Computing Curriculum

EYFS
Educational programmes: Revised EYFS framework 2021

Despite computing **not being explicitly mentioned within the Early Years Foundation Stage (EYFS) statutory framework**, which focuses on the learning and development of children from birth to age five, there are many opportunities for young children to use technology to solve problems and produce creative outcomes.

Computing in the EYFS is centred around play-based, unplugged (no computer) activities that focus on building children's listening skills, curiosity and creativity and problem solving.

Technology in the Early Years Foundation Stage means for example:

- taking a photograph with a camera or tablet
- searching for information on the internet
- playing games on the interactive whiteboard
- using a Beebot
- watching a video clip
- listening to music
- controlling toys with a remote control
- using technology though role play eg mobile phone, camera, microwave, ovens, broken devices
- using technology equipment to measure units of time eg stop watches.

Allowing children the opportunity to explore technology in this carefree and often child-led way, means that not only will they develop a familiarity with equipment and vocabulary, but they will have a strong start in Key Stage 1 Computing and all that it demands.

KS1 National Curriculum
- 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

Kapow Primary

Year 1	Autumn A	Autumn B	Spring A	Spring B	Summer A	Summer B
Concepts of computer science	Computing systems and networks Improving mouse skills	Programming 1 Algorithms unplugged	Skills showcase Rocket to the moon	Programming 2 Programming Bee-bots	Creating media Digital imagery	Data handling Introduction to data
Knowledge and understanding	Learning how to login and navigate around a computer; developing mouse skills; learning how to drag, drop, click and control a cursor to create works of art	Algorithms, decomposition and debugging are made relatable to familiar contexts, following directions, learning why instructions need to be specific	Developing keyboard and mouse skills through designing, building and testing. Creating a digital list of materials, using drawing software and recording data.	Introducing programming through the use of a Bee-Bot and exploring its functions.	Taking and editing photos, searching for and adding images to a project.	Learning what data is and the different ways it can be represented. Learning why data is useful and the ways it can be gathered and recorded.
Learning outcomes	Pupils who are secure will be able to: • Use computers more purposefully. • Log in and navigate around a computer. • Drag, drop, click and control a cursor using a mouse. • Use software tools to create art on the computer.	Pupils who are secure will be able to: • Explain what an algorithm is. • Write clear algorithms. • Follow an algorithm. • Explain what inputs and outputs are. • Create an achievable program. • Decompose a design into steps. • Identify bugs in an algorithm and how to fix them.	Pupils who are secure will be able to: • Use a computer to make a list. • Explain the benefits of making a list on the computer. • Use a basic range of tools on graphics editing software to design a rocket. • Sequence instructions. • Follow instructions to build their model rocket. • Input data about their rockets into a table or spreadsheet.	Pupils who are secure will be able to: • Recognise cause and effect when pressing buttons on a Bee-Bot. • Discuss and demonstrate how the Bee-Bot works. • Record video ensuring everyone is in the shot. • Give a number of clear instructions in sequence. • Program a Bee-Bot to reach a destination. • Identify and correct mistakes in their programming.	Pupils who are secure will be able to: • Plan a pictorial story using photographic images in sequence. • Explain how to take clear photos. • Take photos using a device. • Edit photos by cropping, filtering and resizing. • Search for and import images from the internet. • Explain what to do if something makes them uncomfortable online. • Organise images on the page, orientating where necessary.	Pupils who are secure will be able to: • Represent animal-themed data in different ways, using objects and technology. • Log in and use mouse and keyboard skills to navigate the computer. • Represent the same data as a pictogram and a table or chart. • Collect data about minibeasts using a tally chart and represent their data digitally. • Click and drag objects to sort data using a branching database. • Consider the types of input that would be used to gather different forms of data when designing an invention.

Every Child a
Confident Learner

Year 1	Autumn A	Autumn B	Spring A	Spring B	Summer A	Summer B
Sequence of Learning	• Learn how to explore and tinker with hardware to find out how it works. • Learn where keys are located on the keyboard. • Use a basic range of tools within graphic editing software. • Develop control of the mouse through dragging, clicking and resizing of images to create different effects. • Develop understanding of different software tools. • Recognise devices that are connected to the internet. • Log in and out and saving work on their own account.	• Recognise that some devices are input devices and others are output devices. • Learn that decomposition means breaking a problem down into smaller parts. • Use decomposition to solve unplugged challenges. • Develop the skills associated with sequencing in unplugged activities. • Follow a basic set of instructions. • Assemble instructions into a simple algorithm. • Learn to debug instructions when things go wrong. • Learn to debug an algorithm in an unplugged scenario.	• Learn where keys are located on the keyboard. • Learn how to operate a camera to take photos and videos. • Use logical reasoning to predict the behaviour of simple programs. • Develop the skills associated with sequencing in unplugged activities. • Follow a basic set of instructions. • Assemble instructions into a simple algorithm. • Learn to debug instructions when things go wrong. • Learn to debug an algorithm in an unplugged scenario. • Use a basic range of tools within graphic editing software. • Take and edit photographs. • Develop control of the mouse through dragging, clicking and resizing of images to create different effects. • Develop understanding of different software tools. • Recognise devices that are connected to the internet. • Understand that technology can be used to represent data in different ways: pictograms, tables, pie charts, bar charts, block graphs etc. • Log in and out and saving work on their own account.	• Learn how to explore and tinker with hardware to find out how it works. • Learn how to operate a camera to take photos and videos. • Use decomposition to solve unplugged challenges. • Use logical reasoning to predict the behaviour of simple programs. • Develop the skills associated with sequencing in unplugged activities. • Follow a basic set of instructions. • Assemble instructions into a simple algorithm. • Programme a floor robot to follow a planned route. • Learn to debug instructions when things go wrong. • Use programming language to explain how a floor robot works. • Learn to debug an algorithm in an unplugged scenario. • Take and edit photographs.	• Learn how to explore and tinker with hardware to find out how it works. • Learn where keys are located on the keyboard. • Learn how to operate a camera to take photos and videos. • Develop the skills associated with sequencing in unplugged activities. • Use a basic range of tools within graphic editing software. • Take and edit photographs. • Develop control of the mouse through dragging, clicking and resizing of images to create different effects. • Develop understanding of different software tools. • Search and download images from the internet safely. • When using the internet to search for images, learn what to do if they come across something online that worries them or makes them feel uncomfortable.	• Learn how to explore and tinker with hardware to find out how it works. • Recognise that some devices are input devices and others are output devices. • Learn where keys are located on the keyboard. • Develop control of the mouse through dragging, clicking and resizing of images to create different effects. • Develop understanding of different software tools. • Recognise devices that are connected to the internet. • Understand that technology can be used to represent data in different ways: pictograms, tables, pie charts, bar charts, block graphs etc. • Use data representations to answer questions about data. • Use software to explore and create pictograms and branching databases.
Retrieval Vocabulary		Mouse Log in Device	Mouse Log In Input	Mouse Input Log in Program Debug Instructions	Mouse Image Drag and drop Resize	Mouse Resize Input Keyboard
New Vocabulary	Login Ctrl Log out / off Tools Mouse Right click Mouse pointer Menu Click Layers Keyboard Username Screen Password Drag Account Drag and drop Software Digital photograph Duplicate Undo Resize Cursor	Algorithm Organise Automatic Output Bug Precise Chunks Programming Clear Problem Code Robot Debug Sensor Decompose Sequence Decomposition Solution Directions Specific Steps Input Tasks Instructions Virtual Assistant Manageable Motion Order	Annotate Folder Cells Instructions Components Photo Create Program Data Order Debug Robot Designing Save Digital content Sequence Digital image Share Document Software E-document Spreadsheet Edit Table Editing program Evaluate	Algorithm Pause Artificial intelligence Precise Predict Bee-Bot Tinker Clear Video Code Video recording Demonstration Filming	Background Import Blurred Internet Camera Keyword Clear Online Crop Photograph Delete Save as Device Screen Digital camera Search engine Download Sequence Edit Software Editing software Storage space Filter Visual effects	Bar chart Line graph Block graph Information Branching database Label Categorise Pictogram Chart Pie chart Click and drag Process Compare Record Count Sort Data Table Data collection Tally Data record Values Data representation Edit
Online Safety	Online safety – runs 1 lesson per half term Learning how to stay safe online and how to manage feelings and emotions when someone or something has upset us.					

KS1 National Curriculum
- 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

Kapow Primary

Year 2	Autumn A	Autumn B	Spring A	Spring B	Summer A	Summer B
Concepts of computer science	Computing systems and networks 1 What is a computer?	Programming 1 Algorithms and debugging	Computing systems and networks 2 Word processing	Programming 2 Programming: ScratchJr	Creating media Stop Motion	Data handling International Space Station
Knowledge and understanding	Exploring what a computer is by identifying how inputs and outputs work and how computers are used in the wider world to design their own computerised invention.	Developing an understanding of; what algorithms are, how to program them and how they can be developed to be more efficient, introduction of loops.	Developing touch typing skills, learning keyboard shortcuts and simple editing tools.	Exploring what 'blocks' do' by carrying out an informative cycle of predict > test > review. Programming a familiar story and make a musical instrument.	Learning how to create simple animations from storyboarding creative ideas.	Learning how data is collected, used and displayed and the scientific learning of the conditions needed for plants and humans, to survive.
Learning outcomes	Pupils who are secure will be able to: • Name some computer peripherals and their function. • Recognise that buttons cause effects. • Explain that technology follows instructions. • Recognise different forms of technology. • Design an invention which includes inputs and outputs. • Explain the role of computers in the world around them.	Pupils who are secure will be able to: • Decompose a game to predict the algorithms. • Give a definition for 'decomposition'. • Write clear and precise algorithms. • Create algorithms to solve problems. • Use loops in their algorithms to make their code more efficient. • Explain what abstraction is.	Pupils who are secure will be able to: • Explain which are the home row keys and how to find them for typing. • Use the spacebar and backspace correctly. • Type and make simple alterations to text using buttons on a word processor. • Search for, import and alter appropriate images for a text document. • Modify text in a document. • Use copy and paste to copy text from one document to another. • Explain what information is safe to be shared online.	Pupils who are secure will be able to: • Explore a new application independently. • Explain what the blocks on ScratchJr do and use them for a purpose. • Recognise a loop in coding and why it is useful. • Use a code to create an animation of an animal moving. • Use code to follow and create an algorithm. • Program code to run 'on tap'. • Explain the role of the blocks in a program they have created.	Pupils who are secure will be able to: • Create a flip book animation. • Decompose a story into smaller parts to plan a stop motion animation. • Create stop motion animations with small changes between images.	Pupils who are secure will be able to: • Describe and explain how astronauts' survival needs are met aboard the ISS. • Identify and digitally draw items which fulfil basic human needs when aboard the ISS. • Read the correct temperature on a thermometer. • Design a display showing everything that needs to be monitored by sensors on the ISS. • Create an algorithm that addresses all plants' needs. • Explain how space exploration can benefit life on Earth. • Read data to identify whether a planet might be habitable.

Year 2	Autumn A	Autumn B	Spring A	Spring B	Summer A	Summer B
Sequence of Learning	• Understand what a computer is and that it's made up of different components. • Recognise that buttons cause effects and that technology follows instructions. • Learn how we know that technology is doing what we want it to do via its output. • Use greater control when taking photos with cameras, tablets or computers. • Develop word processing skills, including altering text, copying and pasting and using keyboard shortcuts. • Use word processing software to type and reformat text. • Create and label images. • Learn how computers are used in the wider world	• Develop confidence with the keyboard and the basics of touch typing. • Articulate what decomposition is. • Decompose a game to predict the algorithms used to create it. • Learn that there are different levels of abstraction. • Explain what an algorithm is. • Follow an algorithm. • Create a clear and precise algorithm. • Learn that programs execute by following precise instructions. • Incorporate loops within algorithms. • Use logical thinking to explore software, predicting, testing and explaining what it does. • Use an algorithm to write a basic computer program. • Develop word processing skills, including altering text, copying and pasting and using keyboard shortcuts.	• Develop confidence with the keyboard and the basics of touch typing. • Develop word processing skills, including altering text, copying and pasting and using keyboard shortcuts. • Use word processing software to type and reformat text. • Search for appropriate images to use in a document. • Understand what online information is. • Identify whether information is safe or unsafe to be shared online.	• Recognise that buttons cause effects and that technology follows instruction • Explain what an algorithm is. • Follow an algorithm. • Create a clear and precise algorithm. • Learn that programs execute by following precise instructions. • Incorporate loops within algorithms. • Use logical thinking to explore software, predicting, testing and explaining what it does. • Use an algorithm to write a basic computer program. • Use loop blocks when programming to repeat an instruction more than once. • Use software (and unplugged means) to create story animations.	• Use greater control when taking photos with cameras, tablets or computers. • Use logical thinking to explore software, predicting, testing and explaining what it does. • Understand that an animation is made up of a sequence of photographs. • Know that small changes in my frames will create a smoother looking animation. • Understand what software creates simple animations and some of its features e.g. onion skinning.	• Develop confidence with the keyboard and the basics of touch typing. • Create and label images. • Collect and inputting data into a spreadsheet. • Interpret data from a spreadsheet. • Learn how computers are used in the wider world.
Retrieval Vocabulary	Laptop Input Computer Output Desktop Keyboard Device Screen Mouse	Bug Debug Algorithm Decomposition	Keyboard Online	Algorithm Debug Instructions	Decompose Device Drag and drop Resize Image	Algorithm Keyboard Data

New Vocabulary	Battery Monitor
	Buttons Paying till
	Camera Scanner
	Digital System
	Digital recorder Tablet
	Electricity Technology
	Function Video
	Invention Wires
		Abstraction
		Artificial intelligence
		Clear
		Correct
		Data
		Error
		Key features
		Loop
		Predict
		Unnecessary
			Backspace character
			Bold Keyboard
			Copy shortcut
			Copyright Keyword
			Cut Layout
			Delete Navigate
			Forward button Paste
			Highlight Redo
			Home row Search
			Home screen Space bar
			Image Text
			Import Text effects
			Italics Touch typing
			Keyboard Underline
			  Undo
			  Word processing
				Animation Icon
				Blocks Imitate
				Bug Loop
				Button 'On tap'
				CGI Programming
				Computer code Repeat
				Code ScratchJR
				Fluid Sequence
				  Sound recording
					Animation
					Background
					Digital device
					Drawing
					Flipbook
					Frames
					Moving images
					Object
					Onion skinning
					Plan
					Still images
						Astronaut International
						Digital Space Station
						Digital content Interpret
						Experiment Laboratory
						Galaxy Monitor
						Insulation Planet
						Interactive map Satellite
						International Space Centre Sensor
						  Space
						  Temperature
						  Thermometer
						  Water reservoir

Online Safety	Online safety – runs 1 lesson per half term Learning: how to keep information safe and private online; who we should ask before sharing things online and how to give, or deny permission online.					

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 (word processing, sound, data handling)

Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact.

Kapow Primary

Year 3	Autumn A	Autumn B	Spring A	Spring B	Summer A	Summer B
Concepts of computer science	Computing systems and networks Networks and the internet	Programming 1 Scratch	Computing systems and networks Emailing	Computing systems and networks Journey inside a Computer	Creating media Video Trailers	Data handling Comparison card databases
Knowledge and understanding	Learning what a network and how devices communicate and share information.	Exploring the programme Scratch, following the predict > test > review cycle. Learning about 'loops' and programming an animation, story and game.	Sending emails with attachments and understanding what cyberbullying is.	Assuming the role of computer parts and creating paper versions of computers to consolidate understanding of how a computer works.	Developing digital video skills to create trailers, with special effects and transitions.	Learning about records, fields and data and sorting and filtering data.
Learning outcomes	Pupils who are secure will be able to: • Recognise that a network is two or more devices connected. • Explain how information moves around a network and the role of the server. • Understand that networks connect to the internet via a router. • Explain some of the journey a website goes through to reach your computer. • Explain that websites are split into small pieces (packets) to be sent via the internet.	Pupils who are secure will be able to: • Explain what some of the blocks do in Scratch. • Explain what a loop is and include one in their program. • Suggest possible additions to an existing program. • Recognise where something on screen is controlled by code. • Use a systematic approach to find bugs. • Explain what an algorithm is and its purpose.	Pupils who are secure will be able to: • Log in and out of email. • Send a simple email with a subject plus 'To' and 'From' in the body of the text. • Edit an email. • Type in the email address correctly and send the email. • Add an attachment to an email. • Write an email using positive language, with an awareness of how it will make the recipient feel. • Recognise unkind behaviour online and know how to report it. • Offer advice to victims of cyberbullying. • Recognise when an email may be fake and explain how they know.	Pupils who are secure will be able to: • Recognise inputs and outputs and that the computer sends and receives information. • Explain that the parts of a laptop work together and the purpose of each part. • Explain what an algorithm is. • Suggest what memory is for inside a computer. • Make comparisons between different types of computer.	Pupils who are secure will be able to: • Describe the purpose of a trailer. • Create a storyboard for a book trailer. • Consider camera angles when taking photos or videos. • Import videos and photos into film editing software. • Add text to a video. • Incorporate transitions between images. • Evaluate their own and others' trailers.	Pupils who are secure will be able to: • Explain what is meant by 'field,' 'record,' and 'data.' • Compare paper and computerised databases. • Put values into a spreadsheet. • Sort, filter and interpret data in a spreadsheet. • Create a graph on Microsoft Excel. • Explain the purpose of visual representations of data.

Year 3	Autumn A	Autumn B	Spring A	Spring B	Summer A	Summer B
Sequence of Learning	• Learning about the purpose of routers. • Understanding the role of the key components of a network. • Understanding that websites & videos are files that are shared from one computer to another. • Learning about the role of packets. • Understanding how networks work and their purpose. • Identifying the key components within a network, including whether they are wired or wireless. • Recognising links between networks and the internet. • Learning how data is transferred.	• Using decomposition to explore the code behind an animation. • Using repetition in programs. • Using logical reasoning to explain how simple algorithms work. • Explaining the purpose of an algorithm. • Forming algorithms independently. • Using logical thinking to explore more complex software; predicting, testing and explaining what it does. • Incorporating loops to make code more efficient. • Continuing existing code. • Making reasonable suggestions for how to debug their own and others' code.	• Learning to log in and out of an email account. • Writing an email including a subject, 'to' and 'from'. • Sending an email with an attachment. • Replying to an email. • Understanding the purpose of emails. • Learning about cyberbullying. • Learning that not all emails are genuine, recognising when an email might be fake and what to do about it.	• Understanding what the different components of a computer do and how they work together. • Drawing comparisons across different types of computers. • Using decomposition to explain the parts of a laptop computer. • Explaining the purpose of an algorithm • To know the roles that inputs and outputs play on computers. • To know what some of the different components inside a computer are e.g. CPU, RAM, hard drive, and how they work together. • To know what a tablet is and how it is different from a laptop/desktop computer.	• Using logical thinking to explore more complex software; predicting, testing and explaining what it does. • Taking photographs and recording video to tell a story. • Using software to edit and enhance their video adding music and text on screen with transitions. • To know that different types of camera shots can make my photos or videos look more effective. • To know that I can edit photos and videos using film editing software. • To understand that I can add transitions and text to my video.	• Using logical thinking to explore more complex software; predicting, testing and explaining what it does. • Understanding the vocabulary associated with databases: field, record, data. • Learning about the pros and cons of digital versus paper databases. • Sorting and filtering databases to easily retrieve information. • Creating and interpreting charts and graphs to understand data.
Retrieval Vocabulary	Laptop Online Internet Website	Algorithm Loop Debug Decompose Program	Log In	Algorithm Screen Mouse Tablet Device Desktop Data Keyboard Laptop Input Output Decompose Program	Video Application Recording	Data Database Spreadsheets Graph Information
New Vocabulary	Cables Component Connection Corrupted Data Desktop Device DSL Fibre File Network Network map Network switch Packets Radio waves Router Server Submarine cables Tablet Text map The Cloud Web server Website trackers WiFi Wired Wireless Wireless Access Points World Wide Web	Animation Application Code Code block Coding application Interface Predict Remixing code Repetition code Review Scratch Sprite Tinker	Attachment Bcc (Blind carbon copy) Cc (Carbon copy) Compose Content Cyberbully-ing Document Domain Download Email Email account Email address Emoji Emotions Fake Font Genuine Hacker Icons Inbox Information Link Log out Negative language Password Personal information Positive language Reply Responsible digital citizen Scammer Settings Send Sign in Spam email Subject bar Theme Tone Username Virus WiFi	Assemble CPU (central processing unit) Disassemble GPU (graphics processing unit) Hard drive HDD (hard disk drive) Infinite loop Memory Microphone Monitor Photocopier QR Code RAM (random access memory) ROM (read only memory) Storage Technology Touchscreen Touchpad	Camera angle Clip Cross dissolve Edit Fade to black Fade to white Film Film editing software Graphics Import Key events Music Photo Plan Slide Sound effects Storyboard Time code Trailer Transition Voiceover Wipe	Categorise Category Chart Excel Fields Filter Interpret PDF Questionnaire Record Representation Sort
Online Safety	Online safety – runs 1 lesson per half term Learning: the difference between fact, opinion and belief; and how to deal with upsetting online content. Knowing how to protect personal information online.					

KS2 National Curriculum

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 (word processing, sound, data handling)

Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact.

Kapow Primary

Year 4	Autumn A	Autumn B	Spring A	Spring B	Summer A	Summer B
Concepts of computer science	Programming 2 HTML	Computing systems and networks Collaborative Learning	Programming 1 Further coding with Scratch	Skills showcase Website design	Creating media Computational thinking	Data handling Investigating weather
Knowledge and understanding	Learning about the markup language behind a webpage; becoming familiar with HTML tags, changing HTML and CSS code to alter images and 'remix' a live website	Learning how to work collaboratively and exploring a range of collaborative tools.	Revisiting the key features and beginning to use 'variables' in code scripts.	Learning how web pages and sites are created and how to embed media and links.	Solving problems effectively using the four areas of abstraction, algorithm design, decomposition and pattern recognition	Researching and storing data on spreadsheets and designing a weather station.
Learning outcomes	Pupils who are secure will be able to: • Add text between the heading and paragraph tags. • Easily activate the goggles to investigate a web page. • Explain how they altered the HTML to create their own posters. • Change the colours and sizes of their object elements. Explain how they created their story. • Adapt the basic elements of a story within a web page using the 'Inspect Elements' tool. • Change an image within a web page and create their own news story, replacing the text and images of a webpage.	Pupils who are secure will be able to: • Understand the need to be thoughtful when working on a collaborative document. • Use comments to suggest changes to a document and understand how to resolve comments. • Plan a survey for Microsoft Form with a range of different questions types that will provide different types of answer, e.g. text, multiple choice or numerical values. • Create a Microsoft Form with a range of different question types that will provide different types of answer, e.g. text, multiple choice or numerical values. • Export data to a spreadsheet, highlighting data, using conditional formatting and calculating averages and sums of numbers.	Pupils who are secure will be able to: • Understand how to create a simple script in Scratch – be able to change sprite and prevent the sprite from rotating. • Use decomposition to identify key features and understand how to decipher actions that make the quiz game work. • Understand what a variable is and how to use the 'say' and 'ask' blocks. • Create a variable and be able to use a variable to record a score. • Understand what a variable is and how it works within a program.	Pupils who are secure will be able to: • Create a Sway with a title, image and a completed first header section. • Create a clear plan for their web page and beginning to create it. • Create a professional-looking web page with useful information and a clear style, which is easy for the user to read and find information from. • Create a clear plan by referring back to their checklist to include a range of features. • Create a web page with clear sections and with a range of features in.	Pupils who are secure will be able to: • Understand that problems can be solved more easily using computational thinking. • Understand what the different code blocks do and create a simple game. • Understand the terms 'pattern recognition' and 'abstraction' and how they help to solve a problem. • Create a Scratch program which draws a square and at least one other shape. • Understand how computational thinking can help to solve problems and apply computational thinking to problems they face.	Pupils who are secure will be able to: • Search the web efficiently to find temperatures of different cities and record this accurately. • Design a weather station that gathers and records sensor data, explaining how it works and the units of measurement it would use. • Design an automated machine that uses selection to respond to sensor data. • Search for and record weather forecast information in a spreadsheet and explain how this data is collected. • Create a video which includes weather forecast information.

Year 4	Autumn A		Autumn B		Spring A		Spring B		Summer A		Summer B	
Sequence of Learning	• Remixing existing code. • Building a web page and creating content for it. • Understanding that information found by searching the internet is not all grounded in fact. • Recognising that information on the Internet might not be true or correct and that some sources are more trustworthy than others.		• Understanding that computer networks provide multiple services, such as the World Wide Web, and opportunities for communication and collaboration. • Use online software for documents, presentations, forms and spreadsheets. • Using software to work collaboratively with others. • Understanding that software can be used collaboratively online to work as a team. • Recognising what appropriate behaviour is when collaborating with others online.		• Using decomposition to solve a problem by finding out what code was used. • Using decomposition to understand the purpose of a script of code. • Creating algorithms for a specific purpose. • Coding a simple game. • Incorporating variables to make code more efficient. • Remixing existing code.		• Building a web page and creating content for it. • Designing and creating a webpage for a given purpose. • Using software to work collaboratively with others. • To know that a website is a collection of pages that are all connected. • To know that websites usually have a homepage and subpages as well as clickable links to new pages, called hyperlinks. • To know that websites should be informative and interactive.		• Using decomposition to solve a problem by finding out what code was used. • Using decomposition to understand the purpose of a script of code. • Identifying patterns through unplugged activities. • Using past experiences to help solve new problems. • Using abstraction to identify the important parts when completing both plugged and unplugged activities. • Creating algorithms for a specific purpose. • Using abstraction and pattern recognition to modify code.		• Using tablets or digital cameras to film a weather forecast. • Understanding that weather stations use sensors to gather and record data which predicts the weather. • Using keywords to effectively search for information on the internet. • Searching the internet for data. • Designing a device which gathers and records sensor data. • Recording data in a spreadsheet independently. • Sorting data in a spreadsheet to compare using the 'sort by…' option. • Understanding that data is used to forecast weather.	
Retrieval Vocabulary	Code Fake news Hacking Heading HTML Input Internet Browser	Output Paragraph Permission Webpage	Animation Application Code Code block Coding application Interface	Predict Remixing code Repetition code Review Scratch Sprite Tinker	Decomposition Game Negative Numbers	Tinker Program Coordinates	Assessment Homepage Hyperlinks Images	Insert Online Plan Web page Website World Wide Web	Algorithm Code Decomposition Input Output Script		Collaboration	
New Vocabulary	Component Content Copyright CSS End tag Headline Hex code	Remixing Script Start tag Tags Text URL	Average Bar chart Comment Contribution Data Edited Email account Format Freeze Icon Images Insert Link Multiple Choice	Numerical data Pie chart Presentations Resolved Reviewing comments Share Slides Software Spreadsheets Suggestions Survey Teamwork Themes Transitions	Broadcast Block Code blocks Conditional Features Information	Orientation Parameters Position Project Script Sprite Stage Variables	Audience Checklist Collaboration Content Contribution Create Design Embed Evaluate Features Google Sites Hobby	Progress Published Record Review Style Subpage Tab Theme	Abstraction Computational thinking Logical reasoning Pattern recognition Sequence Variable		Accurate Backdrop Climate zone Cold Condensation Cylinder Degrees Evaporation Extreme weather Forecast Heat sensor Lightning Measurement Pinwheel	Presenter Rain Satellite Script Sensitive Sensor data Solar panel Tablet/Digital camera Temperature Thermometer Tornado Warm Weather Weather forecast Wind
Online Safety	Online safety – runs 1 lesson per half term Searching for information and making a judgement about the probable accuracy; recognising adverts and pop-ups; understanding that technology can be distracting.											

KS2 National Curriculum

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 (word processing, sound, data handling)

Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact.

Kapow Primary

Year 5	Autumn A	Autumn B	Spring A	Spring B	Summer A	Summer B
Concepts of computer science	Computing systems and networks Search engines	Programming 1 Programming music	Data Handling Mars Rover Part 1	Programming 2 Mirco:bit	Creating media Stop Motion	Skills Showcase Mars Rover 2
Knowledge and understanding	Learning about how page rank works and how to identify inaccurate information.	Building-on programming and music skills to create different sounds, beats and melodies which are put to the test with a Battle of the Bands performance!	Learning about the Mars Rover, exploring how and why it transfers data including instructions, and how messages can be sent using binary code.	Creating algorithms and programs that are used in the real world. Using the 'predict, test and evaluate' cycle to create and debug programs with specific aims.	Creating animations, storyboard ideas and decomposing a story into small parts before putting together to create the illusion of a moving image.	Exploring how the Mars rover: moves, follows instructions, collects and sends data; understanding how computers work, what data is and how it is transferred.
Learning outcomes	Pupils who are secure will be able to: • Explain what a search engine is, suggesting several search engines to use and explain how to use them to find websites and information. • Suggest that things online aren't always true and recognise what to check for. • Explain why keywords are important and what TASK stands for, using these strategies to search effectively. • Recognise the terms 'copyright' and 'fair use' and combine text and images in a poster. • Make parallels between book searching and internet searching, explaining the role of web crawlers and recognising that results are rated to decide rank.	Pupils who are secure will be able to: • Iterate ideas, testing and changing throughout the lesson. Explain what the basic commands do. • Explain how their program links to the theme. Include a loop in their work. Correct their own simple mistakes. • Explain their scene in the story. Link musical concepts to their scene. Include a repeat and explain its function to enhance music. • Code a piece of music that combines a variety of structures. Use loops in their programming. • Recognise that programming music is a way to apply their skills	Pupils who are secure will be able to: • Identify some of the types of data that the Mars Rover could collect (for example, photos). • Explain how the Mars Rover transmits the data back to Earth and the challenges involved in this. • Read any number in binary, up to eight bits. • Identify input, processing and output on the Mars Rovers. • Read binary numbers and grasp the concept of binary addition. • Relate binary signals (Boolean) to a simple character-based language, ASCII.	Pupils who are secure will be able to: • Clip blocks together and predict what will happen. Make connections with previous programming interfaces they've used, e.g. Scratch. • Create their own images to make the animation and recognise the difference between 'on start' and 'forever'. • Recognise blocks they've used previously, identifying inputs and outputs used and make predictions about how variables work. • Choose appropriate blocks to complete the program and attempt the challenges independently. • Break a program down into smaller steps, suggesting appropriate blocks and match the algorithm to the program.	Pupils who are secure will be able to: • Create a toy with simple images with a single movement. • Create a short stop motion with small changes between images. • Think of a simple story idea for their animation then decompose it into smaller parts to create a storyboard with simple characters. • Make small changes to the models to ensure a smooth animation and delete unnecessary frames. • Add effects such as extending parts and titles. • Provide helpful feedback to other groups about their animations.	Pupils who are secure will be able to: • Create a pixel picture, explaining that a pixel is the smallest element of a digital image and that binary is used to code and transfer this data. • Save a JPEG as a bitmap and recognise the difference in file size as well as explaining how pixels are used to transfer image data. • Explain the 'fetch, decode, execute' cycle in relation to real-world situations. • Create a profile with a safe and suitable username and password and begin to use 3D design tools. • Independently take tutorial lessons, applying what they have learnt to their design and understand the importance of using an online community responsibly.

Year 5	Autumn A	Autumn B	Spring A	Spring B	Summer A	Summer B
Sequence of Learning	• Developing searching skills to help find relevant information on the internet. • Learning how to use search engines effectively to find information, focussing on keyword searches and evaluating search returns. • Learn about different forms of communication that have developed with the use of technology. • Recognising that information on the Internet might not be true or correct and learning ways of checking validity.	• Predicting how software will work based on previous experience. • Writing more complex algorithms for a purpose. • Iterating and developing their programming as they work. • Confidently using loops in their programming. • Using a more systematic approach to debugging code, justifying what is wrong and how it can be corrected. • Writing code to create a desired effect. • Using a range of programming commands. • Using repetition within a program. • Amending code within a live scenario. • Using logical thinking to explore software more independently, making predictions based on their previous experience. • Using a software programme (Scratch) to create music. • Identify ways to improve and edit programs, videos, images etc.	• Learning that external devices can be programmed by a separate computer. • Recognising how the size of RAM affects the processing of data. • Learning the vocabulary associated with data: data and transmit. • Recognising that computers transfer data in binary and understanding simple binary addition. • Relating binary signals (Boolean) to the simple character-based language, ASCII. • Learning that messages can be sent by binary code, reading binary up to eight characters and carrying out binary calculations. • Understanding how data is collected in remote or dangerous places. • Understanding how data might be used to tell us about a location. • Learn about different forms of communication that have developed with the use of technology.	• Pupils who are secure will be able to: • Clip blocks together and predict what will happen. Make connections with previous programming interfaces they've used, e.g. Scratch. • Create their own images to make the animation and recognise the difference between 'on start' and 'forever'. • Recognise blocks they've used previously, identifying inputs and outputs used and make predictions about how variables work. • Choose appropriate blocks to complete the program and attempt the challenges independently. • Break a program down into smaller steps, suggesting appropriate blocks and match the algorithm to the program.	• Decomposing animations into a series of images. • Decomposing a story to be able to plan a program to tell a story. • Using video editing software to animate. • To know that decomposition of an idea is important when creating stop-motion animations. • To understand that stop motion animation is an animation filmed one frame at a time using models, and with tiny changes between each photograph. • To know that editing is an important feature of making and improving a stop motion animation.	• Learning the difference between ROM and RAM. • Recognising how the size of RAM affects the processing of data. • Understanding the fetch, decode, execute cycle. • Learning how the data for digital images can be compressed. • Recognising that computers transfer data in binary and understanding simple binary addition. • Understanding how bit patterns represent images as pixels. • Using logical thinking to explore software more independently, making predictions based on their previous experience. • Independently learning how to use 3D design software package TinkerCAD. • Learn about different forms of communication that have developed with the use of technology.
Retrieval Vocabulary	Algorithm Website	Coding Command Debug Decompose Error Output Play Predict Programming Repeat Tinker	Communicate CPU Technology Research Input Instructions Internet Output	Algorithm Animation App Connection Create Debug Decompose Desktop Input Instructions Laptop Outputs Predict Program Repetition Tablet Tinkering USB	Animation Decomposition	Algorithm Compression CPU Data Drag and drop Operating system Output
New Vocabulary	Appropriate Copyright Correct Credit Data leak Deceive Fair Fake Inappropriate Incorrect Index Information Keywords Network Privacy Rank Real Search engine TASK Web crawler	Beat Bugs Instructions Loop Melody Mindmap Music Performance Pitch Plan Rhythm Scratch Soundtrack Spacing Tempo Timbre Tutorials Typing	8-bit binary Addition ASCII Binary code Boolean Byte Construction Data transmission Decimal numbers Design Discovery Distance Hexadecimal Mars Rover Moon Numerical data Planet Radio signal RAM Scientist Sequence Signal Simulation Space Subtraction Transmit	Blocks Bluetooth Code block Designing Device Download Images Load Loop Micro:bit Pairing Pedometer Polling Reset Sabotage Scoreboard Screen Systematic Variables Wifi Wireless Wires	Animator Background Character Design Digital device Edit Evaluate Flip book Fluid movement Frames Model Moving images Onion skinning Still images Stop motion Storyboard Thaumatrope Zoetrope	3D Binary image CAD Fetch, decode, execute ID card Input JPEG Memory Online community Pixels RAM Responsible RGB ROM Safe
Online Safety	Online safety – runs 1 lesson per half term Learning about app permissions; the positive and negative aspects of online communication; that online information is not always factual; how to deal with online bullying and managing our health and wellbeing.					

KS2 National Curriculum

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 (word processing, sound, data handling)

Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact.

Year 6	Autumn A	Autumn B	Spring A	Spring B	Summer A	Summer B
Concepts of computer science	Computing systems and networks Bletchley Park	Programming Intro to Python	Data handling Big data 1	Creating Media History of Computers	Data handling Big Data 2	Skills Showcase Inventing a Product
Knowledge and understanding	Discovering the history of Bletchley and learning about code breaking and password hacking. Demonstrating digital literacy skills by creating presentations.	Using the programming language 'Python' to create designs and art. Learning how to create loops and nested loops to make their code more efficient.	Identifying how barcodes and QR codes work. Learning how infrared waves are used for the transmission of data while recognising the uses of RFID.	Writing, recording and editing radio plays set during WWII, learning about how computers have evolved.	Further developing understanding of how networks and the Internet are able to share information. Learning how big data can be used to design smart buildings.	Designing a product, pupils: evaluate, adapt and debug code to make it suitable for their needs and designing products in CAD and creating a website and video.
Learning outcomes	Pupils who are secure will be able to: • Explain that codes can be used for a number of different reasons and decode messages. • Explain how to ensure a password is secure and how this works. • Create a simple poster with information about Bletchley Park including the need to build electronic thinking machines to solve cipher codes. • Explain the importance of historical figures and their contribution towards computer science. • Present information about their historical figure in an interesting and engaging manner.	Pupils who are secure will be able to: • Iterate ideas, testing and changing throughout the lesson and explain what their program does. • Use nested loops in their designs, explaining why they need two repeats. • Alter the house drawing using Python commands; use comments to show a level of understanding around what their code does. • Use loops in Python and explain what the parts of a loop do. • Recognise that computers can choose random numbers; decompose the program into an algorithm and modify a program to personalise it.	Pupils who are secure will be able to: • Understand why barcodes and QR codes were created. • Create (and scan) their own QR code using a QR code generator website. • Explain how infrared can be used to transmit a Boolean type signal. • Explain how RFID works, recall a use of RFID chips, and type formulas into spreadsheets. • Take real-time data and enter it effectively into a spreadsheet. • Presenting the data collected as an answer to a question. • Recognising the value of analysing real-time data. • Analyse and evaluate transport data and consider how this provides a useful service to commuters.	Pupils who are secure will be able to: • Explain how to record sounds and add in sound effects over the top. • Produce a simple radio play with some special effects and simple edits which demonstrate an understanding of how to use the software. • Create a document that includes correct date information and facts about the computers and how they made a difference. • Demonstrate a clear understanding of their device and how it affected modern computers, including well-researched information with an understanding of the reliability of their sources. • Describe all of the features that we'd expect a computer to have including RAM, ROM, hard drive and processor, but of a higher specification than currently available.	Pupils who are secure will be able to: • Recognise that data can become corrupted within a network and that data sent in packets is more robust, as well as identify the need to update devices and software. • Recognise differences between mobile data and WiFi and use a spreadsheet to compare and identify high-use data activities and low-use data activities. • Make links between the Internet of Things and Big Data and give a basic example of how data analysis/analytics can lead to improvement in town planning. • Explain ways that Big Data or IoT principles could be used to solve a problem or improve efficiency within the school and prepare a presentation about their idea, considering the privacy of some data. • Present their ideas about how Big Data/IoT can improve the school and provide feedback to others on their presentations.	Pupils who are secure will be able to: • Evaluate code, understanding what it does and adapt existing to code for a specific purpose. • Debug programs and make them more efficient using sequence, selection, repetition or variables. • Design appropriate housing for their product using CAD software, including any input or output devices needed to make it work. • Create an appealing website for their product, aimed at their target audience which explains what their product is and what it does, using persuasive language. • Create an edited video of their project, articulating the key benefits. • Describe and show how to search for information online and be aware of the accuracy of the results presented.

Year 6	Autumn A	Autumn B	Spring A	Spring B	Summer A	Summer B
Sequence of Learning	• Learning about the history of computers and how they have evolved over time. Using past experiences to help solve new problems. • Writing increasingly complex algorithms for a purpose. • Debugging quickly and effectively to make a program more efficient. • Remixing existing code to explore a problem. • Changing a program to personalise it. • Evaluating code to understand its purpose. • Predicting code and adapting it to a chosen purpose. • Using search and word processing skills to create a presentation. • Understanding how search engines work. • Understanding the importance of secure passwords and how to create them. • Using search engines safely and effectively.	• Decomposing a program into an algorithm. • Writing increasingly complex algorithms for a purpose. • Debugging quickly and effectively to make a program more efficient. • Remixing existing code to explore a problem. • Using and adapting nested loops. • Programming using the language Python. • Changing a program to personalise it. • Evaluating code to understand its purpose. • Using logical thinking to explore software independently, iterating ideas and testing continuously.	• Understanding and identifying barcodes, QR codes and RFID. • Identifying devices and applications that can scan or read barcodes, QR codes and RFID. • Understanding how barcodes, QR codes and RFID work. • Gathering and analysing data in real time. • Creating formulas and sorting data within spreadsheets. • Learning how 'big data' can be used to solve a problem or improve efficiency.	• Learning about the history of computers and how they have evolved over time. • Using the understanding of historic computers to design a computer of the future. • Using search and word processing skills to create a presentation. • Planning, recording and editing a radio play. • Creating and editing sound recordings for a specific purpose.	• Understanding how corruption can happen within data during transfer (for example when downloading, installing, copying and updating files). • Understanding that computer networks provide multiple services. • Using search and word processing skills to create a presentation. • Creating formulas and sorting data within spreadsheets. • Learning about the Internet of Things and how it has led to 'big data'. • Learning how 'big data' can be used to solve a problem or improve efficiency.	• Using past experiences to help solve new problems. • Writing increasingly complex algorithms for a purpose. • Debugging quickly and effectively to make a program more efficient. • Remixing existing code to explore a problem. • Changing a program to personalise it. • Evaluating code to understand its purpose. • Predicting code and adapting it to a chosen purpose. • Using logical thinking to explore software independently, iterating ideas and testing continuously. • Creating and editing videos, adding multiple elements: music, voiceover, sound, text and transitions. • Using design software TinkerCAD to design a product. Creating a website with embedded links and multiple pages. • Understanding how search engines work. Using search engines safely and effectively.
Retrieval Vocabulary	Password	Algorithm Output Code Patterns Repeat Input Shape Instructions Loop	Algorithms	Computer Devices File	Bluetooth Corrupted Data	Algorithm Inputs Bugs Loops Output Coding Photos Debugging Product Design Program Edit Repetition Information Software
New Vocabulary	Acrostic Code Hero Brute force Discovery hacking Nth Letter Cipher Caesar cipher Pig Latin Chip and pin Pigpen cipher system Present Cipher Scrambled Code Secret Combination Secure Contribute Technological Convince advancement Date shift Trial and error Cipher Invention	Command Design Import Indentation Random Remix	Barcode Privacy Binary Proximity Boolean QR code Brand QR scanner Chips Radio waves Graphics RFID Commuter Signal Contactless Systems/data Data analyst Encrypted Transmission Infrared Wireless MagicBand	Background RAM noise Raspberry Pi Byte Record FX Reverb Gigabyte ROM Graphics Script Hard drive Smartphone Hardware Sound Kilobytes Sound Megabyte effects Memory storage Terrabytes Mouse Touch Operating screen system Track Overlay Trackpad Play Processor Trailer Radio play	Big Data QR codes Energy Revolution GPS RFID Improve SIM Infrared Simulation Internet of Smart city Things Smart Personal school Privacy Stop motion Threat WiFi Wireless	Adapt Manipulation Advert Opinions Electronic Screenshot Evaluate Search engine Facts Selection Image rights Sequence Images Snippets Influence Structures Variables Video Website
Online Safety	Online safety – runs 1 lesson per half term Learning to deal with issues online; about the impact and consequences of sharing information online; how to develop a positive online reputation; combating and dealing with online bullying and protective passwords.					

HEALD PLACE

Every Child a
Confident Learner