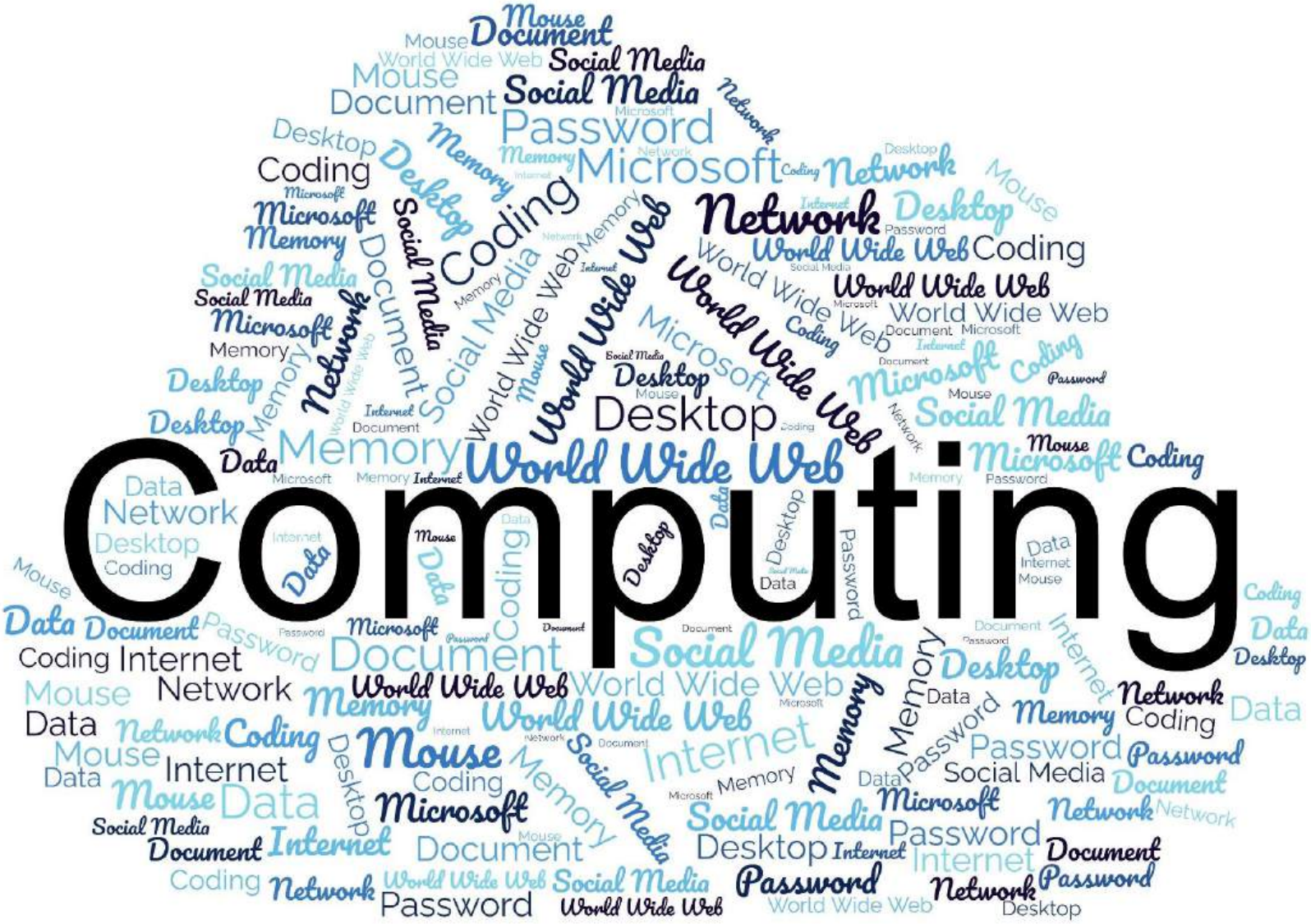


Computing at Sir Martin Frobisher Academy

Subject Leadership
2022/23

CONTENTS

Subject Leaders at SMFA

- Subject Leaders provide professional leadership for a subject or group of subjects to secure high-quality first teaching, a rich curriculum, and the effective use of resources. The success of this will be measured by the impact on learning and progress for pupils.
- We do not expect Subject Leaders to be an 'expert' in the subject they lead. What is important is that they have the overview of what is going well and what needs to be improved – based on evidence.
- Subject leaders at SMFA are part of both the Middle Leadership and the SMFA Extended Leadership Teams
- Each Subject Leader has an assigned Mentor (from SLT)

All Subject Leaders will

- Be part of our distributed leadership
- Utilise the expertise, passion, pedagogical awareness, and strengths of other leadership team members.
- Establish a collective responsibility for demonstrating that everyone makes a difference.
- Moving the school forward through driving the implementation aspect of each subject
- Professionally develop themselves and other staff team members
- Raise standards across all aspects of the curriculum.
- Enrich the curriculum.
- Share knowledge, expertise, skill, passion, and enthusiasm.

How does the role of Subject Leader fit into SMFA's Ofsted Statement of Action?

The staff, pupils and school community are working on areas identified in June 2023's Ofsted inspection as areas that need to be developed. The actions below link directly to the role of school Middle Leaders.

AFI 1 – Curriculum

"Most of the curriculum has been reviewed and newly implemented to take into account what pupils know. This process is further ahead in its development in reading and mathematics. In these areas, leaders consider the starting points of pupils carefully, so they build knowledge and understanding step by step. Teachers receive effective training and support. As a result, teachers plan learning that helps pupils build on prior learning. This ensures that pupils deepen their learning and are consequently generally achieving well."

Most of the curriculum has been reviewed and newly implemented. Aside from English and mathematics, leaders have identified gaps in pupils' learning and are further refining the curriculum to include what knowledge pupils need to learn to catch up. This includes pupils' knowledge of subject-specific vocabulary. Leaders should ensure that the curriculum they intend to offer is planned well, using the information they know about what pupils need to learn. Leaders should ensure that teachers are trained to implement the curriculum so that their delivery adheres to leaders' specification, ensuring that pupils catch up and are ready for the next stages of their education.

Aligning INTENT, IMPLEMENTATION AND IMPACT to ensure we meet the criteria for a good quality of education in the Education Inspection Framework

Intent:

Install a sense of enjoyment around using technology and to develop pupil's appreciation of its capabilities and the opportunities technology offers to, create, manage, organize, and collaborate.
Tinkering' with software and programs forms a part of the ethos of the scheme as we want to develop pupils' confidence when encountering new technology, which is a vital skill in the ever evolving and changing landscape of technology. Through our curriculum, we intend for pupils not only to be digitally competent and have a range of transferable skills at a suitable level for the future workplace, but also to be responsible online citizens.

This guidance was created to help equip children for life in the digital world, including developing their understanding of appropriate online behavior, copyright issues, being discerning consumers of online information and healthy use of technology.

Implementation:

Teacher's plan:

- To incorporate computing vocabulary
- Themed learning environments to immerse children in the subject.
- Links to good quality text
- A cycle of lessons for each topic, which plans for progression and depth.
- Progression in skills using relevant Milestones.
- Challenging questions to develop thinking skills.

Impact:

Our Computing Curriculum is high quality, well thought out and is planned to demonstrate progression. We measure the impact of our curriculum through the following methods:

- A reflection on standards achieved against the planned outcomes.
- Pupils make good progress through computing l skills demonstrated in learning.
- Pupil discussions about their learning

Long Term Plan

	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2	Online safety
EYFS	Set up continuous provision in your classroom: Computing through continuous provision	**Computing systems and networks** Using a computer	**Programming 1** All about instructions	**Computing systems and networks** Exploring hardware	**Programming 2** Programming Bee-Bots	**Data handling** Introduction to data	
Year 1	**Computing systems and networks** Improving mouse skills	**Programming 1** Algorithms unplugged	**Skills showcase** Rocket to the moon	**Programming 2** Programming Bee-bots Option 1: Bee-Bots Option 2: Virtual Bee-bots	**Creating media** Digital imagery	**Data handling** Introduction to data	**Online safety** Online safety Y1 (5 lessons)
Year 2	**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 Option 1: Using tablets Option 2: Using desktops/laptops	**Data handling** International Space Station	**Online safety** Online safety Y2 (4 lessons)

	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2	Online safety
Year 3	**Computing systems and networks 1**	**Programming**	**Computing systems and networks 2**	**Computing systems and networks 3**	**Creating media**	**Data handling**	**Online safety**
	Networks	Programming: Scratch	Emailing Option 1: Google Option 2: Microsoft Office 365	Journey inside a computer	Video trailers Option 1: Using devices other than iPads , Option 2: Using iPads	Comparison cards databases	Online safety Y3 (5 lessons)
Year 4	**Computing systems and networks**	**Programming 1**	**Creating media**	**Skills showcase**	**Programming 2**	**Data handling**	**Online safety**
	Collaborative Learning Option 1: Google Option 2: Microsoft Office 365	Further coding with Scratch	Website design Option 1: Google Option 2: Microsoft Office 365	HTML	Computational thinking	Investigating weather	Online safety Y4 (5 lessons)
Year 5	**Computing systems and networks**	**Programming 1**	**Data handling**	**Programming 2**	**Creating media**	**Skills showcase**	**Online safety**
	Search engines	Programming music Option 1: Sonic Pi , Option 2: Scratch	Mars Rover 1	Micro:bit	Stop motion animation Option 1: Stop motion studio Option 2: Using cameras	Mars Rover 2	Online safety Y5 (5 lessons)
Year 6	**Computing systems and networks**	**Computing systems and networks**	**Data handling**	**Programming**	**Data handling**	**Skills showcase**	**Online safety**
	Bletchley Park and the history of computers	**AI To be published October 2024.**	Big data 1	Intro to Python	Big data 2	Inventing a product	Online safety Y6 (6 lessons)

Progression of Skills

Progression of skills

Computer science

	EYFS	Year 1	Year 2
Hardware	Learning how to operate a camera to take photographs of meaningful creations or moments. Learning how to explore and tinker with hardware to develop familiarity and introduce relevant vocabulary. Recognising and identifying familiar letters and numbers on a keyboard. Developing basic mouse skills such as moving and clicking.	Learning how to operate a camera or tablet to take photos and videos. Learning how to explore and tinker with hardware to find out how it works. Learning where keys are located on the keyboard.	Understanding what a computer is and that it's made up of different components. Recognising that buttons cause effects and that technology follows instructions. Learning how we know that technology is doing what we want it to do via its output. Developing confidence with the keyboard and the basics of touch typing.
Networks and data representation	N/A	N/A	N/A

Progression of skills

Computer science

	EYFS	Year 1	Year 2
Computational thinking	Using logical reasoning to understand simple instructions and predict the outcome.	Learning that decomposition means breaking a problem down into smaller parts. Using decomposition to solve unplugged challenges. Using logical reasoning to predict the behaviour of simple programs. Developing the skills associated with sequencing in unplugged activities. Following a basic set of instructions. Assembling instructions into a simple algorithm.	Articulating what decomposition is. Decomposing a game to predict the algorithms used to create it. Learning that there are different levels of abstraction. Explaining what an algorithm is. Following an algorithm. Creating a clear and precise algorithm.
Programming	Following instructions as part of practical activities and games. Learning to give simple instructions. Learning to debug instructions, with the help of an adult, when things go wrong.	Programming a Floor robot to follow a planned route. Learning to debug instructions when things go wrong. Learning to debug an algorithm in an unplugged scenario.	Using logical thinking to explore software, predicting, testing and explaining what it does. Using an algorithm to write a basic computer program.

Progression of skills

Computer science

	Year 3	Year 4	Year 5	Year 6
Hardware	Understanding what the different components of a computer do and how they work together. Drawing comparisons across different types of computers. Learning about the purpose of routers.	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.	Learning that external devices can be programmed by a separate computer.	Learning about the history of computers and how they have evolved over time. Understanding and identifying barcodes, QR codes and RFID. Identifying devices and applications that can scan or read barcodes, QR codes and RFID.
Networks and data representation	Understanding the role of the key components of a network. Identifying the key components within a network, including whether they are wired or wireless. Understanding that websites and videos are files that are shared from one computer to another. Learning about the role of packets. Understanding how networks work and their purpose. Recognising links between networks and the internet. Learning how data is transferred.	Understanding that computer networks provide multiple services, such as the World Wide Web, and opportunities for communication and collaboration.	Learning the vocabulary associated with data: data and transmit. Recognising that computers transfer data in binary and understanding simple binary addition. Learning that messages can be sent by binary code, reading binary up to eight characters and carrying out binary calculations.	N/A

Progression of skills

Computer science

	Year 3	Year 4	Year 5	Year 6
Computational thinking	Using decomposition to explain the parts of a laptop computer. 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 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 abstraction to identify the important parts when completing both plugged and unplugged activities.	Decomposing animations into a series of images. Decomposing a story to be able to plan a program to tell a story. Predicting how software will work based on previous experience. Writing more complex algorithms for a purpose.	Decomposing a program into an algorithm. Using past experiences to help solve new problems. Writing increasingly complex algorithms for a purpose.
Programming	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.	Creating algorithms for a specific purpose. Coding a simple game. Using abstraction and pattern recognition to modify code. Incorporating variables to make code more efficient.	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.	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. Predicting code and adapting it to a chosen purpose.

Progression of skills

Information technology

	EYFS	Year 1	Year 2
Using software	Using a simple online paint tool to create digital art.	Using a basic range of tools within graphic editing software. Taking and editing photographs. Developing control of the mouse through dragging, clicking and resizing of images to create different effects. Developing understanding of different software tools.	Developing word processing skills, including altering text, copying and pasting and using keyboard shortcuts. Using word processing software to type and reformat text. Using software (and unplugged means) to create story animations. Creating and labelling images.
Using email and internet searches	N/A	Recognising devices that are connected to the internet. Understanding that we are connected to others when using the internet.	Searching for appropriate images to use in a document.
Using data	Representing data through sorting and categorising objects in unplugged scenarios. Exploring branch databases through physical games.	N/A	Collecting and inputting data into a spreadsheet. Interpreting data from a spreadsheet.
Wider use of technology	N/A	Recognising common uses of information technology, including beyond school. Understanding some of the ways we can use the internet.	Learning how computers are used in the wider world.

Progression of skills	Information technology			
	Year 3	**Year 4**	**Year 5**	**Year 6**
Using software	Taking photographs and recording video to tell a story. Using software to edit and enhance their video adding music, sounds and text on screen with transitions.	Use online software for documents, presentations, forms and spreadsheets. Using software to work collaboratively with others.	Using logical thinking to explore software more independently, making predictions based on their previous experience. Using software programme Sonic Pi/Scratch to create music. Using the video editing software to animate. Identify ways to improve and edit programs, videos, images etc. Independently learning how to use 3D design software package TinkerCAD.	Using logical thinking to explore software independently, iterating ideas and testing continuously. Using search and word processing skills to create a presentation.
Using email and internet searches	N/A	Understanding why some results come before others when searching. Understanding that information found by searching the internet is not all grounded in fact. Searching the internet for data.	Developing searching skills to help find relevant information on the internet.	Understanding how search engines work.

Progression of skills

Information technology

	Year 3	Year 4	Year 5	Year 6
Using data	N/A	Understanding that data is used to forecast weather. Recording data in a spreadsheet independently. Sorting data in a spreadsheet to compare using the 'sort by…' option. Designing a device which gathers and records sensor data.	Understanding how data is collected in remote or dangerous places. Understanding how data might be used to tell us about a location.	Understanding how barcodes, QR codes and RFID work. Gathering and analysing data in real time. Creating formulas and sorting data within spreadsheets.
Wider use of technology	Recognising how social media platforms are used to interact.	Understanding that software can be used collaboratively online to work as a team.	Learn about different forms of communication that have developed with the use of technology.	Learning how 'big data' can be used to solve a problem or improve efficiency.

Progression of skills

Digital Literacy

EYFS	Year 1	Year 2	
Recognising that a range of technology is used for different purposes. Learning to log in and log out.	Logging in and out and saving work on their own account. When using the internet to search for images, learning what to do if they come across something online that worries them or makes them feel uncomfortable. Understanding how to interact safely with others online. Recognising how actions on the internet can affect others. Recognising what a digital footprint is and how to be careful about what we post.	Learning how to create a strong password. Understanding how to stay safe when talking to people online and what to do if they see or hear something online that makes them feel upset or uncomfortable Identifying whether information is safe or unsafe to be shared online. Learning to be respectful of others when sharing online and ask for their permission before sharing content. Learning strategies for checking if something they read online is true.	
Year 3	**Year 4**	**Year 5**	**Year 6**
Recognising that different information is shared online including facts, beliefs and opinions. Learning how to identify reliable information when searching online. Learning how to stay safe on social media. Considering the impact technology can have on mood. Learning about cyberbullying. Learning that not all emails are genuine, recognising when an email might be fake and what to do about it.	Recognising that information on the internet might not be true or correct and that some sources are more trustworthy than others. Learning to make judgements about the accuracy of online searches. Identifying forms of advertising online. Recognising what appropriate behaviour is when collaborating with others online. Reflecting on the positives and negatives of time spent online. Identifying respectful and disrespectful online behaviour.	Identifying possible dangers online and learning how to stay safe. Evaluating the pros and cons of online communication. Recognising that information on the internet might not be true or correct and learning ways of checking validity. Learning what to do if they experience bullying online. Learning to use an online community safely	Learning about the positive and negative impacts of sharing online. Learning strategies to create a positive online reputation. Understanding the importance of secure passwords and how to create them. Learning strategies to capture evidence of online bullying in order to seek help. Using search engines safely and effectively. Recognising that updated software can help to prevent data corruption and hacking.

Vocabulary

	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2	Online Safety
Year 1	Account	Algorithm	Annotate	Algorithm	Bar chart	Background	App
	Click	Artificial intelligence	Cells	Bee-Bot	Block graph	Blurred	Appropriate
	Clipart	Bug	Components	Code	Branching database	Camera	Device
	Computer	Chunks	Create	Debug	Categorise	Clear	Digital footprint
	Drag	Code	Data	Demonstration	Chart	Crop	Feelings
	Drag and drop	Computer	Debug	Explain	Click and drag	Delete	Going online
	Layers	Debug	Designing	Explore	Compare	Device	In-person interactions
	Log off	Decompose	Digital content	Filming	Count	Digital camera	Internet
	Log on	Device	Digital image	Inputting	Data	Download	Kindness
	Mouse	Directions	Document	Instructions	Data collection	Drag and drop	Offline activity
	Password	Input	E-document	Precise	Data record	Edit	Online activity
	Predict	Instructions	Edit	Predict	Data representation	Editing software	Online experience
	Resize	Manageable	Editing software	Program	Edit	Filter	Online interactions
	Screen (monitor)	Order	Editing program	Review	Input	Image	Online safety
	Software	Organise	Evaluate	Test	Keyboard	Import	Personal information
	Tool	Output	Folder	Tinker	Line graph	Internet	Pop-up
	Username	Program	Graphics	Video	Mouse	Keyword	Posting online
		Problem	Input		Information	Online	Report
		Solution	Instructions		Label	Photograph	Responsible digital citizen
		Specific	Log in		Pictogram	Resize	Screen time
		Tasks	Photo		Pie chart	Save as	Sharing online
		Virtual assistant	Program		Process	Screen	Stranger
			Order		Record	Search engine	Technology
			Robot		Resize	Sequence	Trusted adult
			Save		Sort	Software	Unkind
			Sequence		Table	Storage space	Website
			Share		Tally	Visual effects	
			Software		Values		
			Spreadsheet				
			Table				

	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2	Online Safety
Year 2	battery	Abstraction	back button	algorithm	Animation	algorithm	accepting
	buttons	Algorithm	backspace	block coding	Background	astronaut	consent
	computer	Artificial intelligence	bold	bug	Decompose	column	denying permission
	desktop	Bug	copy	debug	Digital device	data	fake
	electricity	Clear	copyright	evaluate	Drawing	digital content	giving permission
	input	Correct	cut	micro:bit	Flipbook	essential	offline
	invention	Data	delete	MakeCode	Frames	experiment	online
	keyboard	Debug	forward button	program	Moving images	Goldilocks zone	password
	laptop	Decompose	highlight	programming	Object	interactive map	permission
	mouse	Error	image	sequence	Onion skinning	International Space Station	personal information
	output	Key features	import		Plan	input	pop-up
	robot	Loop	italic		Still images	monitor	pressure
	screen (monitor)	Predict	keyboard			row	private information
	tablet	Unnecessary	keyboard shortcut			satellite	real
	technology		layout			sensor	reliable
	wire		navigate			space	sharing online
			paste			spreadsheet	source
			redo			survival	trusted adult
			search			temperature	
			space bar			thermometer	
			text				
			text effects				
			touch typing				
			underline				
			undo				
			word processing				

	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2	Online Safety
Year 3	device	accepting	algorithm	Attachment	Algorithm	Categorise	accurate
	file	consent	animation	Bcc (Blind carbon copy)	Assemble	Category	age restrictions
	internet	denying permission	application	Cc (Carbon copy)	CPU (central processing unit)	Chart	autocomplete
	network	fake	code	Compose	Data	Data	belief
	network switch	giving permission	code block	Content	Decompose	Database	charity
	packet data	offline	debug	Cyberbullying	Desktop	Excel	content
	router	online	decompose	Document	Disassemble	Fields	digital device
	server	password	game	Domain	GPU (graphics processing unit)	Filter	fact
	the cloud	permission	interface	Download	Hard drive	Graph	fake news
	user	personal information	loop	Email	HDD (hard disk drive)	Information	hoax
	WiFi	pop-up	predict	Email account	Infinite loop	Interpret	internet
	wired	pressure	program	Email address	Input	PDF	internet of things
	wireless	private information	remixing code	Emoji	Keyboard	Questionnaire	opinion
	wireless access point	real	repetition code	Emotions	Laptop	Record	online emotions
		reliable	review	Fake	Memory	Representation	organisation
		sharing online	Scratch	Font	Microphone	Sort	permission
		source	sprite	Genuine	Monitor	Spreadsheet	privacy settings
		trusted adult	tinker	Hacker	Mouse		reliable
				Icons	Output		search
				Inbox	Photocopier		search engine
				Information	Program		share
				Link	QR Code		smart devices
				Log in	RAM (random access memory)		social media platforms
				Log out	ROM (read only memory)		
				Negative language	Storage		
				Password	Tablet device		
				Personal information	Technology		
				Positive language	Touchscreen		
				Reply	Touchpad		
				Responsible digital citizen			

				Scammer			
				Settings			
				Send			
				Sign in			
				Spam email			
				Subject bar			
				Theme			
				Tone			
				Username			
				Virus			
				WiFi			

	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2	Online Safety
Year 4	average	code block	Assessment	code	abstraction	accurate	accuracy
	collaborate	conditional statement	Audience	content	algorithm	climate zone	ad
	comment	coordinates	Checklist	copyright	code	condensation	advantage
	data	decompose	Collaboration	CSS	computational thinking	cylinder	advertisement
	data representation	feature	Content	end tag	decomposition	degree Celsius	belief
	edit	information	Contribution	fake news	input	evaporation	bot
	e-document	negative number	Create	hacker	logical reasoning	extreme weather	computer
	email	orientation	Design	heading	output	filming	disadvantage
	insert (file)	position	Embed	HTML	pattern recognition	forecast	distraction
	multiple choice	program	Evaluate	HTML tags	script	heat sensor	fact
	numerical data	project	Features	internet browser	sequence	lightning	hashtag
	online	script	Google SItes	paragraph	variable	measurement	Implications
	presentation	sprite	Hobby	remixing		pinwheel	in-app purchases
	rating	stage	Homepage	start tag		presenter	influencer
	reply	tinker	Hyperlinks	text		rain	opinion
	resolve	variable	Images	unplugged		satellite	program
	reviewing comments		Insert	URL		script	recommendation
	share		Online	web page		sensor data	reliable
	slide		Plan	web page elements		solar panel	risk
	spreadsheet		Progress			temperature	screen time
	suggestion		Published			thermometer	search results
	survey		Record			tornado	snippets
	teamwork		Review			weather	sponsored
	transition		Style			weather forecast	trustworthy
			Subpage			wind speed	
			Tab				
			Theme				
			Web page				
			Website				
			World Wide Web				

	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2	Online Safety
Year 5	algorithm	Beat	8-bit binary	Algorithm	animation	3D	accurate
	copyright	Buffer	addition	Animation	animator	Algorithm	advice
	credit	Bugs	ASCII	App	background	Binary image	app
	fake news	Coding	binary code	Blocks	character	CAD	application
	inaccurate	Commands	Boolean	Bluetooth	decomposition	Compression	app permissions
	index	Debug	byte	Code block	design	CPU	biography
	keywords	Decompose	CPU	Connection	digital device	Data	bullying
	online	Error	data	Create	edit	Drag and drop	communication
	page rank	Format	data transmission	Debug	evaluate	Fetch, decode, execute	emojis
	search engine	Instructions	decimal numbers	Decompose	flipbook	ID card	health
	TASK	Live loops	discovery	Designing	fluid movement	Input	in-app purchases
	web crawler	Loop	distance	Desktop	frames	JPEG	information
	website	Melody	Hexadecimal	Device	model	Memory	judgement
	www	Mind map	input	Download	moving images	Online community	meme
		Music	Mars Rover	Images	onion skinning	Operating system	mental health
		Output	the Moon	Input	still images	Output	mindfulness
		Performance	numerical data	Instructions	stop motion	Pixels	negative contribution
		Pitch	output	Laptop	storyboard	RAM	online
		Play	planet	Load	thaumatrope	Responsible	online communication
		Predict	radio signal	Loop	zoetrope	RGB	opinion
		Programming	RAM	Micro:bit		ROM	organisation
		Rehearsal	scientist	Outputs		Safe	password
		Repetition	sequence	Pairing			personal information
		Rhythm	signal	Pedometer			positive contribution
		Sleep	simulation	Polling			real world
		Sonic Pi	space	Predict			strong password
		Soundtrack	subtraction	Program			summarise
		Spacing		Repetition			support
		Tempo		Reset			trusted adult
		Timbre		Sabotage			well-being
		Tinker		Scoreboard			
		Tutorials		Screen			
		Typing		Systematic			
		Typo		Tablet			
				Tinkering			
				USB			
				Variables			
				Wifi			
				Wireless			
				Wires			

	Autumn 1	Autumn 2	Spring 1	Spring 2	Summer 1	Summer 2	Online Safety
Year 6	acrostic code	TBC	Algorithm	algorithm	Big Data	adapt	anonymity
	audio advert		Code	barcode	Bluetooth	advert	antivirus
	brute force hacking		Command	Boolean	Corrupted	algorithm	biometrics
	Caesar cipher		Design	brand	Data	bugs	block
	chip and PIN system		Import	chip	Energy	coding	consent
	cipher		Indentation	commuter	GPS	debugging	digital footprint
	combination		Input	contactless	Improve	design	digital personality
	date shift cipher		Instructions	data	Infrared	edit	financial information
	discovery		Loop	encrypt	Internet of Things	electronic	hacking
	invention		Output	infrared	Personal	evaluate	inappropriate
	Nth letter cipher		Patterns	proximity	Privacy	facts	malware
	password		Random	QR code	QR codes	image rights	online bullying
	pigpen cipher		Remix	QR scanner	Revolution	images	online reputation
	scrambled		Repeat	radio waves	RFID	influence	password
	script		Shape	RFID	SIM	information	personal information
	secret			signal	Simulation	inputs	phishing
	secure			spreadsheet	Smart city	loops	privacy settings
	technological advancement			systems analyst	Smart school	manipulation	private
	trial and error			transmission	Stop motion	opinions	reliable source
				wireless	Threat	output	report
					WiFi	photos	respect
					Wireless	product	scammers
						program	screen grab
						repetition	screenshot
						screenshot	secure
						search engine	selfie
						selection	software updates
						sequence	two-factor authentication
						snippets	URL
						software	username
						structures	
						variables	
						video	
						website	

Planning (examples)

Brute force hacking – Year 6

Learning objective	Success criteria
• To understand the importance of having a secure password.	• I can describe what is meant by brute force hacking. • I can understand why it is important to have a secure password. • I can explain why a longer password is more secure than a short one.

Before the lesson

Have ready:

- Presentation: Gimmie Five!
- Presentation: Code Challenge.
- Presentation: Brute force.
- Presentation: Website passwords.
- Digital devices (one between two).
- Link: Scratch – Brute force emulator.

Print in advance:

- Resource: Four-digit brute force emulator

Recap and recall

Display the presentation: Gimmie five! And allow time for paired discussion.

- Take feedback on key points the children recalled from last lesson.

Attention grabber

Display slide 1 of the *Presentation: Code challenge* and arrange the children in pairs. Ask one child in each pair to write down a three-digit code on a whiteboard and explain that their partner will try to guess what it is.

Encourage them to think logically about how to solve the code (e.g. starting with 000, 001, 002, 003 and so on). Link the exercise to working systematically through maths problems. Allow five to ten minutes for the children to solve the code. Explain that if they solve it quickly, they can swap roles and try again.

Show slide 2 and ask if anyone was able to solve the code. Discuss the methods tried and whether the children approached the problem methodically or using random guesswork.

Main event

Explain that using trial and error to guess passwords is called **brute force hacking**. It involves methodically trying every possible combination until the correct answer is guessed.

Use the link: Scratch - Brute force emulator to show the children the Scratch game (a similar activity that they have just done). Ask the children to try password 253 and see how long the emulator takes to figure it out.

Display slide 1 of the *Presentation: Brute force* and explain to the children that a computer can find a solution faster than a person can. Ask the children:

- **Why do you think this is?** (Answers may include that computers can think faster than humans, computers do not get tired and work without stopping; computers do not make simple mistakes; computers can process lots of information.)

Using slide 2, explain to the children that they will remix the code for the brute force emulator to try out different numbers of digits in their password. Start by asking the children the questions on the slide.

Show slide 3 and invite the children to look at the existing code and explain what they think each line of it means. For a detailed explanation of the code, see the *Teacher video: Brute force*.

Allow the children time to experiment with the existing code to create their emulator and explore both making a password easier to guess and a more secure password.

Talk the children through the following steps if they need more guidance:

1. Show slide 4 and explain that the children need to change the text that asks for a three-digit number and then create a new variable called 4digit (or similar).
2. Using slide 5, show them that the code within the **repeat** block must be changed so that the fourth digit increases by one each time instead of the third digit. Another **if** block must be added before the existing ones: **if 4digit = 10, then change 3digit by 1 and set 4digit to 0**.
3. Show slide 6 to explain that a slight amendment is needed to **set current guess** block near the end of the script. It assumes the first digit is in the hundreds but with four digits, the first digit would be in the thousands.

Refer to or hand out the *Resource: Four-digit brute force emulator*, which shows one way you can achieve this code to any children who require support (see Adaptive teaching).

Wrapping up

Display slide 1 of the *Presentation: Website passwords* and discuss a real-world example of when all debit and credit cards in the UK switched to a chip and PIN system because it was more secure than a signature that could be forged.

Show slide 2 and ask the children how many different combinations of passwords there are for a four-digit PIN (personal identification number). Click to reveal the answer is 10,000 because each digit has ten possible combinations, so the calculation is 10 x 10 x 10 x 10. The children may recognise that the highest number used is 9999 but explain that 0000 can also be used as a password, which makes it up to 10,000.

Using slide 3, ask the children why some websites require a password at least eight characters long.

Show slides 4–6 and discuss with the children why longer passwords are more secure and why it takes longer for a computer to decrypt using trial and error.

Invention – Year 2

Learning objective	Success criteria
• To create a design for an invention.	• I can include and input and an output as part of my invention. • I can explain how it works, including how to control it. • I can label my design clearly.

Before the lesson

Have ready:

- Presentation: Gimmie Five!
- Presentation: Rocket-powered chair.
- Presentation: Invention.

Print in advance:

- Activity: Invention design (one each).

Recap and recall

Display the *Presentation: Gimme five* and allow time for paired discussion.

Take feedback on key points the children recalled from the last lesson.

Attention grabber

Display slide 1 of the *Presentation: Rocket-powered chair* to discuss with the children what an invention is.
Arrange the children into pairs and give them time to formulate questions about the image. For example:

- How does it work?
- Is it safe?
- How do you control it?
- Can it only go up?
- Do you need a special suit to use it?

Allow time for the children to think of some questions and ask some of the children to pose their questions to the rest of the class.
Ensure the children recognise that technology follows instructions and is designed by humans.

Main event

Display slide 1 of the *Presentation: Invention* and explain to the children that they will be inventors. Explain that they will create their own inventions using all the knowledge they have learnt about computers. Point out that their invention should have a computer inside but what it does is entirely up to them.

Using slide 2, remind the children of the technology safari they went on in the last lesson and ask them the questions to prompt a discussion.

Show slide 3 and, as a class, create an example invention together. Ask the children for ideas about what they could design or use one of the examples on slide 4. Draw a sketch to depict your shared class invention on the board for the children to refer to.

Display slides 5 and 6. Ask the children to consider the shared invention's inputs and outputs and use the questions below to start a discussion.

- **Will we have buttons or a microphone to start the invention?**
- **Is there a touchscreen to input more detailed instructions?**
- **Do we need a lever or joystick to operate part of the invention?**
- **How do we know the invention is operating? Is there a screen or flashing lights showing progress?**

Demonstrate how to annotate the drawing on the board of the shared invention with the information elicited from the questions. Display slide 7 and ask the children why annotation is so important.

Hand out copies of the *Activity: Invention design* to each child and ask them to plan their inventions. Circulate the classroom and question them about their inputs and outputs to ensure they consider them.

Stop the class before the end and model how to write an explanation of their invention. Link this writing to explanation texts in English if appropriate. Allow the children to finish their inventions and write an explanation of how they work.

Wrapping up

Ask children to self-assess their inventions based on the following criteria:

- I have an input.
- I have an output.
- I have a clear drawing.
- I have annotated my drawing.
- I have explained how it works.

Knowledge Organisers (examples)

Computing - Bletchley Park and the history of computers

acrostic code	A type of code where the first letter of each word, line or paragraph spells out a hidden message.
audio advert	An advert for a product using only sound recordings that might be heard on radio or podcasts.
background noise	A (secondary) sound that is not the focus beacuse there is another primary sound.
brute force hacking	When a hacker uses different methods, such as trial and error, to break into secured information.
chip and PIN system	A secure payment system where a plastic bank card (like a debit or credit card) has a chip that the card owner can access by entering a personal identification number (PIN).
cipher	A way of writing information in a secret code.
CPU (central processing unit)	The brain of the computer that processes all the data from input and output devices and runs the programs on the computer.
password	A unique mix of letters, numbers or symbols used to protect personal information online.
trial and error	Trying different methods to solve a problem until one works.

Use headphones to monitor audio and minimise background noise. During recording, speak clearly and at a consistent volume. After recording, use editing software to remove any mistakes or unwanted sounds.

Ada Lovelace was a mathematician known for her work on Charles Babbage's early computer, the Analytical Engine.

Tim Berners-Lee invented the World Wide Web.

Steve Jobs co-founded Apple Inc. and was a key figure in the personal computing revolution.

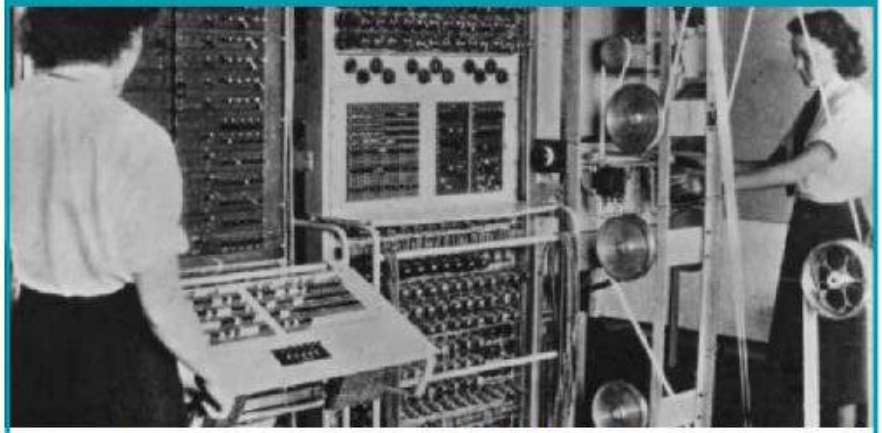
Computing - Bletchley Park and the history of computers

Z3 (1941): German engineer Konrad Zuse created the world's first working, programmable, fully automatic, digital computer.

Apple I (1976): one of the first personal computers that helped to launch the personal computing revolution.

IBM PC (1981): set the standard for personal computers and led to their widespread adoption in homes and businesses.

The Colossus



Codebreakers during World War 2 constructed the Colossus computer in 1943. This enormous machine was the world's first electronic, programmable computer.

Date shift cipher

Date used to encrypt the message: 1st January 1984. In number format this would read: 01 01 1984

Original message:	h	e	l	l	o
Each letter shifted:	0	1	0	1	1
Encrypted message:	h	f	l	m	p

a b c d e f g h i j k l m n o p q r s t u v w x y z

Inventing a product

Adapt	To adjust something or someone, in order to improve a situation or a purpose, to become more effective.
Advertisement	Used to promote or announce something to the public through videos, voice or images broadcast on television and radio. 
Algorithm	A sequence of instructions which, when followed, solve a problem.
Bug	A mistake or error in the code, stopping the program from working as intended.
CAD	Computer-aided design software used to create graphics, diagrams or other visuals.
Computer code	A set of instructions written in programming language, to tell a computer what to do.
Code (verb)	To write in programming language (code).
Design	To make, draw or write plans for something. 
Edit	To change and amend something.
Electronic components 	The parts that make up an electrical device or product with processing capabilities.
Image rights	The picture or image belongs to someone or a company. 
Image	A picture of people or objects.
Input	Information sent to a computer by an input device such as a keyboard or mouse for processing.

Kapow Primary

Information 	Knowledge which can be remembered, written in documents or stored in different forms as data, such as in video files and audio recordings.
Invention	A new device or process that solves a problem.
Loop	A repeated sequence of instructions. 
Output	Information or data that is sent by the computer to an output device such as a printer or speakers.
Photo 	Images that are taken by a device called a camera.
Program	A series of code that instructs the computer to perform specific tasks.
Repetition (code)	To create loops in your program, to make it more efficient.
Screenshot	A captured image of what is currently on the screen or monitor.
Selection (programming)	Where an algorithm or program branches off. It allows the computer to change what it does, depending on the information received.
Sequence	A set order or pattern for something to follow. 
Variable	This could be a number or text, that can change each time the program is run and often in combination with selection to change the end result of the program.
WWW	The acronym used to express the 'World Wide Web'. It is found at the beginning of website addresses e.g. www.kapowprimary.com

Computing - Programming: Scratch

algorithm	Steps or instructions to solve a problem or complete a task.
animation	Bringing concepts to life through 2D or 3D moving pictures or photographs, e.g. cartoons.
coding	Writing instructions for a computer.
code block	Similar to puzzle pieces, they can be dragged, dropped and snapped together to create an algorithm.
debug	To find and fix errors in code.
decomposition	Breaking a problem into smaller parts.
loop	A repeated sequence of instructions.
remixing code	Altering code that already exists.
sprite	An image or character that moves or reacts to commands.
tinker	To explore and play with something to discover its key functions.

Scratch code blocks colour key:

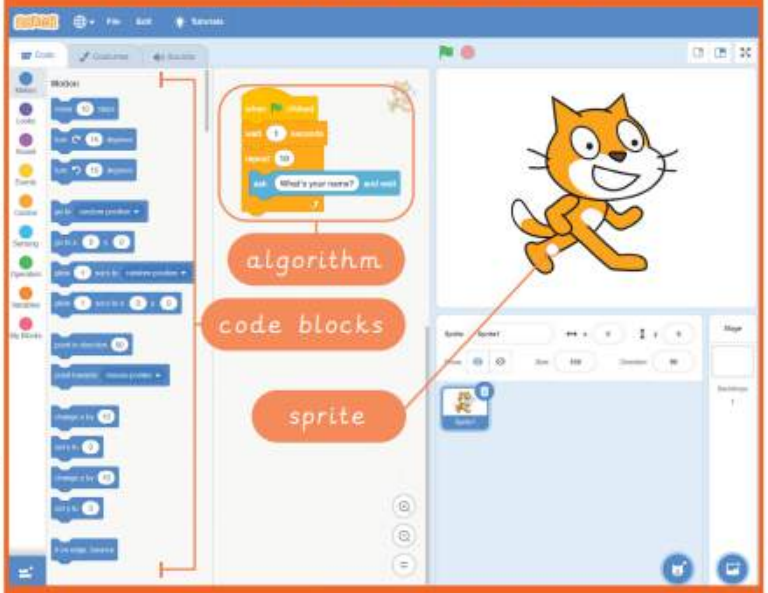
Motion | Sound | Control | Operators | My Blocks

Looks | Events | Sensing | Variables

Key facts

Kapow Primary

Scratch is a coding program in which you can develop interactive games and animations.

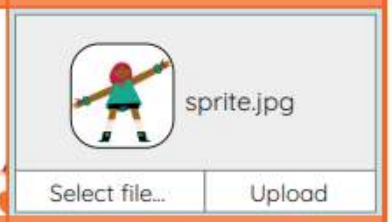


There are four different ways to add a sprite.

choose a sprite

paint your own sprite

generate a random (surprise) sprite



upload a sprite