Computing Curriculum
Year 5 and 6 – Cycle A

Purpose of study

A high-quality computing education equips pupils to use computational thinking and creativity to understand and change the world. Computing has deep links with mathematics, science, and design and technology, and provides insights into both natural and artificial systems. The core of computing is computer science, in which pupils are taught the principles of information and computation, how digital systems work, and how to put this knowledge to use through programming. Building on this knowledge and understanding, pupils are equipped to use information technology to create programs, systems and a range of content. Computing also ensures that pupils become digitally literate – able to use, and express themselves and develop their ideas through, information and communication technology – at a level suitable for the future workplace and as active participants in a digital world.

Aims

The national curriculum for computing aims to ensure that all pupils:

♣ can understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation

♣ can analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems

♣ can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems

♣ are responsible, competent, confident and creative users of information and communication technology.

Attainment targets

By the end of each key stage, pupils are expected to know, apply and understand the matters, skills and processes specified in the relevant programme of study. Schools are not required by law to teach the example content in [square brackets].

Key stage 1 Pupils should be taught to:

♣ 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

♣ 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.

Key stage 2 Pupils should be taught to:

♣ 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

♣ 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.

	Autumn		Spring		Summer	
	Non- Negotiables C13: Collaborate with others online on sites approved and moderated by teachers. C15: Understand and demonstrate knowledge that it is illegal to download copyrighted material, including music or games, without express written permission, from the copyright holder. C16: Understand the effect of online comments and show responsibility and sensitivity when online. C18: Choose the most suitable applications and devices for the purposes of communication. C19: Use many of the advanced features in order to create high quality, professional or efficient communications.					
	C17: Understand how simple networks are set up and used.	C2: Change the position of objects between screen layers (send to back, bring to front) .C15: Understand and demonstrate knowledge that it is illegal to download copyrighted material, including music or games, without express written permission, from the copyright holder. C4: Combine the use of pens with movement to create interesting effects.	C16: Understand the effect of online comments and show responsibility and sensitivity when online. C14: Give examples of the risks of online communities and demonstrate knowledge of how to minimise risk and report problems.	C20: Select appropriate applications to devise, construct and manipulate data and present it in an effective and professional manner.	C1: Set IF conditions for movements. Specify types of rotation giving the number of degrees. C2: Change the position of objects between screen layers (send to back, bring to front) C6: Use IF THEN ELSE conditions to control events or objects.	C3: Upload sounds from a file and edit them. Add effects such as fade in and out and control their implementation. C5: Set events to control other events by 'broadcasting' information as a trigger. C9: Use the Boolean operators () < () , () = (), () > (), ()and(), ()or(), Not() to define conditions. C10: Use the Reporter operators () + (), () - (), () * (), () / () to perform calculations. C11: Pick Random () to (), Join () (), Letter () of (), Length of (), () Mod () C12: This reports the remainder after a division calculation Round () () of ().

Topic	Computing Networks – Sharing Information	Creating Media - Vector Drawing	Creating Media – Video Editing	Data and Information – Flat Field Data bases	Programming A – Selecting in physical computing	Programming B – Selection in quizzes
Resources	Laptops, internet access, iPads, lesson slides,	Laptops, internet access, iPads, GoogleDrawings, Microsoft Publisher/PowerPoint, lesson slides,	Laptops, internet access, iPads, lesson slides,	Laptops, internet access, iPads, lesson slides, Support with navigating the databases can be found at http://www.j2e.com/help/videos/datags4. how to carry out a flight search using https://www.expedia.co.uk/Flights,	Laptops, internet access, iPads, lesson slides, redfernelectronics.co.uk/crumble-software. The unit has been designed to make use of the components provided in the Crumble starter kit, which are as follows: 1 Crumble controller, 12 crocodile leads, 2 Sparkles (A Sparkle is an RGB LED — red, green, blue light-emitting diode. The D connector allows the Crumble to use an electronic signal to control the Sparkle. The signal sets the colour and brightness of the LED.) 1 push switch suitable for Crumble, 1 light sensor suitable for Crumble, 1 buzzer suitable for Crumble, 1 micro USB cable, 1 switched battery box suitable for Crumble, Motors,	Laptops, internet access, iPads, lesson slides, visit the Scratch website (scratch.mit.edu/educators/faq).
Vocabulary	Input, output, processes,	Vector, resize, reshape, zoom	Storyboard, volume, camera lens, zoom, pan, angle, theme, setting, characters, colour, sound, and dialogue. store, retrieve, and export	Database, record, field, sorting, grouping, AND, OR, filter,	microcontroller (Crumble controller) output devices — LEDs and motors, input device – push switch, algorithms, count controlled loop,	Algorithms, if, then, else, binary question, code, infinite loops
Lesson 1	**Year 5/6:** To explain that computers can be connected together to form systems **Activities:** This lesson introduces learners to the concept of a system. Learners will develop their understanding of components working together to make a whole. They will outline how digital systems might work and the physical and electronic connections that exist. **Outcomes:** **Year 5/6:** I can explain that systems are built using a number of parts I can describe that a computer system features inputs, processes, and outputs I can explain that computer systems communicate with other devices	**Year 5/6:** To identify that drawing tools can be used to produce different outcomes **Activities:** In this lesson learners will be introduced to vector drawings and begin to have an understanding that they are made up of simple shapes and lines. Learners will use the main drawing tools within a software package. This unit is written assuming the use of Google Drawings (docs.google.com/drawings/) but other packages such as Microsoft Publisher, or Microsoft PowerPoint can be used if preferred. Learners will discuss how vector drawings differ from paper-based drawings. **Outcomes:** **Year 5/ 6:** I can recognise that vector drawings are made using shapes I can identify the main drawing tools I can discuss how a vector drawing is different from paper-based drawings	**Year 5/6:** To recognise video as moving pictures, which can include audio **Activities:** In this lesson, learners have the opportunity to explore a brief history of moving images and video. Through the lesson, they learn that the purpose of recorded video is to engage the audience and share a message. Learners explore the benefits of adding audio to a video and, in groups, begin to develop ideas for their own video project. **Outcomes:** **Year 5/6:** I can explain that a video can include both visual and audio media I can explain the benefits of adding audio to a video I can plan a video project using a storyboard	**Year 5/6:** To use a form to record information **Activities:** In the first lesson, pupils create a paper version of a record card database. Using a card template, they create a data set, with each pupil creating eight to ten cards linked to a theme, eg animals. They complete records for each of the animals in their database and then physically sort the cards to answer questions about the data. **Outcomes:** **Year 5/6:** I can create multiple questions about the same field I can explain how information can be recorded I can order, sort, and group my data cards	**Year 5/6:** To control a simple circuit connected to a computer **Activities:** In this lesson, learners will become familiar with the Crumble controller, some of its associated components, and the programming environment used to control it. They will explore how the items connect together to create a complete circuit, and how to construct programs that turn an LED on and off and set its colour. Learners will apply their understanding of repetition by identifying how their programs can be modified to make an LED flash continuously. **Outcomes:** **Year 5/6:** I can build a simple circuit to connect a microcontroller to a computer I can program a microcontroller to light an LED I can explain why I used an infinite loop	**Year 5/6:** To explain how selection is used in computer programs **Activities:** In this lesson, learners revisit previous learning on 'selection' and identify how 'conditions' are used to control the flow of actions in a program. They are introduced to the blocks for using conditions in programs using the Scratch programming environment. They modify the conditions in an existing program and identify the impact this has. **Outcomes:** **Year 5/6:** I can recall how conditions are used in selection I can identify conditions in a program I can modify a condition in a program
Lesson 2	**Year 5:** To recognise the role of computer systems in our lives **Year 6:** **Activities:** In this lesson, learners will consider how larger computer systems work. Learners will consider how devices and processes are connected. They will	**Year 5/ 6:** To create a vector drawing by combining shapes **Activities:** During this lesson learners will begin to identify the shapes that are used to make vector drawings. They will be able to explain that each element of a vector drawing is called an **object**. Learners will create their own vector	**Year 5/6:** To identify digital devices that can record video **Activities:** This lesson provides learners with opportunities to explore devices and apps that record audio and video. Opportunities are included for learners to investigate the pros and cons of audio devices such as dictation machines or mobile sound recorders versus fully integrated AV (audiovisual) devices.	**Year 5/6:** To compare paper and computer-based databases **Activities:** In this lesson, pupils use a computer-based database to examine how data can be recorded and viewed. They learn that a database consists of 'records', and that each record contains 'fields'. In addition, they will order records in different ways and compare this	**Year 5/6:** To write a program that includes count-controlled loops **Activities:** In this lesson, learners will develop their knowledge of a Crumble controller further by connecting additional devices (another Sparkle and a motor) to the controller, and they will construct programs to control more	**Year 5/6:** To relate that a conditional statement connects a condition to an outcome **Activities:** In this lesson, learners will develop their understanding of selection by using the 'if… then… else...' structure in algorithms and programs. They will revisit the need to use repetition in selection to ensure that conditions are repeatedly checked. They identify the two outcomes in given programs

	also reflect on how computer systems can help us. **Outcomes:** **Year 5:** 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 **Year 6:**	drawing by moving, resizing, rotating, and changing the colours of a selection of objects. They will also learn how to duplicate the objects to save time. **Outcomes:** **Year 5/6:** 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	Learners can explore devices and locate working features such as the on/off button, record button (start/stop), volume, camera lens, and zoom. Opportunities are provided to develop their project through the storyboard and script. **Outcomes:** **Year 5/6:** I can identify and name digital devices that can record video and sound I can choose the most suitable digital device for recording my project I can locate and identify the working features of a digital device that can record video	database to the paper database they created in lesson 1. **Outcomes:** **Year 5/6:** I can navigate a flat-file database to compare different views of information I can explain what a 'field' and a 'record' is in a database I can choose which field to sort data by to answer a given question	than one of these. They will design sequences of actions for these output devices. They will then apply their understanding of repetition by using count-controlled loops when implementing their design as a program. **Outcomes:** **Year 5/6:** I can connect more than one output device to a microcontroller I can design sequences for given output devices I can decide which output devices I control with a count-controlled loop	and how the condition informs which outcome will be selected. Learners use this knowledge to write their own programs that use selection with two outcomes. **Outcomes:** **Year 5/6:**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 with different outcomes using selection
Lesson 3	**Year 5:** To recognise how information is transferred over the internet **Year 6:** **Activities:** This lesson introduces the idea that parts of a computer system are not always in the same place or country. Instead, those parts of a system must transfer information using the internet. This lesson builds on the introduction to the internet in the Year 4 'What is the internet?' unit, adding awareness of IP addresses and the rules (protocols) that computers have for communicating with one another. **Outcomes:** **Year 5:** I can recognise that data is transferred using agreed methods I can explain that networked digital devices have unique addresses I can explain that data is transferred over networks in packets **Year 6:**	**Year 5/6:** To use tools to achieve a desired effect **Activities:** During this lesson learners will continue to increase the complexity of their vector drawings by using the zoom tool to help them add detail. They will begin to understand how grids and resize handles can be used to improve consistency in their drawings and use tools to modify objects, creating different effects. **Outcomes:** **Year 5/6:** 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 different effects	**Year 5/6:** To capture video using a digital device **Activities:** Learners explore devices and apps, becoming familiar with the devices, functions, and apps. Working collaboratively, they begin to record their video content, considering the use of zoom, angle, and movement (pan). **Outcomes:** **Year 5/6:**I can select a suitable device and software to capture my video I can demonstrate suitable methods of using a digital device to capture my video I can demonstrate the safe use and handling of devices	**Year 5/6:** To outline how grouping and then sorting data allows us to answer questions **Activities:** In this lesson, pupils investigate how records can be grouped, using both the paper record cards created in lesson 1 and a computer based database from J2E. They use 'grouping' and 'sorting' to answer questions about the data. **Outcomes:** **Year 5/6:** I can explain how information can be grouped I can group information to answer questions I can combine grouping and sorting to answer more specific questions	**Year 5/6:** To explain that a loop can stop when a condition is met, eg number of times **Activities:** In this lesson, learners will be introduced to conditions, and how they can be used in algorithms and programs to control their flow. They will identify conditions in statements, stating if they are true or false, and learn how they can be used to start and stop a set of actions. Learners will be introduced to a Crumble switch, and learn how it can provide the Crumble controller with an input that can be used as a condition. They will explore how to write programs that use an input as a condition, and use this knowledge to write a program that uses a condition to stop a repeating light pattern. **Outcomes:** **Year 5/6:** I can explain that a condition is something that can be either true or false (eg whether a value is more than 10, or whether a button has been pressed) I can experiment with a 'do until' loop I can program a microcontroller to respond to an input	**Year 5/6:** To explain how selection directs the flow of a program **Activities:** In this lesson, learners consider how the 'if… then... else…' structure can be used to identify two responses to a binary question (one with a 'yes or no' answer). They identify that the answer to the question is the 'condition', and use algorithms with a branching structure to represent the actions that will be carried out if the condition is true or false. They learn how questions can be asked in Scratch, and how the answer, supplied by the user, is used in the condition to control the outcomes. They use an algorithm to design a program that uses selection to direct the flow of the program based on the answer provided. They implement their algorithm as a program and test whether both outcomes can be achieved. **Outcomes:** **Year 5/6:** I can explain that program flow can branch according to a condition I can design the flow of a program which contains 'if… then… else…' I can show that a condition can direct program flow in one of two ways
Lesson 4	**Year 5:** To explain how sharing information online lets people in different places work together **Year 6:** **Activities:** In this lesson, learners will consider how people can work together when they are not in the same location. They will discuss ways of working and	**Year 5/6:** To recognise that vector drawings consist of layers **Activities:** During this lesson learners will gain an understanding of layers and how they are xused in vector drawings. They will learn that each object is built on a new layer and that these layers can be	**Year 5/6:** To recognise the features of an effective video **Activities:** This lesson provides learners with opportunities to investigate further the features of an effective video, including the use of theme, setting, characters, colour, sound, and dialogue. They learn to apply their knowledge as	**Year 5/6:** To explain that tools can be used to select specific data **Activities:** In this lesson, pupils develop their search techniques to answer questions about the data. They use advanced techniques to search for more than one field, and practise doing this through both unplugged methods (without using computers), and using a	**Year 5/6:** To conclude that a loop can be used to repeatedly check whether a condition has been met **Activities:** In this lesson, learners will develop their understanding of how the flow of actions in algorithms and programs can be controlled by conditions. They will be introduced to selection, and learn to represent conditions and actions using the	**Year 5/6:** To design a program which uses selection **Activities:** In this lesson, learners will be provided with a task: to use selection to control the outcomes in an interactive quiz. They will outline the requirements of the task and use an algorithm to show how they will use selection in the quiz to control the outcomes based on the answer given. Learners

	start a collaborative online project. The online activity assumes that learners can make simple slides including text and images. If your learners are unsure how to do this, you may wish to spend some time on the Year 3 'Desktop publishing' unit before this lesson. **Outcomes:** **Year 5/6:** I can recognise that connected digital devices can allow us 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	moved forward and backward to create effective vector drawings. **Outcomes:** **Year 5/ 6:** I can identify that each added object creates a new layer in the drawing I can identify which objects are in the front layer or in the back layer of a drawing I can change the order of layers in a vector drawing	they record their video content in their groups. **Outcomes:** **Year 5/6:** I can list some of the features of an effective video I can record a video that demonstrates some of the features of an effective video I can explain why lighting and angle are important in creating an effective video	computer database. **Outcomes:** **Year 5/ 6:** 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	'if… then…' structure. They will apply their understanding by using selection in an algorithm created to meet the requirements of a task. They will discover that infinite repetition is required when programming input devices to repeatedly check if a condition has been met. **Outcomes:** **Year 5/6:** 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	will complete their designs by using storyboards to identify the questions that will be asked, and the outcomes for both correct and incorrect answers. To demonstrate their understanding of how they are using selection to control the flow of the program, learners will identify which outcomes will be selected based on given responses. **Outcomes:** **Year 5/6:** 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
Lesson 5	**Year 5/6** To contribute to a shared project online **Activities:** In this lesson, learners will reflect on how they worked together in the previous lesson and how their working together might be improved. Learners will work together on an unplugged activity and use that experience to develop their own ideas of good collective working practices. **Outcomes:** **Year 5/6:** I can suggest strategies to ensure successful group work I can make thoughtful suggestions on my group's work I can compare working online with working offline	**Year 5/6:** To group objects to make them easier to work with **Activities:** During this lesson learners will be taught how to duplicate multiple objects. They will learn how to group objects to make them easier to work with, how to copy and paste these images, and then make simple alterations. **Outcomes:** **Year 5/6:** I can copy part of a drawing by duplicating several objects I can group to create a single object I can reuse a group of objects to further develop my vector drawing	**Year 5/6:** To identify that video can be improved through reshooting and editing **Activities:** This lesson focuses on the technical aspects of exporting video to a computer. It guides learners through the process of making edits to their video, including choosing the best recording, clipping videos, and adding transition effects. It provides learners with opportunities to add images and overlay text. This lesson may be broken down into two smaller parts depending on the available time and the location of the computers that will be used for importing content. **Outcomes:** **Year 5/ 6:** 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	**Year 5/6:** To explain that computer programs can be used to compare data visually **Activities:** In this lesson, pupils consider what makes a useful chart, and how charts can be used to compare data. They create charts from their data in order to answer questions about it. **Outcomes:** **Year 5/6:** 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 graphs	**Year 5/6:** To design a physical project that includes selection **Activities:** In this lesson, learners will apply their understanding of microcontrollers, output devices, and selection when designing a project to meet the requirements of a given task. To ensure their understanding, they will identify how selection might be used in real-world situations, then they will consider how they can apply this knowledge when designing their project. They will produce detailed drawings to show how their model will be made and how they will connect the microcontroller to its components. **Outcomes:** **Year 5/6:** I can identify a condition to start an action (real world) I can describe what my project will do (the task) I can create a detailed drawing of my project	**Year 5/6:** To create a program which uses selection **Activities:** In this lesson, learners will use the Scratch programming environment to implement the first section of their algorithm as a program. They will run the first section of their program to test whether they have correctly used selection to control the outcomes, and debug their program if required. They will then continue implementing their algorithm as a program. Once completed, they will consider the value of sharing their program with others so that they can receive feedback. Learners conclude the lesson by using another learner's quiz and providing feedback on it. **Outcomes:** **Year 5/6:** 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
Lesson 6	**Year 5:** To evaluate different ways of working together online **Year 6:** **Activities:** In the previous two lessons, learners worked together online on a shared project. This lesson introduces another approach to online working: reusing and modifying work done by someone else. (Using someone else's work needs to be done within the	**Year 5/6:** To evaluate my vector drawing **Activities:** During this lesson learners will understand how digital images can be made from shapes or pixels. They will suggest and implement improvements to vector drawings and complete the unit by creating their own labels for the classroom using the skills they have learned. **Outcomes:**	**Year 5/6:** To consider the impact of the choices made when making and sharing a video **Activities:** The unit concludes by enabling learners to review the content of their videos and finalise them by adding special effects, titles, and end credits. The latter part of the lesson prompts learners to discuss what was good about the videos and content, what could be done to improve them, and what did not	**Year 5/6:** To apply my knowledge of a database to ask and answer real-world questions **Activities:** The final lesson requires pupils to use a real-life database to ask questions and find answers in the context of a flight search based on set parameters. They take on the role of a travel agent and present their findings, showing how they arrived at their chosen options. Presentations may be given between groups of pupils, or by	**Year 5/6:** To create a controllable system that includes selection **Activities:** In this final lesson of the unit, learners will build on the designs that they developed in Lesson 5 by creating an algorithm to meet the requirements of the given task. They will identify how they are going to use selection before writing their algorithm. They will then move into the code level to test their	**Year 5/6:** To evaluate my program **Activities:** In this lesson, learners will return to their completed programs and identify ways in which the program can be improved. They will focus on issues where answers similar to those in the condition are given as inputs, and identify ways to avoid such problems. Learners will also consider how the outcomes may change the program for subsequent users, and identify how they can make use of setup to provide all users with the same experience.

	bounds of copyright and with the relevant permissions.) This lesson uses the Scratch programming tool, which allows learners to use other people's work. **Outcomes:** **Year 5:** I can suggest strategies to ensure successful group work I can make thoughtful suggestions on my group's work I can compare working online with working offline **Year 6:**	**Year 5/6:** I create alternatives to vector drawings I can suggest improvements to a vector drawing I can apply what I have learned about vector drawings	work so well. Learners are encouraged to both give and respond to feedback from their peers and teacher using a peer-assessment rubric. TIme permitting, videos could be presented to a wider audience at a red carpet assembly. **Outcomes:** **Year 5/6:** I can make edits to my video and improve the final outcome I can recognise that my choices when making a video will impact on the quality of the final outcome I can evaluate my video and share my opinions	each group to the whole class, depending on the time available. **Outcomes:** **Year 5/6:** 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	algorithm by implementing it as a program, running it, identifying any bugs, and returning to the algorithm to debug it where necessary. Finally, to conclude the unit, they will evaluate their algorithms and other areas of their designs. **Outcomes:** **Year 5/6:** I can write an algorithm to control lights and a motor I can use selection to produce an intended outcome I can test and debug my project	They will implement their identified improvements by returning to the Scratch programming environment and adding to their programs. They conclude the unit by identifying how they met the requirements of the given task, and identifying the aspects of the program that worked well, those they improved, and areas that could improve further. **Outcomes:** **Year 5/6:** 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

	Computing and networks – Sharing Information	Creating Media – Vector Drawings	Creating Media – Video Editing	Data Information – Flat Field Data-Bases	Programming A – Selecting in physical computing	Programming B – Selection in quizzes
Progression	This unit progresses learners' knowledge and understanding of computing systems and online collaborative working.	This unit progresses students' knowledge and understanding of digital painting and has some links to desktop publishing in which learners used digital images. They are now creating the images that they could use in desktop publishing documents.	This unit progresses learners' knowledge and understanding of creating media by guiding them systematically through the process involved in creating a video. By the end of the unit, learners will have developed the skills required to plan, record, edit, and finalise a video.	This unit progresses pupils' knowledge and understanding of why and how information might be stored in a database, and looks at how tools within a database can help us to answer questions about our data. It moves on to demonstrate how a database can help us display data visually, and how real-life databases can be used to help us solve problems. Finally, the pupils create a presentation showing understanding and application of all the tools used within the unit.	This unit assumes that learners will have prior experience of programming using block-based construction (eg Scratch) and understand the concepts of sequence and repetition. The National Centre for Computing Education key stage 1 units focus on floor robots and ScratchJr, however, experience of other languages or environments may also be useful.	This unit assumes that learners will have prior experience of programming using block-based construction (eg Scratch), understand the concepts of 'sequence' and 'repetition', and have some experience of using 'selection'. Ideally, learners will have completed 'Programming A – Selection in physical computing' before undertaking this unit, as this will provide them with the required knowledge of 'selection'.

Curricular Links	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. 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 Education for a Connected World links I can assess and justify when it is acceptable to use the work of others I can give examples of content that is permitted to be reused	**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. **Education for a Connected World links** **Copyright and ownership** I can explain why copying someone else's work from the internet without permission can cause problems.	**Computing** https://www.computingatschool.org.uk/data/uploads/CASPrimaryComputing.pdf (page 12) 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** Recognise inappropriate content, contact, and conduct and know how to report concerns. Use technology safely, respectfully, and responsibly; recognise acceptable/unacceptable behaviour. Identify a range of ways to report concerns about content and contact. **Education for a Connected World links (Years 7–11) Self-image and Identity** I can explain how I can represent myself in different ways online . Knowing this, I can describe the right decisions about how I interact with others and how others perceive me. **Online relationships** I can recognise some ways in which the internet can be used to communicate. I can give examples of how to be respectful to others online. **Online reputation** I can search for information about an individual online and create a summary report of the information I find. I can explain ways that some of the information about me online could have been created, copied, or shared by others. **Managing online information** I can evaluate digital content (and can explain how I make choices from search results).	**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.	**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	**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.

Assessment	Please see the assessment question and answer documents for this unit.	**Formative assessment** Assessment opportunities are detailed in each lesson plan. The learning objective and success criteria are introduced in the slide deck at the beginning of each lesson and then reviewed at the end. Learners are invited to assess how well they feel they have met the learning objective using thumbs up, thumbs sideways, or thumbs down. **Summative assessment** Learners are invited to assess how well their videos met the objectives of the unit. Please see the assessment rubric document for this unit.	**Formative assessment** Assessment opportunities are detailed in each lesson plan. The learning objectives and success criteria are introduced in the slide deck at the beginning of each lesson and then reviewed at the end. Learners are invited to assess how well they feel they have met the learning objective using thumbs up, thumbs sideways, or thumbs down. **Summative assessment** Learners are invited to assess how well their videos met the objectives of the unit. Please see the assessment rubric document for this unit.	**Summative assessment** Please see the assessment question and answer documents for this unit.	**Formative assessment** Assessment opportunities are detailed in each lesson plan. The learning objective and success criteria are introduced in the slide deck at the beginning of each lesson and then reviewed at the end. Learners are invited to assess how well they feel they have met the learning objective using thumbs up, thumbs sideways, or thumbs down. **Summative assessment** See the assessment rubric to support summative assessment for this unit.	**Formative assessment** Assessment opportunities are detailed in each lesson plan. The learning objectives and success criteria are introduced in the slide deck at the beginning of each lesson, and then reviewed at the end. Pupils are invited to assess how well they feel they have met the learning objectives using thumbs up, thumbs sideways, or thumbs down. We recommend the use of teacher accounts in Scratch to help with assessment throughout this unit. For guidance on setting up teacher accounts, please visit the Scratch website (scratch.mit.edu/educators/faq). **Summative assessment** Please see the assessment question and answer documents for this unit.

Subject Knowledge	Enhance your subject knowledge to teach this unit through the following training opportunities: **Online training courses** Raspberry Pi Foundation online training courses **Face-to-face courses** National Centre for Computing Education face-to-face training courses	**Lesson 1:** Teachers will need an understanding of the tools available in the chosen software such as shape and line drawing tools: undo, redo, select, fill, and delete. **Lesson 2:** Teachers would benefit from knowing that vector drawings are created using shapes and that each shape used is called an object. It would be helpful to know how to move, resize, rotate, duplicate, and change the colour of objects within Google Drawings. **Lesson 3:** Teachers would benefit from an understanding of the tools within Google Drawings, including the ability to add colour effects to shapes and lines. Sharing Google Drawing files with the learners will be required for this lesson. **Lesson 4:** Teachers need to understand that each added object creates a new layer in the drawing. It would help to know how to move shapes to a different layer using 'bring to front' or 'send to back'. **Lesson 5:** Teachers would benefit from understanding that objects can be grouped. Knowing how to group objects and copy/paste or duplicate them using keyboard commands would be an advantage. **Lesson 6:** Teachers would benefit from an understanding of how digital images can be made, either using shapes or pixels. It would be an advantage for teachers to have an understanding of the drawing tools available in Google Drawings to support learners in showcasing their skills in this lesson.	This unit focuses on the skills associated with planning, recording, editing, and creating a video. **Lesson 1:** You will need to have a brief understanding of the history of video (and animation) by way of introduction. As learners will progress through this unit in working groups, a sound understanding of the role of groups in conventional roles is essential. Learners could be assigned roles such as facilitator, recorder, summariser, presenter, and timekeeper. Prepare for the lesson by having some appropriate video clips ready. This should include animation/CGI, black-and-white video, silent movie, and/or full-colour video. **Lesson 2:** You will need to have access to some devices and be able to identify their key features (buttons, microphone, on/off switches) and know how to demonstrate effective use. **Lesson 3:** You will need to confidently use a range of devices for recording and where necessary, link the devices to your interactive whiteboard to demonstrate to the learners. Prepare a selection of short videos that demonstrate each of the techniques, e.g. videos (https://blog.storyblocks.com/video-tutorials/7-basic-camera-movements/). **Lesson 4**: You will need to have a good understanding of what makes an effective video. This lesson makes reference to YouTubers and YouTube influencers and you should reference or use videos from appropriate YouTube channels to support the learning. **Lesson 5:** You will need to have a clear understanding of the various devices and apps or programs that are used to import and edit video content, such as Windows Movie Maker. You need to know how to store, transfer, and retrieve files and be able to demonstrate video editing techniques. **Lesson 6:** You will need to have a clear understanding of how to edit and complete the video creation progress by adding titles, end credits, and transitions. Finally, you should be able to demonstrate how to export the video project into an .mp4 format for viewing.	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, eg 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.	This unit focuses on physical computing that allows learners to control real-life events 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 play a video. 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 videos to see what to demonstrate, then show learners with a live demonstration, however, videos 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 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 peripherals. Additionally, you will be connecting the Crumble controller with 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 program and how it works: • Task — this is what is needed • Design — this is what it should do	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 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,

					• 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 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. *Infinite loop* 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. *Count-controlled loop* 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 need to be repeated. *Condition-controlled loop* 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 '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	the condition would only be checked once following the sequence of the code.

					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** When designing programs, there are often points where a decision must be made. BBC Bitesize defines selection as: Selection – a decision within a computer program when the program decides to move on based on the results of an event (source: BBC Bitesize) These decisions are known as selection, and are implemented in programming using if 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) have to be used to instruct the device to check the condition repeatedly. Without using loops, the condition would only be checked once. 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.	