



COMPUTING

Purpose of study

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

Aims

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

- ♣ can understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation
- ♣ can analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems
- ♣ can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems
- ♣ are responsible, competent, confident and creative users of information and communication technology.

KS 1 Statements
KS 2 Statements

Key Stage 1 pupils should be taught to:

- understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions
- create and debug simple programs
- use logical reasoning to predict the behaviour of simple programs
- use technology purposefully to create, organise, store, manipulate and retrieve digital content
- recognise common uses of information technology beyond school
- use technology safely and respectfully, keeping personal information private; identify where to go for help and support when they have concerns about content or contact on the internet or other online technologies.

Key Stage 2 pupils should be taught to:

- design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts
- use sequence, selection, and repetition in programs; work with variables and various forms of input and output
- use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs
- understand computer networks including the internet; how they can provide multiple services, such as the world wide web; and the opportunities they offer for communication and collaboration
- use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content

		<ul style="list-style-type: none"> 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 <p>use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact.</p>
<p>Early Years- Three- and four-Year Olds</p>	<p><u>Communication and language</u> Use a wide range of vocabulary</p> <p><u>Personal, Social and Emotional Development</u> Remember rules without needing an adult to remind them.</p> <p><u>Physical Development</u> Match their developing physical skills to tasks and activities in the setting</p> <p><u>Understanding the World</u> Explore how things work.</p>	
<p>Early Years – Reception</p>	<p><u>Communication and language</u> Understand how to listen carefully and why listening is important. Learn new vocabulary. Use new vocabulary through the day. Listen to and talk about stories to build familiarity and understanding.</p> <p><u>Personal, Social and Emotional Development</u> Show resilience and perseverance in the face of a challenge. Know and talk about the different factors that support their overall health and wellbeing: -sensible amounts of ‘screen time’</p> <p><u>Physical Development</u> Develop their small motor skills so that they can use a range of tools competently, safely and confidently.</p> <p><u>Expressive Arts and Design</u> Explore, use and refine a variety of artistic effects to express their ideas and feelings.</p>	<p>ELG</p> <p><u>Listening, Attention and Understanding</u> Listen attentively and respond to what they hear with relevant questions, comments and actions when being read to and during whole class discussions and small group interactions</p> <p><u>Speaking</u> Participate in small group, class and one-to-one discussions, offering their own ideas, using recently introduced vocabulary. Offer explanations for why things might happen, making use of recently introduced vocabulary. Express their ideas and feelings about their experiences using full sentences.</p> <p><u>Personal, Social and Emotional Development / Managing Self</u> Be confident to try new activities and show independence, resilience and perseverance in the face of challenge. Explain the reasons for rules, know right from wrong and try to behave accordingly.</p> <p><u>Expressive Arts and Design / Creating with Materials</u> Safely use and explore a variety of materials, tools and techniques, experimenting with colour, design, texture, form and function.</p>

Computing Knowledge Categories

Algorithms – Be able to comprehend, design, create and evaluate algorithms.	Design and Development – Understand the activities involved in planning, creating and evaluating computing artefacts.
Computer Networks – Understand how networks can be used to retrieve and share information, and how they come with associated risks	Effective Use of Tools – Use software tools to support computing work.
Computer Systems – Understand what a computer is, and how its constituent parts function together as a whole.	Impact of Technology – Understand how individuals, systems and society as a whole interact with computer systems.
Creating Media – Select and create a range of media including text, images, sounds and video.	Programming – Create software to allow computers to solve problems.
Data and Information – Understand how data is stored, organised, and used to represent real-world artefacts and scenarios.	Safety and Security – Understand risks when using technology and how to protect individuals and systems
The discipline of computing is represented through these categories and the organisation of content. Whilst all strands are present at all phases, they are not always taught explicitly.	



Whole School COMPUTING Delivery Model



Delivered through: DFE Computing Scheme

<https://teachcomputing.org/>

	AUT 1	AUT 2	SPR 1	Online safety week	SPR 2	SUM1	SUM2
EYFS		To take photos on cameras & iPads		Internet Safety	Beebots (Year 1 Unit: Programming A - Moving a robot)		
Y1	Computing Systems: Technology around us: <ul style="list-style-type: none">- Technology in our classroom- Using technology- Developing mouse skills- Using a computer keyboard- Developing keyboard skills- Using a computer responsibly (Algorithms and Computer Systems)		Creating Media: Digital Painting <ul style="list-style-type: none">- How can we paint using computers- Using shapes and lines- Making careful choices- Why did I choose that?- Painting all by myself- Comparing computer art and painting (Effective use of tools and Creating media)	Internet Safety		Programming B: Programming animations. <ul style="list-style-type: none">- Comparing tools- Joining blocks- Make a change- Adding sprites- Project Design- Following my design (Programming and Design and Development)	



Y2	<p>Creating Media: Digital photography</p> <ul style="list-style-type: none"> - Taking photographs - Landscape or portrait? - What makes a good photograph? - Lighting - Effects - Is it real? <p>(Effective use of tools, Computing systems)</p>		<p>Programming B: Programming quizzes</p> <ul style="list-style-type: none"> - Scratch Jr recap - Outcomes - Using a design - Changing a design - Designing and creating a program - Evaluating <p>(Programming, Design and Development)</p>	Internet Safety		<p>Data and Information: Pictograms</p> <ul style="list-style-type: none"> - Counting and comparing - Enter data - Creating pictograms - What is an attribute? - Comparing people - Presenting information <p>(Data and Information, Effective use of tools)</p>	
Y3	<p>Data and information - Branching databases</p> <ul style="list-style-type: none"> - Yes or no questions - Making groups - Creating a branching database - Structuring a branching database - Using a branching database - Two ways of presenting information <p>(Data and information, Effective use of tools)</p>		<p>Creating Media: Desktop publishing</p> <ul style="list-style-type: none"> - Words and pictures - Can you edit it? - Great template! - Can you add content? - Lay it out - Why desktop publishing? <p>(Effective use of tools, Creating Media)</p>	Internet Safety		<p>Programming A - Sequencing sounds</p> <ul style="list-style-type: none"> - Introduction to scratch - Programming sprites - Sequences - Ordering commands - Looking good - Making an instrument <p>(Programming, Design and development)</p>	

Y4	<p>Computing systems and networks: The Internet</p> <ul style="list-style-type: none"> - Connecting networks - What is the internet made of? - Sharing information - What is a website - Who owns the web? - Can I believe what I read? <p>(Networks, Safety and security)</p>		<p>Programming A: Repetition in shapes</p> <ul style="list-style-type: none"> - Programming a screen turtle - Programming letters - Patterns and repeats - Using loops to create shapes - Breaking things down - Creating a program <p>(Algorithms, Programming)</p>	Internet Safety		<p>Creating Media: Photo editing</p> <ul style="list-style-type: none"> - Changing digital images - Changing the composition of images - Changing images for different uses - Retouching images - Fake images - Making and evaluating a publication <p>(Effective use of tools, Creating media)</p>	
Y5	<p>Computing systems and networks: Systems and searching</p> <ul style="list-style-type: none"> - Systems - Computer systems and us - Searching the web - Selecting search results - How search results are ranked - How are searches influenced <p>(Networks, Effective use of tools)</p>		<p>Creating media: Video production</p> <ul style="list-style-type: none"> - What is video? - Filming techniques - Using a storyboard - Planning a video - Importing and editing videos - Video evaluation <p>(Creating Media, Design and development)</p>	Internet Safety		<p>Programming B - Selection in quizzes</p> <ul style="list-style-type: none"> - Exploring conditions - Selecting outcomes - Asking questions - Planning a quiz - Testing a quiz - Evaluating a quiz <p>(Algorithms, Programming)</p>	

Y6	<p>Creating media: Web page creation</p> <ul style="list-style-type: none"> - What makes a good website? - How would you lay out your web page? - Copyright or copywrong? - How does it look? - Follow the breadcrumbs - Think before you link <p>(Creating Media, Design and development)</p>		<p>Data and information: Introduction to Spreadsheets</p> <ul style="list-style-type: none"> - What is a spreadsheet? - Modifying spreadsheets - What's the formula? - Calculate and duplicate - Event planning - Presenting data <p>(Effective use of tools, Data and information)</p>	Internet Safety		<p>Programming A: Variables in games</p> <ul style="list-style-type: none"> - Introducing variables - Variables in programming - Improving a game - Designing a game - Design to code - Improving and sharing <p>(Programming, Design and development)</p>	
----	--	--	---	-----------------	--	--	--

KS3 Links	<p>Pupils should be taught to:</p> <ul style="list-style-type: none"> • design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems • understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem • use 2 or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions • understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming; understand how numbers can be represented in binary, and be able to carry out simple operations on binary numbers [for example, binary addition, and conversion between binary and decimal] • understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems • understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits • undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users • create, reuse, revise and repurpose digital artefacts for a given audience, with attention to trustworthiness, design and usability
-----------	---

- understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct, and know how to report concerns

 <h2 style="text-align: center;">Continuous Provision - Computing</h2> 						
	Y1	Y2	Y3	Y4	Y5	Y6
	Milestone 1		Milestone 2		Milestone 3	
Cross - curricular links	Geography – explore weather and climate in the UK and around the world, describe geographical features Science – observe seasonal changes, investigate differences in living things, identify, name, describe, classify and compare properties and changes Art & Design - explore a variety of techniques		Across the curriculum.			
Games	Create games using Scratch. Ongoing access to these apps.		Create more complex games in a programming language Use commands: motion. Looks, sound, drawings, events, control, sensing, variables and lists and operators			
Devices and Apps	Provide devices and apps to support learning across the curriculum		Use advanced features of apps across the curriculum. Move from adding to and retrieving databases to construction.			
Art & Design	Apps for drawing and painting help children to develop ideas and experiment with techniques		Use advanced features of apps to refine techniques and experiment with effects			
DT	Apps for drawing and painting can be used to model and label designs		Explore games and apps that encourage design			
English	Apps to support phonic knowledge, writing, imaginative description and sentence construction		Use apps that develop fluency in grammar, imaginative description and reading, writing, oracy, drama and film making			
Geography	Apps that contain maps and information about locations. Weather tracking apps from around the world		Apps that help children to understand the nature of places around the world			
History	Access to online resources to aid research and find evidence		Online resources that help children research, find and evaluate the quality of evidence			
Languages	Translators and games that help children develop fluency and vocab use		Apps that help children recognise and use vocab in reading, writing, and speaking as well as learning about the culture of France			
Maths	Apps that promote fluency in number and calculations Position and direction taught within programming Shape geometry practised with gaming apps that require sorting and classification.		Access to maths games on a regular basis. Use programming to create geometric shapes to develop understanding of angles as a measure of turn.			
Music	Use a range of composing apps building layers of sound with different instruments		Use a range of composing apps incorporating digitally created and live recorded sound.			
PE	Use of REAL PE online resources. Heart rate monitors		Use of REAL PE online resources. Heart rate monitors Create databases to interpret and compare data			
Science	Use of online databases for plants, animals, seasons and materials.		Apps to research scientific knowledge			