list-style-type: none"> - Label and describe computer components. - Compare devices for different tasks. - Explain Internet/network functions. - Identify CPU and memory roles in system performance. - Categorise software vs hardware in real contexts. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> • Understand why data is stored in binary. • Convert between binary, denary, and hexadecimal. • Recognise multimedia file representation in computing. <p>Substantive Knowledge:</p> <ul style="list-style-type: none"> • Binary, denary, hexadecimal systems. • ASCII and character encoding. • Pixels, RGB codes, image data. • Bit patterns and file sizes. • File types and compression basics. <p>Procedural Knowledge:</p> <ul style="list-style-type: none"> • Convert binary ↔ denary ↔ hex. • Perform binary addition. • Identify image representation by pixels/colour. • Analyse file types and formats. • Identify compression type and use. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> • Understand logic gates (AND, OR, NOT). • Apply truth tables to logic circuits. • Model decision-making using logical operators. <p>Substantive Knowledge:</p> <ul style="list-style-type: none"> • Boolean logic and logic gates. • Circuits using logic combinations. • Truth table structure and outputs. <p>Procedural Knowledge:</p> <ul style="list-style-type: none"> • Create logic circuits from scenarios. • Complete truth tables. • Simulate circuit operation using software. • Analyse input/output patterns. 	<ul style="list-style-type: none"> • Develop interactive stories/games using Scratch. • Use variables, loops, and conditionals effectively. • Debug and test using structured methods. <p>Substantive Knowledge:</p> <ul style="list-style-type: none"> • Event handling, loops, conditions. • Use of variables and broadcasting. • Collision detection and interactivity. <p>Procedural Knowledge:</p> <ul style="list-style-type: none"> • Design and script interactive animations. • Apply variables to scoring and lives. • Test, debug and refine user input. • Use feedback to iterate and improve. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> • Write basic programs using text-based syntax. • Understand variables, loops, and conditionals in Python. • Apply logic to real-world programming problems. <p>Substantive Knowledge:</p> <ul style="list-style-type: none"> • Syntax of Python (print, input, if, while, for). • Variables, data types, arithmetic, selection. • Program structure and flow. <p>Procedural Knowledge:</p> <ul style="list-style-type: none"> • Write code using sequence and iteration. • Use variables for calculations. • Capture and respond to user input. • Debug and fix syntax errors. • Plan and implement simple projects.
Key ideas/ Themes:	<ul style="list-style-type: none"> - Empathy and responsibility in digital spaces - Online identity and footprint - Preventing and reporting abuse 	<ul style="list-style-type: none"> - Computer systems and performance - Device purpose and use - Network security and functionality 	<ul style="list-style-type: none"> - Abstraction of information - Number systems and conversion - Efficient data storage 	<ul style="list-style-type: none"> - Logical decision-making - System simulation - Predictable behaviours in computing 	<ul style="list-style-type: none"> - Creative coding - Visual programming logic - Collaborative improvement 	<ul style="list-style-type: none"> - Structured problem solving - Text-based programming logic - Syntax, precision, and testing
Prerequisite knowledge:	<p>KS2: Use technology safely and respectfully; understand acceptable/unacceptable behaviour; know how to report concerns about content/contact.</p> <p>Year 7: Created and evaluated E-safety websites; explored digital footprints, password safety, and online risks</p>	<p>KS2: Understand how computer networks can provide multiple services and recognise the role of hardware/software.</p> <p>Year 7: Navigated web design tools; explored input/output in Flowol; developed basic understanding of website hosting and structure.</p>	<ul style="list-style-type: none"> • KS2: Understand how data is stored and used in digital systems; familiarity with units of measurement and basic binary. • Year 7: Used Flowol logic; developed understanding of control and sequence through game design. 	<ul style="list-style-type: none"> • KS2: Logical reasoning to explain predictions; basic understanding of input/output. • Year 7: Used decision structures and iteration in Flowol and Scratch. 	<ul style="list-style-type: none"> • KS2: Design, write, and debug programs using block-based language. • Year 7: Created games using GameMaker; applied sequence, events, conditionals. 	<ul style="list-style-type: none"> • KS2: Write simple programs, use logical reasoning to detect and correct errors. • Year 7: Developed structured logic using Flowol; programmed basic interactivity in Scratch and GameMaker.
Outcomes (Stickable- output that the student produces to demonstrate their knowledge)	<ul style="list-style-type: none"> - Annotated infographic explaining cyber threats and protections - Strong password creation plan with self-assessment checklist - Completed digital footprint reflection worksheet - Green Pen review of assessment questions (WCF/DIRT) - Milestone-assessed test with WWW/EBI - Poster or leaflet on E-safety best practices 	<ul style="list-style-type: none"> - Labelled diagram of a computer system with components explained - Summary sheet on input/output devices with real-world examples - Venn diagram comparing hardware vs software - Short-answer quiz completed and reviewed using Green Pen - Milestone test with personalised misconceptions sheet 	<ul style="list-style-type: none"> - Binary to denary and hexadecimal conversion worksheet (marked and annotated) - ASCII character chart exercise (with extension on compression) - Screenshot and explanation of binary addition working - Milestone-assessed end-of-unit test (WWW/EBI targets) - Green Pen review of misconceptions from lesson recap tasks 	<ul style="list-style-type: none"> - Completed truth tables and logic gate diagrams - Milestone-assessed end-of-unit test (WWW/EBI targets) - Green Pen review of misconceptions from lesson recap tasks 	<ul style="list-style-type: none"> - Scratch project file: working game with variables, scoring and movement - Annotated Scratch code snippets explaining how logic was applied - Peer feedback and improvement log - Project planning sheet showing purpose, audience, and design choices - Milestone assessment (game flow and coding logic) - Green Pen reflection on testing outcomes and debugging strate 	<ul style="list-style-type: none"> - Python mini project (e.g. quiz) including inputs, variables, and selection - Annotated Python code submitted digitally with inline comments - Teacher-assessed milestone quiz (with WWW/EBI) - Debug log identifying common syntax errors and how they were resolved - Green Pen reflection on the Python Challenge task - Peer review of Python project with improvement target

Computing - Long term plans

Year 9 Route 1 (imedia)

Term	Autumn Term		Spring Term		Summer Term	
	HT1	HT2	HT3	HT4	HT5	HT6
Half term	HT1	HT2	HT3	HT4	HT5	HT6
Unit Titles lengths:	Website Review & Planning	Create and Build Websites	Video Review and Storyboards	Video Editing	Comic Review, Planning and Storyboards	Create Characters and Comics
Learning objectives Substantive and procedural knowledge covered in the unit	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Identify and evaluate components of effective multipage websites. Understand audience targeting, user needs, and site structure. Plan a website considering purpose, layout and accessibility. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> UX/UI concepts, accessibility, interactivity, multimedia integration. Website structure: master/landing/site pages, navigation. Planning tools: mood boards, wireframes. Evaluate against client briefs and suggest improvements. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Create webpages using structured layout and multimedia. Develop interactivity and navigation systems. Apply feedback to refine website content and design. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Master pages, hyperlinks, multimedia elements. Tools for creation (Wix, Google Sites, etc.). Interactivity: navigation bars, embedded content. Quality assurance, responsive design. <p>Key Ideas / Themes:</p> <ul style="list-style-type: none"> Real-world application of digital design. Iterative design process. Creativity and communication through media. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Review digital videos for narrative and technical effectiveness. Plan and script a short digital video for a target audience. Develop a storyboard with clear visual sequencing. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Storytelling structure: intro, conflict, resolution. Composition, framing, audio cues. Audience analysis and planning resources. Create scripts and shot lists. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Create and edit a video using appropriate software. Add transitions, text, sound, and effects. Reflect and refine based on feedback. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Editing software tools (e.g., WeVideo, iMovie). Media import, timeline use, audio syncing. Transitions and visual impact. Exporting and reviewing media products. <p>Key Ideas / Themes:</p> <ul style="list-style-type: none"> Crafting narratives through multimedia. Design for audience and purpose. Creativity and technical precision. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Analyse comic strip structure and conventions. Plan characters, scenes and layouts. Review examples and critique narrative and artistic choices. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Panel structure, genre conventions, audience targeting. Dialogue, expression and flow. Graphic novel/storyboard similarities. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Create a multipage comic with effective visual storytelling. Import and arrange characters, text, and panels. Evaluate and refine based on feedback. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Comic software (StoryboardThat, Pixton, Canva). Digital image use, speech bubbles, visual direction. <p>Layout and publishing</p>
Key ideas/ Themes:	<ul style="list-style-type: none"> Purposeful design and planning. User-centred development. <p>Critical evaluation of digital content</p>	<ul style="list-style-type: none"> Real-world application of digital design. Iterative design process. Creativity and communication through media. 	<ul style="list-style-type: none"> Visual storytelling. Effective communication through digital media. <p>Planning for purpose</p>		<ul style="list-style-type: none"> Narrative design. Visual pacing and flow. Characters and world-building 	<ul style="list-style-type: none"> Digital storytelling. Creative expression. Combining visual and written modes.
Prerequisite knowledge:	<ul style="list-style-type: none"> Y7: Evaluating website quality, understanding E-safety and navigation. Y8: Structure of websites, hardware/software understanding, and multimedia use. 	<ul style="list-style-type: none"> Y7: Web design evaluation and planning tools. Y8: Website construction tools, interactivity, and media embedding. 	<ul style="list-style-type: none"> Y7: Game planning and storytelling from GameMaker. Y8: Scratch sequencing, interactive design, and narrative structure. 	<ul style="list-style-type: none"> Y7-Y8: Multimedia integration, Scratch logic and structure. HT3: Storyboarding, scripting and media analysis. 	<ul style="list-style-type: none"> Y7: Visual storytelling and feedback in GameMaker. Y8: Design and planning in Scratch & Python output structure. 	<ul style="list-style-type: none"> Y7-Y8: Storyboarding, visual composition and game logic. HT5: Comic planning and review skills
Outcomes (Stickable-output that the student produces to demonstrate their knowledge)	<ul style="list-style-type: none"> Website review worksheet identifying strengths/weaknesses using criteria Website plan with annotated layout and structure. Wireframe/mock-up annotated with design rationale. Peer-reviewed critique of an existing website. 	<ul style="list-style-type: none"> Completed multipage functioning website. Self and peer evaluation forms. Screenshot portfolio with annotated feedback log. 	<ul style="list-style-type: none"> Video review worksheet identifying strengths/weaknesses using criteria Completed storyboard with written script. Evaluation of existing videos. 	<ul style="list-style-type: none"> Completed edited video. Storyboard vs. final product reflection. 	<ul style="list-style-type: none"> Comic plan with storyboard and character sketches. Analytical comparison of different comic types. Peer feedback form and planning reflections. 	<ul style="list-style-type: none"> Final multipage comic. Annotated evaluation. Self and peer review form with iterative changes documented.

Computing - Long term plans

Year 9 Route 2 (Computer Science)

Term	Autumn Term		Spring Term	Summer Term	
Half term	HT1	HT2	HT3 / 4	HT4 / 5	HT6
Unit Titles lengths:	Website Review & Planning	Create and Build Websites	Data Representation, Logic & Programs	Computational Thinking and Algorithmic Thinking	Intermediate Python Programming & Python Project
Learning objectives Substantive and procedural knowledge covered in the unit	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Identify and evaluate components of effective multi page websites. Understand audience targeting, user needs, and site structure. Plan a website considering purpose, layout and accessibility. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> UX/UI concepts, accessibility, interactivity, multimedia integration. Website structure: master/landing/site pages, navigation. Planning tools: mood boards, wireframes. Evaluate against client briefs and suggest improvements. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Create webpages using structured layout and multimedia. Develop interactivity and navigation systems. Apply feedback to refine website content and design. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Master pages, hyperlinks, multimedia elements. Tools for creation (Wix, Google Sites, etc.). Interactivity: navigation bars, embedded content. Quality assurance, responsive design. <p>Key Ideas / Themes:</p> <ul style="list-style-type: none"> Real-world application of digital design. Iterative design process. Creativity and communication through media. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Understand data units, binary, hex, and character encoding. Interpret logic gate outputs using truth tables. Analyse sorting/searching algorithms and apply number systems. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Binary, denary, hexadecimal conversion. Character encoding (ASCII), image and sound representation. Logic gates: AND, OR, NOT, combined gates. Sorting: bubble, merge; Searching: linear, binary. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Design algorithms using sequence, selection and iteration. Write and refine flowcharts and pseudocode. Apply decomposition and abstraction in problem solving. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Algorithm design: sequence, selection, iteration. Search/sort algorithms, time complexity. Pseudocode, flowcharts, debugging and refinement. Use of logic in branching and iteration. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Apply Python syntax to solve problems. Use variables, inputs, loops, and selection. Build and test a complete mini-project program. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Python syntax: inputs, outputs, selection, iteration. Data types, variables, expressions. Logic-based decision making. Debugging and refining code.
Key ideas/ Themes:	<ul style="list-style-type: none"> Purposeful design and planning. User-centred development. <p>Critical evaluation of digital content</p>	<ul style="list-style-type: none"> Real-world application of digital design. Iterative design process. Creativity and communication through media. 	<ul style="list-style-type: none"> Foundation of digital systems. Binary thinking and logical accuracy. Data compression, efficiency, and structure. 	<ul style="list-style-type: none"> Designing for clarity and efficiency. Step-by-step logical thinking. Refinement through feedback. 	<ul style="list-style-type: none"> Creative solutions through coding. Structured thinking and resilience. Real-world problem solving with Python.
Prerequisite knowledge:	<ul style="list-style-type: none"> Y7: Evaluating website quality, understanding E-safety and navigation. Y8: Structure of websites, hardware/software understanding, and multimedia use. 	<ul style="list-style-type: none"> Y7: Web design evaluation and planning tools. Y8: Website construction tools, interactivity, and media embedding. 	<ul style="list-style-type: none"> Y7: Flowol inputs/outputs, sequences. Y8: Binary representation, Scratch logic and flow, Python basics. 	<ul style="list-style-type: none"> Y7: Scratch sequencing, Flowol logic. Y8: Binary logic, Python sequences and selection 	<ul style="list-style-type: none"> Y7–Y8: Scratch and Python basics, logic, sequence/selection. HT4/5: Algorithm design and planning
Outcomes (Stickable-output that the student produces to demonstrate their knowledge)	<ul style="list-style-type: none"> Website review worksheet identifying strengths/weaknesses using criteria Website plan with annotated layout and structure. Wireframe/mock-up annotated with design rationale. Peer-reviewed critique of an existing website. 	<ul style="list-style-type: none"> Completed multipage functioning website. Self and peer evaluation forms. Screenshot portfolio with annotated feedback log. 	<ul style="list-style-type: none"> Conversion worksheets for binary, hex, and ASCII. Logic gate problems and complete truth tables. Completed algorithm flow diagrams with annotations. Analysis comparing algorithm efficiency. 	<ul style="list-style-type: none"> Written algorithms in pseudocode. Accurate, annotated flowcharts. Refined solutions with evidence of debugging. Completed programming problem scenarios. 	<ul style="list-style-type: none"> Completed Python mini-project with full working code. Internal documentation using comments. Project write-up including problem, design, and evaluation. Screenshot evidence of development stages. Feedback annotation and evidence of improvements.

Computing - Long term plans

Year 10 Computer Science

Term	Autumn Term		Spring Term		Summer Term	
Half term	HT1	HT2	HT3	HT4	HT5 / HT6	
Unit Titles lengths:	Systems Architecture & Memory	Storage & Networks (Part 1)	Networks (Part 2) & Protocols	Network Security & Systems Software	Ethical, Legal, Cultural & Environmental Impact	Algorithms & Programming Fundamentals
Learning objectives Substantive and procedural knowledge covered in the unit	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Understand the purpose and function of CPU components. Explain the stages of the fetch-execute cycle. Describe how common CPU characteristics affect performance. Distinguish between RAM and ROM and explain the purpose of virtual memory. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: CPU architecture, registers (MAR, MDR, PC, Accumulator), Von Neumann architecture, RAM, ROM, storage media. Procedural: Explain system performance factors, compare storage types, analyse RAM vs ROM, describe use of virtual memory. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Describe primary/secondary storage and file storage calculations. Identify network types and their features. Understand LAN/WAN characteristics and hardware. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Optical, magnetic, solid-state storage, capacity, performance. Procedural: Select appropriate storage, analyse networks, identify performance issues. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Evaluate star/mesh topology. Understand protocols like TCP/IP, HTTP(S), IMAP, SMTP. Explain IP/MAC addressing and encryption. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Topologies, protocols, addressing schemes, network layers. Procedural: Interpret scenarios, apply encryption, compare protocols. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Identify and mitigate types of cyber threats. Understand OS functions and utility software roles. Explain risk assessments and access controls. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Malware types, cyberattack prevention, user levels. <p>Procedural: Simulate attacks, configure OS settings, evaluate firewalls</p>	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Understand legislation like DPA, CMA, CDPA. Evaluate cultural and environmental impacts. Compare open-source vs proprietary software. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Ethics, privacy, law, e-waste, sustainability. Procedural: Apply legal knowledge to scenarios, critique case studies. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Apply abstraction, decomposition, sequencing. Write and test Python code using variables, inputs, outputs, loops. Use pseudocode and flowcharts. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Programming constructs, pseudocode, flowcharts. Procedural: Develop and debug Python programs.
Key ideas/ Themes:	<ul style="list-style-type: none"> Computer structure and processing. Performance and memory. Real-world embedded system examples. 	<ul style="list-style-type: none"> Efficient storage and connectivity. Network architecture and security considerations 	<ul style="list-style-type: none"> Secure data transmission. Efficient network communication 	<ul style="list-style-type: none"> Security and resilience. System maintenance and performance 	<ul style="list-style-type: none"> Digital responsibility. Impact of global technology 	<p>Key Ideas / Themes:</p> <ul style="list-style-type: none"> Problem-solving and automation. Thinking algorithmically
Prerequisite knowledge:	<ul style="list-style-type: none"> Y7: Role of input, processing and output. Y8: Binary and basic logic. Y9: Understanding components used in GameMaker and interactive media 	<ul style="list-style-type: none"> HT1: CPU components, memory hierarchy, storage media. Y7: Role of input, processing, and output. Y8: Cloud storage and local saving comparisons. Y9: Multimedia use in websites and data capacity planning. 	<ul style="list-style-type: none"> HT1-2: CPU processing, storage types and calculations, LAN/WAN characteristics. Y7: Role of input, processing, and output. Y8: Internet and online services. Y9: Data transfer in Scratch and GameMaker. 	<ul style="list-style-type: none"> HT1-3: Network types and protocols, addressing and performance, storage and CPU functionality. Y7: E-safety basics. Y8: Online threats, password security. Y9: Website review and interactivity with embedded content. 	<ul style="list-style-type: none"> HT1-4: Systems software, cyber security, network design and protocols. Y7-Y9: Online behaviour, copyright during design tasks. 	<ul style="list-style-type: none"> HT1-5: Understanding of system components, security, legal issues, and software tools. Y7-Y9: GameMaker logic, Scratch sequencing, Storyboarding.
Outcomes (Stickable-output that the student produces to demonstrate their knowledge)	<ul style="list-style-type: none"> Explain and label the components of the CPU and describe how they work together. Describe how clock speed, cache and number of cores affect CPU performance. Compare RAM, ROM and virtual memory with examples. Complete file size calculations using binary values. Apply theoretical knowledge through end-of-unit recap quiz. Sit milestone knowledge check with multiple-choice and structured questions. Demonstrate retrieval via knowledge organisers and graphic organisers. 	<ul style="list-style-type: none"> Perform file size calculations involving sound/image files. Construct and explain comparative storage device tables. Apply knowledge to recommend suitable storage types for real-world contexts. Identify and label LAN hardware components (e.g. switch, router, NIC). Peer-teach scenarios involving LAN/WAN comparisons. Complete network structure diagram activity. milestone review test on file storage and LAN features. 	<ul style="list-style-type: none"> Design comparison tables for topologies with pros/cons and use cases. Match protocols to correct uses and justify selections with written explanation. Complete Venn diagram activity comparing IP and MAC addressing. Model 4-layer protocol stack with real-world examples. Complete layered protocol worksheet using analogies. milestone assessment with questions on protocols and addressing. 	<ul style="list-style-type: none"> Develop a case study write-up describing a fictional cyberattack and response. Design a cyber awareness campaign (poster or infographic) explaining attack types and defences. Carry out practical activity applying user access levels and security settings. Explore anti-virus, firewalls, defragmentation and backups via interactive tasks. milestone assessment including scenario-based questions on cyber security 	<ul style="list-style-type: none"> Write a summary on DPA, CMA, CDPA including real-world applications. Develop a comparative guide explaining open-source and proprietary software. Lead a debate or presentation on digital sustainability and global impacts. Analyse news stories relating to digital ethics and produce annotated articles. Complete end-of-topic milestone covering legislation, impact, and ethics 	<ul style="list-style-type: none"> Create a Python program using inputs, loops and selection constructs. Plan, write and refine pseudocode for a given task. Translate pseudocode into working Python solutions. Produce a flowchart that models a basic algorithm using decisions and repetition. milestone programming project with peer review and annotated code. Demonstrate independent problem solving and code correction

Computing - Long term plans

Year 11 Computer Science

Term	Autumn Term		Spring Term		Summer Term	
Half term	HT1	HT2	HT3	HT4	HT5	HT6
Unit Titles lengths:	Algorithms	Programming Fundamentals		Boolean Logic, Robust Programs & IDEs	Revision and Exam Preparation (Paper 1 and 2)	
Learning objectives Substantive and procedural knowledge covered in the unit	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Understand and apply computational thinking principles. Design and refine algorithms using pseudocode and flowcharts. Analyse, interpret, and improve algorithms including searching and sorting. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Abstraction, decomposition, sequencing, algorithmic thinking, pseudocode, flowcharts, trace tables. Procedural: Design structured algorithms, implement binary/linear search, bubble/merge/insertion sort, evaluate and refine pseudocode. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Master sequence, selection, and iteration through structured programs. Implement variables, constants, input/output, and arithmetic operations. Explore and apply data types, string manipulation, casting, and file handling. Use iteration structures (for/while) for repetitive tasks. Implement and access data from 1D and 2D arrays. Read/write files and use SQL to search and manipulate data. Create and use subprograms for modular design. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Data types (int, real, bool, string, char), constants, variables, input/output operations, basic and logical operators, loops (definite/indefinite), 1D/2D arrays, file handling techniques, SQL basics, subprograms (procedures/functions). Procedural: <ul style="list-style-type: none"> Write structured programs using sequence, selection, and iteration. Apply variables, constants, and user input/output routines. Implement casting and string manipulation. Create and manage data with arrays and file I/O commands (open, read, write, close). Execute SQL SELECT queries to extract data. Design and call subprograms for reusable, modular code. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Use logic gates and truth tables for program decisions. Apply defensive programming techniques. Design test plans and validate with trace tables. Understand programming languages and IDEs. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Boolean logic (AND, OR, NOT), authentication, input validation, test data types, translators, IDE tools. Procedural: Create diagrams, write test cases, simulate program misuse, debug logic errors, use IDEs for diagnostics and code execution. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Prepare for external assessment. Revise all topic areas using metacognitive techniques. Ensure coursework folders are complete and submitted. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> full curriculum. Procedural: Examination technique, extended response structure, 		
Key ideas/ Themes:	<ul style="list-style-type: none"> Problem-solving through algorithm design. Efficient data processing and sequencing. 	<ul style="list-style-type: none"> Reliable code structure and logic. Modular and testable programs. Data-driven solution 		<ul style="list-style-type: none"> Secure and error-resilient systems. Syntax, logic and testing. IDE fluency and translator understanding. 	<ul style="list-style-type: none"> Consolidation of learning. Final improvements and target achievement. 	
Prerequisite knowledge:	<ul style="list-style-type: none"> KS3: Logic, flowcharts, variables, sequencing (Y7 Scratch, Y8 Python basics). Y10 HT1–HT6: CPU processing, memory, networking, security, and programming fundamentals. 	<ul style="list-style-type: none"> KS3: Basic Python syntax, conditional logic, loops. Y10 HT1–HT6: CPU components, memory, storage types, program control structures. Y11 HT1: Algorithm design, searching/sorting algorithms, pseudocode structure. 		<ul style="list-style-type: none"> KS3: Conditional logic, e-safety, block/text coding. Y10 HT1–HT6: System software, security, Python constructs, memory/storage. Y11 HT1–HT2: Programming constructs, SQL, pseudocode, algorithm design. 	<ul style="list-style-type: none"> All previous term content. Familiarity with exam command words and timings. 	
Outcomes (Stickable-output that the student produces to demonstrate their knowledge)	<ul style="list-style-type: none"> Create a step-by-step pseudocode and flowchart for a given scenario. Annotate an algorithm to explain abstraction and decomposition. Complete trace tables and debug syntax/logic errors. Apply binary and linear search to a dataset with justifications. Compare and evaluate two sorting algorithms in terms of performance. Milestone assessment with error analysis and improvement task 	<ul style="list-style-type: none"> Build and test programmes using variables, constants, and arithmetic operators. Create text-programmes using selection and user inputs with validation. Produce annotated pseudocode for structured logic. Write a program using 2D arrays (e.g. seating plan, grade tracker). Add file operations to save and retrieve structured data. Use SQL commands (SELECT, WHERE) to extract data from a table. Create reusable subprograms to reduce repetition and enhance maintainability. Complete Milestone 1 (logic test & debugging challenge) and Milestone 2 (programming task with peer-review and improvement log). 		<ul style="list-style-type: none"> Construct logic diagrams and truth tables for a problem. Create a test plan using normal, boundary, and erroneous data. Simulate user authentication and validation within a coded program. Debug code using IDE error diagnostics and explain solutions. Compare compilers and interpreters and submit an annotated IDE task with reflection. 	<ul style="list-style-type: none"> Mock paper scores with WWW/EBI. Final coursework review sheet. Personal revision action plan. Submission log and folder checklists. 	

Computing - Long term plans

Year 10 Creative imedia

Term	Autumn Term		Spring Term		Summer Term	
Half term	HT1	HT2	HT3	HT4 - R094 External submission	HT5	HT6
Unit Titles lengths:	R094 NEA Task 1 – Planning the Visual Identity and Digital Graphic Product Also Delivered: R093 Topic Area 1 – The Media Industry	R094 NEA Task 1 continued – Planning and Branding Finalisation Also Delivered: R093 Topic Area 2 – Factors Influencing Product Design	R094 Task 2 – Creating the Visual Identity and Digital Graphic Product Also Delivered: Interleaved Practice – R093 Topic Areas 1 and 2	R094 Task 2 continued – Editing, Refining and Saving Digital Graphics Also Delivered: Interleaved Practice – R093 Topic Areas 1 and 2	R095 Task 1 – Planning Character and Comic Product Also Delivered: R093 Topic Area 3 – Pre-production Planning	R095 Task 1 continued – Developing Comic Characters and Planning Tools Also Delivered: R093 Topic Area 3 – Continued Application and Revision
Learning objectives Substantive and procedural knowledge covered in the unit	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Understand visual identity principles including alignment, typography, white space, and colour theory. Apply design conventions to produce visual materials suitable for a client brief. Identify and analyse media industry sectors, evolving products, and job roles. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Visual identity, layout conventions, logo design, typography, media industry sectors and products, creative/technical/senior job roles. Procedural: Research and design using Fireworks, asset sourcing and editing, apply conventions to plan graphics, identify job responsibilities across media phases. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Refine design ideas using industry conventions. Explore audience segmentation, client briefs, and product purpose. Develop annotated assets and apply technical/symbolic codes in planning. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Client requirements, audience segmentation, qualitative vs quantitative data, research methods, media codes. Procedural: Client brief analysis, interpret layout/style, generate research data, annotate graphical intentions. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Create digital graphics using bitmap and vector editing techniques. Source, edit, and optimise assets within a set technical specification. Recall and apply R093 Topic Areas 1 and 2 through spaced retrieval. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Asset types, licensing, file formats, job roles, audience needs. Procedural: Design using layout tools, layer management, asset integration, R093 dual coding and retrieval practice 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Use advanced Fireworks tools (effects, filters, retouching). Edit and refine graphics in response to feedback. Revisit and reinforce R093 topics through spaced practice. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Adjustments, layers, filters, layout precision. Procedural: Editing workflow, exporting assets, flashback quizzes, improving visual effectiveness. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Plan characters using visual tropes and techniques. Plan media products using storyboards and visualisation diagrams. Explore legal and ethical issues linked to pre-production. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Work planning, legal issues, intellectual property, risk, regulation. Procedural: Character sketching, planning storyboards, risk assessment scenarios, visualisation diagram completion. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Refine comic character assets and integrate into the storyboard plan. Complete planning documents including work plan and visualisation. Review and apply R093 Topic 3 legal/ethical content to your own project. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Health and safety, copyright, legal protections, storyboard accuracy. Procedural: Document completion, risk mitigation, client brief alignment.
Key ideas/ Themes:	<ul style="list-style-type: none"> Branding and messaging through digital design. The structure and evolution of the media industry and its workforce. 	<ul style="list-style-type: none"> Designing for purpose and audience. Media literacy and representation techniques. 	<ul style="list-style-type: none"> Combining design skill with theoretical understanding. Practical creation aligned to industry context. 	<ul style="list-style-type: none"> Refinement and iteration. Retention through retrieval and dual coding. 	<ul style="list-style-type: none"> Planning for success and safety. Responsibility and legal awareness in design. <p>Prerequisite Knowledge:</p>	<ul style="list-style-type: none"> Detail and clarity in planning. Final refinement and client readiness.
Prerequisite knowledge:	<ul style="list-style-type: none"> KS3: Digital imaging, online safety, poster/flyer design, colour theory. HT1 links to KS3 media and art projects; HT2 will build upon use of Fireworks and graphic conventions. 	<ul style="list-style-type: none"> KS3: Careers in media, group projects, media language. HT1: Media sectors and job roles, brand brief analysis, audience targeting. 	<ul style="list-style-type: none"> KS3: File formats, saving conventions, folder structure. HT1–HT2: Branding development, Fireworks interface, R093 topics 	<ul style="list-style-type: none"> KS3: Editing tools in Paint.NET/GIMP. HT1–HT3: Initial designs, asset use, R093 recall. 	<ul style="list-style-type: none"> KS3: Character design, comic layout, copyright basics. HT1–HT4: Product planning, visual assets, audience targeting. 	<ul style="list-style-type: none"> KS3: Visual storytelling, media planning tools. HT1–HT5: Pre-production, legal/ethical context, client/audience understanding.
Outcomes (Stickable-output that the student produces to demonstrate their knowledge)	<ul style="list-style-type: none"> Completed visual identity visualisation and justification. A researched visual identity plan showing appropriate design conventions. Completed audience research and planning sketches. Completed R093 Topic 1 fact sheet: sectors, products, and job roles. Fireworks practice assets saved in correct formats. Milestone assessment task: visual identity plan and client response sheet. 	<ul style="list-style-type: none"> Completed mood board with annotated choices. Written response analysing client brief and audience segmentation. Completed research table with primary/secondary methods. Layered mock-up of a digital graphic in Fireworks with key assets shown. Milestone assessment on R093 Topic Area 2: extended written response on product design. 	<ul style="list-style-type: none"> Completed digital asset library with saved file formats. Interleaved R093 quiz recall (Topic 1 + 2). Annotated screenshot walkthrough of design process. Milestone project submission with evaluation of design against brief. 	<ul style="list-style-type: none"> Completed design exported in required formats. Screenshot evidence of filter and retouch tool use. Interleaved worksheet R093 1+2 flashback. Milestone fix-it activity – apply peer feedback to design. 	<ul style="list-style-type: none"> Completed character profile and visualisation. Storyboard for digital comic sequence. Written response on legal scenario from R093 Topic 3. Milestone: Work plan with timeline, risk factors and roles. 	<ul style="list-style-type: none"> Finalised comic narrative, annotated layout, and character sheet. Evaluated comic against target audience and legal considerations. Fix-it task: improving compliance with regulation and IP laws. Topic 3 recap worksheet: regulation bodies, risk mitigation. Peer assessment and improvement log for R095 planning. End-of-year exam-style task on legal and ethical scenarios.

Computing - Long term plans

Year 11 Creative imedia

Term	Autumn Term		Spring Term		Summer Term	
Half term	HT1	HT2	HT3	HT4	HT5	HT6
Unit Titles lengths:	R095 Task 2 (Creating Characters and Comics), R093 Topics 1–3 (Spaced Practice & Interleaving)	R095 Task 2 Continued + Task 3 (Evaluating and Reviewing the Comic Product), R093 Topics 1–3 (Continued Interleaving and Practice)	R093 Topic 3 – Pre-production Planning	R093 Topic 4 – Distribution, Formats and Accessibility	Revision and Exam Preparation (R093 Entire Unit) + Coursework Folder Improvement	
Learning objectives Substantive and procedural knowledge covered in the unit	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Develop the visual style and structure of the comic strip. Create graphical elements and refine character designs. Revisit and reinforce R093 Topics 1–3 using interleaving and retrieval practice. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Comic design principles, character consistency, R093 knowledge on media sectors, job roles, audience, and product design. Procedural: Comic panel layout, graphic editing, character rendering, knowledge organiser tasks, revision strategies. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Finalise and export completed comic product. Evaluate own work and peer contributions. Continue spaced retrieval practice for R093 Topics 1–3. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Review techniques, target setting, improvement planning, regulatory considerations. Procedural: Editing, exporting, evaluation frameworks, mock exam application. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Understand planning phases, documentation and workflows. Identify and apply the purpose of planning documents (e.g. storyboards, scripts, work plans). <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Phases of production, documentation types and their uses, work plan components. Procedural: Document critique, identifying suitable planning tools, applying content to scenarios. 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Understand how media products are distributed across platforms. Explore file formats, compression, and accessibility features. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: Distribution channels, media convergence, file types, accessibility standards. Procedural: Selecting formats for purpose, evaluating distribution strategies 	<p>Learning Objectives:</p> <ul style="list-style-type: none"> Prepare for external R093 assessment. Revise all topic areas using metacognitive techniques. Ensure coursework folders (R094/R095) are complete and submitted. <p>Substantive & Procedural Knowledge:</p> <ul style="list-style-type: none"> Substantive: R093 full curriculum. Procedural: Examination technique, extended response structure, folder organisation. 	
Key ideas/ Themes:	<p>Key Ideas / Themes:</p> <ul style="list-style-type: none"> Visual storytelling through character and scene creation. Spaced learning to support long-term retention. 	<ul style="list-style-type: none"> Refinement and critical reflection. Retrieval as preparation for external assessment. 	<ul style="list-style-type: none"> Planning as a vital stage in production. Justifying choices in response to briefs. 	<ul style="list-style-type: none"> Accessibility and file choices shape user experience. Distribution must suit audience needs and media types. 	<ul style="list-style-type: none"> Consolidation of learning. Final improvements and target achievement. 	
Prerequisite knowledge:	<p>Key Ideas / Themes:</p> <ul style="list-style-type: none"> Visual storytelling through character and scene creation. Spaced learning to support long-term retention. <p>Prerequisite Knowledge:</p> <ul style="list-style-type: none"> KS3: Comic strip layout, digital imaging. Y10 HT1–HT6: R093 full content, character planning from R095 Task 1. 	<ul style="list-style-type: none"> KS3: Storyboarding and evaluation. HT1: Comic development. Year 10 R093 & R094 content. 	<ul style="list-style-type: none"> KS3: Basic pre-production document creation. Y10/11: Creative tasks from R094 and R095 used to apply R093 principles. 	<ul style="list-style-type: none"> KS3: Basic knowledge of file types and media players. Y10–11: Asset creation in R094/R095, export processes. 	<ul style="list-style-type: none"> All previous term content. Familiarity with exam command words and timings. 	
Outcomes (Stickable- output that the student produces to demonstrate their knowledge)	<ul style="list-style-type: none"> Students complete a comic strip layout using pre-designed character assets and speech bubbles. Evidence of consistent application of visual storytelling conventions (e.g. panel flow, camera angles, emotion). digital exports of comic in required file format (e.g. PDF, JPEG). R093 quiz completion sheets (retrieval on job roles, audience, research). Students complete a peer review and refinement task in response to given success criteria. 	<ul style="list-style-type: none"> Students refine and complete their digital comic strip using feedback from HT1. Exported final product in the correct format (e.g. PDF, JPG), with clear narrative structure and visual consistency. Annotated screenshot walkthrough demonstrating how design elements were applied (e.g. mise-en-scène, interactivity, codes). Students complete a self-evaluation of how well their comic meets the client brief and audience needs. R093 retrieval activities embedded into starters/plenaries (e.g. flashback quizzes, dual coding tasks). Peer assessment task on another student's comic strip using provided success criteria. 	<ul style="list-style-type: none"> Students complete structured worksheets on work plans, mind maps, and mood boards. Physical or digital samples of planning documents are stored in folders. Completed planning task showing understanding of how pre-production supports success in creative projects. In-lesson task matching documents to production phases. Students engage in timed retrieval tasks from previous R093 topics (1–3). A planning milestone task where students analyse and improve an example work plan or script. 	<ul style="list-style-type: none"> Students complete a case study task analysing how a chosen media product was distributed across platforms. Group discussion or written response on how target audiences influence marketing strategy. Completion of worksheet comparing online vs traditional distribution methods. Visual timeline or infographic of a product lifecycle (planning → distribution). Mini assessment paper testing all four R093 topics. Students complete a revision journal entry reflecting on their weakest R093 topic and how they'll improve it. 	<ul style="list-style-type: none"> Mock paper scores with WWW/EBI. Final coursework review sheet. Personal revision action plan. Submission log and folder checklists. 	



Computing - Long term plans