Computing Curriculum Structure and Progression

Digital Provision at Abbey Hill is arranged into 3 strands: Digital Literacy, Computing and Information Technology. Pupils have access to Curriculum iPads and laptops to explore and apply their digital learning journey.

Digital Literacy (2 or 3 half terms)	Computing (2 half terms)	Information Technology (1 half term)
Developing our pupils as digital citizens so that they are both content creators and safe and responsible citizens of a digital world.	Developing our pupils' understanding of the systems, networks and infrastructure through which our technologies operate.	Developing computational thinking including programming and coding.
Digital outcomes embedded in our curriculum	Rodocodo	TeachComputing Computing Systems and Networks Unit

Our Digital Literacy Outcomes

When delivering Digital Literacy units, staff should use the Digital Literacy strands to support their intended learning.

Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
• Infographic PicCollage • Video tour • Weather Forecast	• Infographic PicCollage • Photo Map	• Keynote recount of school trip • Cooking demonstration video	• Documentary Video • Podcast	• Sketches app art outcome • Podcast for discussion (GarageBand) • Podcast (Audacity)	• Numbers project report • Green screen demonstration video • Podcast (Audacity)

The Abbey Hill CURRICULUM

Computing Overview

	Autumn	Spring	Summer
1	Video tour of school	Weather Report Infographic	Vlogging TeachComputing Systems and Networks
2	Photo Map Rodocodo	Infographic TeachComputing Systems and Networks	Vlogging
3	E-safety Campaign Green Screen	Keynote TeachComputing Systems and Networks	Cooking Video
4	E-safety Campaign Green Screen	Podcast Rodocodo	Documentary Video TeachComputing Systems and Networks
5	Podcast Rodocodo	Skeches TeachComputing Systems and Networks	Podcast
6	E-safety Campaign Green Screen	Numbers Report Rodocodo	Vlogging TeachComputing Systems and Networks

Achieving our dreams together

Abbey Hill Primary and Nursery School

Disciplinary skills in the strands of our curriculum

Digital Literacy

Creators

Early Years	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Can understand that technology is used at home and school for different reasons.	Can recognise that technology is for creating, storing, manipulating and retrieving content. Can understand how other people in our school & family use technology	Can recognise that technology is for creating, storing, manipulating and retrieving content. Can understand how other people use technology in their community.	Can understand that technology is for creating, storing, manipulating and retrieving content for specific purpose. Can recognise that they can share information internally and use variety of software.	Can recognise that technology is for creating, storing, manipulating and retrieving content. Can understand they can share information internally and use variety of software.	Can recognise that technology is for creating, storing, manipulating and retrieving content. Can understand that content can be shared externally, combining variety of software and range of devices.	Can recognise that technology is for creating, storing, manipulating and retrieving content. Can understand that content can be shared externally, combining variety of software and range of devices.

Citizens

Early Years	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Can understand who their trusted adults are. Can understand that it is ok to say no. Understands what to do when they feel worried online Understands that it is important to be kind online and offline Understands healthy online habits	Can understand what private information is. Can understand who can help them with technology	Can understand that technology can help them and needs to be looked after respectfully. Can understand different ways of keeping information private. Can understand where to go for help with technology using contacts (people they know) and technology.	Can understand their responsibility to be safe online. Can understand respectful, acceptable/ unacceptable behaviour with technology.	Can understand where to report concerns with technology (content & contact) within apps and programmes they are using. Can understand respectful, acceptable/ unaccepta ble behaviour with technology	Can understand various reporting sites/ helplines. Can access different ways of receiving help (linked to responsibility) Can understand that not everyone uses technology safely. Can understand the importance and risks associated with acceptable/	Can understand various reporting sites/ helplines. Can access different ways of receiving help (linked to responsibility) Can understand the importance and risks associated with acceptable/ unacceptable behaviour

Understands how to take care when using devices.					unacceptable behaviour	

Computing

Strand	Early Years	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Sequencing	To understand that a sequence is a set of actions performed in the correct order to achieve something To understand and explain what the commands do To understand that there can be more than 1 solution to a problem	To understand and explain what each command will do To create simple programs using a variety of commands	To understand and explain what each command will do To create simple programs using a variety of commands	To understand that decomposition is the process of breaking down larger problems into smaller, more manageable parts To use decomposition to break down larger problems into multiple parts	To understand that decomposition is the process of breaking down larger problems into smaller, more manageable parts	To write complex programs using Conditional Loops, Functions and Selection. To write complex programs that include Variables, Loops and Functions	To write complex programs that include Variables and Loops (basic and nested) To write complex programs that include Variables, Loops and Functions
Debugging	To understand that bugs are errors in code To understand that debugging is the process of correcting bugs	To debug simple programs with 1 or 2 bugs	To explain the run-step-fix method of debugging To debug programs which include Loops and Functions	To explain the run-step-fix method of debugging To debug programs which include Loops and Functions	To debug programs which include Loops and Functions	To debug programs which include Functions and Loops	To debug programs which include Functions, Loops and Selection

Creating Loops		To understand that Loops are used when you want to repeat actions To identify that using Loops creates more optimal solutions To use Loops with a program to control character's movement	To identify that using Loops creates more optimal solutions To use pattern recognition to identify Loops To create simple programs using Loops To use Loops to control the character's movement and rotation	To identify that Lopps create more optimal solutions To use pattern recognition to identify Loops To use Loops to control the character's movement and rotation	To understand that Loops can be placed within other Loops, and that these are called Nested Loops To use Nested Loops within simple programs To write complex programs using selection and loops (basic and nested) To write complex programs using Selection, Loops and Functions.	To understand the difference between counting loops and conditional loops To understand that Conditional Loops repeat until a certain condition has been reached To create programs using Functions and Loops To write complex programs using Selection, Loops and Functions To write complex programs using Conditional Loops, Functions and Selection To write complex programs that include Variables, Loops and Functions	To understand the difference between Counting Loops and Conditional Loops To understand that Conditional Loops repeat until a certain condition has been reached To write complex programs that include Variables and Loops (basic and nested) To write complex programs that include Variables, Loops and Functions

Functions		To create programs using ready-made Functions	To understand that a function is a section of a program that performs a specific task and can be used multiple times To create programs that use ready-made Functions To debug programs which include Functions	To identify that functions create more optimal solutions To complete ready-made Functions that are not fully complete, and use them in sequences To debug programs which include Functions	To create a Function and use it in a program To write complex programs using Selection, Loops and Functions.	To write complex programs using Selection, Loops and Functions To write complex programs using Conditional Loops, Functions and Selection To write complex programs that include Variables, Loops and Functions	To write complex programs that include Variables and Loops (basic and nested) To write complex programs that include Variables, Loops and Functions
Selection					To understand that Selection is how a computer program makes decisions, and that those decisions are based on conditions To use selection within simple programs To write complex programs using selection and loops (basic and nested) To write complex programs using Selection, Loops and Functions.	To understand that Selection is how a computer program makes decisions, and that those decisions are based on conditions To write complex programs using Selection, Loops and Functions To write complex programs using Conditional Loops, Functions and Selection To write complex programs that include Variables, Loops and Functions	To understand that Selection is how a computer program makes decisions, and that those decisions are based on conditions To use the If Else command to write more complex Selection chains that instruct the character to perform different actions. To write complex programs using Conditional Loops, Functions and Selection

Variables						To understand that variables help computers remember values that can change To write complex programs that include Variables, Loops and Functions To use variables within simple programs	To understand that variables help computers remember values that can change To use variables within a simple program To write complex programs that include Variables and Loops (basic and nested) To write complex programs that include Variables, Loops and Functions

Information Technology

Purpose

Early Years	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Can understand how technology may be used at home or school	Can understand how other people may use technology in our community	Can understand how technology helps us store and retrieve content	Can understand how technology is used for different purposes	Can understand how people collaborate using technology	Can understand how a range of devices or software can be used for one purpose	Can understand how to select devices and/or software for a purpose.

Networks

Early Years	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6

Can understand how devices connect beyond school			Can understand how devices can be connected to make networks	Can understand that the internet is a network of networks including the WWW,	Can understand and recognise how IT systems connect to make a searchable internet	Can understand and explain how data can be collaborated with online

Data

Early Years	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Can organise and group data. Can present data			Can recognise how data can be gathered and assembled	Can recognise how data may be digitally processed	Can understand how data can be digitally analysed	Can understand how data can be organised and presented