COMPUTING SUBJECT INTENT

Our Curriculum Map

		Cycle A				Cycle B		
	EYFS	Year 1/2	Year 3/4	Year 5/6	EYFS	Year 1/2	Year 3/4	Year 5/6
Autumn 1	Can We Be Friends? / Come Rhyme With Me	Why do I love to be beside the seaside?	Sticks and stones	Hurry up, you've Benin there a while!	Can We Be Friends? / Come Rhyme With Me	The great animals	The Vikings on the wall	It's all Greek to me!
Autumn 2	Tell Me A Story	Darling, put the fire out!	We built this mega city on rock and roll!	Lavas all you need	Tell Me A Story	Happily Everest after	You crack me up!	Ain't no mountain high enough
Spring 1	Are We Nearly There Yet?	A toy story!	Bronze ain't bad!	What did the Romans do for us?	Pole To Pole	What's the nurse that can happen?	Where's my mummy?	The big Shang theory
Spring 2	It's A Bug's Life	Why don't penguins need to fly?	National parks	Walking on sunshine	Land Of The Giants	Wicked weather!	Don't rainforest on my parade	How is fair trade fair?
Summer 1	How Does Your Garden Grow?	It's pasture bedtime!	I've got my iron you	Mirror, Royal Signal, Manoeuvre	Commotion in the Ocean	Location, location, location	Great scot!	Only we can save the world!
Summer 2	The Best Show Of Your Life!	On the road again!	Save it for a train-y day	Current affairs	Here Comes The Sun!	Too hot to handle	Ch-ch-changes!	Who do you think you are, Mr Hitler?

What does Computing look like at Downlands?

Volume of content:
1. Each mixed-age class will complete **six** units per year.
2. Each unit will be at least **six** lessons.

Subject delivery:
- Lessons are sourced from the 'Teach Computing' curriculum available at https://teachcomputing.org/ (please note that you will need to create a NCCE login to access).
- The Teach Computing curriculum is structured into units for each year group, and each unit is broken down into lessons. Units can generally be taught in any order, with the exception of programming, where concepts and skills rely on prior knowledge and experiences. Lessons must be taught in numerical order.
- Worksheets are collected and stored at teacher's discretion.
- Milestone work should be added to Seesaw accounts once per unit.
- Any photographs as evidence of work completed should be added to Seesaw.

Computing – progression of skills (disciplinary knowledge)

SKILL	FS2	End of KS1	End of LKS2	End of UKS2
Computing systems and networks **Digital literacy**	Know they can say 'no' to things that make them sad, upset or embarrassed. I can recognise some ways in which the internet can be used to communicate and give examples of how I (might) use it. I can describe ways that some people can be unkind online and give examples of how this can make others feel. I can identify devices I could use to access information on the internet. Can describe the people I can trust and can share this with; I can explain why I can trust them.	Recognise common uses of information technology beyond school . Use technology safely and respectfully, keeping personal information private; identify where to go for help and support when they have concerns about content or contact on the internet or other online technologies.	Identify that digital devices have inputs, processes, and outputs, and how devices can be connected to make networks. Recognise the internet as a network of networks including the WWW, and why we should evaluate online content.	Recognise IT systems in the world and how some can enable searching on the internet. Explore how data is transferred by working collaboratively online.
Creating media **Information technology**	Know and recognise computers at home and at school. Be able to recognise and talk about familiar technology (e.g. apps, websites and online services). Learn how to interact with computers and technology as appropriate.	Choose appropriate tools in a program to create art, and make comparisons with working non-digitally. Capture and change digital photographs for different purposes. Use a computer to create and format text, before comparing to writing non-digitally Use a computer as a tool to explore rhythms and melodies, before creating a musical composition.	Capture and edit digital still images to produce a stop-frame animation that tells a story. Capture and edit audio to produce a podcast, ensuring that copyright is considered. Create documents by modifying text, images, and page layouts for a specified purpose. Manipulate digital images, and reflect on the impact of changes and whether the required purpose is fulfilled.	Plan, capture, and edit video to produce a short film. Design and create webpages, giving consideration to copyright, aesthetics, and navigation. Create images in a drawing program by using layers and groups of objects. Plan, develop, and evaluate 3D computer models of physical objects.
Data and information **Information technology**		Recognise common uses of information technology beyond school . Use technology safely and respectfully, keeping personal information private; identify where to go for help and support when they have concerns about content or contact on the internet or other online technologies. Collect data in tally charts and use attributes to organise and present data on a computer.	Build and use branching databases to group objects using yes/no questions. Recognise how and why data is collected over time, before using data loggers to carry out an investigation.	Use a database to order data and create charts to answer questions. Answer questions by using spreadsheets to organise and calculate data.
Programming **Computer science**	Know how to follow simple instructions. Put pictorial instructions into sequential order. Use Beebots and talk about their discoveries.	Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions. Create and debug simple programs. Use logical reasoning to predict the behaviour of simple programs. Design and program the movement of a character on screen to tell stories. Design algorithms and programs that use events to trigger sequences of code to make an interactive quiz.	Create sequences in a block-based programming language to make music. Use a text-based programming language to explore count-controlled loops when drawing shapes. Write algorithms and programs that use a range of events to trigger sequences of actions. Use a block-based programming language to explore count-controlled and infinite loops when creating a game.	Explore conditions and selection using a programmable microcontroller. Explore variables when designing and coding a game. Explore selection in programming to design and code an interactive quiz. Design and code a project that captures inputs from a physical device.

Computing – EYFS Learning Steps

Early Learning Goal	Vocabulary
Computer Science 1. Listening, Attention and Understanding Listen attentively and respond to what they hear with relevant questions, comments and actions when being read to and during whole class discussions and small group interactions 2. Speaking Participate in small group, class and one-to-one discussions, offering their own ideas, using recently introduced vocabulary; Offer explanations for why things might happen, making use of recently introduced vocabulary from stories, non-fiction, rhymes and poems when appropriate 7. Fine Motor Skills Use a range of small tools, including scissors, paint brushes and cutlery **Information Technology** 7. Fine Motor Skills Use a range of small tools, including scissors, paint brushes and cutlery. Begin to show accuracy and care when drawing. 10. Writing Spell words by identifying sounds in them and representing the sounds with a letter or letters; Write simple phrases and sentences that can be read by others. 15. The Natural World Explore the natural world around them, making observations and drawing pictures of animals and plants. 16. Creating with Materials Safely use and explore a variety of materials, tools and techniques, experimenting with colour, design, texture, form and function. **Digital Literacy** 3. Self-Regulation Show an understanding of their own feelings and those of others, and begin to regulate their behaviour accordingly. 4. Managing Self Explain the reasons for rules, know right from wrong and try to behave accordingly.	direction, control, instructions, steps, robot, microphone, keyboards, keys, letter, number, camera, save, print, video, film, record Technology in the Early Years can mean: • taking a photograph with a camera or tablet • searching for information on the internet • playing games on the interactive whiteboard • exploring an old typewriter or other mechanical toys • using a Beebot • watching a video clip • listening to music.

Computing – curriculum map – KS1 – cycle A

COMPUTER SCIENCE	INFORMATION TECHNOLOGY	DIGITAL LITERACY	INFORMATION TECHNOLOGY

	Autumn 1	Autumn 2	Spring 1
Unit	Why do we like to be beside the seaside?	Darling, put the fire out!	A toy story!
Computing Unit (& yr group)	**Networks and Systems** Parts of a computer from **Y1, lessons 2, 6** Info tech from **Y2, lessons 1, 2, 3, 4, 6** See notes for differentiation suggestions Link to note	**Creating Media first unit** Digital painting **Y1 unit**	**Programming first unit** **Y1 Prog A, Lessons 1, 4** **Y2 Programming A, Lesson 1, 2, 4, 5** (also add in debugging) Link to notes
Skills (disciplinary knowledge)	Identify technology. Identify the toolbar. and use bold and change font and size . Type capital letters. Use the space bar. Find letters on a keyboard to type words. Insert a picture from a picture box. Follow rules for using technology responsibly.	Draw lines and make marks on a screen and explain which tools I used. Make marks with the square and line tools. Use the shape and line tools effectively. Use the shape and line tools to recreate the work of an artist. Explain why I have chosen specific tools.	Use a start block in a program. Use more than one block by joining them together. Compare left and right turns. Experiment with turn and move. commands to move a physical computer. Use event, action and object code blocks. Select appropriate background artwork for a project.
Knowledge (substantive knowledge)	In this unit, pupils will develop their understanding of technology and how it can help them in their everyday lives. They will start to become familiar with the different components of a computer by developing their keyboard and mouse skills. Learners will also consider how to use technology responsibly. Pupils will also develop their understanding of what information technology (IT) is and will begin to identify examples. They will discuss where they have seen IT in school and beyond, in settings such as shops, hospitals, and libraries. Learners will then investigate how IT improves our world, and they will learn about the importance of using IT responsibly.	In this unit, pupils will develop their understanding of a range of tools used for digital painting. They then use these tools to create their own digital paintings, while gaining inspiration from a range of artists' work. The unit concludes with learners considering their preferences when painting with and without the use of digital devices.	In this unit, pupils will be introduced to early programming concepts. Learners will explore using individual commands, both with other learners and as part of a computer program. This unit also develops learners' understanding of instructions in sequences and the use of logical reasoning to predict outcomes. Learners will use given commands in different orders to investigate how the order affects the outcome. They will also learn about design in programming. They will develop artwork and test it for use in a program. They will design algorithms and then test those algorithms as programs and debug them.
Learning Objectives	I can name the main parts of a computer. I can identify rules to keep us safe and healthy when using technology. I can identify some examples and uses of computers. I can identify examples of IT and sort what it is used for, both in school and the wider world. I can recognize common types of technology. I can use IT for different types of activities and explain why.	I can use paint tools to make marks, draw a line or draw a picture. I can use the shape tool and the line tool. I can make careful choices when painting a digital picture. I can explain why I chose the tools I used. I can use a computer on my own to paint a picture. I can compare painting a picture on a computer and on paper.	I can explain what a given command will do. I can combine four direction commands to make sequences. I can describe a series of instructions as a sequence. I can explain what happens when we change the order of instructions. I can explain that programming projects can have code and artwork. I can design an algorithm.
Vocabulary	technology, computer, mouse, trackpad, keyboard, screen, click, drag, input device, shift, spacebar, capital letter, full stop, safely, responsibly, Information technology (IT), computer, barcode, scanner/scan	paint program, tool, paintbrush, erase, fill, undo, Piet Mondrian, primary colours, shape tools, line tool, fill tool, undo tool, Henri Matisse, Wassily Kandinsky, feelings, colour, brush style, George Seurat, Pointillism, prefer, dislike, like	forwards, backwards, turn, clear, go, commands, instructions, directions, left, right, plan, algorithm, route, program, Instruction, sequence, clear, unambiguous, algorithm, program, order, commands, prediction, artwork, design, route, mat, debugging

Computing – curriculum map – KS1 – cycle A

COMPUTER SCIENCE	INFORMATION TECHNOLOGY	DIGITAL LITERACY	INFORMATION TECHNOLOGY

	Spring 2	**Summer 1**	**Summer 2**
Unit	Why don't penguins need to fly?	It's pasture bedtime	On the road again!
Computing Unit (& yr group)	**Data** Pictograms **Y2 unit** - context changes Cycle A and B - introduction to or use of depending on prior learning.	**Creating Media second unit** Digital writing **Y1 unit.**	**Programming second unit** Programming animations **Y1 programming B** (repeated annually but context changed). <u>Link to notes</u>
Skills (disciplinary knowledge)	Count and compare objects (data) using tally charts, comparing totals. Enter data on a computer and view that data in a different format: Use a pictogram to answer simple questions about the data. Use a tally chart to create a pictogram. Answer 'more than'/'less than' and 'most/least' questions about an attribute. Create a pictogram to arrange objects by attributes. Create a pictogram to compare people by a common attribute. Explain that we can present information using a computer and that sometimes it is this data should not be shared.	Find and identify keys on a key pad. Use a computer to write. Add and remove text on a computer using the backspace key. Change the look of the text by using bold, italic and underlining. Make careful choices when changing text, for example, changing the font, selecting a word by double clicking or clicking and dragging. Explain why I used the tools that I chose. Compare writing on a computer with writing on paper.	Compare different programming tools and find and use commands to move a sprite. Use a start block in a program and Join blocks together. Explain what happens when I change a value. Add blocks to my sprite and delete a sprite. Create an algorithm for each sprite to control movement. Test the programs I have created and alter my designs.
Knowledge (substantive knowledge)	In this unit, pupils will begin to understand what the term data means and how data can be collected in the form of a tally chart. They will learn the term 'attribute' and use this to help them organise data. They will then progress onto presenting data in the form of pictograms and finally block diagrams. Learners will use the data presented to answer questions.	In this unit, pupils will develop their understanding of the various aspects of using a computer to create and manipulate text. They will become more familiar with using a keyboard and mouse to enter and remove text. Learners will also consider how to change the look of their text, and will be able to justify their reasoning in making these changes. Finally, learners will consider the differences between using a computer to create text, and writing text on paper. They will be able to explain which method they prefer and explain their reasoning for choosing this.	In this unit, pupils will be introduced to on-screen programming through ScratchJr. Learners will explore the way a project looks by investigating sprites and backgrounds. They will use programming blocks to use, modify, and create programs. Learners will also be introduced to the early stages of program design through the introduction of algorithms.
Learning Objectives	I can recognize that we can count and compare objects using tally charts. I can recognize the objects can be represented as pictures. I can create a pictogram. I can select objects by attributes and make comparisons. I can recognize that people can be described by attributes. I can explain that we can present information using a computer.	I can use a computer to write. I can add and remove text on a computer. I can identify that the look of text can be changed on a computer. I can make careful choices when changing text. I can explain why I used the tools that I chose. I can compare typing on a computer to writing on paper.	I can choose a command for a given purpose. I can show that a series of commands can be joined together. I can identify the effect of changing a value. I can explain that each sprite has its own instructions. I can design the parts of a project. I can use my algorithm to create a program.
Vocabulary	more than, less than, most, least, organise, data, object, tally chart, votes, total, pictogram, enter, data, tally chart, compare, count, explain, attribute, group, same, different, most popular, least popular	word processor, keyboard, keys, letters, Microsoft Word, letters, numbers, space, backspace, text cursor, toolbar, bold, italic, underline, undo, font, toolbar	ScratchJr, Bee-Bot, command, sprite, compare, programming, programming area, block, joining, start, program, background, delete, reset, algorithm, predict, effect, change, value, block, instructions, appropriate, design

Downlands Community School
We CARE, We SHARE, We BELIEVE, We ACHIEVE

Computing – curriculum map – LKS2 – cycle A

COMPUTER SCIENCE	INFORMATION TECHNOLOGY	DIGITAL LITERACY	INFORMATION TECHNOLOGY

	Autumn 1	Autumn 2	Spring 1
Unit	Sticks and stones	We built this mega city on rock and roll!	Bronze aint' bad!
Computing Unit (& yr group)	**Networks and Systems** Inputs and outputs and how networks are connected from **Y3, lessons 1, 2, 4, 5, 6** + The Internet from **Y4, lessons 1, 2** Link to notes	**Creating Media first unit** Desktop publishing **Y3 unit.**	**Programming first unit** Introduction to Scratch **Combine lesson 1,2,3 from Y3 programming A with lesson 1 from Y3 programming B**. Create 2 lessons from these. Repeat annually. Then: Sequence, creating a piano **Y3 Progamming A, Lessons 4 - 6** Link to notes
Skills (disciplinary knowledge)	Classify input and output devices; design a digital device and model a simple process. Recognise similarities and differences between using digital devices and non-digital tools. Explain how a computer network can be used to share information and that messages pass through multiple connections. Explain how digital devices can be connected and what the role of a switch, server and wireless access point is. Recognise the physical components of a network and how they are connected.	Recognise how text and images convey information clearly and that there are some advantages and disadvantages to using them. Change the text layout, including font style, size and colour. Choose appropriate page settings: generating a template to meet my needs with placeholders. Add content to a desktop publishing publication, including adding text and pasting pictures. Change the layout to suit different purposes. Consider the benefits of desktop publishing and identify its use in the real world.	Explain how a sprite moves in an existing project. Create a program to move a sprite in four directions Adapt a program to a new context. Develop my program by adding features. Identify and fix bugs in a program. Design and create a maze-based challenge.
Knowledge (substantive knowledge)	In this unit, pupils will develop their understanding of digital devices, with an initial focus on inputs, processes, and outputs. They will also compare digital and non-digital devices. Next, learners will be introduced to computer networks, including devices that make up a network's infrastructure, such as wireless access points and switches. Finally, learners will discover the benefits of connecting devices in a network. Pupils will also apply their knowledge and understanding of networks, to appreciate the internet as a network of networks which need to be kept secure.	In this unit, pupils will become familiar with the terms 'text' and 'images' and understand that they can be used to communicate messages. They will use desktop publishing software and consider careful choices of font size, colour and type to edit and improve premade documents. Learners will be introduced to the terms 'templates', 'orientation', and 'placeholders' and begin to understand how these can support them in making their own template for a magazine front cover. They will start to add text and images to create their own pieces of work using desktop publishing software. Learners will look at a range of page layouts thinking carefully about the purpose of these and evaluate how and why desktop publishing is used in the real world.	This unit explores the concept of sequencing in programming through Scratch. It begins with an introduction to the programming environment, which will be new to most learners. They will be introduced to a selection of motion, sound, and event blocks which they will use to create their own programs, featuring sequences. The final project is to make a representation of a piano. The unit is paced to focus on all aspects of sequences, and make sure that knowledge is built in a structured manner. Learners also apply stages of program design through this unit.
Learning Objectives	I can explain how digital devices function. I can identify input and output devices. I can explain how a computer network can be used to share information. I can explore how digital devices can be connected. I can recognise the physical components of a network. I can describe how networks physically connect to other networks. I can recognize how networked devices make up the internet.	I can recognize how text and images convey information. I can recognise that text and layout can be edited. I can choose appropriate page settings. I can add content to a desktop publishing publication. I can consider how different layouts can suit different purposes. I can consider the benefits of desktop publishing.	I can explore a new programming environment. I can identify that commands have an outcome. I can explain that a program has a start. I can explain how a sprite moves in an existing project. I can recognise that a sequence of commands can have an order. I can change the appearance of my project. I can create a project from a task description.
Vocabulary	digital device, input, output, process, program, connection, network, network switch, server, wireless access point (WAP)	text, images, advantages, disadvantages, communicate, font, style, template, desktop publishing, copy, paste, layout, purpose, benefits	Scratch, programming, blocks, commands, code, sprite, costume, stage, backdrop, motion, turn, point in direction, go to, glide, event, task, design, code, run the code, order, note, chord, algorithm, bug, debug

Computing – curriculum map – LKS2 – cycle A

COMPUTER SCIENCE	INFORMATION TECHNOLOGY	DIGITAL LITERACY	INFORMATION TECHNOLOGY

	Spring 2	Summer 1	Summer 2
Unit	National parks	I've got my iron you	Save it for a train-y day
Computing Unit (& yr group)	Data Branching databases **Y3 unit.**	Creating Media second unit Audio production **Y4 unit.**	Programming second unit Repetition using Logo **Y4 Programming A.** Link to notes
Skills (disciplinary knowledge)	Create a branching database by grouping groups of objects separated by one attribute. Make up yes/no questions about these groups. Identify the object attributes needed to collect relevant data. Explain why it is helpful for a database to be well structured. Compare the information shown in a pictogram with a branching database	Identify digital devices that can record sound and play it back and that a range of sounds can be recorded. Plan and record a podcast, saving it as a file. Discuss how to improve my podcast and edit sections of an audio recording. Reopen my recording and add sound, using editing tools to rearrange sections of audio.	Create a code snippet for a given purpose, for example controlling a turtle. Write an algorithm for a given outcome, including repetition. Design a program that has a count-controlled loop. Debug my program.
Knowledge (substantive knowledge)	In this unit, pupils will develop their understanding of what a branching database is and how to create one. They will use yes/no questions to gain an understanding of what attributes are and how to use them to sort groups of objects. Learners will create physical and on-screen branching databases. To conclude the unit, they will create an identification tool using a branching database, which they will test by using it. They will also consider real-world applications for branching databases.	In this unit, pupils will identify the input device (microphone) and output devices (speaker or headphones) required to work with sound digitally. Learners will discuss the ownership of digital audio and the copyright implications of duplicating the work of others. In order to record audio themselves, learners will use Audacity to produce a podcast, which will include editing their work, adding multiple tracks, and opening and saving the audio files. Finally, learners will evaluate their work and give feedback to their peers.	In this unit, pupils will create programs by planning, modifying, and testing commands to create shapes and patterns. They will use Logo, a text-based programming language. This unit is the first of the two programming units in Year 4, and looks at repetition and loops within programming
Learning Objectives	I can create questions with yes/no answers. I can identify the attributes needed to collect data about an object. I can create a branching database. I can explain why it is important for a database to be well structured. I can plan the structure of a branching database. I can independently develop an identification tool.	I can identify that sound can be recorded. I can explain that audio recordings can be edited. I can recognise the different parts of creating a podcast project. I can apply audio editing skills independently. I can combine audio to enhance my podcast project. I can evaluate the effective use of audio.	I can identify that accuracy in programming is important. I can create a program in a text-based language. I can explain what 'repeat' means. I can modify a count-controlled loop to produce a given outcome. I can decompose a task into small steps. I can create a program that uses count-controlled loops to produce a given outcome.
Vocabulary	attribute, value, questions, table, objects, branching databases, objects, equal, even, separate, order, organise, j2data, selecting, pictogram, information, decision tree, questions	audio, record, playback, microphone, speaker, headphones, input, output, start, stop, podcast, save, file, selection, edit, mixing, time shift, export, MP3, evaluate, feedback	program, turtle, commands, code, snippet, algorithm, design, debug, logo commands, pattern, repeat, repetition, count-controlled loop, value, decompose, procedure

Computing – curriculum map – UKS2 – cycle A

COMPUTER SCIENCE	INFORMATION TECHNOLOGY	DIGITAL LITERACY	INFORMATION TECHNOLOGY

	Autumn 1	Autumn 2	Spring 1
Unit	Hurry up, you've Benin there a while!	Lavas all you need	What did the Romans do for us?
Computing Unit (& yr group)	Networks and Systems Computer systems and working collaboratively online, **Y5 lessons (all)** Link to notes	Creating Media first unit Vector drawing **Y5 unit.**	Programming first unit Selection in physical computing (Crumble controllers) Y5 **Programming A** Link to notes
Skills (disciplinary knowledge) The skills I have learned..	Explain how computers are connected together to form systems. Explain the role that computers have in our lives and how information is transferred over the internet. Work collectively on a shared project online. Evaluate different ways of working together online.	Use drawing tools to produce different outcomes and for different purposes. Create a vector drawing by combining shapes and I can move, resize, rotate and duplicate them. Use tools to achieve a desired effect, for example using the zoom tool to add detail to my drawing. Create layers bring objects to the front or the back. Evaluate my vector drawing and say how I might improve it.	Explain how selection is used in a program and identify conditions and how to modify them. Create a program with different outcomes using selection and identify the condition and outcome is an if... then... else statement. Explain how selection directs the flow of a program. Design and create a program which uses selection: creating the algorithms, running the program and debugging.
Knowledge (substantive knowledge) The knowledge I have..	In this unit, pupils develop their understanding of computer systems and how information is transferred between systems and devices. Learners consider small-scale systems as well as large-scale systems. They explain the input, output, and process aspects of a variety of different real-world systems. Learners discover how information is found on the World Wide Web, through learning how search engines work (including how they select and rank results) and what influences searching, and through comparing different search engines.	In this unit, pupils start to create vector drawings. They learn how to use different drawing tools to help them create images. Learners recognise that images in vector drawings are created using shapes and lines, and each individual element in the drawing is called an object. Learners layer their objects and begin grouping and duplicating them to support the creation of more complex pieces of work.	In this unit, pupils will use physical computing to explore the concept of selection in programming through the use of the Crumble programming environment. Learners will be introduced to a microcontroller (Crumble controller) and learn how to connect and program it to control components (including output devices — LEDs and motors). Learners will be introduced to conditions as a means of controlling the flow of actions in a program. Learners will make use of their knowledge of repetition and conditions when introduced to the concept of selection (through the 'if...then...' structure) and write algorithms and programs that utilise this concept. To conclude the unit, learners will design and make a working model of a fairground carousel that will demonstrate their understanding of how the microcontroller and its components are connected, and how selection can be used to control the operation of the model. Throughout this unit, learners will apply the stages of programming design.
Learning Objectives	I can explain that computers can be connected together to form systems. I can recognise the role of computer systems in our lives I can identify how to use a search engine I can describe how search engines select results. I can explain how search results are ranked. I can recognise why the order of results is important, and to whom.	I can identify that drawing tools can be used to produce different outcomes. I can create a vector drawing by combining shapes. I can use tools to achieve a desired effect e.g. zoom. I can recognise that vector drawings consist of layers. I can group objects to make them easier to work with. I can apply what I have learned about vector drawings.	I can control a simple circuit connected to a computer. I can write a program that includes count-controlled loops. I can explain that a loop can stop when a condition is met. I can explain that a loop can be used to repeatedly check whether a condition has been met. I can design a physical project that includes selection. I can create a program that controls a physical computing project.
Vocabulary	Searching, search engine, web crawler, content creator, selection, ranking System, connection, digital, input, process, output, protocol, address, packet, chat, explore, slide deck, reuse, remix, collaboration	Vector, drawing tools, shapes, object, icons, toolbar, move, resize, colour, rotate, duplicate/copy, zoom, select, alignment grid, handles, consistency, modify, layers, front, back, copy, paste, group, ungroup, reuse, improvement, evaluate, alternatives	Microcontroller, controller, components, LED, crocodile clips, connect, battery, program, repetition, infinite loop, count-controlled loop, condition, true, false, input, action, selection, motor, switch, algorithm, debug, evaluate

Computing – curriculum map – UKS2 – cycle A

	COMPUTER SCIENCE	INFORMATION TECHNOLOGY	DIGITAL LITERACY	INFORMATION TECHNOLOGY

	Spring 2	Summer 1	Summer 2
Unit	Walking on sunshine	Mirror, Royal Signal, Manoeuvre	Current affairs
Computing Unit (& yr group)	Data Flat file databases **Y5 unit**	Creating Media second unit 3D Modelling **Y6 unit**	Programming second unit Variables in games **Y6 Programming A** (using different examples from Cycle B) Link to notes
Skills (disciplinary knowledge) The skills I have learned..	Create a database, using fields which hold and record the data. Search a database using 'and' and 'or.' Apply filters and select an appropriate chart or graph to visually compare data. Apply knowledge of a database to ask questions that will need more than one field to answer	Use a computer to create and manipulate three-dimensional (3D) digital objects. Compare working digitally with 2D and 3D graphics. Construct a digital 3D model of a physical object. Identify that physical objects can be broken down into a collection of 3D shapes. Design a digital model by combining 3D objects. Develop and improve a digital 3D model.	Define a 'variable' as something that is changeable, variables can hold numbers or letters. Explain why a variable is used in a program; it is a place holder in memory for a single value. Choose how to improve a game by using variables. Design a project that builds on a given example: choosing artwork and creating the algorithm. Use my design to create a project, testing the code that I have written. Evaluate my project
Knowledge (substantive knowledge) The knowledge I have..	This unit looks at how a flat-file database can be used to organise data in records. Learners will use tools within a database to order and answer questions about data. They will create graphs and charts from their data to help solve problems. They will also use a real-life database to answer a question, and present their work to others.	In this unit, pupils will develop their knowledge and understanding of using a computer to produce 3D models. Learners will initially familiarise themselves with working in a 3D space, moving, resizing, and duplicating objects. They will then create hollow objects using placeholders and combine multiple objects to create a model of a desk tidy. Finally, learners will examine the benefits of grouping and ungrouping 3D objects, then go on to plan, develop, and evaluate their own 3D model of a building.	This unit explores the concept of variables in programming through games in Scratch. First, learners find out what variables are and relate them to real-world examples of values that can be set and changed. Then they use variables to create a simulation of a scoreboard. In Lessons 2, 3, and 5, which follow the Use-Modify-Create model, learners experiment with variables in an existing project, then modify them, before they create their own project. In Lesson 4, learners focus on design. Finally, in Lesson 6, learners apply their knowledge of variables and design to improve their games in Scratch.
Learning Objectives	I can use a form to record information. I can compare paper and computer-based databases. I can outline how you can answer questions by grouping and then sorting data. I can explain that tools can be used to select specific data. I can explain that computer programs can be used to compare data visually. I can use a real-world database to answer questions.	I can recognise that you can work in three dimensions on a computer. I can identify that digital 3D objects can be modified. I can To recognise that objects can be combined in a 3D model. I can create a 3D model for a given purpose. I can plan my own 3D model I can create my own 3D model.	I can define a 'variable' as something that is changeable. I can explain why a variable is used in a program. I can choose how to improve a game by using variables. I can design a project that builds on a given example. I can use my design to create a project. I can evaluate my project.
Vocabulary	database, data, information, record, field, sort, order, group, search, criteria, value, graph, chart, axis, compare, filter, presentation	2D, 3D, 3D object, 3D space, view, resize, colour, lift, rotate, position, select, duplicate, dimensions, placeholder, hole, group, ungroup, modify, evaluate, improve	variable, change, name, value, set, design, algorithm, code, task, artwork, program, project, code, test, debug, improve, evaluate, share