

# Y7

## Python

Fundamental programming concepts. Learn to use the `print()` function for outputs, `input()` for user inputs, and variables to store data. It explores control structures like `if-else` for decision-making and introduces data types such as strings, integers, and booleans. The unit emphasises problem-solving through practical coding, teaching foundational skills in writing, debugging, and understanding Python programs.

T6

## Binary (Bits and Bobs)

Looks at binary concepts essential to computing. It introduces binary and decimal number systems, conversion between them, and binary addition. Key topics include ASCII for text representation, bitmap images, and sound sampling, all explaining how data is encoded in binary. Students learn practical skills and theoretical knowledge to understand how computers store, process, and represent various types of data.

T5

## Introduction to HTML

Introduces the basics of web development, focusing on HTML as the language of the web. It covers the use of tags to structure content, including headings, paragraphs, images, and hyperlinks. Students learn about attributes like font styling, alignment, and background colours, as well as the importance of tag pairs and closing tags. The topic builds foundational skills for creating functional, visually structured webpages.

T4

## Back to the Future!

Understanding the history of computer science provides valuable perspective on its foundations and impact. This topic explores key figures like Tim Berners-Lee, George Boole, and Alan Turing, highlighting their contributions to computing. From Boole's logical foundations to Turing's breakthroughs in algorithms and Berners-Lee's creation of the World Wide Web, their work showcases the evolution of computer science and its enduring influence on modern technology and innovation.

T3

## Computer Hardware

The key aspects of computer hardware, including input and output devices, their roles in a computer system, and definitions of essential components such as the CPU, RAM, hard drive, and motherboard. Explaining how computers process instructions, using the Fetch-Decode-Execute cycle, including analogies like the office worker to clarify concepts. Detailed descriptions of components and their functions are also provided, along with explanations of clock speed and its importance.

T2

## Introduction to Scratch

Introduction to the ideas about programming in Scratch. It explores Scratch programming, covering sprites, variables, inputs, outputs, and logic through blocks and IF statements. This integrated approach develops technical understanding, IT skills, and programming logic for practical and creative problem-solving.

T1

# Computer Science

# Y8

## E-Safety

Essential E-Safety skills, focusing on foundational knowledge and topics. Topics include BIG Data, Digital Footprints, Online Presence, Upstanders and Allies as well as Copyright and Fairtrade. The unit offers the chance to examine real world examples of E-Safety, supported by the ongoing research and evidence base from e-safety experts..

T6

## Artificial Intelligence

Introduces artificial intelligence and the basic fundamentals of AI systems. Looking at AI for healthcare, education and daily devices. Learning to use AI within the scope of daily life, whilst also being wary of the issues surrounding current AI systems. The unit explores integrating AI into modern day life, as well as the ongoing development and issues within AI systems.

T5

## Digital World

Explores safe and effective online practices. It covers smart searching, evaluating website reliability, and advanced search techniques using Boolean operators. Key topics include copyright, plagiarism, and protecting personal information online. It addresses cyber safety, managing online risks like malware and cyberbullying, and fostering awareness of responsible digital behaviour. Students learn practical strategies to navigate and utilise the digital world confidently and securely.

T4

## Python

Fundamental programming concepts. Learn to use the `print()` function for outputs, `input()` for user inputs, and variables to store data. It explores control structures like `if-else` for decision-making and introduces data types such as strings, integers, and booleans. The unit emphasises problem-solving through practical coding, teaching foundational skills in writing, debugging, and understanding Python programs.

T3

## HTML and CSS

Introduces web development fundamentals. Looking at HTML for webpage structure using tags and CSS for styling and presentation. Learning to use `<div>` tags to create sections, apply CSS rules with selectors, and style elements like borders, padding, and fonts. The unit explores integrating HTML and CSS to create visually appealing and well-structured web pages, building foundational skills in website design and coding.

T2

## Networks

Introduces key concepts in computer networking. It covers types of networks, including LANs and WANs, and essential equipment like switches and network interface cards. Understanding data packets, IP addresses, DNS, and URLs. The unit explores threats like malware and phishing and prevention methods like firewalls, encryption, and anti-malware software. It emphasises how networks enable data sharing, communication, and the functioning of the internet.

T1

# Computer Science

# Y9

## Algorithms

This unit examines, the key computational approaches to solving real world problems. These include: Abstraction, Decomposition, Thinking concurrently, Thinking procedurally as well as exploring the use of these in development of programs. Learners will be given the chance to engage in a variety of activities linking to problem solving through the application of these algorithmic principles.

T6

## Memory and Storage

Memory and Storage covering key components like the Ram and ROM, as well as HDD, SDD and alternative forms of storage. Memory distinguishing between RAM, ROM, and virtual memory. Secondary storage discussing storage types—magnetic, optical, and solid-state—evaluating their capacities, speeds, durability, and costs for different applications.

T5

## System Architecture

CPU architecture covering key components like the ALU, CU, and registers, along with the fetch-decode-execute cycle. CPU performance exploring factors like clock speed, cores, cache, and embedded systems. Memory distinguishing between RAM, ROM, and virtual memory. Secondary storage discusses storage types—magnetic, optical, and solid-state—evaluating their capacities, speeds, durability, and costs for different applications.

T4

## Advanced Python

Developing robust programming skills, including advanced concepts like count-controlled iterations with for loops, input validation, authentication, and error handling. Exploring procedural programming techniques, learning to design secure and efficient programs that respond dynamically to user inputs. Tasks include creating structured outputs, managing data types, and implementing defensive design strategies. This unit emphasises critical thinking and problem-solving for real-world coding challenges.

T3

## Advanced Python

Developing robust programming skills, including advanced concepts like count-controlled iterations with for loops, input validation, authentication, and error handling. Exploring procedural programming techniques, learning to design secure and efficient programs that respond dynamically to user inputs. Tasks include creating structured outputs, managing data types, and implementing defensive design strategies. This unit emphasises critical thinking and problem-solving for real-world coding challenges.

T2

## HTML, CSS and Java Script

Extension to previous work on web development, focusing on the structure, design, and interactivity of webpages. HTML is taught as the foundational layer for structuring content, CSS for styling and presentation, and JavaScript for adding dynamic and interactive features. Learning to use JavaScript functions, IF statements, and user inputs to create responsive webpages, developing key coding and problem-solving skills applicable in real-world scenarios.

T1

# Computer Science