Computing Curriculum: Year 6

# What are the aims and intentions of this curriculum?

That children:

- design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts
- use sequence, selection, and repetition in programs; work with variables and various forms of input and output
- use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs
- understand computer networks including the internet; how they can provide multiple services, such as the world wide web; and the opportunities they offer for communication and collaboration
- use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content
- select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information
- use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact.

| Term | Topic | Key Learning (Knowledge & Skills) | Key Vocabulary |
|---|---|---|---|
| Autumn 1 | Computing systems:<br>Bletchley Park | • Understand the importance of having a secure password and what "brute force hacking" is.<br>• Know that the first computers were created at Bletchley Park to crack the Enigma code to help the war effort in World War 2.<br>• Know about some of the historical figures that contributed to technological advances in computing.<br>• Understand what techniques are required to create a presentation using appropriate software<br>• Learn about the history of computers and how they have evolved over time.<br>• Using past experiences to help solve new problems<br>• Writing increasingly complex algorithms for a purpose.<br>• Debugging quickly and effectively to make a program more efficient. | acrostic code, brute force hacking, Caesar cipher, cipher, encrypt, Nth letter cipher, pigpen cipher, technological advancement, trial and error |

| | | | |
|---|---|---|---|
| | | • Remixing existing code to explore a problem.<br>• Changing a program to personalise it.<br>• Evaluating code to understand its purpose.<br>• Predicting code and adapting it to a chosen purpose. | |
| Autumn 2 | Programming: Intro to python | • Know that there are text-based programming languages such as Logo and Python.<br>• Know that nested loops are loops inside of loops.<br>• Understand the use of random numbers and remix Python code.<br>• Decomposing a program into an algorithm.<br>• Writing increasingly complex algorithms for a purpose.<br>• Debugging quickly and effectively to make a program more efficient.<br>• Remixing existing code to explore a problem.<br>• Using and adapting nested loops.<br>• Programming using the language Python.<br>• Changing a program to personalise it.<br>• Evaluating code to understand its purpose.<br>• Using logical thinking to explore software independently, iterating ideas and testing continuously. | algorithm, computer command, decompose, import, loop, nested loops, random numbers, remix, script libraries, variable |
| Spring 1 | Data Handling: Big Data 1 | • Know how barcodes and QR codes work<br>• Know that computer networks provide multiple services<br>• Know how data is collected and stored<br>• Know how big data can be used to help people in a variety of different scenarios<br>• Identify and understand barcodes, QR codes and RFID<br>• Identify devices and applications that can scan or read barcodes, QR codes and RFID<br>• Acknowledge that corruption can happen during data transfer<br>• Understand that computer networks provide multiple services | data privacy, encrypt, RFID, transmission, analyst, QR code, Boolean, radio waves, signal |
| Spring 2 | Creating Media: History of Computers | • Know that radio plays are plays where the audience can only hear the action so sound effects are important.<br>• Know that sound clips can be recorded using sound recording software.<br>• Know that sound clips can be edited and trimmed.<br>• Learn about the history of computers and how they have evolved over time. | background noise, byte, CPU, memory storage, OS, RAM, ROM |

| | | | |
|---|---|---|---|
| | | • Use the understanding of historic computers to design a computer of the future.<br>• Use search and word processing skills to create a presentation.<br>• Plan, recording and editing a radio play.<br>• Create and editing sound recordings for a specific purpose. | |
| Summer 1 | Data Handling: Big Data 2 | • Know that data can become corrupted within a network but this is less likely to happen if it is sent in 'packets'.<br>• Know that devices or that are not updated are most vulnerable to hackers.<br>• Know the difference between mobile data and WiFi.<br>• Understanding how corruption can happen within data during transfer (for example when downloading, installing, copying and updating files).<br>• Understanding that computer networks provide multiple services.<br>• Using search and word processing skills to create a presentation.<br>• Creating formulas and sorting data within spreadsheets.<br>• Learning about the Internet of Things and how it has led to 'big data'.<br>• Learning how 'big data' can be used to solve a problem or improve efficiency | Big Data, Bluetooth, Corrupt Data, Digital Revolution, GPS, Infrared Waves, Internet of Things, QR Code, RFID, SIM, Computer Simulation, Smart School/city |
| Summer 2 | Skills showcase: Inventing a Product | • know what designing an electronic product involves.<br>• know which programming software/ language is best to achieve a purpose.<br>• know the building blocks of computational thinking e.g. sequence, selection, repetition, variables and inputs and outputs.<br>• Using past experiences to help solve new problems.<br>• Writing increasingly complex algorithms for a purpose.<br>• Debugging quickly and effectively to make a program more efficient.<br>• Remixing existing code to explore a problem.<br>• Changing a program to personalise it.<br>• Evaluating code to understand its purpose.<br>• Predicting code and adapting it to a chosen purpose<br>• Using logical thinking to explore software independently, iterating ideas and testing continuously.<br>• Creating and editing videos, adding multiple elements: music, voiceover, sound, text and transitions. | Adapt, Advertisement, Algorithm, Bug, CAD, Computer Code, Code, Design, Edit, Electronic components, Image rights, input, invention, loop, output, repetition, sequence, variable |

| | | | |
|---|---|---|---|
| | | • Using design software TinkerCAD to design a product.<br>• Creating a website with embedded links and multiple pages. | |
| Continuous | Online Safety | • know that a digital footprint means the information that exists on the internet as a result of a person's online activity.<br>• know what steps are required to capture bullying content as evidence.<br>• understand that it is important to manage personal passwords effectively<br>• understand what it means to have a positive online reputation.<br>• know some common online scams<br>• Learn about the positive and negative impacts of sharing online.<br>• Learn strategies to create a positive online reputation.<br>• Understand the importance of secure passwords and how to create them<br>• Learn strategies to capture evidence of online bullying in order to seek help.<br>• Use search engines safely and effectively.<br>• Recognise that updated software can help to prevent data corruption and hacking | online, block, privacy settings, report, consent, screen grab, inappropriate, respect, block and report, URL, two factor authentication, hacking, biometrics, personal information, phishing, scammers, antivirus, malware |