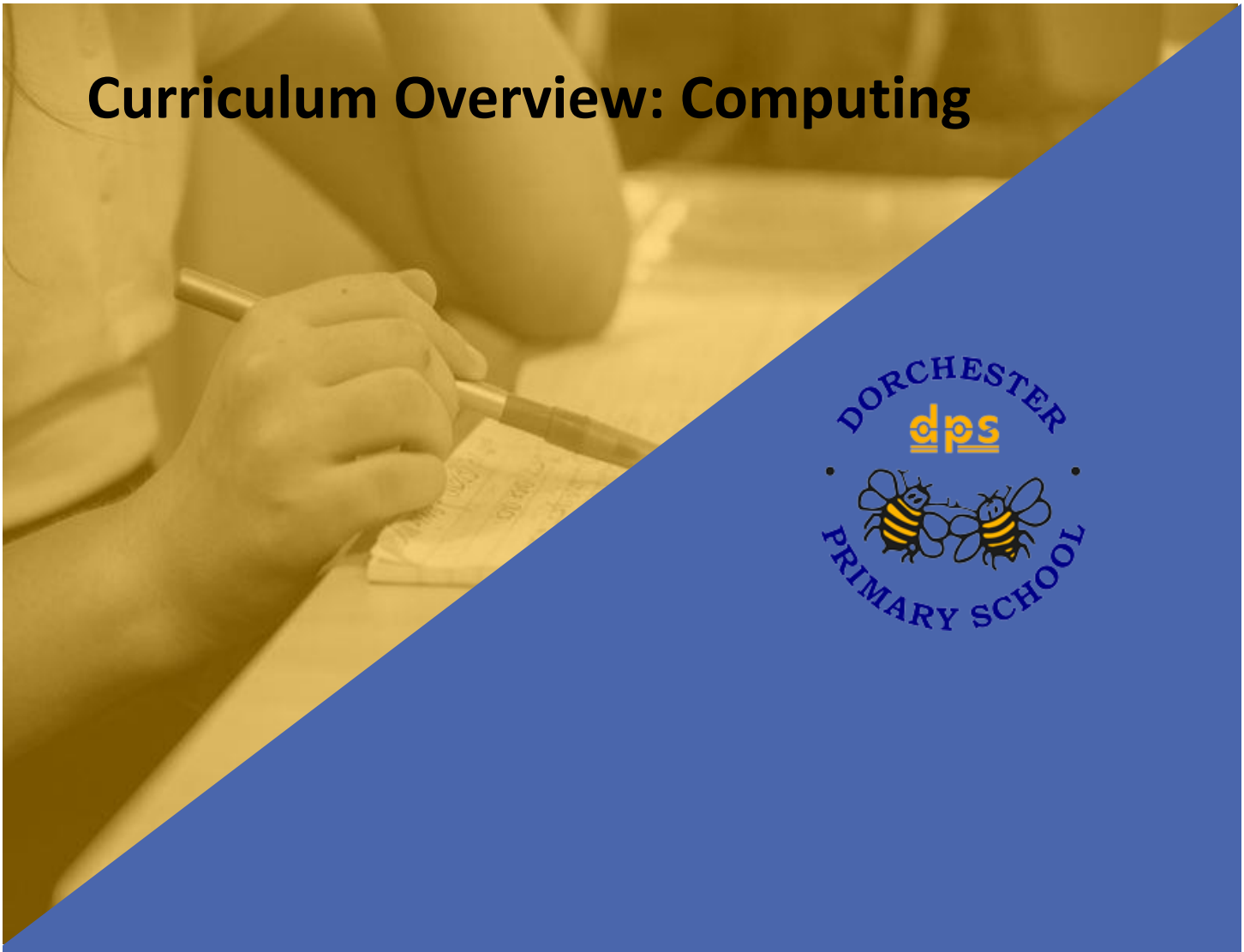
Curriculum Overview: Computing



Dorchester Primary School

A Hull Collaborative Academy Trust school.

Working together in a safe and inclusive environment to develop creative, disciplined and aspirational pupils.

The Dorchester Curriculum

The Vision

Our curriculum vision at Dorchester is to provide children with an ambitious, academic and personal curriculum which will open up future opportunities for success.

Curriculum Aims

Our curriculum aims to provide:

- A cumulative acquisition of concepts, knowledge and skills which enables all children to achieve their goals
- A creative, enquiry based curriculum which promotes inquisitive minds, independence and a love for learning.
- A sense of belonging, identity and community to succeed in the modern world

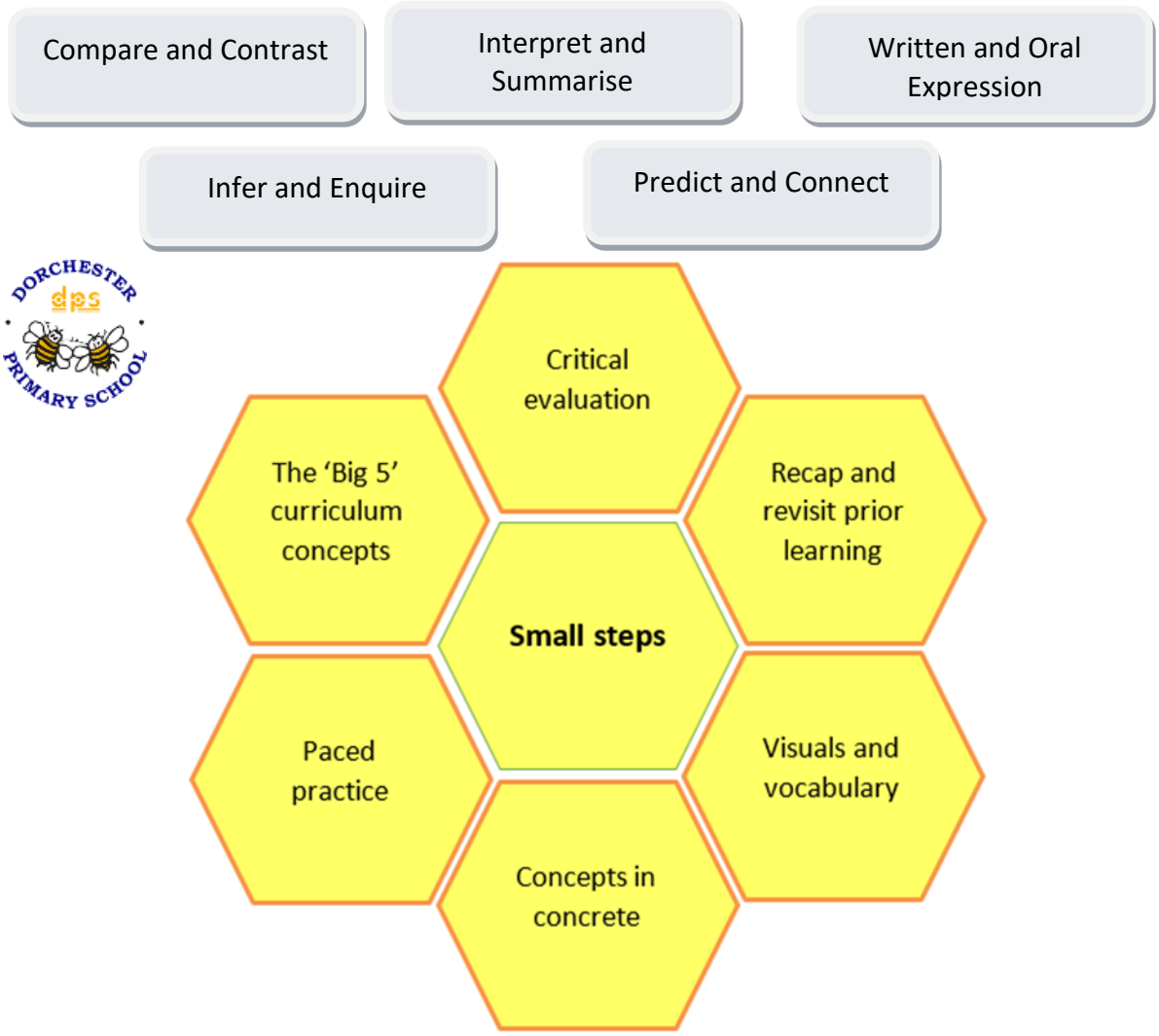
Values

These aims are underpinned by our school values:

- Our school values respect, so our curriculum provides plentiful opportunities for children to work collaboratively, practice active listening, turn taking and acts of service.
- Our school values tolerance therefore our curriculum promotes social awareness and represents diverse voices
- Our school values self-belief so our curriculum promotes a growth mindset and develops independence
- Our school values empathy therefore our curriculum provides opportunities for children to view the world from different perspectives
- Our school values resilience so our curriculum promotes goal setting and problem solving

Curriculum Concepts

Children will also develop their understanding of identified curriculum concepts throughout all subjects. These concepts branch across our whole curriculum, creating horizontal links across all subjects. They aim to develop flexible knowledge and skills that children can apply to multiple curriculum areas.



Dorchester Learning Framework

We have created a learning framework to support our teachers in planning, delivering and assessing the curriculum. This framework has been devised using research into knowledge acquisition and working memory to enable all children to learn and remember what we teach.
Progress at Dorchester means 'knowing more and remembering more'.

Assessment

We have adopted a three-tier assessment model for wider curriculum subjects. Teachers will gather assessment information on what children have learnt (and retained): in the short term (e.g. within / after a lesson), the medium-term (e.g. after a unit of work), and long-term (e.g. at the end of their phase or key stage). Assessments will be used to inform the learning moving forward.

Curriculum Timetabling

At Dorchester, we want to ensure that we celebrate the talents of all pupils and provide everyone with opportunities to shine. Therefore, we have calculated the number of teaching hours available and have ensured that all pupils receive a broad and balanced curriculum based on the starting points of our children

Maths, Reading (including EARS for fluency and book talk KS1) and Writing (including spelling and handwriting are taught daily.

Science, RE, PE, Music, PSHE and MFL (KS2) are taught weekly except where blocking of other subjects is needed. Computing is built within the curriculum subject areas as well as some standalone knowledge

History and Geography along with Art and DT are taught in alternating blocks per half term.

Day	Am (Hours)	Pm (Hours)
Mon	2hrs 45min	2hr 25mins
Tue	2hrs 45min	2hr 25mins
Wed	2hrs 45min	2hr 25mins
Thur	2hrs 45min	2hr 25mins
Fri	2hrs 45min	2hr 25mins

Curriculum Area	**Hours per day**	**Weekly (B = Blocked)**	**Hours Per Year**
English			
Reading	1	5 hrs	185 hours
Reading-Fluency	15 mins	1hr 15 mins	46 hours
			231 hour
Writing	45 mins	4 hr 15 mins	157 hours
- *Handwriting*	10 mins	50 mins	30
- *Spelling*	10mins	50mins	30
			217 hours
Maths			
Maths	1	5	**185 hours**
Computer Sciences			
Science	-	1hr	**35 hours**
Computing	-	45 mins	**26 hours**
Creative			
Art	-	1hr 30mins (B)	**24 hours**
Design and Technology	-	1hr 30 mins (B)	**24 hours**
Music	-	20 mins	**11 hours**
Humanities			
History	-	1hr 30 mins (B)	**24 hours**
Geography	-	1hr 30 mins (B)	**24 hours**
RE	-	45 minutes	**24 hours**
Additional			
Physical Education	-	1 hour	**35 hours**
MFL	-	20 mins	**11 hours**
PSHE	-	20 mins	**11 hours**

The Computing Vision

At Dorchester, we recognise the integral part computers and technology play in everyday life and aim to offer children a broad and progressive curriculum to develop and challenge their computational thinking skills while teaching them how to use technology safely. Using the national curriculum to drive our core planning, we aim to give our children the confidence to enhance their digital skills, allowing them to become creators of digital content rather than simply consumers of it. We aim to offer the children a number of different ways in which they can apply their computational thinking skills from EYFS to KS2 allowing them to improve their skills not only in computing but also allowing them to supplement and enhance their learning in other areas of the curriculum through the use of technology.

Computing Curriculum

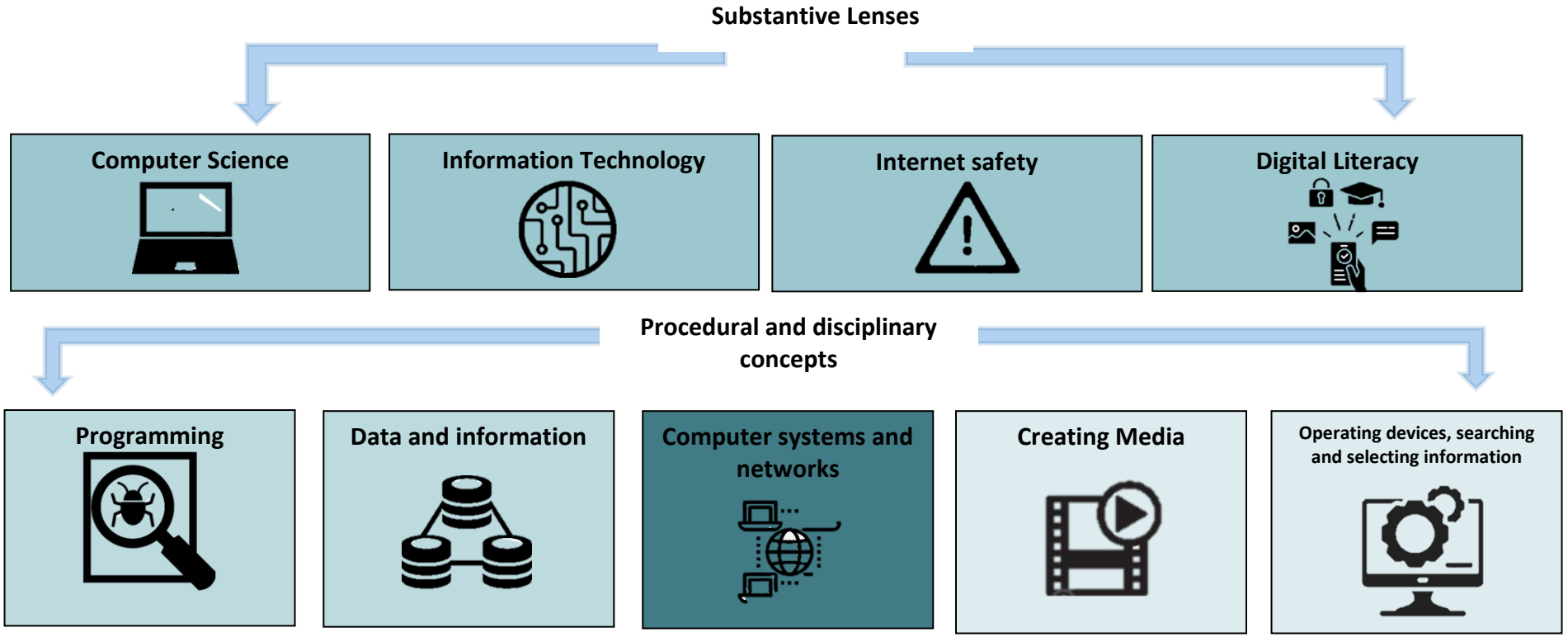
Our computing curriculum aims to:

- Support all pupils develop the appropriate skills and knowledge to use a range of technology in a safely and effectively;
- Provide a comprehensive coverage of the subject which can be described through high-level taxonomy of ten computing strands. The taxonomy provides categories and an organised view of content to encapsulate the discipline of computing:
    - Algorithms
    - Computer networks
    - Computer systems
    - Creating media
    - Data and information
    - Design and development
    - Effective use of tools
    - Impact of technology
    - Programming
    - Safety and security
- Expose pupils to a range of software and hardware with which they can practise and apply skills;

- Provide opportunities for pupils to revisit learning regularly, reducing lost learning and ensuring that connections are made to learning in previous year groups via our 'spiral curriculum' design;
- Use the twelve pedagogical approaches proven to contribute to effective teaching and learning in computing (see appendix 1)
- Engage pupils creatively through 'physical computing' (combining software and hardware to build physical systems that sense and respond to the real word);
- Give pupils the skills and knowledge to use online technology safely and respectfully.

Computing key concepts

Knowledge overview



Curriculum Concepts in Computing

Programming	Data and information	Computer Systems and networks	Creating Media	Digital literacy
Pupils will learn how to interpret, create and evaluate algorithm. They will be taught to program to accomplish specific goals and to detect and correct errors.	Pupils will learn how to collect, analyse, evaluate and present data and information.	Pupils will learn about computer systems, networks and how they are used. They will also learn about the internet and different types of hardware and software.	Pupils will learn about the design and development of digital media in different forms. They will learn how to collaborate online, evaluate online content and how to communicate, create and present content in a respectful way.	This is woven through the other key concepts, ensuring pupils know how to operate devices, how to search and select information, and how to use digital devices safely and responsibly.

Computing Progression Map

	End of EYFS	End of KS1	End of Lower KS2	End of Upper KS2
Programming	☑ Follow a series of instructions ☑ Combine a series of movements ☑ Exposure to programable hardware such as Beebots. ☑ Use logical reasoning in their play and to describe the world around them.	☑ Write short algorithms and programs for floor robots, and predict program outcomes. ☑ Design and program the movement of a character on screen to tell stories. ☑ Create and debug programs. ☑ Use logical reasoning to make predictions. ☑ 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 ☑ Write algorithms and programs that use a range of events to trigger sequences of actions. ☑ Use a text-based programming language to explore count-controlled loops when drawing shapes. ☑ Use a block-based programming language to explore count-controlled and infinite loops.	☑ Explore conditions and selection using a programmable microcontroller. ☑ Explore selection in programming to design and code an interactive quiz. ☑ Explore variable when designing and coding a game. ☑ Designing and coding a project that captures inputs from a physical device.
Data and Information	☑ Organise materials into the correct group (containers). ☑ Orally count groups of objects and identify more or less. ☑ Make comparisons between objects as well as comparing capacity and measure.	☑ Explore object labels and use them to sort and group objects by properties. ☑ Collect data in tally charts and use attributes to present and 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.
Computer Systems and Networks	☑ Name technology in a familiar environment. ☑ Sensible amounts of screentime contributing to a healthy lifestyle.	☑ Recognise technology in school and use it responsibly. ☑ Identify IT and how responsible use improves our world in school, and beyond.	☑ Identify that digital devices have inputs, processes, and outputs, and how devices can be connected to make networks.	☑ Recognise IT systems in the world and how some can enable searching on the internet.

	☑ Compare old and new technology ☑ Handle and experience different devices and think about their purpose.		☑ Recognise the internet as a network of networks including the WWW, and why we should evaluate online content.	☑ Explore how data is transferred by working collaboratively online.
Creating Media	☑ Use technology to take photographs ☑ Experience ways to create music digitally ☑ Experience drawing apps to create artwork to express their ideas and feelings.	☑ Choose appropriate tools in a program to create art, and make comparison when working non-digitally. ☑ Use a computer to create and format text, before comparing to writing non-digitally. ☑ Capture and change digital photographs for different purposes. ☑ Use a computer as a tool to explore rhythms and melodies, before creating a musical composition.	☑ Identify that digital devices have inputs, processes and outputs and how devices can be connected ☑ Capture and edit audio to produce a podcast ensuring that copyright is considered.	☑ Plan, capture and edit video to produce a short film. ☑ Design and create webpages, giving consideration to copyright, aesthetics and navigation.

Computing Long Term Plan

Cycle 1

	Aut 1	Aut 2	Spr 1	Spr 2	Sum 1	Sum 2
EYFS	Data and information	Programming A	Computer systems and networks A	Computer systems and networks B	Creating media	Programming B
KS1	Computing Systems & Networks – Technology Around Us (Y1)	Creating Media – Digital Painting (Y1)	Creating Media – Digital Photography (Y2)	Data & Information – Grouping Data (Y1)	Programming A – Moving a Robot (Y1)	Programming B – Programming Animations (Y1)
LKS2	Computing Systems & Networks – Connecting Computers (Y3)	Creating Media – Photo Editing (Y4)	Programming A – Sequencing Sound (Y3)	Computing Systems & Networks – The Internet (Y4)	Creating Media – Stop-frame Animations (Y3)	Programming B – Events & Actions (Y3)
Y5	Computing Systems & Networks – Systems & Searching	Creating Media – Video Production	Programming A – Selection in Physical Computing	Data & Information – Flat-file Databases	Creating Media – Introduction to Vector Graphics	Programming B – Selection in Quizzes
Y6	Computing Systems & Networks – Communication & Collaboration		Programming A – Variables in Games		Programming B – Sensing Movement	

Cycle 2

	Aut 1	Aut 2	Spr 1	Spr 2	Sum 1	Sum 2
EYFS	Data and information	Programming A	Computer systems and networks A	Computer systems and networks B	Creating media	Programming B
KS1	Computing Systems & Networks – IT Around Us (Y2)	Creating Media – Digital Writing (Y1)	Creating Media – Digital Music (Y2)	Data & Information – Pictograms (Y2)	Programming A – Robot Algorithms (Y2)	Programming B – Programming Quizzes (Y2)
LKS2	Data & Information – Branching Databases (Y3)	Creating Media – Desktop Publishing (Y3)	Programming A – Repetition in Shapes (Y4)	Data & Information – Data Logging (Y4)	Creating Media – Audio Production (Y4)	Programming B – Repetition in Games (Y4)
Y5	Computing Systems & Networks – Systems & Searching	Creating Media – Video Production	Programming A – Selection in Physical Computing	Data & Information – Flat-file Databases	Creating Media – Introduction to Vector Graphics	Programming B – Selection in Quizzes
Y6	Computing Systems & Networks – Communication & Collaboration		Programming A – Variables in Games		Programming B – Sensing Movement	

EYFS Medium Term Plan

FS1	Key Substantive knowledge
Autumn	• I can match objects and pictures that are the same • I can follow a two-step instruction • I use technology in my play • Explore a range of technology
Spring	• Discuss objects that help us • Discuss preferences for different technologies. • Explore a range of technology
Summer	• I notice differences between similar objects • I can explore different ways to make music, including digitally. • Explore toys with digital functions • Explore a range of technology

FS2	Key substantive knowledge		
	Learning objective	**Teachers subject knowledge**	**Component knowledge**
Autumn 1 Data and information	• I can sort by colour • I can sort by shape • I can sort by size • I can sort by abstract criteria e.g. texture, material or function • I can describe how I have matched items • I know that there is more than one way to match items • I can count objects accurately up to 5.	Children may need to practise directly matching by colour, shape, size, and other criteria before they can sort. Children will need to explore similarities and difference before they can sort into groups.	• To sort materials into the correct group • To count groups of objects up to 5. • To explain how objects have been sorted.
Autumn 2 Programming A	• I can follow a three-step instruction • I can follow a four-step instruction • I can copy a single movement • I can copy two movements in a sequence • I can copy three movements in a sequence • I can copy four movements in a sequence • I can copy five movements in a sequence		• To follow a series of instructions • To combine a series of movements
Spring 1 Computer systems and networks A	• I can identify objects that are not technology • I can identify objects that are technology • I can name technology in my home • I can name technology in my school • I know that technology can help us • I know that technology can be used for entertainment • I know that lots of screen time is not healthy • I can name other ways to have fun without screen time	Technology is the application of scientific knowledge to the practical aims of human life or to the change and manipulation of the human environment. Technology can involve the use of tools, devices, systems, or methods that are based on scientific principles and methods. Technology can also refer to the products or outcomes of such application. Technology can be used for various purposes, such as industrial, commercial, or personal objectives. Objects such as a bike can be defined as technology.	• To name technology in a familiar environment • To identify objects that are technology and those that are not. • To know that lots of screen-time in unhealthy

Spring 2 Computer systems and networks B	• I can handle different devices and think about their purpose • I can identify old from new technology • I can think about why the technology was developed	• To compare old and new technology • To think and discuss the purpose of different devices
Summer 1 Creating media	• I can name different devices to take photographs • I can think about why different devices might be used to take photographs • I can name an object, scene or person I want to take a photograph of. • I can use an iPad or digital camera to capture a picture • I can use an iPad or digital camera to capture a picture of object, person or scene I have named. • I can explore music apps to create different sounds e.g. piano, drums • I can use a drawing app to create a picture	• To use technology to take photographs • To experience ways to create music digitally • To experience drawing apps to create artwork to express their ideas and feelings.
Summer 2 Programming B	• I can experiment with the different functions of a BeeBot. • I can predict the different functions of the buttons on a BeeBot. • I can describe what the BeeBot has done. • I can explore programming the BeeBot to move in a desired direction.	• To be exposed to programable hardware such as Beebots.
E-safety	• I can talk about what I am doing on a device • I know when something online has made me feel bad. • I know to tell a trusted adult if something online makes me feel bad. • I know it is not my fault if something on the screen has made me bad. • I know not to tell strangers my school or where I live.	

KS1 Cycle 1 Medium Term Plans

<table>
<tr><td colspan="4">KS1 MTP Cycle 1</td></tr>
<tr><td colspan="4">Autumn 1</td></tr>
<tr><td colspan="4">Unit overview

Learners 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.

Note: This lesson has been planned using desktop computers and the (free) program paintz.app, however, it can be taught with laptops. If you are using laptops for this unit, consider spending more time practising and discussing the trackpad.</td></tr>
<tr><td>Unit</td><td>Learning objectives</td><td>Teachers subject knowledge</td><td>Component knowledge</td></tr>
<tr><td rowspan="6">Computing Systems & Networks – Technology Around Us (Y1)</td><td>• I can explain technology as something that helps us
• I can locate examples of technology in the classroom
• I can explain how these technology examples help us</td><td rowspan="6">Teachers need to know that the definition of technology is something that has been made with a specific purpose to help other people. Teachers should familiarise themselves with objects which are and are not examples of technology.

Teachers will need to be aware that typing is the process of using a keyboard to write words, letters or numbers on a screen.</td><td rowspan="6">• To identify technology
• To identify a computer and its main parts
• To use a mouse in different ways
• To use a keyboard to type on a computer
• To use the keyboard to edit text
• To create rules for using technology responsibly</td></tr>
<tr><td>• I can name the main parts of a computer
• I can switch on and log into a computer
• I can use a mouse to click and drag</td></tr>
<tr><td>• I can name the main parts of a computer
• I can switch on and log into a computer
• I can use a mouse to click and drag</td></tr>
<tr><td>• I can say what a keyboard is for
• I can type my name on a computer
• I can save my work to a file</td></tr>
<tr><td>• I can open my work from a file
• I can use the arrow keys to move the cursor
• I can delete letters</td></tr>
<tr><td>• I can identify rules to keep us safe and healthy when we are using technology in and beyond the home
• I can give examples of some of these rules
• I can discuss how we benefit from these rules</td></tr>
<tr><td colspan="4">National curriculum links:
• Recognise common uses of information technology beyond school
• Use technology purposefully to create, organise, store, manipulate, and retrieve digital content
• 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.</td></tr>
</table>

Autumn 2			
Unit overview Learners 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.			
Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Creating Media – Digital Painting (Y1)	• I can make marks on a screen and explain which tools I used • I can draw lines on a screen and explain which tools I used • I can use the paint tools to draw a picture	Before teaching this unit, you should ensure you are familiar with the following: Lesson 1: The freehand painting tools in Microsoft Paint or the online app Paintz (paintz.app), or another appropriate digital painting program - Lesson 2: The style of Piet Mondrian (or another appropriate artist); primary colours; and the line, shape, fill, and undo tools in the digital painting program you've chosen - Lesson 3: The style of Henri Matisse (or another appropriate artist); the shape, fill, and undo tools in the digital painting program you've chosen - Lesson 4: The following painting tools in the digital painting program: paintbrush, pencil, fill, erase, undo, shape, and brush styles (eg spray can) if available - Lesson 5: The following painting tools in the digital painting program: paintbrush, undo, brush sizes, and brush styles if available - Lesson 6: The following painting tools in the digital painting program: paintbrush, pencil, fill tool, eraser, undo, shape tool, and brush styles if available	To describe what different freehand tools do To use the shape tool and the line tools To make careful choices when painting a digital picture To explain why I chose the tools I used To use a computer on my own to paint a picture To compare painting a picture on a computer and on paper
	• I can make marks with the square and line tools • I can use the shape and line tools effectively • I can use the shape and line tools to recreate the work of an artist		
	• I can choose appropriate shapes • I can make appropriate colour choices • I can create a picture in the style of an artist		
	• I can explain that different paint tools do different jobs • I can choose appropriate paint tools and colours to recreate the work of an artist • I can say which tools were helpful and why		
	• I can make dots of colour on the page • I can change the colour and brush sizes • I can use dots of colour to create a picture in the style of an artist on my own		
	• I can explain that pictures can be made in lots of different ways • I can spot the differences between painting on a computer and on paper • I can say whether I prefer painting using a computer or using paper		
National curriculum links- Computing • Use technology purposefully to create, organise, store, manipulate, and retrieve digital content Art and Design: Pupils should be taught: • To develop a wide range of art and design techniques in using colour, pattern, texture, line, shape, form, and space • About the work of a range of artists, craft makers, and designers, describing the differences and similarities between different practices and disciplines and making links to their own work			
Spring 1			

Unit overview

Learners will learn to recognise that different devices can be used to capture photographs and will gain experience capturing, editing, and improving photos. Finally, they will use this knowledge to recognise that images they see may not be real.

It is recommended that you use digital cameras to take photographs in these lessons, so that learners can experience a range of devices. However, tablets or other devices with cameras will also work. This unit uses screenshots from the website https://pixlr.com/x/, but you could also use the Pixlr app if you're using tablets.

Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Creating Media – Digital Photography (Y2)	• I can recognise what devices can be used to take photographs • I can talk about how to take a photograph • I can explain what I did to capture a digital photo	You should be familiar with the basic principles of photography, including composition, framing, lighting, and how to reduce blur. Lesson 5 uses an online photo editing tool, Pixlr, and knowledge of using photo editing software to apply filters to images is required to use this effectively; you should also be familiar with saving and downloading images.	To use a digital device to take a photograph To make choices when taking a photograph To describe what makes a good photograph To decide how photographs can be improved To use tools to change an image To recognise that photos can be changed
	• I can explain the process of taking a good photograph • I can take photos in both landscape and portrait format • I can explain why a photo looks better in portrait or landscape format		
	• I can identify what is wrong with a photograph • I can discuss how to take a good photograph • I can improve a photograph by retaking it		
	• I can explore the effect that light has on a photo • I can experiment with different light sources • I can explain why a picture may be unclear		
	• I can recognise that images can be changed • I can use a tool to achieve a desired effect • I can explain my choices		
	• I can apply a range of photography skills to capture a photo • I can recognise which photos have been changed • I can identify which photos are real and which have been changed		

National curriculum links- Computing

- Use technology purposefully to create, organise, store, manipulate, and retrieve digital content
- 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

Art and design

- To develop a wide range of art and design techniques in using colour, pattern, texture, line, shape, form, and space

Spring 2

Unit overview

This unit introduces learners to data and information. Labelling, grouping, and searching are important aspects of data and information. Searching is a common operation in many applications, and requires an understanding that to search data, it must have labels. This unit of work focuses on assigning data (images) with different labels in order to demonstrate how computers are able to group and present data.

During this unit, learners will be logging on to the computers, opening their documents, and saving their documents. Depending on how your school's system is set up, additional support and time may be required to facilitate these steps, and consideration should be given as to how this will impact the timings of activities in each lesson.

Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Data & Information – Grouping Data (Y1)	• I can describe objects using labels • I can match objects to groups • I can identify the label for a group of objects	You will need to be aware that the term 'object' is used to describe anything that can be labelled with properties, eg animals, pencils, or trees. When talking about objects, they are named to make it easier for humans to know what other humans are talking about, eg 'tree'. The name may change depending on context (sometimes 'tree' is enough, but sometimes 'oak tree' may be required), but it is always a property that an object can be labelled with. A label is a property used to describe an object, eg 'green'. This is the data that is collected about the object. You will need an understanding that computers are not intelligent. Although they may seem like they are able to complete tasks autonomously, they are using input from humans, for example, searching for images that have been labelled by a person, or 'counting' data that has been grouped by people. Through the unit, teachers will need to be aware that: • Computers can be used to group data for analysis. The analysis in this unit is limited to a simple count of the objects in a group. Grouping is revisited throughout the data and information units. • The term 'property' to describe objects. A label is a property used to describe an object, eg 'green'. This is the data that is collected about the object. • 'Data set' is a term used to describe a collection of related data. • The link between grouping objects in the real world and grouping objects on a computer. To strengthen this link, the language of 'is…' and 'is not…' should be used wherever possible. • Objects can be grouped by different properties, so there are multiple ways of grouping the same objects.	To label objects To identify that objects can be counted To describe objects in different ways To count objects with the same properties To compare groups of objects To answer questions about groups of objects
	• I can count objects • I can group objects • I can count a group of objects		
	• I can describe an object • I can describe a property of an object • I can find objects with similar properties		
	• I can group similar objects • I can group objects in more than one way • I can count how many objects share a property		
	• I can choose how to group objects • I can describe groups of objects • I can record how many objects are in a group		
	• I can decide how to group objects to answer a question • I can compare groups of objects • I can record and share what I have found		

National curriculum links

- Use technology purposefully to create, organise, store, manipulate, and retrieve digital content
- Use technology safely and respectfully

Summer 1

Learners will be introduced to early programming concepts. Learners will explore using individual commands, both with other learners and as part of a computer program. They will identify what each command for the floor robot does, and use that knowledge to start predicting the outcome of programs. The unit is paced to ensure time is spent on all aspects of programming, and builds knowledge in a structured manner. Learners are also introduced to the early stages of program design through the introduction of algorithms.

This unit includes references relating to Bee-Bot and Blue-Bot floor robots, however, other educational floor robots are available. Learners should be given access to a device with a limited range of functions that is designed for young learners. Before starting this unit, ensure you are familiar with your school's floor robots, including charging or battery requirements. You should also know how to switch the devices on and off, as well as key functions such as clearing the memory. It is advisable to use the robots on the floor if possible, as this can reduce damage caused by dropping.

Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Programming A – Moving a Robot (Y1)	• I can predict the outcome of a command on a device • I can match a command to an outcome • I can run a command on a device	This unit focuses on developing learners' understanding of computer programming. It highlights that algorithms are a set of clear, precise, and ordered instructions, and that a computer program is the implementation of an algorithm on a digital device. The unit also introduces reading 'code' to predict what a program will do. Learners will engage in aspects of program design, including outlining the project task and creating algorithms. When programming, there are four levels that can help describe a project, known as 'levels of abstraction'. Research suggests that this structure can support learners in understanding how to create a program and how it works: • Task — what is needed • Design — what it should do • Code — how it is done • Running the code — what it does Spending time at the 'task' and 'design' levels before engaging in writing code aids learners in assessing the achievability of their programs and reduces the cognitive load for learners during programming.	To explain what a given command will do To act out a given word To combine 'forwards' and 'backwards' commands to make a sequence To combine four direction commands to make sequences To plan a simple program To find more than one solution to a problem
	• I can follow an instruction • I can recall words that can be acted out • I can give directions		
	• I can compare forward and backward movements • I can start a sequence from the same place • I can predict the outcome of a sequence involving 'forwards' and 'backwards' commands		
	• I can compare left and right turns • I can experiment with 'turn' and 'move' commands to move a robot • I can predict the outcome of a sequence involving up to four commands		
	• I can explain what my program should do • I can choose the order of commands in a sequence • I can debug my program		
	• I can identify several possible solutions • I can plan two programs • I can use two different programs to get to the same place		

- 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
- Recognise common uses of information technology beyond school

Summer 2

Unit overview

Learners 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.

All the lessons in this unit require access to ScratchJr.

- Download ScratchJr App for tablets (iPad or Android), or install ScratchJr for computers (https://jfo8000.github.io/ScratchJr-Desktop/) before the lesson

Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Programming B – Programming animations (Y1)	• I can find the commands to move a sprite • I can use commands to move a sprite • I can compare different programming tools	The unit focuses on developing learners' understanding of computer programming. It highlights that algorithms are a set of clear, precise, and ordered instructions, and that a computer program is the implementation of an algorithm on a digital device. The unit also introduces reading 'code' to predict what a program will do. Learners will engage in aspects of program design, including outlining the project task and creating algorithms. When programming, there are four levels that can help describe a project, known as levels of abstraction. Research suggests that this structure can support learners in understanding how to create a program and how it works: • Task – what is needed • Design – what it should do • Code – how it is done • Running the code – what it does Spending time at the 'task' and 'design' levels before engaging in code writing aids learners in assessing the achievability of their programs, and reduces a learner's cognitive load during programming.	To choose a command for a given purpose To show that a series of commands can be joined together To identify the effect of changing a value To explain that each sprite has its own instructions To design the parts of a project To use my algorithm to create a program
	• I can use more than one block by joining them together • I can use a **Start** block in a program • I can run my program		
	• I can find blocks that have numbers • I can change the value • I can say what happens when I change a value		
	• I can show that a project can include more than one sprite • I can delete a sprite • I can add blocks to each of my sprites		
	• I can choose appropriate artwork for my project • I can decide how each sprite will move • I can create an algorithm for each sprite		
	• I can use sprites that match my design • I can add programming blocks based on my algorithm • I can test the programs I have created		

National curriculum links

- 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

KS1 MTP- Cycle 2			
Autumn 1			
Unit overview Learners will 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.			
Unit	Key learning	Teachers subject knowledge	Sticky knowledge
Computing Systems & Networks – IT Around Us (Y2)	To recognise the uses and features of information technology To identify the uses of information technology in the school To identify information technology beyond school To explain how information technology helps us To explain how to use information technology safely To recognise that choices are made when using information technology	You will need to have a clear understanding of devices that can be described as information technology (IT). For younger learners, IT can be explained as being a computer or something that has been made to work with computers. Examples could include: • Computers: PCs, laptops, tablets • Devices made to work with computers: scanners, barcode scanners, printers., smart speakers You will also need to be aware that as technology continues to develop rapidly, some devices may fit in multiple categories. For example, a multifunction printer has a computer (processor) inside. It can work with a computer or independently. You will need to know where technology can be found in shops and how it can be used. You should also know which devices can work together, for example: • Barcode scanner, till • Bank card, chip and PIN card reader, till • Traffic light, crossing button, crossing signal You can find some useful information and a short video about barcodes at www.waspbarcode.com/buzz/barcode.	
National curriculum links: • Use technology purposefully to create, organise, store, manipulate, and retrieve digital content • 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			
Autumn 2			

Unit overview

You will need to have a clear understanding of devices that can be described as information technology (IT). For younger learners, IT can be explained as being a computer or something that has been made to work with computers.

Examples could include:

- Computers: PCs, laptops, tablets
- Devices made to work with computers: scanners, barcode scanners, printers., smart speakers

You will also need to be aware that as technology continues to develop rapidly, some devices may fit in multiple categories. For example, a multifunction printer has a computer (processor) inside. It can work with a computer or independently.

You will need to know where technology can be found in shops and how it can be used. You should also know which devices can work together, for example:

- Barcode scanner, till
- Bank card, chip and PIN card reader, till
- Traffic light, crossing button, crossing signal

You can find some useful information and a short video about barcodes at www.waspbarcode.com/buzz/barcode.

Unit	Key learning	Teachers subject knowledge	Sticky knowledge
Creating Media – Digital Writing (Y1)	To use a computer to write To add and remove text on a computer To identify that the look of text can be changed on a computer To make careful choices when changing text To explain why I used the tools that I chose To compare typing on a computer to writing on paper	You will need to be familiar with the word processing software used in your school (Google Docs, Microsoft Word, or other) and the layout of the computer keyboard. In this unit, the key skills covered are adding and removing text; using basic formatting tools such as bold, italic, and underline; using click and drag to select text; and changing the font of text.	

National curriculum links

- Use technology purposefully to create, organise, store, manipulate, and retrieve digital content
- Use technology safely and respectfully, keeping personal information private

Spring 1

Unit overview

In this unit, learners will be using a computer to create music. They will listen to a variety of pieces of music and consider how music can make them think and feel. Learners will compare creating music digitally and non-digitally. Learners will look at patterns and purposefully create music.

Unit	Key learning	Teachers subject knowledge	Sticky knowledge

Creating Media – Digital Music (Y2)	To say how music can make us feel
	To identify that there are patterns in music
	To experiment with sound using a computer
	To use a computer to create a musical pattern
	To create music for a purpose
	To review and refine our computer work
		• You should be familiar with *The Planets* by Gustav Holst:
		  ○ BBC Ten Pieces (includes video recordings of the suite and music/digital art lesson plan ideas): www.bbc.co.uk/programmes/articles/14ZjT5yjnKQRdKVsqrLzk1x/mars-from-the-planets-by-gustav-holst
		  ○ Gustav Holst's *The Planets* : a guide – Classic FM: www.classicfm.com/composers/holst/pictures/holsts-planets-guide
		  ○ Learning to Listen: Gustav Holst's *The Planets* – YourClassical: www.yourclassical.org/story/2014/02/10/gustav-holst-the-planets-on-learning-to-listen
		• You should also be familiar with musical terminology:
		  ○ BBC: www.bbc.co.uk/bitesize/subjects/zwxhfg8
		  ○ BBC Bitesize video (pulse and rhythm): www.bbc.co.uk/bitesize/clips/zmqn34j
		• You should be familiar with Chrome Music Lab (musiclab.chromeexperiments.com/About), including:
		  ○ The Song Maker tool (musiclab.chromeexperiments.com/Song-Maker)
		  ○ Saving and opening work in Chrome Music Lab	

National curriculum links

- Use technology purposefully to create, organise, store, manipulate, and retrieve digital content

Spring 2

Unit overview

Learners 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.

During this unit of work learners will use j2e pictogram tool which can be accessed online using a desktop, laptop or tablet computer. Your school may have access to an equivalent alternative which could be used instead.

Unit	Key learning	Teachers subject knowledge	Sticky knowledge
Data & Information – Pictograms (Y2)	To recognise that we can count and compare objects using tally charts To recognise that objects can be represented as pictures To create a pictogram To select objects by attribute and make comparisons To recognise that people can be described by attributes	This unit builds on prior learning from the Year 1 unit 'Grouping data'. Teachers should understand how tally charts and pictograms are created, and the benefits of organising data in those formats. These different formats allow data to be presented in different ways and will suit different purposes.. Teachers will need to understand how people, animals and objects can be described using different attributes.	

National curriculum links

Computing

- use technology purposefully to create, organise, store, manipulate and retrieve digital content
- 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

Maths

Building on Year 1 number and place value:

- Identify and represent numbers using objects and pictorial representations including the number line, and use the language of: 'equal to', 'more than', 'less than' ('fewer'), 'most', 'least'

Year 2

- interpret and construct simple pictograms, tally charts, block diagrams and simple tables
- ask and answer simple questions by counting the number of objects in each category and sorting the categories by quantity
- ask and answer questions about totalling and comparing categorical data

Summer 1

Unit overview

This unit 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.

There are two Year 2 programming units:

- Programming A – Robot algorithms
- Programming B – Programming quizzes

This is unit A, which should be delivered before unit B.

This unit includes references relating to Bee-Bot and Blue-Bot floor robots, however, other educational floor robots are available. Learners should be given access to a device with a limited range of functions that is designed for young learners. Before delivering this unit, ensure that you are familiar with your school's floor robots, including charging or battery requirements. You should also know how to switch the devices on and off, as well as key functions such as clearing the memory. It is advisable to use the robots on the floor if possible, as this can reduce damage caused by dropping.

Unit	Key learning	Teachers subject knowledge	Sticky knowledge
Programming A – Robot Algorithms (Y2)	To describe a series of instructions as a sequence To explain what happens when we change the order of instructions To use logical reasoning to predict the outcome of a program To explain that programming projects can have code and artwork To design an algorithm To create and debug a program that I have written	This unit focuses on developing learners' understanding of computer programming. It highlights that algorithms are a set of clear, precise, and ordered instructions, and that a computer program is the implementation of an algorithm on a digital device. The unit also introduces reading 'code' to predict what a program will do. Learners will engage in aspects of program design, including outlining the project task and creating algorithms. When programming, there are four levels that can help describe a project, known as 'levels of abstraction'. Research suggests that this structure can support learners in understanding how to create a program and how it works: • Task — what is needed • Design — what it should do • Code — how it is done • Running the code — what it does Spending time at the 'task' and 'design' levels before engaging in writing code aids learners in assessing the achievability of their programs and reduces the cognitive load for learners during programming.	

National curriculum links

- 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

Summer 2

Unit overview

This unit initially recaps on learning from the Year 1 ScratchJr unit 'Programming B – Programming animations'. Learners begin to understand that sequences of commands have an outcome, and make predictions based on their learning. They use and modify designs to create their own quiz questions in ScratchJr, and realise these designs in ScratchJr using blocks of code. Finally, learners evaluate their work and make improvements to their programming projects.

There are two Year 2 programming units:

- Programming A – Robot algorithms
- Programming B – Programming quizzes

This is unit B, which should be delivered after unit A.

Unit	Key learning	Teachers subject knowledge	Sticky knowledge
Programming B – Programming Quizzes (Y2)	To explain that a sequence of commands has a start To explain that a sequence of commands has an outcome To create a program using a given design To change a given design To create a program using my own design To decide how my project can be improved	This unit focuses on developing learners' understanding of computer programming. It highlights that algorithms are a set of clear, precise, and ordered instructions, and that a computer program is the implementation of an algorithm on a digital device. The unit also introduces reading 'code' to predict what a program will do. Learners will engage in aspects of program design, including outlining the project task and creating algorithms. When programming, there are four levels that can help describe a project, known as Levels of abstraction. Research suggests that this structure can support learners in understanding how to create a program and how it works: Task – what is needed Design – what it should do Code – how it is done Running the code – what it does Spending time at the 'task' and 'design' levels before engaging in code-writing aids learners in assessing the achievability of their programs, and reduces a learner's cognitive load during programming.	

National curriculum links

- 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
- Use technology purposefully to create, organise, store, manipulate and retrieve digital content

<table>
<tr><td colspan="4">LKS2 MTP</td></tr>
<tr><td colspan="4">Autumn 1</td></tr>
<tr><td colspan="4">Unit overview

Learners 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.

You will need digital devices for learners to interact with during this unit. Lesson 3 requires digital devices with a painting application. Lesson 6 includes a 'network tour', which involves learners identifying key parts of your school network. You will therefore need access to your school's server, switch, and wireless access points.</td></tr>
<tr><td>Unit</td><td>Learning objectives</td><td>Teachers subject knowledge</td><td>Component knowledge</td></tr>
<tr><td rowspan="5">Computing Systems & Networks – Connecting Computers (Y3)</td><td>• I can explain that digital devices accept inputs
• I can explain that digital devices produce outputs
• I can follow a process</td><td rowspan="5">You will need an understanding of digital and non-digital devices. The key difference between them is that a digital device is capable of some processing, ie it has functions beyond being either on or off. You will also need to be familiar with the concept of input, process, output (IPO), which underpins all digital devices. You will need to understand that devices can have one input that leads to several outputs (eg starting a video leads to outputs from the screen and the speaker) and that many inputs can lead to one output (e.g. using a mouse and a keyboard to produce a document).

You will need a basic understanding of how information (data) flows around a computer network, and how this benefits us. You will also need to know that a network switch manages the way in which data moves around a network. You will need to be familiar with the main parts of a school network, including the server, wireless access points, network switch, router, and output devices such as a printer or copier.</td><td rowspan="5">To explain how digital devices function
To identify input and output devices
To recognise how digital devices can change the way that we work
To explain how a computer network can be used to share information
To explore how digital devices can be connected
To recognise the physical components of a network</td></tr>
<tr><td>• I can classify input and output devices
• I can describe a simple process
• I can design a digital device</td></tr>
<tr><td>• I can explain how I use digital devices for different activities
• I can recognise similarities between using digital devices and using non-digital tools
• I can suggest differences between using digital devices and using non-digital tools</td></tr>
<tr><td>• I can recognise different connections
• I can explain how messages are passed through multiple connections
• I can discuss why we need a network switch</td></tr>
<tr><td>• I can recognise that a computer network is made up of a number of devices
• I can demonstrate how information can be passed between devices
• I can explain the role of a switch, server, and wireless access point in a network</td></tr>
</table>

	• I can identify how devices in a network are connected together • I can identify networked devices around me • I can identify the benefits of computer networks		

National curriculum links:

Computing

- use sequence, selection, and repetition in programs; work with variables and various forms of input and output
- understand computer networks including the internet; how they can provide multiple services, such as the World Wide Web; and the opportunities they offer for communication and collaboration
- select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information

Maths (Lesson 1)

- **Number and place value:** solve number problems and practical problems involving these ideas.

Art (Lesson 3)

- to improve their mastery of art and design techniques, including drawing, painting and sculpture with a range of materials [for example, pencil, charcoal, paint, clay]

Autumn 2

Unit overview

Learners will develop their understanding of how digital images can be changed and edited, and how they can then be resaved and reused. They will consider the impact that editing images can have, and evaluate the effectiveness of their choices.

Throughout this unit, there are opportunities to model with photo editing applications or to demonstrate a concept using the included screen recordings. Pedagogically, it is more beneficial to model the concepts and skills to the learners, which allows for easier questioning and understanding. We recommend that you use the screen recordings to see what needs to be modelled, but give a live demonstration within the lesson. However, the videos are provided on the slides if you wish to use them instead.

Unit	Component learning	Teachers subject knowledge	Component knowledge
Creating Media – Photo Editing (Y4)	• I can improve an image by rotating it • I can explain why I might crop an image • I can use photo editing software to crop an image	• You will need to be familiar with the tools used throughout the unit in paint.net or your chosen image editor, and know how to save a new version of an image from within the editor. You can find a guide to all tools in paint.net at www.getpaint.net/doc/latest/index.html. ○ You should consider how the learners will access the editor. For example, you may wish to create a shortcut to the program for them. **Lesson 1** • You will need to be familiar with the effect that cropping can have on an image. You can find more information at www.dpreview.com/forums/post/56318241. **Lesson 2**	To explain that the composition of digital images can be changed To explain that colours can be changed in digital images To explain how cloning can be used in photo editing To explain that images can be combined
	• I can explain that different colour effects make you think and feel different things • I can experiment with different colour effects • I can explain why I chose certain colour effects		
	• I can add to the composition of an image by cloning • I can identify how a photo edit can be improved • I can remove parts of an image using cloning		

	• I can experiment with tools to select and copy part of an image • I can use a range of tools to copy between images • I can explain why photos might be edited	• You will need to be familiar with how to make image adjustments and change effects in paint.net, or your chosen image editor — there is a video in the lesson for support if you need it. **Lesson 3** • You will need to be familiar with the tools used in this lesson in paint.net or your chosen image editor. For more information about tools in paint.net, visit the following websites: ○ Guide to all tools in paint.net: www.getpaint.net/doc/latest/index.html ○ The 'clone stamp': www.getpaint.net/doc/latest/CloneStamp.html **Lesson 4** • You will need to be familiar with the tools used in this lesson in paint.net or your chosen image editor. For more information about tools in paint.net, visit the following websites: ○ Guide to all tools in paint.net: www.getpaint.net/doc/latest/index.html **Lesson 5** • You will need to be familiar with the tools used in this lesson in paint.net or your chosen image editor. **Lesson 6** • You will need to be familiar with the text tool in paint.net or your chosen image editor. For more information about these tools in paint.net, visit www.getpaint.net/doc/latest/TextShapeTools.html.	To combine images for a purpose To evaluate how changes can improve an image
	• I can describe the image I want to create • I can choose suitable images for my project • I can create a project that is a combination of other images		
	• I can review images against a given criteria • I can use feedback to guide making changes • I can combine text and my image to complete the project		

National curriculum links

- Select, use, and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems, and content that accomplish given goals, including collecting, analysing, evaluating, and presenting data and information
- Use technology safely, respectfully, and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact

Spring 1

Unit overview

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.

Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Programming A – Sequencing Sound (Y3)	• I can identify the objects in a Scratch project (sprites, backdrops) • I can explain that objects in Scratch have attributes (linked to) • I can recognise that commands in Scratch are represented as blocks	This unit focuses on developing learners' understanding of sequences in a new programming language. It highlights that the order of sequences is important. This unit also develops learners' understanding of design in programming, using the approach outlined below. When programming, there are four levels which can help describe a project (known as levels of abstraction). Research suggests that this structure can support learners in understanding how to create a program and how it works: • Task - what is needed • Design - what it should do • Code - how it is done • Running the code - what it does Spending time at the task and design levels before engaging in code-writing can aid learners in assessing the 'do-ability' of their programs. It also reduces a learner's cognitive load during programming.	To explore a new programming environment To identify that commands have an outcome To explain that a program has a start To recognise that a sequence of commands can have an order To recognise that a sequence of commands can have an order To create a project from a task description
	• I can identify that each sprite is controlled by the commands I choose • I can choose a word which describes an on-screen action for my plan • I can create a program following a design		
	• I can start a program in different ways • I can create a sequence of connected commands • I can explain that the objects in my project will respond exactly to the code		
	• I can explain what a sequence is • I can combine sound commands • I can order notes into a sequence		
	• I can build a sequence of commands • I can decide the actions for each sprite in a program • I can make design choices for my artwork		
	• I can identify and name the objects I will need for a project • I can relate a task description to a design • I can implement my algorithm as code		

National curriculum links- Computing

- Design, write, and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts
- Use sequence, selection, and repetition in programs; work with variables and various forms of input and output
- Use logical reasoning to explain how some simple algorithms work, and to detect and correct errors in algorithms and programs
- Select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information

Spring 2

Unit overview

Learners will apply their knowledge and understanding of networks, to appreciate the internet as a network of networks which need to be kept secure. They will learn that the World Wide Web is part of the internet, and will be given opportunities to explore the World Wide Web for themselves in order to learn about who owns content and what they can access, add, and create. Finally, they will evaluate online content to decide how honest, accurate, or reliable it is, and understand the consequences of false information.

This unit requires devices with an internet connection. Chrome Music Lab is used in one lesson to demonstrate content which can be produced on the World Wide Web.

Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Computing Systems & Networks – The Internet (Y4)	• I can describe the internet as a network of networks • I can demonstrate how information is shared across the internet • I can discuss why a network needs protecting • I can describe networked devices and how they connect • I can explain that the internet is used to provide many services • I can recognise that the World Wide Web contains websites and web pages • I can explain the types of media that can be shared on the WWW • I can describe where websites are stored when uploaded to the WWW • I can describe how to access websites on the WWW • I can explain what media can be found on websites • I can recognise that I can add content to the WWW • I can explain that internet services can be used to create content online • I can explain that websites and their content are created by people • I can suggest who owns the content on websites • I can explain that there are rules to protect content	Teachers will need a knowledge of computer networks, including how data is routed around the internet. Teachers will need to be aware that the World Wide Web is one of many services which are offered over the internet. They will need to know the difference between a web page and a website, and a knowledge of where websites are stored. A knowledge of what content you can find on websites will also be useful. An awareness of copyright (and the reasons for it) and that people create and share false and inaccurate information is important for the last two lessons in this unit. The YouTube video titled 'A Packet's Tale' (www.youtube.com/watch?v=ewrBalT_eBM) provides an overview of networks and the internet. That the World Wide Web is part of the internet is explained in this video: www.bbc.co.uk/newsround/47523993	To describe how networks physically connect to other networks To recognise how networked devices make up the internet To outline how websites can be shared via the World Wide Web (WWW) To describe how content can be added and accessed on the World Wide Web (WWW) To recognise how the content of the WWW is created by people To evaluate the consequences of unreliable content

	• I can explain that not everything on the World Wide Web is true • I can explain why some information I find online may not be honest, accurate, or legal • I can explain why I need to think carefully before I share or reshare content		

National curriculum links

- Understand computer networks including the internet; how they can provide multiple services, such as the World Wide Web, and the opportunities they offer for communication and collaboration
- Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content
- Select, use, and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems, and content that accomplish given goals, including collecting, analysing, evaluating, and presenting data and information
- Use technology safely, respectfully, and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact.

Summer 1

Unit overview

Learners will use a range of techniques to create a stop-frame animation using tablets. Next, they will apply those skills to create a story-based animation. This unit will conclude with learners adding other types of media to their animation, such as music and text.

It is recommended that you use a tablet for this unit as this makes it simpler for learners to take the photos and do the editing. However, you could use stop-frame animation software on a desktop or laptop if this is what you have available. This unit uses screenshots from iMotion which is an iPad app, but you could also try Stop Motion Studio if you have Android tablets.

Unit	Key learning	Teachers subject knowledge	Sticky knowledge
Creating Media – Stop-frame Animations (Y3)	• I can draw a sequence of pictures • I can create an effective flip book—style animation • I can explain how an animation/flip book works	Teachers will need to understand that animations are a series of still images stitched together to create a motion video. Animations can be created using on-screen or off-screen (flipbooks) images. Teachers need to understand how to create a simple flipbook (see lesson 1 for support) and how to use software to create an on-screen animation (support is provided in the lessons). Teachers will need to have an understanding of using their chosen software. Within the software, teachers will need to be aware of how to take images, 'onion skinning' (showing a part transparent photo to demonstrate the previous frame to make small movements more consistent), deleting frames and saving.	To explain that animation is a sequence of drawings or photographs To relate animated movement with a sequence of images To plan an animation To identify the need to work consistently and carefully To review and improve an animation To evaluate the impact of adding other media to an animation
	• I can predict what an animation will look like • I can explain why little changes are needed for each frame • I can create an effective stop-frame animation		
	• I can break down a story into settings, characters and events • I can describe an animation that is achievable on screen • I can create a storyboard		
	• I can use onion skinning to help me make small changes between frames • I can review a sequence of frames to check my work • I can evaluate the quality of my animation		
	• I can explain ways to make my animation better • I can evaluate another learner's animation • I can improve my animation based on feedback		
	• I can add other media to my animation • I can explain why I added other media to my animation		

	• I can evaluate my final film		

National curriculum links

- Select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information
- use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact.

Summer 2

Unit overview

This unit explores the links between events and actions, while consolidating prior learning relating to sequencing. Learners begin by moving a sprite in four directions (up, down, left, and right). They then explore movement within the context of a maze, using design to choose an appropriately sized sprite. This unit also introduces programming extensions, through the use of **Pen** blocks. Learners are given the opportunity to draw lines with sprites and change the size and colour of lines. The unit concludes with learners designing and coding their own maze-tracing program.

Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Programming B – Events & Actions (Y3)	• I can explain the relationship between an event and an action • I can choose which keys to use for actions and explain my choices • I can identify a way to improve a program	This unit focuses on the links between 'events' and 'actions' in programming, while also developing learners' understanding of sequencing. It highlights that events cause actions, and that the order of those actions can have an impact on the outcome of a program. This unit also further develops learners' understanding of design in programming, using the approach outlined below. When programming, there are four levels that help to describe the stages of a project, known as levels of abstraction. Research suggests that this structure can support learners in understanding how to create a program and how it works. • Task — this is what is needed • Design — this is what it should do • Code — this is how it is done • Running the code — this is what it does Spending time at the Task and Design levels before engaging in code writing aids learners in assessing the 'do-ability' of their programs and reduces a learner's cognitive load during programming.	To explain how a sprite moves in an existing project To create a program to move a sprite in four directions To adapt a program to a new context To develop my program by adding features To identify and fix bugs in a program To design and create a maze-based challenge
	• I can choose a character for my project • I can choose a suitable size for a character in a maze • I can program movement		
	• I can use a programming extension • I can consider the real world when making design choices • I can choose blocks to set up my program		
	• I can identify additional features (from a given set of blocks) • I can choose suitable keys to turn on additional features • I can build more sequences of commands to make my design work		
	• I can test a program against a given design • I can match a piece of code to an outcome • I can modify a program using a design		
	• I can make design choices and justify them • I can implement my design • I can evaluate my project		

National curriculum links

- Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts
- Use sequence, selection, and repetition in programs; work with variables and various forms of input and output
- Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs
- Select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information

Year 5 MTP			
Autumn 1			
Unit overview Learners 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.			
Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Computing systems and networks- systems and searching	• I can explain that systems are built using a number of parts • I can describe the input, process, and output of a digital system • I can explain that computer systems communicate with other devices	See resources	To explain that computers can be connected together to form systems To recognise the role of computer systems in our lives To identify how to use a search engine To describe how search engines select results To explain how search results are ranked To recognise why the order of results is important, and to whom
	• I can identify tasks that are managed by computer systems • I can identify the human elements of a computer system • I can explain the benefits of a given computer system		
	• I can make use of a web search to find specific information • I can refine my web search • I can compare results from different search engines		
	• I can explain why we need tools to find things online • I can recognise the role of web crawlers in creating an index • I can relate a search term to the search engine's index		
	• I can order a list by rank • I can explain that a search engine follows rules to rank results I can give examples of criteria used by search engines to rank results		
	• I can describe some of the ways that search results can be influenced • I can recognise some of the limitations of search engines • I can explain how search engines make money		

National curriculum links:

- Understand computer networks, including the internet; how they can provide multiple services, such as the World Wide Web, and the opportunities they offer for communication and collaboration
- Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content

Autumn 2

Unit overview

Learners will learn how to create short videos by working in pairs or groups. As they progress through this unit, they will be exposed to topic-based language and develop the skills of capturing, editing, and manipulating video. Learners are guided with step-by-step support to take their idea from conception to completion. At the conclusion of the unit, learners have the opportunity to reflect on and assess their progress in creating a video.

To teach this unit, you will need video recording equipment such as video cameras or tablets with video capabilities. The recommended editing software is Microsoft Video Editor, which is included for free with Windows 10.

Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Creating media-video production	• I can explain that video is a visual media format • I can identify features of videos • I can compare features in different videos	This unit focuses on the skills associated with planning, recording, editing, and creating a video. You will need to be able to explain that video is the recording, reproducing, or broadcasting of moving visual images. You will also need to be familiar with a number of shot types and filming techniques, which are introduced in Lessons 2 and 3. A storyboard is used as a planning tool. It will be useful if you are familiar with the format of the storyboard. Once learners begin filming, you will need to be familiar with the device they are using, including how to start and stop recording, how to zoom in and out, and how to download content from the device to a computer for editing. It is important that you are familiar with the devices and apps or programs that you will use to import and edit video content. Windows 10 Video Editor is demonstrated in this unit, but there are many other free and paid-for apps available that can be used, either on a computer or on a device. Another option with limited function is to use the Microsoft Photos app. You need to know where to locate the video files and where to save them for easy retrieval. Once recording has been completed, learners will need to import their video files to the video editing software, so you will need to be familiar with this process, including where videos will be stored.	To explain what makes a video effective To use a digital device to record video To capture video using a range of techniques To create a storyboard To identify that video can be improved through reshooting and editing To consider the impact of the choices made when making and sharing a video
	• I can identify and find features on a digital video recording device • I can experiment with different camera angles • I can make use of a microphone		
	• I can suggest filming techniques for a given purpose • I can capture video using a range of filming techniques • I can review how effective my video is		
	• I can outline the scenes of my video • I can decide which filming techniques I will use • I can create and save video content		
	• I can store, retrieve, and export my recording to a computer • I can explain how to improve a video by reshooting and editing • I can select the correct tools to make edits to my video		

	• I can make edits to my video and improve the final outcome • I can recognise that my choices when making a video will impact the quality of the final outcome • I can evaluate my video and share my opinions	You will need to have a clear understanding of how to edit and complete the video creation process, deleting or reordering clips. Finally, you should be able to demonstrate how to export the video project into an *.mp4 format for viewing.	

National curriculum links

- Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content
- Select, use, and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems, and content that accomplish given goals, including collecting, analysing, evaluating, and presenting data and information
- Use technology safely, respectfully, and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact

Internet safety

Use technology safely, respectfully, and responsibly; recognise acceptable/unacceptable behaviour

Spring 1

Unit overview

In this unit, learners 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.

Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Programming A- Selection in physical computing	• I can create a simple circuit and connect it to a microcontroller • I can program a microcontroller to make an LED switch on • I can explain what an infinite loop does	This unit focuses on physical computing, which allows learners to control real-life projects through the construction of programs. When learners undertake physical computing, they write programs that control real-world objects, like LEDs and motors, using a computer. The tangible effect of seeing the commands that they entered into a computer being carried out on a physical item, rather than on screen, can be highly motivational for learners. Physical computing also offers the opportunity to take a more project-based approach to learning, and allows learners to make choices about the purpose, design, and program of their product. Throughout this unit, there are opportunities to demonstrate a concept within the Crumble programming software or show a screencast animation on a slide. Pedagogically, it is more beneficial to demonstrate the concepts to learners, as it allows for easier questioning and understanding. We recommend that you use the animations to see what to demonstrate, then show learners with a live demonstration, however, animations are provided on the slides if you wish to use them instead. For this unit, you will need experience of constructing programs using the Crumble programming software (see the 'Resources' section at the end of this document). It uses the	To control a simple circuit connected to a computer To write a program that includes count-controlled loops To explain that a loop can stop when a condition is met To explain that a loop can be used to repeatedly check whether a condition has been met To design a physical project that includes selection To create a program that controls a physical computing project
	• I can connect more than one output component to a microcontroller • I can use a count-controlled loop to control outputs • I can design sequences that use count-controlled loops		
	• I can explain that a condition is either true or false • I can design a conditional loop • I can program a microcontroller to respond to an input		

	• I can explain that a condition being met can start an action • I can identify a condition and an action in my project • I can use selection (an 'if…then…' statement) to direct the flow of a program • I can identify a real-world example of a condition starting an action • I can describe what my project will do • I can create a detailed drawing of my project • I can write an algorithm that describes what my model will do • I can use selection to produce an intended outcome • I can test and debug my project	same drag-and-drop style as Scratch. You will need to write programs that turn LEDs (Sparkles) on and off, change LED colours, spin motors, use push switches as inputs, and combine a number of these components. Additionally, you will connect the Crumble controller to battery packs, Sparkles, motors, and push switches. For further support on using Crumbles, see the Crumble 'Getting Started' guide at redfernelectronics.co.uk/crumble-getting-started. **Levels of abstraction** When programming, there are four levels that can help describe a project (known as 'levels of abstraction'). Research suggests that this structure can support learners in understanding how to create a physical computing project or standalone program and how it works: • Task — this is what is needed • Design — this is what it should do • Build — this is how it is done • Running the code — this is what it does Spending time at the 'Task' and 'Design' levels before engaging in writing code aids learners in assessing the 'do-ability' of their programs and reduces a learner's cognitive load during programming. Learners will move between the different levels throughout the unit, and this is highlighted within each lesson plan. **Repetition** You will need to know that repetition is used in programming to give the same instruction or set of instructions several times. Repetition uses loops as the means to give these instructions. This unit makes use of two types of loops: infinite and count-controlled. These have been defined below. **An infinite loop** is a loop that commands the instruction/set of instructions to repeat forever. When an infinite loop is used in a program, there is no way of ending the program, as the command(s) within the loop will be repeated endlessly. For this reason, infinite loops should only be used when writing a program that is intended to run forever. The exception to this is when using selection in physical computing, as you will see throughout this unit. **A count-controlled loop** is a form of repetition in which a set of commands are carried out a specific number of times. Count-controlled loops should only be used when it is known how many times a set of commands needs to be repeated. **A condition-controlled loop** is a form of repetition in which a set of commands stop being carried out when a condition is met. The condition could be anything from when the 'score' in a game reaches a certain value to when a key on a keyboard has been pressed. **Conditions** Conditions are statements that need to be met for a set of actions to be carried out. They can be used in algorithms and programs to control the flow of actions. When a condition is met, it is referred to as 'true' and when it is not met, it is referred to as 'false'. You will need to be able to identify and use conditions in algorithms in the form of statements to both start and stop sets of action. Additionally, you will need to understand that conditions can be used in loops, and when they are, that the set of actions in the loop will be carried out repeatedly until the condition is true, for example, 'until button A is pressed'.	

		Selection Selection is “part of a program where, if a condition is met, then a set of commands are run”. Selection is implemented in programming using if...then... statements. Selection is used to control the flow of actions in algorithms and programs by checking if a condition (see above) has been met. If it has been met, the identified actions will be carried out. When selection is used in programs, loops (see above) often have to be used to instruct the device to check the condition repeatedly. Without using loops, the condition would only be checked once. It’s important to understand that each loop cycle will complete before the condition is checked again. In the Crumble programming software, selection is implemented through the if...then... command block. In addition to the above, you will also need to understand that programs are an implementation of an algorithm, and that when the program does not produce the required output, the algorithm should be debugged. This should then be implemented in the program.	

National curriculum links- Computing

- Design, write, and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts
- Use sequence, selection, and repetition in programs; work with variables and various forms of input and output
- Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs
- Select, use, and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems, and content that accomplish given goals, including collecting, analysing, evaluating, and presenting data and information

Science – Electricity (Year 4)

- Construct a simple series electrical circuit, identifying and naming its basic parts, including cells, wires, bulbs, switches, and buzzers

Design and Technology (Key stage 2)

Design

- Generate, develop, model, and communicate their ideas through discussion, annotated sketches, cross-sectional and exploded diagrams, prototypes, pattern pieces, and computer-aided design

Make

- Select from and use a wider range of tools and equipment to perform practical tasks [for example, cutting, shaping, joining, and finishing], accurately
- Select from and use a wider range of materials and components, including construction materials, textiles, and ingredients, according to their functional properties and aesthetic qualities

Evaluate

- Evaluate their ideas and products against their own design criteria and consider the views of others to improve their work

Technical knowledge

- Understand and use electrical systems in their products [for example, series circuits incorporating switches, bulbs, buzzers, and motors]
- Apply their understanding of computing to program, monitor, and control their products

Spring 2

Unit overview

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.

Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Data and information	• I can create a database using cards • I can explain how information can be recorded • I can order, sort, and group my data cards	Teachers will need to know that a flat-file database is a collection of data organised in a single table. The term 'database' means 'a collection of organised data that is stored on a computer'. Databases allow people to search and sort large quantities of data to find information. Data can be letters, words, numbers, dates, images, sounds, etc. In addition, teachers will need to be familiar with the basic structure of a database, and the concept of 'grouping' and 'sorting' data records based on different fields. For example, grouping objects by colour, or sorting into alphabetical order. A database is composed of 'records', which are sets of data on a particular object. Records are formed from one or more 'fields' of data. A field is one specific piece of data in a database record. For example, a record all about a country could have fields such as 'country name' and 'country population'. The value within the record is the 'answer' to each field, e.g. Mexico is the value in the 'country name' field and '126.2 million' is the value in the 'country population' field. Teachers will also need to be aware that all objects have attributes. An attribute includes its 'name' and a 'value'. For example, a ball will have a 'colour', which might be 'red'. 'Colour' is the attribute 'name'; 'red' is the attribute 'value'. In a flat-file database the attribute names become the fields when the data about the object is stored as a record. The values of the attributes become the values that are saved in the database fields. Teachers will need to be familiar with using J2Data sample databases. Support with navigating the databases can be found at http://www.j2e.com/help/videos/datags4. Knowledge of how to carry out a flight search using https://www.expedia.co.uk/Flights, and the ability to screenshot flight details from a web browser would also be beneficial.	To use a form to record information To compare paper and computer-based databases To outline how you can answer questions by grouping and then sorting data To explain that tools can be used to select specific data To explain that computer programs can be used to compare data visually To use a real-world database to answer questions
	• I can explain what a field and a record is in a database • I can navigate a flat-file database to compare different views of information • I can choose which field to sort data by to answer a given question		
	• I can explain that data can be grouped using chosen values • I can group information using a database • I can combine grouping and sorting to answer specific questions		
	• I can choose which field and value are required to answer a given question • I can outline how 'AND' and 'OR' can be used to refine data selection • I can choose multiple criteria to answer a given question		
	• I can select an appropriate chart to visually compare data • I can refine a chart by selecting a particular filter • I can explain the benefits of using a computer to create charts		
	• I can ask questions that will need more than one field to answer • I can refine a search in a real-world context • I can present my findings to a group		

National curriculum links

- Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content
- Select, use, and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems, and content that accomplish given goals, including collecting, analysing, evaluating, and presenting data and information

Summer 1

Unit overview

In this unit, learners 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.

Note: This unit is planned using the Google Drawings application. All Google Drawings files are provided in the lesson plans (the links create a copy of the drawing that can be edited). However, if you wish to use a different vector drawing program, all of the resources are included in the folders as Google Slides presentations. The contents of these files can be copied across to your preferred program.

In order to demonstrate the tools and skills involved in vector drawings, it is recommended that you use a vector drawing program, such as Vectr. While the resources could be used in the Google Slides files, it is important that learners recognise that true vector drawings are made using a vector drawing program.

Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Creating media- Introduction to vector graphics	• I can recognise that vector drawings are made using shapes • I can experiment with the shape and line tools • I can discuss how vector drawings are different from paper-based drawings	Teachers need a good understanding of the tools available in their chosen software. These tools include shape and line drawing tools, fill, undo and redo, select, and delete. In the unit, learners also need to move, resize, and rotate shapes, including the use of size and alignment guides. Many of these tools are available in presentation software such as Google Slides and Microsoft PowerPoint. Teachers need to know that vector drawings are created using shapes and lines. In vector drawings, all these shapes and lines are called objects. Within vector drawings, each object is created using a new layer. These can be rearranged using the menu tool, which allows the objects to be sent backwards and forwards (one at a time) or sent to the back or front. Objects within these programs can be grouped, which enables the objects to be treated as though they are a single object. It is important to recognise that although they act as though they are a single object, grouped objects are still a number of individual objects. Teachers need to have an understanding of how digital images can be made. This could be using shapes and lines in a vector drawing, or using pixels in a freehand paint program.	To identify that drawing tools can be used to produce different outcomes To create a vector drawing by combining shapes To use tools to achieve a desired effect To recognise that vector drawings consist of layers To group objects to make them easier to work with To apply what I have learned about vector drawings
	• I can identify the shapes used to make a vector drawing • I can explain that each element added to a vector drawing is an object • I can move, resize, and rotate objects I have duplicated		
	• I can use the zoom tool to help me add detail to my drawings • I can explain how alignment grids and resize handles can be used to improve consistency • I can modify objects to create a new image		
	• I can identify that each added object creates a new layer in the drawing • I can change the order of layers in a vector drawing • I can use layering to create an image		
	• I can copy part of a drawing by duplicating several objects • I can recognise when I need to group and ungroup objects • I can reuse a group of objects to further develop my vector drawing		
	• I can create a vector drawing for a specific purpose • I can reflect on the skills I have used and why I have used them • I can compare vector drawings to freehand paint drawings		

National curriculum links

- Select, use, and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems, and content that accomplish given goals, including collecting, analysing, evaluating, and presenting data and information.

Summer 2

Unit overview

Learners will develop their knowledge of 'selection' by revisiting how 'conditions' can be used in programming, and then learning how the 'if… then… else...' structure can be used to select different outcomes depending on whether a condition is 'true' or 'false'. They represent this understanding in algorithms, and then by constructing programs in the Scratch programming environment. They learn how to write programs that ask questions and use selection to control the outcomes based on the answers given. They use this knowledge to design a quiz in response to a given task and implement it as a program. To conclude the unit, learners evaluate their program by identifying how it meets the requirements of the task, the ways they have improved it, and further ways it could be improved.

Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Programming B- Selection in quizzes	• I can recall how conditions are used in selection • I can identify conditions in a program • I can modify a condition in a program • I can use selection in an infinite loop to check a condition • I can identify the condition and outcomes in an 'if… then… else…' statement • I can create a program that uses selection to produce different outcomes • I can explain that program flow can branch according to a condition • I can design the flow of a program that contains 'if… then… else…' • I can show that a condition can direct program flow in one of two ways • I can outline a given task • I can use a design format to outline my project • I can identify the outcome of user input in an algorithm • I can implement my algorithm to create the first section of my program • I can test my program • I can share my program with others • I can identify ways the program could be improved • I can identify the setup code I need in my program • I can extend my program further	This unit focuses on developing learners' understanding of selection in an on-screen context. It highlights what 'conditions' are and how they are used as part of 'selection'. This unit also develops learners' understanding of design in programming, using the approach outlined below. **Levels of abstraction** When programming, there are four levels which can help describe a project (known as levels of abstraction). Research suggests that this structure can support learners in understanding how to create a program and how it works: • Task - this is what is needed • Design - this is what it should do • Code - this is how it is done • Running the code - this is what it does Spending time at the 'Task' and 'Design' levels before engaging in code-writing aids learners in assessing the 'do-ability' of their programs and reduces a learner's cognitive load during programming. Learners will move between the different levels throughout the unit and this is highlighted within each lesson plan. **Conditions** 'Conditions' are statements that need to be met for a set of actions to be carried out. They can be used in algorithms and programs to control the flow of actions. When a condition is met it is referred to as 'true' and when it is not met it is referred to as 'false'. You need to be able to identify and use conditions in algorithms in the form of statements to both start and stop sets of action. Additionally, you need to understand that conditions can be used in loops, and when they are, that the set of actions in the loop will be carried out repeatedly until the condition is true. For example, 'until button 'A' is pressed'. **Selection** When designing programs, there are often points where a decision must be made. These decisions are known as 'selection', and are commonly implemented in programming using 'if' statements. Selection is used to	To explain how selection is used in computer programs To relate that a conditional statement connects a condition to an outcome To explain how selection directs the flow of a program To design a program that uses selection To create a program that uses selection To evaluate my program

		control the flow of actions in algorithms and programs by checking whether a condition (see above) has been met. If it has been met, the identified actions will be carried out. When selection is used in programs, infinite loops (see above) are often used to instruct the device to check the condition repeatedly. Without using loops, the condition would only be checked once following the sequence of the code.	

National curriculum links

- design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts
- use sequence, selection, and repetition in programs; work with variables and various forms of input and output
- use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs
- select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information

Year 6 MTP			
Autumn 1			
Unit overview In this unit learners explore how data is transferred over the internet. Learners initially focus on addressing, before they move on to the makeup and structure of data packets. Learners then look at how the internet facilitates online communication and collaboration; they complete shared projects online and evaluate different methods of communication. Finally, they learn how to communicate responsibly by considering what should and should not be shared on the internet. **Note:** Some of the content in this unit was previously included in the Year 5 – 'Computer systems and networks' unit, so some learners may have already completed similar activities. Where this is the case, the context for the activity has been changed.			
Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Computing systems and networks- communication and collaboration	• I can recognise that data is transferred using agreed methods • I can explain that internet devices have addresses • I can describe how computers use addresses to access websites	In this unit, you will need to have an understanding of the way data is sent over the internet. Some key terms you will need to be familiar with are Internet Protocol (IP) addresses; Domain Name Server (DNS); and data packets, including the main parts of a packet (header and data payload). The terms are discussed in more detail within the lesson plans. Part of this unit focuses on collaboration over the internet. These activities assume the use of Google Slides, a free web-based app, which is part of Google Workspace for Education. You will need a school Google account to access this. If your school doesn't have Google Workspace, you can sign up for a free account. Alternative collaborative presentation tools are available, such as Microsoft PowerPoint used with Office 365. However, if you wish to use different software, the lesson slides will need to be adapted to suit any changes. We recommend the use of teacher accounts in Scratch for certain activities within this unit. For guidance on setting up teacher accounts, please visit the Scratch website. (https://scratch.mit.edu/educators/faq). It is possible for learners to make changes without 'remixing' the activities, however these changes will not be saved.	To explain the importance of internet addresses To recognise how data is transferred across the internet To explain how sharing information online can help people to work together To evaluate different ways of working together online To recognise how we communicate using technology To evaluate different methods of online communication
	• I can identify and explain the main parts of a data packet • I can explain that data is transferred over networks in packets • I can explain that all data transferred over the internet is in packets		
	• I can recognise how to access shared files stored online • I can send information over the internet in different ways • I can explain that the internet allows different media to be shared		
	• I can identify different ways of working together online • I can recognise that working together on the internet can be public or private • I can explain how the internet enables effective collaboration		
	• I can explain the different ways in which people communicate • I can identify that there are a variety of ways to communicate over the internet • I can choose methods of communication to suit particular purposes		

	• I can compare different methods of communicating on the internet • I can decide when I should and should not share information online • I can explain that communication on the internet may not be private		

National curriculum links:

- Understand computer networks, including the internet; how they can provide multiple services, such as the World Wide Web, and the opportunities they offer for communication and collaboration
- Select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information
- Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact

Autumn 2

Unit overview

Learners will be introduced to creating websites for a chosen purpose. Learners identify what makes a good web page and use this information to design and evaluate their own website using Google Sites. Throughout the process, learners pay specific attention to copyright and fair use of media, the aesthetics of the site, and navigation paths.

It is recommended that learners use laptop or desktop computers for this unit of work. The unit has been based on the use of Google Sites, which is free to use with any Google account. If your school uses the free Google Workspace for Education, your Google administrator can create accounts for pupils and also ensure that the Google Sites feature is enabled. If you don't have a school Google Workspace account, your school may choose to set one up or you may opt to create individual Google accounts for your learners to use. Whichever option you choose, it should be in line with your school's policies.

Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Creating media- Web page creation	• I can explore a website • I can discuss the different types of media used on websites • I know that websites are written in HTML	You will need to ensure that you and learners have access to Google Sites. Experience with using Google Sites would support you in delivering this unit, but the unit will also deliver support within the slides. Familiarity with the implications of linking to other people's work online is needed, and an understanding of the terms 'fair use' and 'copyright' is important. You should be aware of your school's procedures related to children searching for images and how to report any issues. You will need to be able to access websites and have some understanding of HTML and the differences between browsers,	To review an existing website and consider its structure To plan the features of a web page To consider the ownership and use of images (copyright) To recognise the need to preview pages To outline the need for a navigation path
	• I can recognise the common features of a web page • I can suggest media to include on my page • I can draw a web page layout that suits my purpose		
	• I can say why I should use copyright-free images • I can find copyright-free images • I can describe what is meant by the term 'fair use'		

	• I can add content to my own web page • I can preview what my web page looks like • I can evaluate what my web page looks like on different devices and suggest/make edits.	websites, and web pages. You should also have an understanding of the terms 'breadcrumb trail' and 'navigation', and how websites are generally structured.	To recognise the implications of linking to content owned by other people
	• I can explain what a navigation path is • I can describe why navigation paths are useful • I can make multiple web pages and link them using hyperlinks		
	• I can explain the implication of linking to content owned by others • I can create hyperlinks to link to other people's work • I can evaluate the user experience of a website		

National curriculum links

- Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content
- Select, use, and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems, and content that accomplish given goals, including collecting, analysing, evaluating, and presenting data and information.
- use technology safely, respectfully, and responsibly; recognise acceptable/unacceptable behaviour.

English links

- Writing composition: Identifying the audience for and purpose of the writing, selecting the appropriate form, and using other similar writing as models for their own.

Spring 1

Unit overview

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.

Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Programming A-Variables in games	• I can identify examples of information that is variable • I can explain that the way a variable changes can be defined • I can identify that variables can hold numbers or letters	This unit focuses on developing learners' understanding of variables in Scratch, a block-based programming language. It emphasises where variables can be used and how they can be set and changed through the running of a program. This unit also develops learners' understanding of design in programming, using the approach outlined below. When programming, there are four levels that can help describe a project (known as 'levels of abstraction'). Research suggests that this structure can support learners in understanding how to create a program and how it works: • Task — what is needed • Design — what it should do • Code — how it is done	To define a 'variable' as something that is changeable To explain why a variable is used in a program To choose how to improve a game by using variables To design a project that builds on a given example
	• I can identify a program variable as a placeholder in memory for a single value • I can explain that a variable has a name and a value • I can recognise that the value of a variable can be changed		

	• I can decide where in a program to change a variable • I can make use of an event in a program to set a variable • I can recognise that the value of a variable can be used by a program	• Running the code — what it does Spending time at the 'task' and 'design' levels before engaging in writing code can aid learners in assessing the 'do-ability' of their programs. It also reduces the cognitive load for learners during programming. Learners will move between the different levels throughout the unit, and this is recognised within each lesson plan. During this unit, learners are required to save their work in Scratch. We recommend the use of teacher and pupil accounts to manage this process. You can find detailed guidance on setting up and managing accounts in Scratch on the Scratch website (scratch.mit.edu/educators/faq).	To use my design to create a project To evaluate my project
	• I can choose the artwork for my project • I can create algorithms for my project • I can explain my design choices		
	• I can create the artwork for my project • I can choose a name that identifies the role of a variable • I can test the code that I have written		
	• I can identify ways that my game could be improved • I can use variables to extend my game • I can share my game with others		

National curriculum links- Computing

- Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts
- Use sequence, selection, and repetition in programs; work with variables and various forms of input and output
- Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs
- Select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information

Spring 2

Unit overview

This unit introduces the learners to spreadsheets. They will be supported in organising data into columns and rows to create their own data set. Learners will be taught the importance of formatting data to support calculations, while also being introduced to formulas and will begin to understand how they can be used to produce calculated data. Learners will be taught how to apply formulas that include a range of cells, and apply formulas to multiple cells by duplicating them. Learners will use spreadsheets to plan an event and answer questions. Finally, learners will create charts, and evaluate their results in comparison to questions asked.

It is recommended that you use laptop or desktop computers which have access to a spreadsheet application. The screenshots and videos in this unit are based on Google Slides, however the unit can be adapted for other applications.

Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Data and information-	• I can collect data • I can suggest how to structure my data • I can enter data into a spreadsheet	It would be beneficial for teachers to have an understanding of a spreadsheet application e.g. 'Google Sheets' or alternative software such as 'Microsoft Excel' or 'Purple Mash – 2Calculate'.	To create a data set in a spreadsheet To build a data set in a spreadsheet

Introduction to Spreadsheets	• I can explain what an item of data is • I can choose an appropriate format for a cell • I can apply an appropriate format to a cell	An understanding that data can be words, numbers, dates, images, sounds, etc. without context is important. Just as words need to be in a sentence to give them meaning, data items need to be part of a structure. For example, the number 6 isn't data unless it's part of a larger structure, such as included in a spreadsheet with data headings. Understanding that a data set is a collection of related data that can be modified using a computer is helpful, as learners will be creating their own data sets throughout the unit. Knowledge of why data headings are important and an understanding of how data is organised in columns and rows would be beneficial. Organising data is an important aspect of data and information. It supports the use of calculations and provides the opportunity to use sorting and filtering, which enables ease of use and reduces human error. This unit focuses on the learners applying number formats to alter cells. It is important to understand that this type of formatting changes how a spreadsheet interacts with the data and is different to applying style formatting (bold, italics, etc.), which only changes the appearance of data. In Lesson 5 of this unit, learners have been provided with the mathematical calculations they need to complete the activities in the unit, the calculations can be found in the 'Data calculations' handout. It is important that learners are given the opportunity to demonstrate their ability to use the computational skills required, regardless of their mathematical ability.	To explain that formulas can be used to produce calculated data To apply formulas to data To create a spreadsheet to plan an event To choose suitable ways to present data
	• I can explain which data types can be used in calculations • I can construct a formula in a spreadsheet • I can identify that changing inputs changes outputs		
	• I can calculate data using different operations • I can create a formula which includes a range of cells • I can apply a formula to multiple cells by duplicating it		
	• I can use a spreadsheet to answer questions • I can explain why data should be organised • I can apply a formula to calculate the data I need to answer questions		
	• I can produce a chart • I can use a chart to show the answer to a question • I can suggest when to use a table or chart		

National curriculum links

- Select, use, and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems, and content that accomplish given goals, including collecting, analysing, evaluating, and presenting data and information

National curriculum maths links

Number – addition, subtraction, multiplication, and division:

- Solve problems involving addition, subtraction, multiplication, and division

Statistics:

- Interpret and construct pie charts and line graphs, and use these to solve problems
- Calculate and interpret the mean as an average

Summer 1

Unit overview

Learners 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.
For this sequence of lessons, learners will be using Tinkercad (https://www.tinkercad.com), a web-based 3D modelling application. Learners will need accounts to save their work and access the resources. We recommend signing up for a teacher account at https://www.tinkercad.com/join, which enables learner accounts to be created and the website accessed with a class code. For guidance on setting up your class, please visit https://www.tinkercad.com/teach. Please ensure your school's online safety policy is considered when creating accounts.

Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Creating media- 3D modelling	• I can add 3D shapes to a project • I can view 3D shapes from different perspectives • I can move 3D shapes relative to one another	Teachers will need to be familiar with the main concepts associated with 3D modelling. During the unit the following skills and concepts are introduced: • Working with and viewing shapes in three dimensions • Adding, resizing, and moving (including lifting and lowering) 3D shapes • Rotating 3D shapes • Combining 3D shapes • Grouping and ungrouping objects	To recognise that you can work in three dimensions on a computer To identify that digital 3D objects can be modified To recognise that objects can be combined in a 3D model To create a 3D model for a given purpose To plan my own 3D model To create my own digital 3D model
	• I can resize an object in three dimensions • I can lift/lower 3D objects • I can recolour a 3D object		
	• I can rotate objects in three dimensions • I can duplicate 3D objects • I can group 3D objects		
	• I can accurately size 3D objects • I can show that placeholders can create holes in 3D objects • I can combine a number of 3D objects		
	• I can analyse a 3D model • I can choose objects to use in a 3D model • I can combine objects in a design		
	• I can construct a 3D model based on a design • I can explain how my 3D model could be improved • I can modify my 3D model to improve it		

National curriculum links

Computing – KS2

- Select, use, and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems, and content that accomplish given goals, including collecting, analysing, evaluating, and presenting data and information
- Use technology safely, respectfully, and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact

Art and design – KS2

- To improve their mastery of art and design techniques, including drawing, painting, and sculpture with a range of materials

Design and technology – KS2

- Generate, develop, model, and communicate their ideas through discussion, annotated sketches, cross-sectional and exploded diagrams, prototypes, pattern pieces and computer-aided design

Mathematics – KS2 (Y6)

- Recognise, describe, and build simple 3D shapes, including making nets

Summer 2

Unit overview

This unit is the final KS2 programming unit and brings together elements of all the four programming constructs: sequence from Year 3, repetition from Year 4, selection from Year 5, and variables (introduced in Year 6 – 'Programming A'. It offers pupils the opportunity to use all of these constructs in a different, but still familiar environment, while also utilising a physical device — the micro:bit. The unit begins with a simple program for pupils to build in and test within the new programming environment, before transferring it to their micro:bit. Pupils then take on three new projects in Lessons 2, 3, and 4, with each lesson adding more depth.

Design features prominently in this unit. A design template is introduced in Lesson 3, initially scaffolded to give pupils the opportunity to create code from a given design. In Lesson 4 that scaffolding is gradually reduced, then in Lesson 5, pupils create their own design, using the same template. In the final lesson, pupils will apply their knowledge of the programming constructs and use their design to create their own micro:bit-based step counter.

Unit	Learning objectives	Teachers subject knowledge	Component knowledge
Programming B-Sensing movement	• I can apply my knowledge of programming to a new environment • I can test my program on an emulator • I can transfer my program to a controllable device • I can identify examples of conditions in the real world • I can use a variable in an if, then, else statement to select the flow of a program • I can determine the flow of a program using selection • I can use a condition to change a variable • I can experiment with different physical inputs • I can explain that checking a variable doesn't change its value • I can use an operand (e.g. <>=) in an if, then statement • I can explain the importance of the order of conditions in else, if statements • I can modify a program to achieve a different outcome • I can decide what variables to include in a project • I can design the algorithm for my project • I can design the program flow for my project • I can create a program based on my design • I can test my program against my design • I can use a range of approaches to find and fix bugs	This unit focuses on developing pupils' understanding of variables in a different programming environment and using a physical device. It also enables pupils to combine their knowledge and understanding of programming constructs introduced in previous years. This unit continues to advance pupils' understanding of design in programming, using the approach outlined below. When programming, there are four levels that can help describe a project (known as 'levels of abstraction'). Research suggests that this structure can support pupils in understanding how to create a program and how it works: • Task — what is needed • Design — what it should do • Code — how it is done • Running the code — what it does Spending time at the 'task' and 'design' levels before engaging in writing code can aid pupils in assessing the 'do-ability' of their programs. It also reduces the cognitive load for pupils during programming. Pupils will move between the different levels throughout the unit, and this is highlighted within each lesson plan: • Lesson 3 - pupils work at the 'code' and 'running the code' levels from a given design • Lesson 4 - pupils move from 'design' to 'code', to 'running the code' with some scaffolding • Lesson 5 - pupils work at the 'design' level with increasing independence • Lesson 6 - pupils work at the 'code' and 'running the code' levels, using their own design	To create a program to run on a controllable device To explain that selection can control the flow of a program To update a variable with a user input To use an conditional statement to compare a variable to a value To design a project that uses inputs and outputs on a controllable device To develop a program to use inputs and outputs on a controllable device

National curriculum links

- Design, write, and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts
- Use sequence, selection, and repetition in programs; work with variables and various forms of input and output
- Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs
- Select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information

Appendix 1- Pedagogy

Computing is a broad discipline, and computing teachers require a range of strategies to deliver effective lessons to their pupils. These 12 key principles are underpinned by research; each principle has been shown to contribute to effective teaching and learning in computing.

Lead with concepts

Support pupils in the acquisition of knowledge, through the use of key concepts, terms and vocabulary, providing opportunities to build a shared and consistent understanding. Glossaries, concept maps, and displays, along with regular recall and revision, can support this approach.

Work together

Encourage collaboration, specifically using pair programming and peer instruction, and also structured group tasks. Working together stimulates classroom dialogue, articulation of concepts, and development of shared understanding.

Get hands-on

Use physical computing and making activities that offer tactile and sensory experiences to enhance learning. Combining electronics and programming with arts and crafts (especially through exploratory projects) provides pupils with creative, engaging context to explore and apply computing concepts.

Unplug, unpack, repack

Teach new concepts by first unpacking complex terms and ideas, exploring these ideas in unplugged and familiar contexts, then repacking this new understanding into the original concept. This approach, called ‘semantic waves’, can help pupils develop a secure understanding of complex concepts.

Model everything

Model processes or practices- everything from debugging code to binary number conversations- using techniques such as worked examples and live coding. Modelling is particularly beneficial to novices, providing scaffolding that can be gradually taken away.

Foster program comprehension

Use a variety of activities to consolidate knowledge and understanding of the function and structure of programs, including debugging, tracing, and Parson’s Problems. Regular

comprehension activities will help secure understanding and build connections with new knowledge.

Create projects

Use project-based learning activities to provide pupils with the opportunity to apply and consolidate their knowledge and understanding. Design is an important, often overlooked aspect of computing. Pupils can consider how to develop an artefact for a particular user or function, and evaluate it against a set of criteria.

Add variety

Provide activities with different levels of direction, scaffolding, and support that promote learning, ranging from highly structured to more exploratory tasks. Adapting your instruction to suit different objectives will help keep all pupils encourage greater independence.

Challenge misconceptions

Use formative questioning to uncover misconceptions and adapt teaching to address them as the occur. Awareness of common misconceptions alongside discussion, concept mapping, peer instruction, or simple quizzes can help identify areas of confusion.

Make concrete

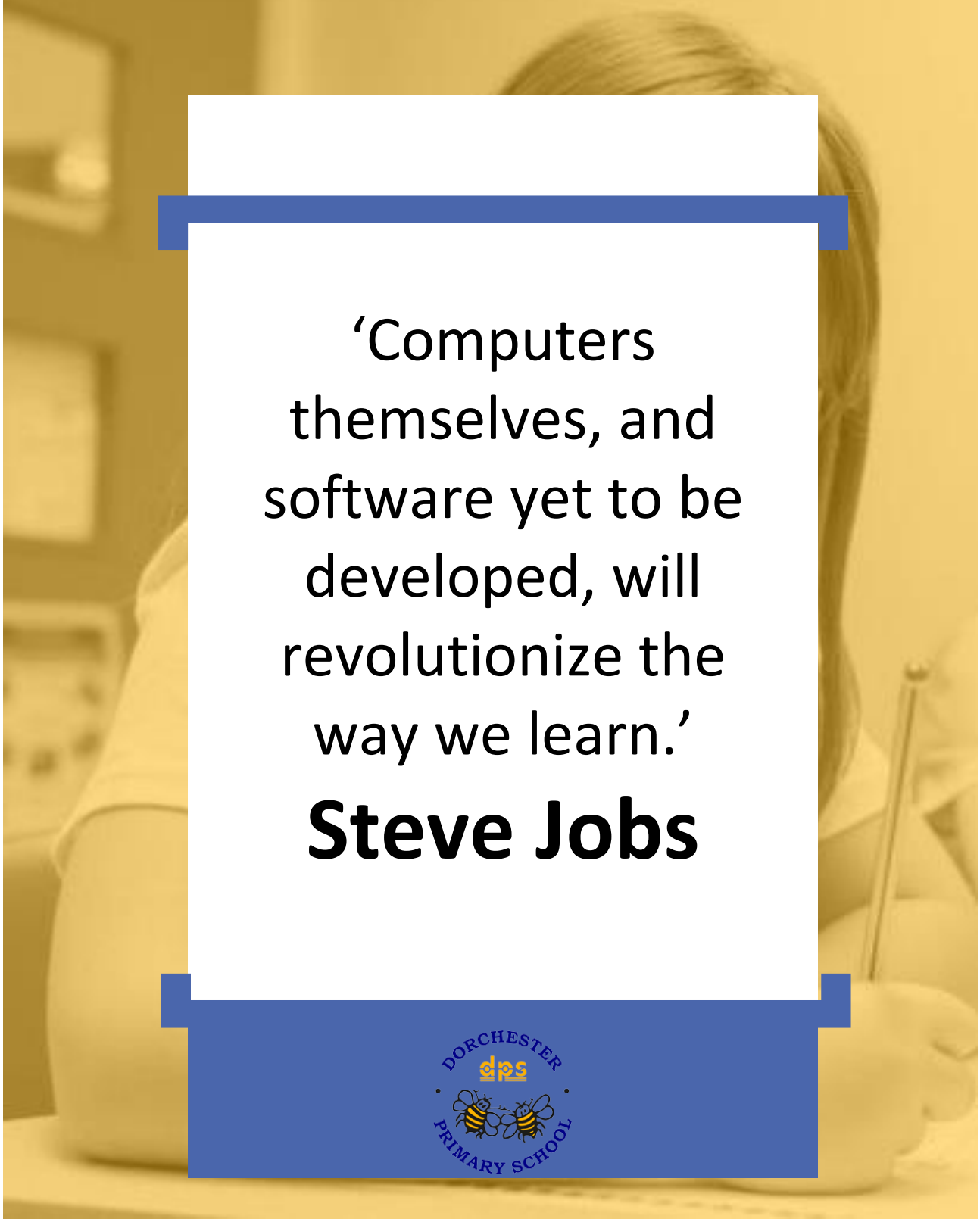
Bring abstract concepts to life with real-world, contextual examples, and a focus on interdependencies with other curriculum subjects. This can be achieved through the use of unplugged activities, proposing analogies, storytelling around concepts, and finding examples of the concepts in pupils' lives.

Structure lessons

Use supportive frameworks when planning lessons, such as PRIMM (predict, run, investigate, modify, make) and (use-modify-create). These frameworks are based on research and ensure that differentiation can be built in at various stages of the lesson.

Read and explore code first

When teaching programming, focus first on code 'reading' activities, before code writing. With both block-based and text-based programming, encourage pupils review and interpret blocks of code. Research has shown that being able to read, trace and explain code augments pupils' ability to write code.



‘Computers themselves, and software yet to be developed, will revolutionize the way we learn.’

Steve Jobs