YEAR A						
Vocabulary	**Question**	**Key Concepts**	**Common Misconceptions**	**Substantive Knowledge**	**Disciplinary Knowledge**	**Oracy**
Tier 2: use help safe kind follow different **Tier 3:** technology computer device mouse keyboard monitor/screen input output	**<u>What is technology and how can I use it responsibly?</u>** What is technology? What parts does a computer have? What is a mouse and how can it be used in different ways? What is a keyboard and how do you use it? How do you edit text on a keyboard? What rules make using technology responsibly look like? ARC	Technology is all around us – it helps people in different places (e.g. school, home, shops). A computer is a type of technology that follows instructions to do tasks. Computers have parts (e.g. keyboard, mouse, screen) that we use to give instructions or see results. We use computers for different purposes, like writing, drawing, or playing. Technology should be used responsibly – safely, kindly, and with permission.	Technology only means computers or iPads. A screen is a computer. All technology is for games. You can touch any technology. Computers work by themselves.	What technology is and that it can be found in many places. That a computer is a kind of digital device that follows instructions. Examples of technology at school and at home (e.g. tablets, printers, cameras). The names and purposes of computer parts: mouse, keyboard, monitor, screen. That we use computers for different tasks: typing, drawing, playing, watching. That we must use technology safely, kindly, and with adult help if needed.	Observation identifying where technology is used in the classroom or school. Classification - Grouping items into 'technology' and 'not technology'. Exploration - Using parts of a computer (e.g. mouse to draw). Discussion - Talking about how and why we use technology. Responsibility - Learning to care for devices and use them kindly.	"I can see technology in the…" "This is technology because…" "We use this to…" "I used the computer to…" "Next time, I will try to…" "I stayed safe by…"
Tier 2: move change next before try fix **Tier 3:** program instruction sequence animation sprite	**<u>What is programming and how can I make an animation?</u>** What is a command and what will it do? How can you join a series of commands together? What is the effect of when you change a value?	Programs are sets of instructions that computers follow exactly. Animations can be made by sequencing instructions to move or change a character (sprite). Order matters – changing the order of instructions	The computer knows what I mean. All commands happen at once. If it doesn't work, it's broken. The character moves by itself. If I click 'Go' again, it will fix it."	That a program is a sequence of instructions a computer can follow. That animations are made by programming movements or actions in a specific order. That a sprite (character or object) can be told to move,	Sequencing - Creating a step-by-step set of instructions to move a sprite or object. Programming - Using blocks to make a character move or animate. Testing - Running the program to see what happens. Debugging - Spotting and fixing	"My program makes the sprite…" "This block tells the sprite to…" "The order is important because…" "It didn't work because…" "I fixed it by…"

debug	Why does each sprite have its own instructions? What designs will inform your project? How can you use an algorithm to create a program? ARC	changes the outcome. Computers need clear, step-by-step commands (no guessing or interpreting). Debugging is part of programming – mistakes help us improve		turn, or show emotions. That sequencing means putting instructions in the correct order. That debugging means fixing mistakes in the code. That a computer only does what it's told – nothing more, nothing less.	mistakes in the code (e.g. wrong direction or extra steps). Predicting - Saying what will happen before running a program. Explaining - Talking about what each block or instruction does.	"Next time I will remember to…"
Tier 2: choose change make try shape colour **Tier 3:** digital paint tool brush tool eraser fill tool undo mouse control	**<u>How can I create a digital painting?</u>** What is a freehand tool? What are shape and line tools and what do they do? How can you make careful choices when painting a digital picture? How can you explain your choice of tools? How can you paint a picture on a computer What comparison can we make between a computer and paper painting? ARC	Digital tools can be used to create art, just like pens or paint. Choosing the right tool helps artists achieve their creative goals. Digital art can be edited or improved without starting over. Artists can express ideas through colours, shapes, and tools. Technology is a tool we use to create, not just to consume	Drawing on a screen is just playing. All paint tools do the same thing. Mistakes can't be fixed. You need to copy to make good art. Computers make the art by themselves.	What a digital painting program is and what it can do. That they can select and use tools (e.g. brush, fill, eraser) to create artwork. That colours and shapes can express ideas and feelings. That they can change and improve a digital painting without starting again. That digital paintings can be saved, named, and shared. That different digital tools are suited to different artistic effects.	Exploring - Trying different paint tools to see their effects. Creating - Making a picture with colour, shape, and digital tools. Editing - Using the undo or eraser to improve a digital painting. Evaluating - Talking about what they like or would change about their artwork Sketching - Choosing a tool or colour based on artistic purpose	"I used the ___ tool to…" "My picture shows…" "I chose this colour because…" "Next time I will try…" "I liked my painting because…"

***Comprehensive working versions of these documents are available to school staff, outlining arrangements for fieldwork, educational trips, and associated resourcing.**

Year 1 and 2 Year A

YEAR B						
Vocabulary	**Question**	**Key Concepts**	**Common Misconceptions**	**Substantive Knowledge**	**Disciplinary Knowledge**	**Oracy**
Tier 2: common popular organise data tally chart votes total , data compare objects count explain group **Tier 3:** attribute block diagram pictogram	**<u>What is pictogram and how does it present information?</u>** What is a tally? How can objects be represented as pictures? What is a pictogram? How can I select objects by attribute and make comparisons? How can people be described by attributes? How can we present information using a computer?	A pictogram (or pictograph) is a way of showing data using pictures. Each picture represents a certain number of items. Helps people easily see and understand information. Teach children how to gather data by asking questions or conducting a survey. Emphasize tallying and counting correctly.	Misunderstanding the Key Incorrect Counting or Interpreting Symbols Used Inconsistently No Labels or Headings Drawing Too Many or Too Few Pictures Confusing Tallying with Pictogram Symbols Assuming Pictograms Are Always Better Than Bar Charts Not Checking for Accuracy	What a pictogram is and how it is used to represent data. The meaning of a key in a pictogram (e.g., 1 picture = 2 people). That data can be collected, sorted, and represented visually. Different symbols or pictures can be used to represent different quantities. Data can show things like most popular, least popular, and total. Software tools (e.g. 2Count, J2Data) can be used to make pictograms digitally.	Designing a survey or data collection method to answer a question (e.g., "What is the class's favourite fruit?"). Gathering and recording data accurately (e.g., using tallies or checklists). Choosing appropriate symbols and designing a pictogram that communicates clearly. Using digital tools to input and visualize data. Interpreting pictograms to draw conclusions or answer questions (e.g., "Which fruit was the most popular?"). Evaluating the effectiveness of a pictogram (e.g., "Is this easy to understand?").	"A pictogram is a type of chart that shows data using ___." "Each picture in this pictogram stands for ___." "The key tells us that ___." "This pictogram shows that the most popular ___ is ___." "I chose this symbol because it represents ___."
Tier 2: pattern repeat repetition value **Tier 3: Added as relevant**	**<u>What is an algorithm and how can it program a robot?</u>** What is a sequence? How can you change the order of instructions?	Algorithm = a set of precise instructions to achieve a goal. Algorithms can be followed by humans or programmed into robots/computers.	Children often think an algorithm *is* a program. Pupils may think steps can be put in *any* order and still work. Children may think robots are "smart" and will do what they	An algorithm is a sequence of step-by-step instructions to solve a problem or complete a task. Programs are written instructions that a computer or robot can follow.	Designing algorithms - Pupils can create their own algorithms to solve a simple problem (e.g., getting from A to B). Programming - Pupils can input instructions into a robot or coding app (e.g.,	An algorithm is a set of instructions to ___." "My program tells the robot to ___." "I chose these steps because ___."

debug count-controlled loop trace decompose procedure	How can we predict the outcome of a program? What is code? How can you design an algorithm? What is debugging and how can it help a program?	Programs are how we give instructions to computers or robots. Debugging means finding and fixing errors in a program. Computers/robots follow instructions exactly, so instructions must be clear and in the correct sequence. Logical reasoning helps predict what a program will do.	*mean*, not what they *program*. Turning instructions confuse children (left/right) Debugging means starting over Multiple instructions = one big action Not understanding what an algorithm is without a computer Input/output confusion	The order of instructions matters – changing it changes the outcome. If a program doesn't work as expected, it can be fixed by checking and changing the instructions. Computers and robots do not think – they follow what we tell them, exactly.	Beebots, ScratchJr, Blue-Bots). Debugging - Pupils can spot when a sequence doesn't work and try to fix it. Logical reasoning - Pupils can explain why something happened and predict what will happen next in a program. Evaluating - Pupils can talk about what went well and what could be improved in their program.	"The robot will do ___ because that's what the program says." "I think the robot will move ___ because the instructions are ___." "If I change the order, it will ___." "I know this is wrong because ___." My program didn't work because ___." "I fixed the mistake by ___." "Next time, I would ___." "I found it easy/hard when ___." "I liked programming because ___." "I learned that instructions must be ___."
Tier 2: music quiet loud feelings emotions pattern create emotion instrument open edit **Tier 3:** tempo rhythm pulse	**<u>How can I make digital music?</u>** How does music make you feel? What patterns are there in music? How can I use a computer to experiment with sounds? How can I use a computer to create a musical pattern?	Music can be created using technology. Computers can be used to compose, edit, and play back digital music. Sounds can be arranged in patterns (like beats or melodies). You can change instruments, tempo, and volume to	Computers make music by themselves. Faster music means louder music. All digital music sounds the same. You can't change music once it's made. You need to be good at music to use a music app. All sounds must follow the same pattern.	Sound - Music is made from sounds that can be high/low (pitch), loud/quiet (volume), and fast/slow (tempo). Music tools - Computers can be used to create music using software that mimics instruments. Composition - Music can be created by arranging notes or sounds in a specific sequence or pattern.	Creating - Pupils can use digital tools to create a simple music sequence or rhythm. Editing - Pupils can change sounds by altering pitch, speed (tempo), and instruments. Evaluating - Pupils can listen to their music and say what they like or what they could improve. Reasoning - Pupils can explain how	"I created this sound by using the ___ instrument." "I made my music faster/slower by changing the ___." "This pattern repeats because ___." "I used high/low sounds to make it sound like ___." "I changed the instrument because ___."

pitch tempo rhythm notes beat	How can I create music for a purpose? How can you review and refine your computer work?	create different effects. Digital music is a form of media that can be shared or saved.		Editing - Digital music can be changed easily – we can undo, reorder, or replace sounds. Media - Digital music is one form of digital media, like images or videos.	they made their music and why they made certain choices. Saving/sharing - Pupils know how to save, play back, and share their digital music.	"I didn't like how that sounded, so I ___." "Next time, I would try to ___."

***Comprehensive working versions of these documents are available to school staff, outlining arrangements for fieldwork, educational trips, and associated resourcing.**

YEAR A						
Vocabulary	**Question**	**Key Concepts**	**Common Misconceptions**	**Substantive Knowledge**	**Disciplinary Knowledge**	**Oracy**
Tier 2: connect process device function share collaborate communicate identify respond present **Tier 3:** network router internet server input output digital device collaboration tool data Wi-fi	**<u>How do you connect computers?</u>** How do digital devices function? What are input and output devices? How can digital devices change the way we work? How can a computer network be used to share information? How can digital devices be connected? What are the physical components of a network? ARC	A device that uses computer technology (e.g., tablet, laptop, smartwatch) Devices receive input (keyboard, microphone), process it, and produce output (screen, speakers). A group of connected digital devices that can share information. Router, switch, server, and the devices (computers, printers, etc.). The internet is a global network of networks. Sending messages, files, or pictures via email, shared documents, or messaging platforms. Collaborating online while following safety rules and showing respect.	The internet is not stored on one computer – it's a network that many devices connect to. Wi-Fi is a way to connect to a network, not the internet itself. Different devices have different functions, inputs, and outputs. Data is transmitted via networks using specific protocols. Online sharing can be public; students need to understand permissions and safety.	Identify digital devices and their inputs/outputs. Compare digital and non-digital devices. Demonstrate how devices connect in a network (e.g., role-play or drawing). Log in and collaborate in a shared digital space (e.g., Google Docs or Microsoft 365). Use a search engine with guidance. Send and receive a message safely (within a closed platform or simulation). Use correct terminology (device, network, server, input/output).	Decomposition: Breaking down systems into input, process, and output. Logical reasoning: Understanding how actions (e.g., sending a message) have predictable outcomes in a network. Abstraction: Using models (diagrams or role-play) to represent complex networks. Evaluation: Reflecting on how effective or safe their communication is.	"I think this is a digital device because…" "This part of the system helps by…" "The output of this device is…" "Next time, I would change…" "I learned that networks are important because…"
Tier 2: sequence instruction create change control check **Tier 3:** code program block sound block	**<u>What is Scratch?</u>** How do you program a sprite with commands? What is a sequence in Scratch? How do you order commands?	Computers follow instructions in the order they're given. Students can instruct computers when and how to play sounds. In block-based languages, each block has a specific function Instructions run in sequence and can	Computers only do *exactly* what they're told in the code. Some blocks only work in specific ways (e.g., a 'play sound' block needs a sound to exist). Code runs in order, top to bottom, unless controlled otherwise.	A program is a set of instructions written using code. A block is a visual piece of code used in block-based programming tools. Sequence means the order in which instructions run – changing the order	Sequencing Ordering instructions to make a sound program work Debugging Testing the program and fixing problems if sounds don't play correctly Decomposition	"I used this block to…" "The sound plays after this block because…" "My program works by…" "I found a mistake because…"

sequence debug timing event	How can you change appearances? How can you make an instrument using Scratch? ARC	include delays, repeats, or events. Code can be used to create stories, music, or animations	Coding is logical and structured, but not always mathematical. It's likely a problem with the code – debugging helps fix it.	changes the outcome. A sound block can be used to play music, effects, or voices. Wait blocks control timing – they pause the program between instructions. You can create and debug your own simple sound program. Programming is not just technical – it can also be creative and fun.	Breaking down the problem into parts. Evaluation Reviewing if the program works as intended and thinking about improvements Abstraction Focusing only on the important blocks needed to get the desired output	"Next time, I would…" "I debugged my program by…"
Tier 2: sequence create change plan organise review careful **Tier 3:** animation frame stop-motion animation storyboard timeline camera angle capture playback	**<u>What is an animation?</u>** What is a sequence of images? How can you plan an animation? Why do you need to work consistently and carefully? Why should you review and improve an animation? How can you add other media to an animation? ARC	Stop-frame animation is created by showing a sequence of images very quickly. Each frame is a single image—small changes between frames create the illusion of movement. Digital devices can capture and play back animations using software tools. Planning and sequencing are key to successful animation. Media creation is both technical and creative—it involves storytelling, editing, and digital skills	Animation is drawn like cartoons only. Moving the object a lot will look better. It doesn't matter where the camera is placed. All the frames are the same. Making an animation is quick and easy.	What stop-frame animation is and how it works. That each frame is a still image, and small movements between frames create the effect of motion. That animations are planned using storyboards. How to use a digital device (tablet, app, or camera) to capture images. How to edit and sequence frames using digital tools. The importance of keeping the camera still for a smooth animation.	Sequencing - Placing images in a logical, time-based order Evaluation - Watching and improving the animation after playback. Decomposition - Breaking the animation into smaller steps: planning, capturing, editing Iterating - Making small changes and improvements through trial and error. Precision - Keeping the camera steady and making small, intentional movements of objects.	"In our storyboard, the character starts by…" "We chose this object because…" "We planned to move it slowly so that…" "Next time, I would change…" "It didn't look right because…" "We improved our animation by…"

				That playback lets you preview and improve your work.		

***Comprehensive working versions of these documents are available to school staff, outlining arrangements for fieldwork, educational trips, and associated resourcing.**

Year 3 and 4 B

YEAR B						
Vocabulary	**Question**	**Key Concepts**	**Common Misconceptions**	**Substantive Knowledge**	**Disciplinary Knowledge**	**Oracy**
Tier 2: group sort choose clear correct describe **Tier 3:** data database branching databases field record criteria classifying	**<u>What is a branching database?</u>** How do I create questions with yes/no answers? What attributes are needed to collect data about an object? How can I create a branching database? Why is it helpful for a database to be well structured? How can I plan the structure of a branching database? How can I create an identification tool? ARC	Data can be organised and sorted using questions. Branching databases use yes/no questions to divide data. The quality of a database depends on clear, accurate questions. Digital tools help us classify and find information efficiently. We can use branching databases to identify, compare and group information.	All questions can go in any order. A branching database gives opinions. The first question doesn't matter. You can only make a branching database with animals. All data must go under the same branch."	What a branching database is: a structure that uses yes/no questions to sort data. That data can be grouped and organised using characteristics. The importance of using clear, specific yes/no questions. That each question should split the data in a meaningful way. That digital tools can be used to create and test branching databases. That databases help us identify and classify objects or information quickly.	Classifying - Group objects based on observable features. Questioning - Generate clear yes/no questions to split data. Sequencing - Order questions so the database works logically. Testing - Use branching databases to identify items. Evaluating - Reflect on the clarity and accuracy of questions.	"I chose this question because it helps to…" "The database works by asking…" "This item goes here because…" "Next time, I would ask…" "Our question didn't work well because…" "We could make the database clearer by…"
Tier 2: repeat change pattern step clear efficient **Tier 3:** algorithm loop/repetition code command debug sequence turtle angle	**<u>How do you repeat shapes in Logo?</u>** Why is accuracy in programming important? How can I create a program in text-based language? What is 'repeat' and what does it mean in coding? How can I modify count-controlled loop to produce a given outcome?	Programs can use loops to repeat instructions instead of writing them multiple times. Repetition makes code more efficient, clearer, and easier to change. Loops can create patterns and shapes by repeating movements. Algorithms must be precise and in the correct order to achieve the desired outcome.	The computer understands what I want to draw. Loops make the shape automatically. All shapes need the same instructions. You don't need to test your code. You can only repeat full shapes.	A loop is a programming construct that repeats a set of instructions. Repetition makes code more efficient and easier to manage. Shapes can be drawn by combining forward movements and turns. The number of repeats and turn	Programming - Writing code to create shapes using repeat commands. Decomposition - Breaking shapes into steps (e.g. a square = 4 lines and 4 turns) Sequencing - Placing commands in the correct order for intended results. Abstraction - Identifying patterns and using loops to simplify code.	"My code tells the turtle to…" "This loop repeats the instruction to…" "I used a repeat block because…" "It didn't work because…" "I fixed the error by…" "Next time, I will remember to…"

	How can I chunk a task into small steps? How can I create a program that uses count-controlled loops to produce a given outcome? ARC	Debugging is essential when code doesn't behave as expected.		angle affect the final output. Programming languages like Logo or Scratch can be used to create shapes with repetition. Debugging means identifying and fixing errors in a program.	Testing and debugging - Running code, identifying errors, and making corrections. Logical reasoning - Predicting the outcome of loops or sequences.	
Tier 2: plan record clear edit choose mistake **Tier 3:** audio microphone sound wave editing export volume clip timeline	**<u>How do you produce audio on a digital device?</u>** How can sounds be recorded? How can audio recordings be edited? What is a podcast and how what different parts create them? How can I edit audio independently? How can I combine audio? How can I evaluate the effective use of audio? ARC	Audio is a form of digital media that can inform, entertain, or persuade. Recording and editing tools help us shape and refine sound. Sound quality, clarity, and planning affect how well a message is communicated. Digital media can be created for a specific audience and purpose. Creating media involves a process: plan → create → edit → publish.	You just press record and talk. Any microphone will sound the same. You can't change a mistake once recorded. Audio can only be used for music. Once it's saved, it can't be changed.	Audio is sound recorded and stored digitally (voice, music, effects). Sound can be captured using a microphone and saved as a file. Audio editing software allows changes, such as trimming, reordering, and adjusting volume. Planning and scripting improve clarity and structure of digital audio. Audio content can be created for specific audiences (e.g. younger children, classmates). Audio projects go through stages: planning → recording → editing → evaluating → publishing.	Media creation - Plan, record, and edit a digital audio file. Composition - Structure audio content with purpose and audience in mind Evaluation -. Review audio quality and suggest improvements Digital editing - Cut, move or adjust clips using audio software. Collaboration - Work in teams to plan, produce and review work. Problem solving - Troubleshoot issues with recording, background noise, or exporting files.	"Our audio is about…" "We chose this topic because…" "The audience for our recording is…" "What worked well was…" "Next time we would improve…" "I like how the sound effect adds…"

Year 3 and 4 B

***Comprehensive working versions of these documents are available to school staff, outlining arrangements for fieldwork, educational trips, and associated resourcing.**

YEAR A						
Vocabulary	**Question**	**Key Concepts**	**Common Misconceptions**	**Substantive Knowledge**	**Disciplinary Knowledge**	**Oracy**
Tier 2: system component process efficient reliable appropriate evaluate **Tier 3:** network server client router search engine indexing algorithm key word web crawler	**<u>How can computers be connected together?</u>** How are computer systems in our lives? What is a search engine and how can I explore it? How do search engines select results? How are search results ranked? Why is the order of search results important? ARC	A system is a group of parts working together – in computing, this includes hardware, software, and networks. Computer systems can be connected together (e.g. in a network) to share data and resources. Search engines use algorithms to find and rank web pages. Searching effectively requires choosing appropriate keywords. Information online is not always accurate or reliable	The internet is the same as the World Wide Web. Search engines search the whole internet live. The top search result is always the best. All computers work on their own. Any information on the internet is true.	A system is made up of connected components that work together. Computer systems include hardware (devices) and software (programs). A network connects devices and allows them to share data. The internet is a network of networks. A search engine helps users find information online. Search engines use web crawlers, indexing, and algorithms to rank results. The choice of keywords affects the relevance of search results. Not all online information is reliable – pupils should think critically when evaluating sources.	Abstraction – focusing on important information. Decomposition – breaking problems into smaller parts. Logical reasoning - making sense of how things work. Evaluation - judging the usefulness or accuracy. System thinking - understanding how parts interact.	"This part of the system helps by…" "The server stores the files and…" "The router connects…" "I used the keyword(s) '___' to find…" "The search engine showed these results because…" "Next time, we could try using…" "Let's compare our search results and see which are better."
Tier 2: control detect cause respond check depend **Tier 3:** selection	**<u>What is selection in physical computing?</u>** How can I control a simple circuit from a computer? How can I write a program to include a count-controlled loop?	Selection allows programs to make choices based on conditions (e.g. "if a button is pressed, then turn on a light"). Physical computing means using programming to control real-world	The program understands what I want it to do. If I write 'if button pressed', it will always work. Outputs happen without code. Selection means repeating.	What selection is in programming (using if, if...else, or when blocks). That inputs provide data to a system (e.g. button pressed, light level).	Selection - Writing conditional statements (e.g. if input A = true, then do B) Decomposition - Breaking down a physical system into code components (e.g. "first, wait for input")	"This block checks if…" "The program uses selection to…" "I used an input to trigger the output by…"

condition input output physical computing debugging loop	Why does a loop stop when a condition is met? How can a loop be used to repeatedly check whether a condition has been met? How can I design a physical project that includes selection? How can I create a program that controls a physical computing project? ARC	devices (e.g. motors, LEDs, sensors). Inputs (like sensors or buttons) and outputs (like lights or buzzers) allow a computer system to interact with the environment. Code must be specific and logical for the hardware to behave as expected. Debugging and iteration are key parts of creating working systems	The device is broken if it doesn't work	That outputs show a response (e.g. an LED lights up, a motor spins). That physical computing involves programming a device to react to the environment. The importance of using clear, logical sequences in code. How to test and debug simple programs connected to physical components. That computers cannot think – they only follow instructions exactly as given.	Logical reasoning - Predicting how changing a condition will change the system's behaviour. Debugging - Identifying and correcting problems in code or hardware. Evaluation - Judging how well a physical system performs and suggesting improvements.	"Next time, I would change…" "I improved the system by…" "We debugged it by checking…"
Tier 2: adjust organise create combine design layer object **Tier 3:** vector graphic bitmap object layering grouping alignment scaling rotation	How can drawing tools produce different outcomes? What is a vector drawing and how can I create one by combining shapes? How can I use tools to achieve a desired effect? What layers create vector drawings? How can grouping objects help in my vector drawing? How can I apply my learning to make a vector drawing? ARC	Vector graphics are made of shapes (objects) that can be resized without losing quality.Each object can be edited independently – you can change size, colour, position, and layering. Vector images are different from bitmap images, which are made from pixels. Designing with vector graphics involves layering, grouping, and aligning objects. Vector tools can be used to create effective, clear designs such as icons, posters, and diagrams	Vector graphics are just drawings. If I stretch a picture, it will always get blurry. All parts of the image move together. 'The image is just one flat shape.	Vector images are made using individual objects, which can be moved, resized, and edited without losing quality. Each object in a vector graphic is separate and can be layered and grouped. Vector images are used in graphic design, especially for logos, icons, and diagrams. Bitmap images are made from pixels and can lose quality when resized. Graphic design principles such as balance, alignment,	Decomposition - Breaking an image into shapes (e.g. eyes = 2 circles) Abstraction - Using basic shapes to represent more complex ideas (e.g. person = rectangle + circle) Evaluation - Judging the effectiveness of a design (e.g. "Does the layout make sense?") Iteration - Making small changes to improve a design (e.g. changing colours or alignment) Precision - Carefully positioning and adjusting objects for	"I used these shapes because…" "I changed the size of the object so that…" "This object is layered above the others to show…" "Next time, I would adjust…" "I found it difficult to…, but I improved by…" "This design works well because…"

				and composition help create visually effective designs.	clean, balanced visuals	

***Comprehensive working versions of these documents are available to school staff, outlining arrangements for fieldwork, educational trips, and associated resourcing.**

YEAR B						
Vocabulary	**Question**	**Key Concepts**	**Common Misconceptions**	**Substantive Knowledge**	**Disciplinary Knowledge**	**Oracy**
Tier 2: organise calculate change result compare total **Tier 3:** spreadsheet cell cell reference formula function range chart/graph data type	**<u>What are spreadsheets and how can they organise data and information?</u>** How do you create data in a spreadsheet? How do you build data in a spreadsheet? What is a formula and how can it produce calculated data? How do you apply formulas to data? How do you create a spreadsheet to plan an event? How do you present data in different ways? ARC	Spreadsheets store, organise, and analyse data using tables made of cells. Formulas and functions help automate calculations. Spreadsheets are used in real life for budgeting, planning, recording results, and making decisions. Accuracy in inputting data and formulas is essential. Charts and graphs help us understand and communicate data clearly.	Spreadsheets are just for maths. All cells are the same. You have to do all the maths in your head. If the formula looks right, it will work. Graphs just happen by themselves.	A spreadsheet is made up of cells arranged in rows and columns. Each cell has an address (e.g. A1, B2) used to identify it. Data types can be text, number, or formulas. Formulas begin with an equals sign (=) and can include operations like +, -, *, /. Functions like SUM(), AVERAGE(), and MAX() perform common calculations. Data can be visually represented through charts or graphs. Spreadsheets can be used to model real-life problems (e.g. budgets, scores, timetables).	Data handling - Input, organise, and format data in a spreadsheet. Problem solving - Use spreadsheets to model and solve real-life scenarios (e.g. planning an event). Logical thinking - Use cell references and formulas accurately. Analysis - Use functions and charts to draw conclusions from data. Debugging - Identify and fix formula errors. Abstraction - Recognise patterns and simplify calculations with functions	“This spreadsheet is used to…” “The formula in this cell does…” “I chose this chart because…” “The spreadsheet is clear because…” “One improvement could be…” “Next time, I would use a different formula because…”
Tier 2: change increase track rule result cause **Tier 3:** variable sprite event loop	**<u>What are variables and how can I use them in a game?</u>** What is a variable? Why are variables used in a program? How can variables improve a game? How can I design a project?	Programs use variables to store changing information. Variables can be updated as a game progresses (e.g. score increasing, lives decreasing).	A variable is just a number. Variables always stay the same. You don’t need variables to make a game. You can use any name for a variable. All variables work the same.	A variable is a named location in memory that stores information. Variables can be used to store numbers, text, or other data. Variables can increase or decrease based on game	Programming - Create and modify programs that use variables. Decomposition - Break down a game into parts: score system, timer, sprite movement. Sequencing - Ensure that code blocks execute in the right order.	“This variable is used to…” “When the player touches the object, the variable…” “I used a loop to…” “The variable didn’t update because…”

condition function debug input	How can I use my design to create a project? What ways can I evaluate my project? ARC	User input and events can affect variables in real time. Testing and debugging are essential for games to work as expected. Games are made up of sequences, loops, conditions, and variables.		actions (e.g. score = score + 1). Events (like key presses or collisions) can change a variable's value. Loops can be used to constantly check a variable's state (e.g. "While lives > 0"). If...then conditions can trigger actions when a variable reaches a certain value. Variables are used in real-world games and apps (e.g. timers, scores, health).	Logical reasoning - Predict and explain how variables will change Debugging - Identify and fix errors in variable use or logic. Abstraction - Use meaningful names for variables to represent concepts (e.g. "score" for performance).	"I fixed the error by changing..." "Next time, I'll check if..."
Tier 2: design shape object adjust view combine **Tier 3:** 3D model CAD (computer aided design) axis scale group rotate render viewport	**<u>What is 3D modelling?</u>** How do 3 dimensions look on a digital device? How can 3D objects be modified? How can objects be combined in a 3D model? What is the purpose of 3D models? What plan do I need to make a 3D model? How will I create my own digital 3D model? ARC	3D models are digital representations of real or imaginary objects. Computer-aided design (CAD) tools allow precise creation and manipulation of shapes. Objects in 3D space have width, height, and depth (x, y, z axes). 3D models are used in real-world applications such as architecture, engineering, games, and animation.	3D modelling is the same as drawing. You can only use cubes or basic shapes. The object is flat because it looks flat. All 3D models are just for printing. It's only about making things look good.	3D model is a digital representation of an object with three dimensions: height, width, and depth. Objects in 3D space are built using basic geometric shapes like cubes, cylinders, and spheres. Shapes can be moved, scaled, rotated, and grouped using software tools. Computer-aided design (CAD) is used in various industries to design products, buildings, and systems. 3D models can be viewed from different angles and rendered	Digital design - Create and manipulate 3D shapes in a modelling environment. Spatial reasoning - Understand and work with objects in 3D space (x, y, z coordinates). Iteration - Modify and refine designs through testing and feedback. Problem solving - Adjust components to ensure a model is balanced, functional, or realistic. Abstraction - Simplify real-world objects into geometric forms.	"This 3D model represents..." "I chose this shape because..." "I changed the angle so that..." "One thing that worked well was..." "Next time, I would improve..." "The final model shows that..."

		Digital design involves testing, adjusting, and refining		for printing or animation. Good 3D design involves precision, planning, and iteration.	Evaluation - Reflect on the effectiveness of the model for its intended use.	

***Comprehensive working versions of these documents are available to school staff, outlining arrangements for fieldwork, educational trips, and associated resourcing.**