

## Computing : Knowledge Progression and Overview

**INTENT:** We recognise the integral role technology plays in our lives. It is paramount, that we provide our students with a deep-rooted knowledge and understanding of the digital world. It is our aim, for all students to develop a secure skillset, that allows them to navigate safely throughout the digital world; and empowers them to lead the way in responsible online behaviour. Although technology is certain to play a pivotal role in their lives; what it will look like, its purpose and how they interact with it – remains uncertain. Therefore, encouraging students to think creatively, with open minds, and with a resilience to overcome problems is vital to a safe secure future. An understanding of computer science provides them with the tools to manipulate and engage with the starting blocks of technology. Whereas information technology allows students to select appropriate technology and interact with it accurately and with purpose. Finally, digital literacy provides the skills, not only to remain safe online, with a clear understanding of what to do in specific situations; but also an awareness of how to behave, and understand consequences to inappropriate behaviour.



### Computing Progression – Computer Science (National Curriculum in bold)

YEAR 1	YEAR 2
<p><b>Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions.</b></p> <p><i>Children understand that an algorithm is a set of instructions used to solve a problem or achieve an objective. They know that a computer program turns an algorithm into code that the computer can understand.</i></p> <p><b>Create and debug simple programs.</b></p> <p><i>Children can work out what is wrong with a simple algorithm when the steps are out of order, e.g. The Wrong Sandwich in Purple Mash and can write their own simple algorithm, e.g. Colouring in a Bird activity. Children know that an unexpected outcome is due to the code they have created and can make logical attempts to fix the code, e.g. Bubbles activity in 2Code.</i></p> <p><b>Use logical reasoning to predict the behaviour of simple programs.</b></p> <p><i>When looking at a program, children can read code one line at a time and make good attempts to envision the bigger picture of the overall effect of the program. Children can, for example, interpret where the turtle in 2Go challenges will end up at the end of the program.</i></p>	<p><b>Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions.</b></p> <p><i>Children can explain that an algorithm is a set of instructions to complete a task. When designing simple programs, children show an awareness of the need to be precise with their algorithms so that they can be successfully converted into code.</i></p> <p><b>Create and debug simple programs.</b></p> <p><i>Children can create a simple program that achieves a specific purpose. They can also identify and correct some errors, e.g. Debug Challenges: Chimp. Children’s program designs display a growing awareness of the need for logical, programmable steps.</i></p> <p><b>Use logical reasoning to predict the behaviour of simple programs.</b></p> <p><i>Children can identify the parts of a program that respond to specific events and initiate specific actions. For example, they can write a cause and effect sentence of what will happen in a program.</i></p>

YEAR 3	YEAR 4	YEAR 5	YEAR 6
<p><b>Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts.</b></p> <p><i>Children can turn a simple real-life situation into an algorithm for a program by deconstructing it into manageable parts. Their design shows that they are thinking of the desired task and how this translates into code. Children can identify an error within their program that prevents it following the desired algorithm and then fix it.</i></p> <p><b>Use sequence, selection and repetition in programs; work with variables and various forms of input and output.</b></p> <p><i>Children demonstrate the ability to design and code a program that follows a simple sequence. They experiment with timers to achieve repetition effects in their programs. Children are beginning to understand the difference in the effect of using a timer command rather than a repeat command when creating repetition effects.</i></p> <p><b>Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs.</b></p> <p><i>Children's designs for their programs show that they are thinking of the structure of a program in logical, achievable steps and absorbing some new knowledge of coding structures. For example, repetition and use of timers. They make good attempts to 'step through' more complex code in order to identify errors in algorithms and can correct this. e.g. In programs such as Logo, they can 'read' programs with several steps and predict the outcome accurately.</i></p> <p><b>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.</b></p>	<p><b>Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts.</b></p> <p><i>When turning a real-life situation into an algorithm, the children's design shows that they are thinking of the required task and how to accomplish this in code using coding structures for selection and repetition. Children make more intuitive attempts to debug their own programs.</i></p> <p><b>Use sequence, selection and repetition in programs; work with variables and various forms of input and output.</b></p> <p><i>Children's use of timers to achieve repetition effects are becoming more logical and are integrated into their program designs. They understand 'IF statements' for selection and attempt to combine these with other coding structures including variables to achieve the effects that they design in their programs. As well as understanding how variables can be used to store information while a program is executing, they are able to use and manipulate the value of variables. Children can make use of user inputs and outputs such as 'print to screen'. e.g. 2Code</i></p> <p><b>Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs.</b></p> <p><i>Children's designs for their programs show that they are thinking of the structure of a program in logical, achievable steps and absorbing some new knowledge of coding structures. For example, 'IF' statements, repetition and variables. They can trace code and use step-through methods to identify errors in code and make logical attempts to correct this. In programs such as Logo, they can 'read' programs with several steps and predict the outcome accurately.</i></p> <p><b>Understand computer networks, including the internet; how they can provide multiple services, such as the World Wide Web, and the</b></p>	<p><b>Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts.</b></p> <p><i>Children may attempt to turn more complex real-life situations into algorithms for a program by deconstructing it into manageable parts. Children are able to test and debug their programs as they go and can use logical methods to identify the approximate cause of any bug but may need some support identifying the specific line of code.</i></p> <p><b>Use sequence, selection and repetition in programs; work with variables and various forms of input and output.</b></p> <p><i>Children can translate algorithms that include sequence, selection and repetition into code with increasing ease and their own designs show that they are thinking of how to accomplish the set task in code utilising such structures. They are combining sequence, selection and repetition with other coding structures to achieve their algorithm design.</i></p> <p><b>Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs.</b></p> <p><i>When children code, they are beginning to think about their code structure in terms of the ability to debug and interpret the code later, e.g. the use of tabs to organise code and the naming of variables.</i></p> <p><b>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.</b></p> <p><i>Children understand the value of computer networks but are also aware of the main</i></p>	<p><b>Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts.</b></p> <p><i>Children are able to turn a more complex programming task into an algorithm by identifying the important aspects of the task (abstraction) and then decomposing them in a logical way using their knowledge of possible coding structures and applying skills from previous programs. Children test and debug their program as they go and use logical methods to identify the cause of bugs, demonstrating a systematic approach to try to identify a particular line of code causing a problem.</i></p> <p><b>Use sequence, selection and repetition in programs; work with variables and various forms of input and output.</b></p> <p><i>Children translate algorithms that include sequence, selection and repetition into code and their own designs show that they are thinking of how to accomplish the set task in code utilising such structures, including nesting structures within each other. Coding displays an improving understanding of variables in coding, outputs such as sound and movement, inputs from the user of the program such as button clicks and the value of functions.</i></p> <p><b>Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs.</b></p> <p><i>Children are able to interpret a program in parts and can make logical attempts to put the separate parts of a complex algorithm together to explain the program as a whole.</i></p>

<p>Children can list a range of ways that the Internet can be used to provide different methods of communication. They can use some of these methods of communication, e.g. being able to open, respond to and attach files to emails using 2Email. They can describe appropriate email conventions when communicating in this way</p>	<p><b>opportunities they offer for communication and collaboration.</b></p> <p>Children recognise the main component parts of hardware which allow computers to join and form a network. Their ability to understand the online safety implications associated with the ways the internet can be used to provide different methods of communication is improving.</p>	<p>dangers. They recognise what personal information is and can explain how this can be kept safe. Children can select the most appropriate form of online communications contingent on audience and digital content, e.g. 2Blog, 2Email, Display Boards.</p>	<p><b>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.</b></p> <p>Children understand and can explain in some depth the difference between the internet and the World Wide Web. Children know what a WAN and LAN are and can describe how they access the Internet in school.</p>
--	---	---	--

## Computing Progression – Information Technology (National Curriculum in bold)

YEAR 1		YEAR 2	
<p><b>Use technology purposefully to create, organise, store, manipulate and retrieve digital content.</b></p> <p>Children are able to sort, collate, edit and store simple digital content e.g. children can name, save and retrieve their work and follow simple instructions to access online resources, use Purple Mash 2Quiz example (sorting shapes), 2Code design mode (manipulating backgrounds) or using pictogram software such as 2Count.</p>		<p><b>Use technology purposefully to create, organise, store, manipulate and retrieve digital content.</b></p> <p>Children demonstrate an ability to organise data using, for example, a database such as 2Investigate and can retrieve specific data for conducting simple searches. Children are able to edit more complex digital data such as music compositions within 2Sequence. Children are confident when creating, naming, saving and retrieving content. Children use a range of media in their digital content including photos, text and sound.</p>	
YEAR 3	YEAR 4	YEAR 5	YEAR 6
<p><b>Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content.</b></p> <p>Children can carry out simple searches to retrieve digital content. They understand that to do this, they are connecting to the internet and using a search engine such as Purple Mash search or internet-wide search engines.</p> <p><b>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.</b></p> <p>Children can collect, analyse, evaluate and present data and information using a selection of software, e.g. using a branching database (2Question), using software such as 2Graph.</p>	<p><b>Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content.</b></p> <p>Children understand the function, features and layout of a search engine. They can appraise selected webpages for credibility and information at a basic level.</p> <p><b>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.</b></p> <p>Children are able to make improvements to digital solutions based on feedback. Children make informed software choices when presenting information and data. They create linked content using a range of</p>	<p><b>Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content.</b></p> <p>Children search with greater complexity for digital content when using a search engine. They are able to explain in some detail how credible a webpage is and the information it contains.</p> <p><b>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.</b></p> <p>Children are able to make appropriate improvements to digital solutions based on feedback received and can confidently comment on the success of the solution. e.g. creating their own program to meet a design brief using 2Code. They objectively review solutions from others.</p>	<p><b>Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content.</b></p> <p>Children readily apply filters when searching for digital content. They are able to explain in detail how credible a webpage is and the information it contains. They compare a range of digital content sources and are able to rate them in terms of content quality and accuracy. Children use critical thinking skills in everyday use of online communication.</p> <p><b>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.</b></p> <p>Children make clear connections to the audience when designing and creating digital content. The children design and create their own blogs to</p>

Children can consider what software is most appropriate for a given task. They can create purposeful content to attach to emails, e.g. 2Respond.	software such as 2Connect and 2Publish+. Children share digital content within their community, i.e. using Virtual Display Boards.	Children are able to collaboratively create content and solutions using digital features within software such as collaborative mode. They are able to use several ways of sharing digital content, i.e. 2Blog, Display Boards and 2Email.	become a content creator on the Internet, e.g. 2Blog. They are able to use criteria to evaluate the quality of digital solutions and are able to identify improvements, making some refinements.
--	--	---	--

## Computing Progression – Digital Literacy (National Curriculum in bold)

YEAR 1		YEAR 2	
<p><b>Recognise common uses of information technology beyond school.</b></p> <p><i>Children understand what is meant by technology and can identify a variety of examples both in and out of school. They can make a distinction between objects that use modern technology and those that do not e.g. a microwave vs. a chair.</i></p> <p><b>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.</b></p> <p><i>Children understand the importance of keeping information, such as their usernames and passwords, private and actively demonstrate this in lessons. Children take ownership of their work and save this in their own private space such as their My Work folder on Purple Mash.</i></p>		<p><b>Recognise common uses of information technology beyond school.</b></p> <p><i>Children can effectively retrieve relevant, purposeful digital content using a search engine. They can apply their learning of effective searching beyond the classroom. They can share this knowledge, e.g. 2Publish example template. Children make links between technology they see around them, coding and multimedia work they do in school e.g. animations, interactive code and programs.</i></p> <p><b>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.</b></p> <p><i>Children know the implications of inappropriate online searches. Children begin to understand how things are shared electronically such as posting work to the Purple Mash display board. They develop an understanding of using email safely by using 2Respond activities on Purple Mash and know ways of reporting inappropriate behaviours and content to a trusted adult.</i></p>	
YEAR 3	YEAR 4	YEAR 5	YEAR 6
<p><b>Use technology safely, respectfully and responsibly; recognise acceptable/ unacceptable behaviour; identify a range of ways to report concern about content and contact.</b></p> <p><i>Children demonstrate the importance of having a secure password and not sharing this with anyone else. Furthermore, children can explain the negative implications of failure to keep passwords safe and secure. They understand the importance of staying safe and the importance of their conduct when using familiar communication tools</i></p>	<p><b>Use technology safely, respectfully and responsibly; recognise acceptable/ unacceptable behaviour; identify a range of ways to report concern about content and contact.</b></p> <p><i>Children can explore key concepts relating to online safety using concept mapping such as 2Connect. They can help others to understand the importance of online safety. Children know a range of ways of reporting inappropriate content and contact.</i></p>	<p><b>Use technology safely, respectfully and responsibly; recognise acceptable/ unacceptable behaviour; identify a range of ways to report concern about content and contact.</b></p> <p><i>Children have a secure knowledge of common online safety rules and can apply this by demonstrating the safe and respectful use of a few different technologies and online services. Children implicitly relate appropriate online behaviour to their right to personal privacy and mental wellbeing of themselves and others.</i></p>	<p><b>Use technology safely, respectfully and responsibly; recognise acceptable/ unacceptable behaviour; identify a range of ways to report concern about content and contact.</b></p> <p><i>Children demonstrate the safe and respectful use of a range of different technologies and online services. They identify more discreet inappropriate behaviours through developing critical thinking, e.g. 2Respond activities. They recognise the value in preserving their privacy when online for their own and other people's safety</i></p>

such as 2Email in Purple Mash. They know more than one way to report unacceptable content and contact.			
--	--	--	--

Numbers in brackets refer to the planning unit within each year group.

Computer Science (Impact – End points)						
EYFS	Key Stage 1		Key Stage 2			
Year R	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
In Reception, children will	In Year 1, children will	In Year 2, children will	In Year 3, children will	In Year 4, children will	In year 5, children will	In Year 6, children
	<p>Explain that an algorithm is a set of instructions. (1.4, 1.5)</p> <p>Know that a computer turns an algorithm into code that a computer can understand. (1.4, 1.7)</p> <p>Work out what is wrong when the steps are out of order in instructions. (1.4, 1.5)</p> <p>Know that if something does not work as it should, it is because the code is incorrect. (1.7)</p> <p>Be able to try to fix code that isn't working properly. (1.7)</p>	<p>Know that an algorithm is a set of instructions to complete a task. (2.1)</p> <p>Know to take care when writing an algorithm so that it will work when it is converted into code. (2.1)</p> <p>Design a simple program using 2Code that achieves a purpose. (2.1)</p> <p>Find and correct some errors in a program. (2.1)</p> <p>Say what happens in a program. (2.1)</p> <p>I can spot something that has an action or effect (does something). (2.1)</p>	<p>Make a real-life situation into an algorithm for a program. ((3.1)</p> <p>Design an algorithm carefully, thinking about what they want it to do and how they can turn it into code. (3.1)</p> <p>Identify an error in their program and fix it. (3.1)</p> <p>Experiment with timers in their programs. (3.1)</p> <p>Identify the difference between using a timer and a repeat command in their code. (3.1)</p> <p>Know that q variable stores information while a program is running. (3.1)</p>	<p>Turn a real-life situation to solve into an algorithm, using a design that shows how they can accomplish this in code. (4.1, 4.5)</p> <p>Use repetition in their code. For example, using a loop that continues until a condition is met such as the correct answer being entered. (4.1)</p> <p>Use timers within their program designs more accurately to create repetition effects. For example, I can create a counting machine. (4.1)</p> <p>Use selection (decision) in their programming. For example, using an 'if</p>	<p>Can make more complex real-life problems into algorithms for a program. (5.1)</p> <p>Can test and debug their programs as they work. (5.1, 5.5)</p> <p>Can convert (translate) algorithms that contain sequence, selection and repetition into code that works. (5.1)</p> <p>Can use sequence, selection, repetition, and some other coding structures in my code. (5.1)</p> <p>Can organise their code carefully for example, naming variables and using</p>	<p>Can turn a complex programming task into an algorithm. (6.1)</p> <p>Can identify the important aspects of a programming task (abstraction). (6.1)</p> <p>Can decompose important aspects of a programming task in a logical way, identifying appropriate coding structures that would work. (6.1)</p> <p>Can test and debug my program as I work on it and use logical methods to identify a cause of a bug. (6.1)</p> <p>Can identify a specific line of code that is causing a</p>

	<p>Be able to make a good guess as to what is going to happen in a program (e.g. where the turtle might move to). (1.5, 1.7)</p>		<p>Identify 'IF' statements, repetition and variables. (3.1)</p> <p>Read a piece of code with several steps and predict what will happen. (3.1)</p> <p>Identify the different ways the internet can be used for communication. (3.5)</p> <p>Use email (2email) to respond to others and attach files. (3.5)</p>	<p>statement' for a question being asked and the program takes one of two paths. (4.1)</p> <p>Use variables within their program and know how to change the value of variables. (4.1)</p> <p>Use the user inputs and output features within their program, such as 'Print to screen'. (4.1)</p> <p>Can identify errors in their code by using different methods, such as stepping through lines of code and fixing them. (4.1)</p> <p>Can read programs that contain several steps and predict the outcomes with increasing accuracy. (4.1, 4.5)</p> <p>Recognise the main component parts of hardware which allow computers to join and form a network. (4.8)</p> <p>Understand that network and communication components can be found in many different devices which allow them to join the internet. (4.2, 4.7, 4.8)</p>	<p>tabs; and to know this will help them debug more efficiently. (5.1)</p> <p>Can use logical methods to identify the cause of any bug with support to identify the specific line of code. (5.1)</p> <p>Know the importance of computer networks and how they help solve problems and enhance communication. (5.2)</p> <p>Recognise the main dangers that can be perpetuated via computer networks. (5.2)</p> <p>Can explain what personal information is and know strategies for keeping this safe. (5.2)</p> <p>Can use the most appropriate form of online communication according to the digital content. For example, use 2Email, 2Blog and Display Boards. (5.2 &amp; others)</p>	<p>problem in my program and attempt a fix. (6.1)</p> <p>Can translate algorithms that include sequence, selection and repetition into code and nest these structures within each other. (6.1)</p> <p>Can use inputs and outputs within my coded programs such as sound, movement and buttons and represent the state of an object (6.1, 6.7)</p> <p>Can interpret (understand) a program in parts and can make logical attempts to put the separate parts together in an algorithm to explain the program as a whole. (6.1)</p> <p>Can explain the difference between the internet and the World Wide Web. (6.2, 6.4,6.6)</p> <p>Can explain what a WAN and LAN is and describe the process of how access to the internet in school is possible. (6.2,6.6)</p>
--	--	--	---	--	---	---

Numbers in brackets refer to the planning unit within each year group.

EYFS	Key Stage 1		Key Stage 2			
Year R	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
In Reception, children will	In Year 1, children will	In Year 2, children will	In Year 3, children will	In Year 4, children will	In year 5, children will	In Year 6, children
	Sort sound, pictures and text. (1.2)	Organise data (e.g using a database such as 2investigate). (2.3, 2.4)	Carry out searches to find digital content on a range of online systems (within PM or on a search engine). Collect data and input it into software. (all units)	Understand the purpose of a search engine and the main features within it. (4.7)	Can search precisely when using a search engine. For example, knowing they can add additional words or removes words to help find better results. (5.2)	Can use filters when searching for digital content. (6.2,6.9)
	Add sound, pictures and text to a program such as 2Create a story. (1.6)	Find data using specific searches (e.g using 2investigate). (2.4, 2.5)	Analyse data using features within software to help such as formula in 2calculate (spreadsheets). (3.8, 3.6, 3.8)	Can look at information on a webpage and make predictions about the accuracy of information contained within it. (4.7)	Can explain in detail how accurate, safe and reliable the content is on a webpage. (5.2)	Can explain in detail how accurate and reliable a webpage and its content is. (6.2)
	Change the content on a file such as text, sound and images. (1.3, 1.6, 1.7, 1.8)	Use several programs to organise information (e.g using binary trees such as 2Question or spreadsheets such as 2calculate). (2.4, 2.8)	Present data and information using different software such as 2Question or 2Graph. (3.3, 3.6, 3.8)	Can create and improve my solutions to a problem based on feedback. For example, create a program using 2Code. (4.1, 4.2)	Can make appropriate improvements to digital work they have created. (Across units)	Can compare a range of digital content sources and rate them in terms of content quality and accuracy. (6.1, 6.3, 6.4, 6.5, 6.7,6.9)
	Name, save and retrieve their work. (1.2, 1.3, 1.6, 1.7, 1.8)	Edit digital data such as data in music composition software (2sequence). (2.7 and most units)	Consider what the most appropriate software to use when given a task. (3.3, 3.6, 3.8, 3.9)	Can review solutions that others have created, using a checklist of criteria. (4.1, 4.2)	Can comment on how successful a digital solution is that they have created. For example, a program built in 2Code that sorts decimals numbers. (Across units)	Can consider the intended audience carefully when they design and make digital content. (6.1, 6.3, 6.4, 6.5, 6.7,6.9)
		Name, save and find their work. (2.3, 2.4, 2.6, 2.7, 2.8 and most units)	Create purposeful (appropriate) content and attach this to emails. (3.3, 3.5, 3.6, 3.7, 3.8, 3.9)	Can work collaboratively to create content and solutions. (4.1, 4.3, 4.4,4.8)	Can work collaboratively with others creating solutions to problems using appropriate software such as 2Code. (Across units).	Can design and create their own online blogs. (6.4)
		Include photos, text and sound in their creations. (2.6, 2.8)		Can share digital content using a variety of applications such as: 2Blog, 2Email and Display Boards. (Across units)	Can work collaboratively modes such as within 2Connect to work with others and share it. (5.7)	Can use criteria to evaluate the quality of their own and others digital solutions, suggesting refinements. (6.1, 6.3, 6.4, 6.5, 6.7,6.9)

Numbers in brackets refer to the planning unit within each year group.

**Digital Literacy (Impact – End points)**

<b>EYFS</b>	<b>Key Stage 1</b>		<b>Key Stage 2</b>			
<b>Year R</b>	<b>Year 1</b>	<b>Year 2</b>	<b>Year 3</b>	<b>Year 4</b>	<b>Year 5</b>	<b>Year 6</b>
In Reception, children will	In Year 1, children will	In Year 2, children will	In Year 3, children will	In Year 4, children will	In year 5, children will	In Year 6, children
	Say what technology is. (1.9)	Use search engines to find information. (2.5)	Create a secure password. (3.2)	Have a good understanding of the online safety rules we learn at school. (4.2 & across curriculum)	Have a secure knowledge of online safety rules taught at school. (5.2 & across units)	Can demonstrate safe and respectful use of a range of different technologies and online services. (6.2, 6.4)
	Say what examples of technology are in school. (1.9)	Know the consequences of not searching online safely. (2.2, 2.5)	Explain the importance of having a secure password and not sharing it with others. (3.2, 3.5)	Can demonstrate how to use different online technologies safely. (4.2 & across curriculum)	Can demonstrate the safe and respectful use of different online technologies and online services. (5.2 & across units)	Can identify more discrete inappropriate behaviours online. For example, someone who may be trying to groom me or someone else. (6.2)
	Say what examples of technology are at home. (1.9)	Share their work and communicate electronically (2Email or the Display boards). (2.2 and others)	Explain the negative consequences of not keeping passwords safe and secure. (3.2, 3.5)	Can demonstrate how to use a few different online services safely. (4.2 & across curriculum)	Always relate appropriate online behaviour to my right to have personal privacy. (5.2 & across units)	Can use critical thinking to help me stay safe online. (6.2)
	Know that a chair uses old technology and a smart phone uses new technology. (1.9)	Know how to report unkind behaviour to a trusted adult. (2.2)	Understand the importance of keeping safe online and behaving respectfully. (3.2)	Know I have a right to privacy both on and offline. (4.2 & across curriculum)	Know how to not let my mental wellbeing or others be affected by use of online technologies and services. (5.2 & across units)	Know the value of protecting my privacy and others online. (6.2, 6.4)
	Keep their login information safe. (1.1 and most units)	Recognise where technology is used throughout the school. (2.2)	Use communication tools such as 2Email respectfully and use good etiquette. (3.2, 3.5)	Recognise that my wellbeing can be affected by how I use technology. (4.2 & across curriculum)		
	Save their work in a safe place (e.g. 'My Folder'). (1.1 and most units)	Know that skills used in their creations are also used in the adult world. (2.1)	Report unacceptable content and contact online in more than one way to a trusted adult. (3.2)	Can report with ease any concerns with content and contact online and know immediate strategies to keep safe. (4.2 & across curriculum)		



