OUR

Computing

CURRICULUM

SUPPORT • ACHIEVE • CELEBRATE

The teaching of Computing at Cherry Lane Primary School is underpinned by the principles of the Cherry Lane Way.

INTENT

At Cherry Lane Primary School, we aim to deliver an exceptional computing curriculum that empowers every pupil with the skills and knowledge required to thrive in an increasingly digital world for the 21st Century. Our intent is constructed around the following key principles:

1. ***Holistic Development**: At Cherry Lane, we recognise that computing is not just about technical skills; it fosters critical thinking, problem-solving, and creativity. Our curriculum is designed to support the overall personal development of pupils. We support this with workshops designed to explore Computing from a more holistic view by allowing pupils to explore technology from the 19th century to current day.*
2. ***Curriculum Underpinned by National Standards**: Our computing curriculum is delivered through Purple Mash which is in line with the National Curriculum and has been tailored to meet the specific needs of our pupils. We ensure that all pupils experience a well-structured progression through the Key Stages, focusing on three core areas: computer science, digital literacy, and information technology. Our exploration of computing begins towards the end of the Early Years Foundation Stage by introducing children to use of a mouse, keyboard*

and gain access to a login as part of their transition into Key Stage 1 (KS1). Progressing from Years 1 to 6, we follow the Purple Mash scheme, providing consistent weekly 1 hour lessons that cover the 10 computing skills throughout their academic journey.

3. ***Inclusive and Engaging Learning**: We aim to inspire all pupils to develop a love for computing. Activities are inclusive, catering to diverse learning styles and abilities, ensuring every child can engage and succeed.*
4. ***Preparation for the Future**: We strive to equip our pupils with the skills to navigate a rapidly changing technological landscape, fostering resilience and adaptability in the face of emerging digital challenges.*

IMPLEMENTATION

Our implementation strategy is taught through the programme, 2Simple Purple Mash. To deliver high-quality computing education in line with the National Curriculum:

1. ***Structured Curriculum Design**: The computing curriculum is sequenced to build upon prior knowledge systematically, ensuring pupils develop a solid foundation in programming, computational thinking, and digital literacy. We utilise a range of resources, including interactive platforms and age-appropriate software, to enhance learning experiences.*
2. ***Cross-Curricular Integration**: Computing skills are embedded across the curriculum, promoting interdisciplinary links and real-world application of technology. For instance, using coding to create animations and blogs or data analysis in Maths through pictograms enhances engagement and reinforces learning.*

3. ***Professional Development for Staff***: *Continuous professional development (CPD) is prioritised for teaching staff, ensuring they remain confident and competent in delivering the computing curriculum. Regular training sessions, workshops, and collaboration with technology experts provide staff with the latest pedagogical strategies and tools.*
4. ***Parental Engagement***: *We actively involve parents with the home learning of their children the Purple Mash, TTRockstars and Learning with Parents programmes. This partnership encourages pupils to explore technology outside of the classroom environment.*
5. ***Assessment and Feedback***: *A robust system for formative assessment is in place, enabling both teachers and pupils to track progress and identify areas for improvement. Assessments are varied, including practical projects, quizzes, and presentations, ensuring a comprehensive evaluation of pupils' understanding.*

CONSOLIDATION

By combining structured learning pathways, interactive tools, and creative opportunities, we can use our Purple Mash platform to ensure that computing concepts are not only introduced effectively but also effectively reinforced through repetition, practice, and application.

1. Engaging Tools and Resources

- *Interactive Activities: It offers a wide array of computing activities such as coding exercises, simulations, and games. These tools reinforce concepts by allowing students to practice skills in a hands-on, engaging way.*
- *Step-by-Step Guidance: Many tasks come with scaffolding, such as tutorials and instructions, helping students gradually build their skills.*

2. Progressive Learning Pathways

- *Sequenced Lessons: It includes a structured curriculum for computing that ensures concepts are introduced in a logical sequence. This builds a strong foundation before progressing to advanced topics.*
- *Cross-Curricular Integration: The platform integrates computing skills into other subjects, reinforcing learning by applying knowledge in diverse contexts.*

3. *Practice and Repetition*
    - *Self-Paced Learning: Students can revisit tasks and practice skills multiple times, which helps reinforce learning and ensures mastery.*
    - *Adaptive Content: Activities are designed to cater to different ability levels, allowing for personalized repetition and reinforcement.*
4. *Collaborative and Creative Learning*
    - *Project-Based Learning: Students work on creative projects, such as building apps, creating animations, or designing digital art, which apply their computational thinking in meaningful ways.*
    - *Collaboration Tools: Features like collaborative writing and shared coding projects help students learn from peers, reinforcing concepts through teamwork.*
5. *Feedback and Assessment*
    - *Immediate Feedback: Many activities provide instant feedback, helping students identify errors and correct them in real-time.*
    - *Teacher Assessment Tools: Teachers can monitor student progress, provide feedback, and tailor activities to individual needs, ensuring consolidation of learning.*
6. *Revisiting and Reviewing Concepts*
    - *Saved Work and Portfolios: Students can save their work, revisit it, and improve upon it, which reinforces prior learning and builds confidence.*
    - *Skill Checkers: Regular quizzes and challenges help consolidate understanding and prepare students for new topics.*

KEY VOCABULARY

The following table sets out the key vocabulary used in our computing programme from EYFS to Year 6. It is divided in terms of essential (must) and advanced (could) vocabulary.

Focus	Early Years	Year One	Year Two	Year Three	Year Four	Year Five	Year Six
Must	**Computer Science** Equipment Buttons Movement **Information Technology** Screen Mouse Keyboard Paint **Digital Literacy** Technology Internet	**Online safety** Login Log out Username Password Private **Grouping and sorting** Sort Algorithm Criteria **Pictograms** Pictogram Data Compare **Lego Builders** Algorithm Code Debugging Program	**Coding** Algorithm Command Debug/Debugging Event **Online safety** Email Digital footprint Internet Search Sharing **Spreadsheet** Cells Columns Rows Equals Data **Questioning** Binary Tree Pictogram	**Coding** Algorithm Bug Code Command Action **Online safety** Blog Appropriate Inappropriate Reliable source Verify **Spreadsheets** Data Bar graph Cell address Table **Touch typing** Keys Space bar	**Coding** Action Algorithm Code block Command Debug **Online safety** Cookies Plagiarism Malware Phishing Virus Attachment **Spreadsheets** Formula Charts Equals tool Format cell Random number tool	**Coding** Input Output Abstraction Decomposition Simplify Repeat Sequence **Online safety** Copyright Encrypt Malware Password Validity **Spreadsheets** Data Formula bar Variable Format **Databases**	**Coding** Algorithm Command Debug Coordinates Event **Online safety** Digital footprint Screen time Inappropriate Data analysis **Spreadsheets** Format cell Formula bar Formula wizard Budget **Blogging** Blog post Blog Vlog

		Maze explorers Command Arrow Undo Instruction Route **Animated story books** Animation Copy Paste e-book Font Sound **Coding** Algorithm Code Code blocks Command Debug Event **Spreadsheets** Columns Rows Cells Data Select	Data Database Question **Effective Searching** Browser Internet Search engine **Creating pictures** Clipart Palette Style Fill **Making Music** Compose Sound effect Beat Tune Speed **Presenting Ideas** Quiz Presentation Mind map	Typing **Emails** Communication Attachment Address book Inbox **Branching databases** Binary tree Branching database Data Database **Simulation** Analysis Simulation Evaluation **Graphing** Chart Axis Data **Presenting** Animation Text box Font formatting	**Writing for different audiences** Format Font Genre Opinion **Logo** Debugging grid Pen up Pen down **Animation** Frame Animation Pause Stop motion **Effective searching** Search engine Reliability Results page **Hardware Investigators** Components CPU Motherboard	Arrange Avatar Chart Database Sort Search **Game Creator** Animation Image Texture Interactive Customise **3D Modelling** 3D Printing CAD Net Template **Concept Maps** Concept Concept map Connection **Word processing** Cursor Copy/paste Document Font	Approval **Text adventures** Debug Sprite Selection Function **Networks** Internet WAN WiFi Network **Quizzing** Audience Audio Quiz **Binary** Digit Bit Integer **Spreadsheets** Cell Formula Axis Chart Graph

				Layer	Input Output Software **<u>Making Music</u>** BPM Dynamics Melody Synths	Formatting **<u>Using external devices</u>** Algorithm Input Output External device	
Could	Images Share Create	Alert Avatar Criteria Collate File Delete Search Sequence Software	Sequence Attachment Copy and Paste Database Collate	PEGI rating Spoof Website Home row keys CC BCC Media Transition Reliable source	Number variable Cookies Copyright Flipbook Stop motion Easter egg Graphics card RAM	Decomposition Physical stem Variable Citations Bibliography Binary Tree Perspective CAD Modelling Node Text wrapping	String Tab Phishing Icon Local Area Network (LAN) Transistor Byte Megabyte/Terabyte

PROGRESSION OVERVIEW

The first table below sets out the Purple Mash Computing scheme of work long term plan summarises curriculum coverage at Cherry Lane Primary across Years 1-6. The subsequent tables outline the detailed skills and knowledge covered in each year group.

Theme Key:							
Coding and Computational Thinking	Spreadsheets	Internet and Email	Art and Design	Music	Databases and Graphing	Writing and Presentation	Communication and Networks

Year Group	Autumn 1							Autumn 2								Spring 1						Spring 2						Summer 1						Summer 2						
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3	4	5	6	7	8	9	10	11	12	13
1		**Unit 1.1** Online Safety & Exploring Purple Mash Weeks – 4 Programs – Various				**Unit 1.2** Grouping and Sorting Weeks -2 Programs - 2DIY			**Unit 1.3** Pictograms Weeks – 3 Programs – 2Count				**Unit 1.4** Lego Builder Weeks – 3 Programs – 2DIY				**Unit 1.5** Maze Explorers Weeks – 3 Programs – 2Go				**Unit 1.9** Technology outside School Weeks – 2 Programs - Various			**Unit 1.6** Animated Story Books Weeks – 5 Programs – 2CreateAStory						**Unit 1.8** Spreadsheets Weeks – 3 Programs – 2Calculate							**Unit 1.7** Coding Weeks – 6 Programs – 2Code						
2			**Unit 2.1** Coding Weeks – 5 Programs – 2Code						**Unit 2.2** Online Safety Weeks – 2 Programs - Various			**Unit 2.3** Spreadsheets Weeks – 4 Programs – 2Calculate					**Unit 2.4** Questioning Weeks – 5 Programs – 2Question, 2Investigate						**Unit 2.5** Effective Searching Weeks – 3 Programs - Browser					**Unit 2.7** Making Music Weeks – 3 Programs – 2Sequence			**Unit 2.6** Creating Pictures Weeks – 5 Programs – 2PaintAPicture							**Unit 2.8** Presenting Ideas Weeks – 4 Programs - Various						
3		**Unit 3.1** Coding Weeks – 6 Programs – 2Code							**Unit 3.2** Online Safety Weeks – 2 Programs – Various			**Unit 3.3** Spreadsheets Weeks – 3 Programs - 2Calculate					Unit 3.4 Touch Typing Weeks 4 Programs – 2Type						**Unit 3.5** Email Weeks – 6 Programs – 2Email, 2Connect, 2DIY							**Unit 3.6** Branching Databases Weeks 4 Programs – 2Question							**Unit 3.7** Simulations Weeks – 3 Programs – 2Simulate, 2Publish			Unit 3.8 Graphing Weeks – 3 Programs – 2Graph			
4		**Unit 4.7** Effective Search Weeks – 3 Programs - Browser			**Unit 4.8** Hardware Investigators Weeks 2				**Unit 4.2** Online Safety Weeks – 2 Programs – Various			**Unit 4.4** Writing for different audiences Weeks – 5 Programs – 2Email, 2Connect, 2DIY				**Unit 4.3** Spreadsheets Weeks – 6 Programs – 2Calculate						**Unit 4.5** Logo Weeks – 4 Programs - Logo							**Unit 4.6** Animation Weeks – 3 Programs - 2Animate							**Unit 4.1** Coding Weeks – 6 Programs – 2Code						
5		**Unit 5.1** Coding Weeks – 6 Programs – 2Code							**Unit 5.2** Online Safety Weeks – 2 Programs – Various			**Unit 5.4** Databases Weeks – 4 Programs – 2Question, 2Investigate				**Unit 5.3** Spreadsheets Weeks – 6 Programs – 2Calculate						**Unit 5.5** Games Creator Weeks – 5 Programs – 2DIY 3D							**Unit 5.7** Concept Maps Weeks – 4 Programs – 2Connect							**Unit 5.6** 3D Modelling Weeks – 4 Programs – 2Design &Make						
6		**Unit 6.1** Coding Weeks – 6 Programs – 2Code							**Unit 6.2** Online Safety Weeks – 2 Programs - Various			**Unit 6.4** Blogging Weeks – 5 Programs – 2Blog					**Unit 6.3** Spreadsheets Weeks – 5 Programs – 2Calculate						**Unit 6.5** Text Adventures Weeks 5 Programs – 2Code, 2Connect									**Unit 6.6** Networks Weeks - 3						**Unit 6.7** Quizzing Weeks – 6 Programs – 2Quiz, 2DIY, Text Toolkit, 2Investigate						

2simple

Aspect	★ Statement
Computer Science	Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions.
	Create and debug simple programs.
	Use logical reasoning to predict the behaviour of simple programs.
Information Technology	Use technology purposefully to create, organise, store, manipulate and retrieve digital content.
Digital Literacy	Recognise common uses of information technology beyond school.
	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.

Year 1

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

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.

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.

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.

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.

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.

Aspect	Statement
Computer Science	Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions.
	Create and debug simple programs.
	Use logical reasoning to predict the behaviour of simple programs.
Information Technology	Use technology purposefully to create, organise, store, manipulate and retrieve digital content.
Digital Literacy	Recognise common uses of information technology beyond school.
	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.

Year 2

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.

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.

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.

Children demonstrate an ability to organise data using, for example, a database such as 2Invesitigate 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.

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.

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.

Aspect	Statement
Computer Science	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.
Information Technology	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.
Digital Literacy	Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concern about content and contact.

Year 3

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.

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.

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.

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.

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.

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. Children can consider what software is most appropriate for a given task. They can create purposeful content to attach to emails, e.g. 2Respond.

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

Aspect	★ Statement
Computer Science	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.
Information Technology	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.
Digital Literacy	Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concern about content and contact.

Year 4

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.

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.

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.

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.

Children understand the function, features and layout of a search engine. They can appraise selected webpages for credibility and information at a basic level. .

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 software such as 2Connect and 2Publish+. Children share digital content within their community, i.e. using Virtual Display Boards.

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.

Aspect	★ Statement
Computer Science	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.
Information Technology	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.
Digital Literacy	Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concern about content and contact.

Year 5

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.

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.

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.

Children understand the value of computer networks but are also aware of the main 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.

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.

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. 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.

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.

Aspect	★ Statement
Computer Science	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.
Information Technology	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.
Digital Literacy	Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concern about content and contact.

Year 6

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.

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.

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.

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.

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.

Children make clear connections to the audience when designing and creating digital content. The children design and create their own blogs to 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.

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.

INCLUSION

At Cherry Lane Primary School our children work at varying levels appropriate to their abilities. Provision for children with SEND in relation to Computing is the responsibility of the Class Teacher, Learning Support Assistants (LSAs), Teaching Assistants (TAs) and the SENCO Team as appropriate.

Computing provides an opportunity for SEND children to present and develop their work easily. Where possible, a computer/iPad/Chrome Book will be provided for children with specific special educational needs to work with support staff so that work can be drafted and redrafted. Programmes to assist such children with specific needs include; Nessy for children with dislexia and Learning Village for EAL children.

Each class is provided with an iPad and at least two Chrome Books for children to use during lessons and interventions with SEND children. Kindles are also available for children in KS1 and upper Key Stage 2 and laptops are available for lower Key Stage 2.

By providing a flexible, engaging, and inclusive platform, Purple Mash helps SEND pupils overcome barriers to learning and offers extended learning opportunities for more able pupils, building their confidence, independence, and digital skills. It therefore supports individual needs while fostering a sense of achievement and belonging for all.

This is achieved through:

1. Accessibility Features

- ***Adaptable Interface:** The platform is designed to be user-friendly and customizable, allowing for adjustments to suit individual needs, such as font size, background color, or layout.*
- ***Text-to-Speech:** Many tools include text-to-speech functionality, enabling children with reading difficulties or visual impairments to access content more easily.*
- ***Simplified Navigation:** Icons and intuitive menus make it easier for children with cognitive or processing challenges to navigate the platform independently*

2. Personalized Learning

- ***Customizable Activities:*** *Teachers can modify tasks to match the ability levels of individual students, ensuring that SEND learners engage at a pace and level that suits them.*
- ***Self-Paced Progression:*** *Students can work through activities at their own speed, which reduces stress and allows for mastery of concepts.*
- ***Differentiated Resources:*** *Purple Mash offers a variety of activities and levels, making it easy to provide content appropriate for a range of needs.*

3. Multi-Sensory Engagement

- ***Interactive Tools:*** *Activities incorporate visual, auditory, and kinesthetic elements, which help SEND children engage through their preferred learning styles.*
- ***Creative Outputs:*** *Tools like painting, animation, and music composition enable children to express themselves in non-verbal ways, building confidence and creativity.*
- ***Game-Like Features:*** *Gamified tasks make learning fun and engaging, holding the attention of children who may struggle with traditional teaching methods.*

4. Teacher Support for Individual Needs

- ***Monitoring and Assessment Tools:*** *Teachers can track progress and identify specific challenges faced by SEND learners, enabling timely interventions.*
- ***Task Assignments:*** *Teachers can assign differentiated tasks to cater to specific needs, ensuring all students remain engaged and challenged appropriately.*

5. Safe, Low-Pressure Environment

- ***Mistake-Friendly Learning:*** *Activities encourage experimentation without penalties, reducing anxiety and fostering a growth mindset.*
- ***Non-Competitive Setting:*** *The platform emphasizes personal progress over competition, which is particularly supportive for SEND children.*

IMPACT

Assessment of Computing is an on-going process which focuses on the progress and achievement of pupils throughout each Key Stage. Computing is assessed through scrutiny of progress as well as discussions with children. Ongoing formative assessments inform differentiated input for individuals and groups of pupils.

Attainment is measured summatively during termly assessments, the results of which are recorded and analysed. Class teachers report annually on pupils' attitudes to computing where they are responsible for identifying children's strengths as well as areas for development.

The impact of our computing curriculum is evident through indicators that align with Ofsted's criteria for excellence:

1. ***Pupil Outcomes**: Teachers' assessment data shows progress in computing skills across all year groups, with some pupils meeting or exceeding age-related expectations. Pupil confidence and enthusiasm for computing are evident, with many expressing a desire to further develop their skills.*
2. ***Engagement and Attitudes**: Observations have revealed high levels of engagement during computing lessons, with pupils actively participating and demonstrating curiosity and resilience when faced with challenges. Feedback from a sample of pupils indicate enjoyment they find in problem-solving and collaborating on technology projects.*

In conclusion, at Cherry Lane Primary School, our computing curriculum embodies our mission to inspire, engage, and equip pupils with the skills necessary to navigate and contribute to a digital future. By embedding best practices and continually evaluating our impact, we ensure that our pupils are not just consumers of technology, but skilful creators and responsible digital citizens.