Ickleford Primary School **Progression Map: Computing**

	Reception	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Vocabulary		Alert, avatar, button, device, file name, filter, home screen, icon, login, log out, menu, my work area, notification, password, private, purple mash tools, saving, search, shared folder, textbox, think about box, topic area, tool bar, typing, writing template, algorithm, criteria, describe, equal, groups, less than, more than, sort, animation, background, category, clip-art gallery, copy, drop down menu, e-book, edit, eraser, features, font, sound, overwrite, paint tools, paste, play mode, redo, save, sound effect, text, undo, voice recording, computer, technology, action, code, click code blocks, coding, command, debug, debugging, design view, event, execute, instruction, object, output, plan, programmer, properties, run, scale, scene, software, sound, when clicked.	Attachment, digital footprint, display board, email, filter, identifying, internet, personal information, private information, protection, reply, search, secure, sharing, action, algorithm, background bug, button, click events, collision detection, Collison detection action, collision detection event, command, debug, event, execute, image, implement, instructions, interaction, interval, object, object name, output, predict, properties, run, scale, scene, sequence, test, text, timer, turtle object, when clicked, when key event, when swiped event, addition, block graph, cell, coins, column, copy, count tool, cut, data, drag, equals, equals tool, image value, label, paste, price, row, speak tool, table, browser, device, digital footprint, domain, internet, network, search engine, URL, web address, web page, website, world wide web, e-book, fact file, fiction, mind map, multiple-choice, node, non-fiction, presentation, quiz	Appropriate, blog, inappropriate, internet, password, personal information, permission, reliable source, reputable source, spoof, verify, vlogs, website, action, alert, algorithm, back ground, bug, button, click event, code, collison detection event, command, debug, degrees, event, flowchart, implement, input, interval, nest, object, predict, properties, repeat, right angle, run, scene, sequence, test, timer, turtle object, keys, posture, spacebar, typing, address book, attachment, BCC – Blind Carbon Copy, CC – Carbon Copy, communication, compose, email, inbox, link, mind mapping, node, password, personal information, save to draft, trusted contact, axis, chart, column, data, graph, investigation, row, sorting, survey, tally chart, title	Adfly, attachment, citation, collaborate, collaborative database, cookies, copyright, data analysis, digital footprint, malware, phishing, plagiarism, ransomware, report, SMART rules, software, spam, virus, watermark, action, alert, algorithm, background, button, code blocks, command, co-ordinates, debug, design, event, execute, flowchart, 'if' statements, 'if/else' statements, input, nest, object, prompt, implement, predict, repeat, repeat until, run, properties, selection, sequence, timer, variable, value, BPM, dynamics, harmonious, melody, pitch, pulse, rhythm, tempo, texture, synths, average, budget, calculations, chart, column, data, decimal place, equals to tool, format cell, formula, formula wizard, line graph, percentage, place value, random number tool, resize, spinner tool, timer, totals, balanced view, easter eggs, internet, key words, reliability, results page, search engine	Appropriate, avatar, bibliography, citation, collaborate, communication, copyright, creative commons, license, critical thinking, digital footprint, encrypt, identify theft, image manipulation, malware, ownership, PEGI ratings, phishing, password, personal information, plagiarism, reliability, responsibility, reliable source, screenshot, SMART rules, spoof, validity, Abstraction, action, algorithm, command, concatenation, co-ordinates, debug, decomposition, efficient, event, flowchart, friction, function, input, nest, object, output, physical system, predict, print to screen, properties, random,repeat, selection, sequence, simplify, simulation, string, tabs, timer, variable, evaluation, feedback, image, instructions, promotion, quest, scene, screenshot, texture, theme, 2D, 3D, 3D, printing, CAD – computer aided design, design brief, net, pattern fill, points, attributing, bulleted lists, breaks, caps lock, captions, column (table, column (newspaper), copy and paste, copyright, creative commons, cropping, cursor, distributing columns, documents drop capitals, editor	Data analysis,digital footprint, inappropriate, location sharing, password, PEGI rating, phishing, print screen, screen time, secure websites, spoof, action, algorithm, command, concatenation, co-ordintes, debug, decomposition, event, execute, flowchart, function, input, launch command, object, output, predict, procedure, properties, repeat, repeat until, selection, sequence, simulation, string, tabs, text object, timer, turtle object, variable, x and y properties, data, Domain Name Server, Ethernet, hosting, hub/swtich, internet, IP address, Internet Service Provider, Local Area Network, Network, Router, Search engine, Wide Area Network, Web page, web server, website, WLAN Wireless Local Area Network, Wi-Fi, World Wide Web, binary, bit, decimal, denary, digit, games states, integer, microprocessor, nanotechnology, nibble, byte, kilobyte, megabyte, gigabyte, tetrabyte, switch, transistor, variable, auto-fit, average, budget, calculation, catergories ribbon, cell, cell reference, chart, column, computational model, conditional formatting, currency, data, delimiter, expense, filter, flash-fill, formatting, formula,

						options, font, front screen, grammar check, hyperlink, image editing, image transparency, merge cells, numbered lists, page orientation, readability, row, selecting/highlighting, sharing, spell check, styles, template, text box, text formatting, text wrapping, word art, word processing tools, zoom	formula bar, graph, horizontal axis, maximum, minimum, profit, range, row, series, sheet, sorting, spreadsheet, template, text wrapping, vertical axis, workbook.
Computer Science	In the Early Years Foundation Stage (EYFS), teachers introduce computing and online safety, supporting young children's development in a digital world. It's important to introduce them to the concepts of technology in a way that is safe, engaging, and age-appropriate. At Ickleford School we also use our digital leaders to share stories that give basic understanding of technology and online safety. The EYFS is centred around play based activities that focus on listening skills, curiosity, creativity and problem solving. In EYFS this may look like:- · Using an iPad to paint pictures · Listen to stories on the white board	To understand that an algorithm is a set of instructions used to solve a problem or achieve an objective. To know that a computer program turns an algorithm into code that the computer can understand. To work out what is wrong with a simple algorithm when the steps are out of order and can write their own simple algorithm. To know that an unexpected outcome is due to the code they have created and can make logical attempts to fix the code. To read code one line at a time and make good attempts to envision the bigger picture of the overall effect of the program.	To explain that an algorithm is a set of instructions to complete a task. To show an awareness of the need to be precise with their algorithms so that they can be successfully converted into code. To create a simple program that achieves a specific purpose. To identify and correct some errors. To ensure that program designs display a growing awareness of the need for logical, programmable steps. To 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.	To turn a simple real-life situation into an algorithm for a program by deconstructing it into manageable parts. To ensure that design shows that they are thinking of the desired task and how this translates into code. To identify an error within their program that prevents it following the desired algorithm and then fix it. To demonstrate the ability to design and code a program that follows a simple sequence. To experiment with timers to achieve repetition effects in their programs. To begin to understand the difference in the effect of using a timer command rather than a repeat command when creating repetition effects. To ensure that designs for their programs show	To turn a real-life situation into an algorithm, designs to show thinking of required task and how to accomplish this in code using coding structures for selection and repetition. To make more intuitive attempts to debug their own programs. Use of timers to achieve repetition effects are becoming more logical and are integrated into their program designs. To 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. To understand how variables can be used to store information while a program is executing, they are able to use and manipulate the value of variables.	To turn more complex real-life situations into algorithms for a program by deconstructing it into manageable parts. To test and debug programs and can use logical methods to identify the approximate cause of any bug but may need some support identifying the specific line of code. To translate algorithms that include sequence, selection and repetition into code with increasing ease and designs show thinking of how to accomplish the set task in code utilising such structures. To combine sequence, selection and repetition with other coding structures to. To begin 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	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. To 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. To translate algorithms that include sequence, selection and repetition into code and designs show thinking of how to accomplish the set task in code utilising such structures, including nesting structures within each other. To ensure that coding displays an improving understanding of variables in coding, outputs such as sound

	· Playing and learning to programme devices such as a beebot · Phonics and maths games through an app At Ickleford School we believe that in allowing children the opportunity to explore technology in this carefree and often child-led way, means that will they develop a familiarity with equipment and vocabulary, ready to start their computing journey in Year 1.			that they are thinking of the structure of a program in logical, achievable steps and absorbing some new knowledge of coding structures. To make good attempts to 'step through' more complex code in order to identify errors in algorithms and can correct this. To 'read' programs with several steps and predict the outcome accurately. To list a range of ways that the Internet can be used to provide different methods of communication. To use some of these methods of communication, e.g. being able to open, respond to and attach files to emails using 2Email. To describe appropriate email conventions when communicating in this way.	To make use of user inputs and outputs such as 'print to screen'. e.g. 2Code. To ensure that designs show 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. To trace code and use step-through methods to identify errors in code and make logical attempts to correct this. In programs such as Logo, 'read' programs with several steps and predict the outcome accurately. To recognise the main component parts of hardware which allow computers to join and form a network. To understand the online safety implications associated with the ways the Internet can be used to provide different methods of communication is improving.	To understand the value of computer networks but are also aware of the main dangers. To recognise what personal information is and can explain how this can be kept safe. To select the most appropriate form of online communications contingent on audience and digital content, e.g. 2Blog, 2Email, Display Boards.	and movement, inputs from the user of the program such as button clicks and the value of functions. To interpret a program in parts and be able to make logical attempts to put the separate parts of a complex algorithm together to explain the program as a whole. To understand and be able to explain in some depth the difference between the internet and the World Wide Web. To know what a WAN and LAN are and can describe how they access the internet in school.
Information Technology		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.	To demonstrate an ability to organise data and can retrieve specific data for conducting simple searches. To be able to edit more complex digital data. To confidently create, name, save and retrieve content.	To carry out simple searches to retrieve digital content. To 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.	To understand the function, features and layout of a search engine. To appraise selected webpages for credibility and information at a basic level. To make improvements to digital solutions based on feedback.	To search with greater complexity for digital content when using a search engine. To explain in some detail how credible a webpage is and the information it contains. To make appropriate improvements to digital solutions based on	To apply filters when searching for digital content. To explain in detail how credible a webpage is and the information it contains. To compare a range of digital content sources and are able to rate them in terms of

			To use a range of media in their digital content including photos, text and sound.	To collect, analyse, evaluate and present data and information using a selection of software. To consider what software is most appropriate for a given task. To create purposeful content to attach to emails.	To make informed software choices when presenting information and data. To create linked content using a range of software. To share digital content within their community.	feedback received and can confidently comment on the success of the solution. To review solutions from others. To collaboratively create content and solutions using digital features within software such as collaborative mode. To use several ways of sharing digital content.	content quality and accuracy. To use critical thinking skills in everyday use of online communication. To make clear connections to the audience when designing and creating digital content. To use criteria to evaluate the quality of digital solutions and are able to identify improvements, making some refinements.
Digital Literacy		To understand what is meant by technology and can identify a variety of examples both in and out of school. To make a distinction between objects that use modern technology and those that do not e.g. a microwave vs. a chair. To understand the importance of keeping information, such as their usernames and passwords, private and actively demonstrate this in lessons. To take ownership of their work and save this in their own private space such as their My Work folder on Purple Mash.	To retrieve relevant, purposeful digital content using a search engine. To apply their learning of effective searching beyond the classroom. Share this knowledge, e.g. 2Publish example template. To make links between technology they see around them, coding and multimedia work they do in school e.g. animations, interactive code and programs. To know the implications of inappropriate online searches. To begin to understand how things are shared electronically such as posting work to the Purple Mash display board. To develop an understanding of using email safely and know	To demonstrate the importance of having a secure password and not sharing this with anyone else. To explain the negative implications of failure to keep passwords safe and secure . To understand the importance of staying safe and the importance of their conduct when using familiar communication tools such as 2Email in Purple Mash. To know more than one way to report unacceptable content and contact.	To explore key concepts relating to online safety using concept mapping. To help others to understand the importance of online safety. To know a range of ways of reporting inappropriate content and contact.	To 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. To implicitly relate appropriate online behaviour to their right to personal privacy and mental wellbeing of themselves and others.	To demonstrate the safe and respectful use of a range of different technologies and online services. To identify more discreet inappropriate behaviours through developing critical thinking. To recognise the value in preserving their privacy when online for their own and other people's safety.

			ways of reporting inappropriate behaviours and content to a trusted adult.				
Vocabulary							

National Curriculum KS1

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 (1.2, 1.7 and 2.1)
- create and debug simple programs (1.7 and 2.1)
- use logical reasoning to predict the behaviour of simple programs (1.7 and 2.1)
- use technology purposefully to create, organise, store, manipulate and retrieve digital content (1.6, 2.3 and 2.8)
- recognise common uses of information technology beyond school (1.9 and 2.5)
- 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. (1.1 and 2.2)

National Curriculum KS2

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 (3.1, 4.1, 5.1, 5.5 and 6.1)
- use sequence, selection, and repetition in programs; work with variables and various forms of input and output (3.1, 4.1, 5.1 and 6.1)
- use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs (3.1, 4.1, 5.1 and 6.1 and 6.8)
- 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 (3.5, 4.2, 4.7, 5.2 and 6.2)

- use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content (3.8, 4.9, 4.3, 5.8 and 6.2)
- 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 (3.4, 3.8, 4.9, 4.3, 5.5, 5.6, 5.8, 6.1, 6.8, 6.9)
- use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact. (3.2, 3.5, 4.2, 5.2 and 6.2)