

The Meadows Primary Academy



Computing

Logic	Computational logic is the process of working step-by-step to understand a problem and develop a solution. It describes the decision-making process used in programming and writing algorithms.
Abstraction	Abstraction is an important part of computer programming. In computing, abstraction is the technique used to arrange computer systems and hide the complexity of programs to make it more accessible to the everyday user.
Machines	A computing machine is a device used to perform calculations and process data.
Algorithms	An algorithm is a process or set of rules followed in calculations or other problem-solving operations, especially by a computer.
Program	A computing program is a collection of instructions that performs a specific task when executed by a computer.
Data	Data is any sequence of one or more symbols given meaning by specific acts of interpretation. Computer data is information processed or stored by a computer.

The Meadows Primary Academy - COMPUTING progression through EYFS

Understanding the World: Computing Overview

Playing & Exploring - Engagement	Active Learning - Motivation	Creating & Thinking Critically - Thinking
- Finding out & exploring - Playing with what they know - Being willing to 'have a go'	- Being involved & concentrating - Keep on trying - Enjoying achieving what they set out to do	- Having their own ideas (creative thinking) - Making links (building theories) - Working with ideas (critical thinking)

ELG

NO ELGs are represented for this area.

Focus	Electronic Communication Understanding Technologies	Text and Multimedia	Research and E-Safety	Digital images and audio	Algorithms Handling information	Vocabulary- To be used daily.
Nursery Skills	Shows an interest in technological toys with knobs or pulleys, real objects such as cameras, and touchscreen devices such as interactive screen, table top computer and tablets	Knows how to operate simple equipment, e.g. turn on CD player, uses a remote control, can navigate touch-capable technology with support	- Know how to handle equipment safely - Begin to know that they shouldn't use devices without supervision	Knows that information can be retrieved from digital devices and the internet	Shows skill in making toys work by pressing parts or lifting flaps to achieve effects such as sound, movements or new images	Choices, equipment, buttons, movement, screen, keyboard, count, organise,
Nursery Knowledge	Autumn 1 Ourselves	Autumn 2 celebrations	Spring 1 Moving on up	Spring 2 What a wonderful world	Summer 1 What's the story	Summer 2 Rescue me
	Explore different toys in role play such as telephones, cameras, keyboards.	Can operate a simple CD player by pressing start and stop to play music.	Understands that we can search for information on 'google' by typing in a word to find out more.	Can use a simple I board touch programme to draw a picture by changing tools and colours using the on-screen options.	Can operate simple games on the iPad and know to open and end a programme.	Can follow instructions when operating a device

Children to be exposed to key vocabulary daily in provision. High quality resources will be provided for daily accessibility.

Role-play will be a key area where a range of technologies will be used in play - telephones, microwaves, cookers, keyboards. These should be modelled.

Interactive screen and table top computer as part of continuous provision

Core Values: Resilience, Respect, Team Work, Aspiration, Kindness, Curiosity
Curriculum Intent: RAISE

The Meadows Primary Academy - COMPUTING progression through EYFS

Understanding the World: Computing Overview

Playing & Exploring - Engagement		Active Learning - Motivation		Creating & Thinking Critically - Thinking		
- Finding out & exploring - Playing with what they know - Being willing to ‘have a go’		- Being involved & concentrating - Keep on trying - Enjoying achieving what they set out to do		- Having their own ideas (creative thinking) - Making links (building theories) - Working with ideas (critical thinking)		
ELG NO ELGs are represented for this area.						
Focus	Electronic Communication Understanding Technologies	Text and Multimedia	Research and E-Safety	Digital images and audio	Algorithms Handling information	Vocabulary- To be used daily.
Reception Skills	Completes a simple program on electronic devices	Begin to list different IT in their home	- Begin to give reasons why we need to stay safe online - Can use the internet with adult supervision to find and retrieve information of interest to them	Can create content such as a video recording, stories, and/or draw a picture on screen	Develops digital literacy skills by being able to access, understand and interact with a range of technologies	Internet, website, mouse, images, paint, technology, share, collect, set, sound, communicate, videos, photos, programme
Reception Knowledge	Autumn 1 “Who am I?”	Autumn 2 “Who am I?”	Spring 1 “Food to fork”	Spring 2 “Food to fork”	Summer 1 “where will we go now? Water water everywhere”	Summer 2 “where will we go now? Water water everywhere”
	- Can turn on an iPad or a Kindle to open a programme and follow instructions. - Can explain how to stay safe when using the internet.	Can follow the teacher’s instructions when using an online interactive programme such as paint or draw.	To collect information about the measurement of plants and see which was the best environment for growing in.	Can write a variety of CVC words using a keyboard.	- Can use the iPad or Kindle to take their own images - Can send a group class email to a different class and wait for a response.	Can use ‘Google’ to find out more information about animals and use the images to support their own representations.
E-Safety		Computer Skills		Programming		Word Processing skills
						Data Collection
Children to be exposed to key vocabulary daily in provision. High quality resources will be provided for daily accessibility. Role-play areas will be a key area where a range of technologies will be used in play- telephones, microwaves, cookers, keyboards, tills. These should be modelled.						

Our Mission: To drive personal and academic excellence; everyone, every day.

Year 1

NC states that in 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
- create and debug simple programs
- use logical reasoning to predict the behaviour of simple programs
- use technology purposefully to create, organise, store, manipulate and retrieve digital content
- 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			
Unit	End points	Key skills	Key vocab
Creative computing	• Art can be created digitally • Computers can be controlled with mouse or touch • Digital tools can create games • ☑ Work can be shared	• Select colours and painting effects • Use mouse or touch • Drag and drop objects • Create jigsaws and games • Share and access work	create, design, digital, tool, program, save, open, edit, image, text
Data Explorers	• Items can be grouped using criteria • Sorting organises items • Sorting and grouping differ • Data is information • Data can be shown visually • ☑ Data collection requires planning	• Identify sorting criteria • Group and sort items • Complete grouping/sequencing tasks • Decide whether to sort or group • Use data to answer questions • Create pictograms • Collect and record data • ☑ Interpret pictograms	Data, information, object, group, sort, category, label, compare, criteria, same, different, chart, pictogram, block diagram, tally, count
Animated stories	• Digital books differ from traditional books • Images and animations can be added • Copy and paste moves content • Audio can be added • ☑ Backgrounds and fonts can change	• Use drawing tools • Edit images • Add animation • Copy and organise pages • Add sound and music • Add backgrounds • Change fonts and text size	animation, story, character, scene, background, movement, sequence, frame, play, stop
Creating & following instructions	• Clear instructions are needed • Programs follow algorithms • Vague instructions cause different results • Order of instructions matters • ☑ Debugging fixes errors	• Give clear instructions • Test instructions • Follow instructions on devices • Identify errors • Correct sequences • Apply debugging	algorithm, instruction, command, sequence, order, direction, forward, turn, route, debug
Technology around us	• Technology solves problems • Technology is everywhere • Not all technology is digital • Hardware is physical parts	• Identify technology • Describe uses • Compare digital and non-digital • Name hardware	technology, device, computer, keyboard, mouse, screen, click, drag, type, icon

Core Values: Resilience, Respect, Team Work, Aspiration, Kindness, Curiosity
Curriculum Intent: RAISE

Our Mission: To drive personal and academic excellence; everyone, every day.

	Technology has risks	- Use technology safely - Identify risks	
Making Beats	- Digital and traditional music differ - Digital tools create music - Music tools can change sound - Multiple sounds can combine ☑ Different tools suit different tasks	- Compare music types - Compose melodies and beats - Select sounds - Change tempo and volume - Combine sounds ☑ Choose appropriate tools	music, beat, rhythm, tempo, sound, instrument, pattern, loop, play, compose
Coding	- Programs use algorithms - Code uses blocks - Commands are instructions - Code must be run - Events trigger actions - Debugging fixes code ☑ Programs are designed first	- Create algorithms - Interpret code - Use code blocks - Drag and arrange code - Run programs - Use event blocks - Debug code - Design programs ☑ Test and improve	code, program, algorithm, command, object, action, event, sequence, debug, predict

Year 2

NC states that in 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
- create and debug simple programs
- use logical reasoning to predict the behaviour of simple programs
- use technology purposefully to create, organise, store, manipulate and retrieve digital content
- 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			
Unit	End points	Key skills	Vocabulary
Creating pictures	- Digital tools have painting effects. - Effects can be combined. - Brush size can change. - Colour intensity can change. ☑ Outlines help compose images.	- Explore painting effects. - Create images in styles. - Change brush size. - Adjust colour intensity. - Use outlines, resize and edit.	digital art, paint tool, brush size, effect, style, colour intensity, outline, resize, edit
Spreadsheets	- Spreadsheets organise data in rows/columns. - Cells have names. - Rows = horizontal, columns = vertical. - Formatting improves clarity. - Images can have values. - Calculations can be automated.	- Identify and use cells. - Enter and move between data. - Resize rows/columns. - Format data. - Insert images with values. - Use calculations (add, subtract, etc.).	spreadsheet, cell, row, column, data, format, alignment, wrap text, calculate, formula, equals, count, tool, block diagram

Core Values: Resilience, Respect, Team Work, Aspiration, Kindness, Curiosity
Curriculum Intent: RAISE

Our Mission: To drive personal and academic excellence; everyone, every day.

	- Tools include: Is Equals, Quiz, Count. - Block diagrams present data.	- Use Is Equals, Quiz, Count tools. - Create block diagrams.	
The internet	- The internet is a global network. - Internet ≠ World Wide Web. - Internet connections can be wired/wireless. - A browser accesses websites. - The web is what you see in a browser. - Smart devices connect to the internet. - Homepage = front page. - Links connect webpages. - Search engines find websites.	- Explain internet vs web. - Recognise Wi-Fi. - Identify a browser. - Navigate websites. - Identify smart devices. - Use links. - Search using keywords.	internet, world wide web, browser, website, webpage, homepage, link, search engine, keyword, online, wifi, smart device
Route explorers	- A command = direction + distance. - Commands control movement in 2Go. - Planning a route is important. - Routes can be sequences. - An algorithm is a list of instructions. - Errors are bugs; fixing them is debugging	- Input commands in 2Go. - Move a turtle to a goal. - Plan routes (e.g. finger tracing). - Create sequences of commands. - Reset and re-run code. - Write algorithms before inputting. - Debug and fix errors.	algorithm, instruction, command, direction, distance, forward, backward, turn, quarter turn, route, sequence, plan, debug, bug
Questioning	- Data is collected information. - Tally charts record data. - Charts present data visually. - Open vs closed questions differ. - Yes/No questions classify. - Branching databases organise info.	- Collect and record data. - Use tally charts. - Create pictograms/bar charts. - Form Yes/No questions. - Use branching databases.	data, tally chart, pictogram, bar chart, branching database, yes/no question, open question, closed question, classification
Making music	- Music can be live or digital. - Digital music uses programs. - Sounds create tunes. - Notes of same colour = same tone. - Tempo = speed (BPM). - Volume can change. - Looping repeats music. - Programs can record sounds.	- Identify sound types. - Use 2Sequence. - Play compositions. - Add and organise sounds. - Compare notes. - Change tempo. - Adjust volume. - Loop tracks. - Record and play back sounds.	music, beat, tempo, bpm, track, volume, loop, instrument, sound, composition, record, playback
Coding	- An algorithm is a set of instructions. - Collision detection triggers actions. - Programs follow sequences. - Timers delay actions. - Objects have attributes. - Events trigger code. - Debugging fixes bugs.	- Follow algorithms. - Use collision detection. - Run sequential code. - Use timers. - Modify object attributes. - Use event blocks. - Debug programs	algorithm, instruction, sequence, command, program, event, action, object, collision detection, timer, debug, bug, attribute

Core Values: Resilience, Respect, Team Work, Aspiration, Kindness, Curiosity
Curriculum Intent: RAISE

Our Mission: To drive personal and academic excellence; everyone, every day.

Year 3

NC states that in 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
- 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
- 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
- use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact

Year 3			
Unit	End points	Key skills	Vocabulary
Route planners	• a command combines direction and distance • algorithms are sets of instructions • routes use sequences of commands • repetition can be used in routes • ☒ bugs are errors in code	• Input commands to control movement • plan routes before coding • create sequences of commands • use repeat functions • debug and fix errors • ☒ test and rerun programs	algorithm, instruction, command, direction, distance, sequence, route, plan, turn, repeat, debug, bug
Email	• email is a form of digital communication • different communication methods have pros and cons • emails are sent instantly • emails have features (inbox, subject, message, etc.) • attachments can be added to emails • there are risks when using email • personal information should be protected	• send and receive emails • use inbox, compose, and reply • add attachments • use an address book • recognise unsafe emails • report concerns • choose appropriate communication methods	email, inbox, compose, reply, subject, message, attachment, address book, contact, communication, online safety
Coding	• flowcharts represent algorithms • timers control when code runs • repeat loops repeat actions • nesting affects how code runs • debugging fixes errors	• read and follow flowcharts • use timers and repeats • create nested code • test and debug programs • predict outcomes of code	flowchart, algorithm, timer, repeat, sequence, nesting, debug, bug, event, action
Presentations	• presentation software communicates information • slides can include text and images • design, animations, and transitions enhance slides • effective presentations are clear and organised	• create slides • insert text and images • apply designs, animations, transitions • add notes • present clearly to an audience	presentation, slide, text box, image, design, layout, animation, transition, notes, audience
Micro:bit	• a micro:bit is a programmable device • it has inputs and outputs • code controls behaviour • ☒ sequence affects outcomes	• write and transfer code to a micro:bit • use inputs (buttons, sensors) • create outputs (LEDs, sounds) • test code using a simulator	micro:bit, input, output, program, code, sequence, event, sensor, LED, button, debug, repeat

Core Values: Resilience, Respect, Team Work, Aspiration, Kindness, Curiosity
Curriculum Intent: RAISE

Our Mission: To drive personal and academic excellence; everyone, every day.

Branching databases	- a database stores organised data - binary questions have yes/no answers - branching databases use binary choices - good questions split data effectively ☑ debugging ensures accuracy	- debug and improve programs - create and use a database - ask binary questions - sort and classify data - build a branching database - test and debug databases	database, data, binary question, yes/no, branching, classify, sort, record, search, criteria, debug
---------------------	---	---	---

Year 4

NC states that in 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
- 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
- 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

use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact

Year 4			
Unit	End points	Key skills	Vocabulary
Animation	- Some animations are created by hand, while others use technology. - Animation software includes features such as static backgrounds, onion skinning, and frame copying. - Sound effects, timing, and frame rates can improve animations. ☑ Storyboarding helps plan animations.	- Explain how hand-drawn animation is created. - Make a simple flipbook animation. - Compare hand-drawn and digital animation methods. - Use 2Animate to create simple animations. - Select suitable sound effects and animation speeds. ☑ Use storyboards to plan animations.	Animation, Frame, Flipbook, Flick Book, Storyboard, Onion Skinning, Static Background, Copy Frame, Frames Per Second (FPS), Sound Effect, Timing, Sequence
Unpacking hardware	- Technology uses scientific knowledge to design tools, systems, or machines that solve problems or make tasks easier. - Electrical, digital, and smart technology are types of technology. - Hardware refers to the physical parts of a computer. - Software refers to programs that instruct computers to perform tasks. ☑ Hardware and software work together to complete processes and task	- Identify examples of technology. - Classify items as electrical, digital, or smart technology. - Define hardware, components, and peripherals. - Name computer hardware components. - Describe the function of hardware parts. - Identify the functions and components of software tools. ☑ Describe processes using inputs, processing, hardware, software, and outputs.	Technology, Electrical Technology, Digital Technology, Smart Technology, Hardware, Software, Component, Peripheral, Computer System, Input, Process, Output, Device
Sound stories	- An audiobook is a recorded story or book. - Audiobooks can make texts more accessible. - Expression, voices, sound effects, and music improve audiobooks.	- Explain differences between audiobooks and physical books. - Describe advantages and disadvantages of audiobooks. - Identify features of engaging audiobooks.	Audiobook, Narration, Script, Sound Effect, Background Music, Tone, Pitch, Expression, Pronunciation, Editor, Narrator, Sound Technician, Sound Effects Manager, Recording, Timeline, Edit, Copy, Paste, Duplicate, Evaluate

Core Values: Resilience, Respect, Team Work, Aspiration, Kindness, Curiosity
Curriculum Intent: RAISE

Our Mission: To drive personal and academic excellence; everyone, every day.

	• Audiobook scripts include speaker notes, spoken text, and sound-effect instructions. • Audiobook production involves different team roles. • Effective narration uses clear pronunciation, tone, pitch, and pauses. • Editing improves the quality of an audiobook. • ☑ Reviewing and evaluating are important parts of creating digital content.	• Identify places where sound effects or voices could improve a story. • Create sound effects using everyday objects. • Use the 2Cast sound and music library. • Read and follow audiobook script cues. • Help write audiobook scripts. • Explain roles such as narrator, editor, sound technician, and sound-effects manager. • Take part in recording and editing an audiobook. • Experiment with voice effects and narration styles. • Use editing tools such as cut, copy, paste, duplicate, and move. • ☑ Evaluate and improve audiobook work.	
Logo	• Logo is a text-based programming language that controls a screen turtle. • Logo commands include movement, rotation, and operational commands. • Correct spacing is important in Logo code. • Visual effects can be created using commands such as PU, PD, SETPC, and SETPS. • The repeat command allows commands to run multiple times. • A procedure is a named set of commands. • ☑ Errors in code are called bugs and fixing them is called debugging.	• Enter Logo commands to move the turtle. • Use directional commands (FD, BK, RT, LT). • Use commands such as PU and PD. • Apply correct spacing and debug spacing errors. • Modify line properties and visual effects. • Use repeat commands efficiently. • Create regular shapes using repeat. • Predict the effects of repeat commands. • Create, save, and use procedures. • Use procedures efficiently. • Debug Logo programs logically. • ☑ Use multi-line mode to support debugging.	Logo, Coding, Programming Language, Turtle, Command, Forward (FD), Backward (BK), Right Turn (RT), Left Turn (LT), Pen Up (PU), Pen Down (PD), Repeat, Procedure, Debug, Bug, Sequence, Angle, Degrees
Composing beats	• Music contains pulse, rhythm, tempo, pitch, melody, texture, and dynamics. • Rhythm is a pattern of sounds and silences. • Digital rhythm uses samples. • A sample is a short recorded sound. • Tempo is the speed of music. • BPM controls tempo in Busy Beats. • Melody is the tune made from notes. • Digital melodies use synths. • Pitch describes how high or low a sound is. • Higher synth numbers produce higher pitches. • Music in Busy Beats uses samples and synths. • ☑ Music can be arranged using multiple patches.	• Recognise and explain musical elements. • Clap the pulse. • Identify instruments and sounds. • Discuss likes and dislikes about music. • Identify rhythms. • Clap simple rhythms. • Create rhythms using samples. • Adjust tempo using BPM. • Identify and sing simple melodies. • Create melodies using synths. • Change pitch using the synth board. • Experiment with samples and synths to create moods. • Create and record music in Busy Beats. • ☑ Use multiple patches to build and arrange music.	Pulse, Rhythm, Tempo, Pitch, Melody, Texture, Dynamics, Sample, Synth, Beat, BPM (Beats Per Minute), Electronic Music, Patch, Instrument, Mood, Composition, Recording
Coding	• Different object types in 2Code have different attributes. • Selection uses decision-making commands. • Coordinates determine object positions on a screen. • If and if/else statements create selection. • Loops can be created using repeat, repeat until, and timers. • Variables store information in computer memory.	• Change object attributes in Design View. • Choose suitable object types. • Interpret flowcharts containing selection. • Identify and modify coordinates. • Use if and if/else blocks. • Read and predict programs with selection. • Create and debug selection programs.	Program, Algorithm, Object, Attribute, Property, Coordinate, Selection, Condition, If Statement, If/Else Statement, Repeat, Repeat Until, Timer, Loop, Variable, Value, Debug, Flowchart

Core Values: Resilience, Respect, Team Work, Aspiration, Kindness, Curiosity
Curriculum Intent: RAISE

Our Mission: To drive personal and academic excellence; everyone, every day.

	☐ Variables have names and values that can change	• Nest if/else statements. • Use repeat loops. • Use repeat until loops. • Decide when to use repeat or timers. • Create and initialise variables. • Change variable values. • Use variables within programs. • ☐ Monitor variables using variable watch	
--	---	---	--

Year 5

NC states that in 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
- 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
- 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

use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact

Year 5			
Unit	End points	Key skills	Vocabulary
Quizzing	• There are different types of quizzes for different purposes. • 2Quiz contains different question types. • A successful quiz considers difficulty, interest and audience capability. • Enjoyment, visual variety and feedback affect quiz quality. • Additional features such as title and content screens improve quizzes. • ☐ Quiz settings affect the look and feel of a quiz.	• Select the best quiz type for a task. • Choose suitable question types. • Design quizzes that meet a brief. • Use additional quiz features effectively. • ☐ Select appropriate quiz settings.	Quiz, Question Type, Audience, Feedback, Title Screen, Content Screen, Multiple Choice, Text Question, Image Question, Timer, Settings, Design Brief, Evaluation, User Experience
Databases	• Databases contain data that can be searched and sorted. • Table-based databases contain records and fields. • Databases support sorting, filtering, grouping and searching. • Collaborative databases allow multiple users to work on data. • Well-designed databases use validation. • Queries answer specific questions about data. • Some databases use linked tables. • ☐ Databases can generate automated reports.	• Use 2Investigate. • Create and edit records and fields. • Sort, group and search data. • Answer questions using a database. • Add records to collaborative databases. • Design databases with validation. • Build and use queries. • Use conditional operators in queries. • Create queries across multiple tables. • Produce automated reports.	Database, Data, Record, Field, Table, Search, Sort, Filter, Group, Validation, Query, Condition, Conditional Operator, Collaborative Database, Linked Tables, Report, Information

Core Values: Resilience, Respect, Team Work, Aspiration, Kindness, Curiosity
Curriculum Intent: RAISE

Our Mission: To drive personal and academic excellence; everyone, every day.

Game creator	• Certain criteria determine whether a game is playable. • 3D games can be made using 2DIY 3D. • Planning improves game quality. • Visual design should support the game theme. • Audio design should support immersion. • Character and quest item design are important. • Good instructions improve player experience. • ☑ Evaluation helps improve games.	• Evaluate game playability. • Create a 3D game. • Use a design document. • Design textures and environments. • Select or create sound effects and music. • Design collectable items and sprites. • Position game elements effectively. • Choose penalties for enemies. • Create instruction screens. • ☑ Evaluate and improve games.	Game Creator, Playability, 3D Game, Design Document, Theme, Character, Sprite, Quest Item, Texture, Scenery, Hazard, Lighting, Sound Effect, Music, Instructions, Evaluation, Immersion
Word processing	• Word processors can create many document types. • Document layouts can be changed. • Images can be inserted and edited. • Text appearance and placement can be modified. • ☑ Tables can present information effectively.	• Create and open documents. • Use templates. • Navigate toolbars. • Save and rename documents. • Format text. • Change page layout. • Insert and edit images. • Crop and resize images. • Wrap text around images. • Use copy, paste and cut. • Add headings and bullet points. • Use font formatting. • Insert and edit tables. • Format table appearance.	Word Processor, Document, Template, Toolbar, Formatting, Font, Heading, Subheading, Bullet Point, Numbered List, Margin, Portrait, Landscape, Image, Crop, Resize, Wrap Text, Table, Row, Column, Cell
Coding	• Good code structure improves efficiency and debugging. • Simulations model real or imaginary situations. • Abstraction removes unnecessary detail. • Decomposition breaks tasks into manageable parts. • Functions reduce repeated code. • Datatypes affect how data can be used. • Variables should be initialised. • ☑ Concatenation joins items together	• Use tags, tabs and functions. • Explain simulations. • Plan and code simulations. • Test and evaluate programs. • Apply abstraction. • Use decomposition. • Create flowcharts with sub-routines. • Create and call functions. • Choose suitable datatypes. • Monitor variables. • Initialise variables. • Use concatenation. • Debug concatenation errors.	Code, Program, Algorithm, Simulation, Physical System, Abstraction, Decomposition, Sub-routine, Function, Variable, Datatype, Number, Text, Operator, Initialisation, Concatenation, Debugging, Flowchart, Tag, Tab
Concept maps	• Concept maps organise ideas and show links. • Concept maps begin with a main idea and branch out. • Concept maps have different purposes. • Different layouts improve clarity. • Concept maps can be adapted for different tasks. • Good organisation improves understanding.	• Create concept maps in 2Connect. • Add and connect nodes. • Organise ideas logically. • Improve layout and formatting. • Choose suitable map structures. • Refine concept maps.	Concept Map, Node, Link, Connection, Main Idea, Category, Sub-category, Layout, Structure, Organisation, Planning, Presentation, Audience, Feedback, Formatting

Core Values: Resilience, Respect, Team Work, Aspiration, Kindness, Curiosity
Curriculum Intent: RAISE

Our Mission: To drive personal and academic excellence; everyone, every day.

	• Concept maps can support presentations. • ☑ Feedback improves work.	• Explain concept maps to others. • Present ideas clearly. • ☑ Improve work using feedback.	
--	--	---	--

Year 6

NC states that in 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
- 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
- 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
- use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact

Year 6			
Unit	End points	Key skills	
Blogging	• A blog is a regularly updated webpage, written about a particular topic. • Blogs consist of several blog posts. • A well written blog post has certain features that make the blog clear and easy to understand and increase reader engagement. • The 'hook' draws the reader into the blog. • The look and feel of a blog post makes it clear for the reader to access the information. • The conclusion of a blog post ties the information in the post together. • The process of writing a blog post requires planning, drafting, revising and editing before publication. • Engaging with readers is crucial to the success of a blog. • Moderation exists to make the blogging environment a safe place for its readership and authors.	• Give examples of topics for existing or prospective blogs. • Create a blog post. • Plan the hook, look and feel, conclusion and reader engagement for a blog post. • Use an appropriate hook for a blog post by including either a quote, a story, a question or an observation to grab the reader's interest. • Write a blog post that is easy to follow, uses lists or bullets, bolds key information and uses an appropriate conversational style. • Write a conclusion that summarises the main points of the post and might give the reader advice. • Follow the plan, draft, revise and edit process before publishing a blog post. • Read and respond to comments on their blog post. • Use commenting to increase engagement and guide future blog posts. • Decide whether content conforms to appropriate netiquette guidance. • Report posts or comments that violate community or legal guidelines.	blog, blog post, hook, conclusion, audience, engagement, comment, moderation, netiquette, draft, edit, revise, publish, layout, formatting, bullet points
Networks	• A network describes a group of connected computers that can share information and hardware resources. • LAN and WAN are different kinds of networks. • Certain hardware is required to create a network.	• Identify types of computer networks locally and globally. • Explain the hardware resources that a network might share.	network, LAN, WAN, internet, World Wide Web, router, server, device, Wi-Fi, mobile data, 5G, browser, hardware, communication, filtering, censorship

Core Values: Resilience, Respect, Team Work, Aspiration, Kindness, Curiosity
Curriculum Intent: RAISE

Our Mission: To drive personal and academic excellence; everyone, every day.

	• Networks can be wired or wireless or a combination of both. • There is a difference between the World Wide Web and the Internet. • Web browsers are used to access the World Wide Web. • The existence of networks has opened online communication. • Internet filtering and censorship are both used to make parts of the internet less accessible for different reasons.	• Explain the difference between LAN and WAN. • Create a network diagram that includes hardware such as a router and different connected devices and peripherals. • Identify the terms Wi-Fi, mobile data and 5G as pertaining to wireless network connections. • Describe the difference between the Internet and World Wide Web giving examples of the services that both provide. • Give examples of web browser tools. • Give examples of online communication. • Give safety tips related to online communication • Explain the differences between internet filtering and censorship and why they are used.	
Data Detectives	• A database contains data organised in such a way that it can be queried to find useful information. • Query tools exist to help users of a database find information from data stored within it. • Chart tools can be used to make graphs from data contained within a database. • Some databases will need multiple tables of data. • Queries can be created that query data from multiple tables. • Bespoke multiple conditions can be made that help a user find the exact information being asked.	• Identify tables, records and fields. • Explain the data types contained within each field on a record. • Identify any format types applied to fields. • Use query tools to find useful information. • Use the filter tool to create conditions. • Use the grouping tool to group related information together as part of a query. • Use the calculate tool to apply calculations to selected data that provides meaningful results. • Use the sort tool to change the order records are presented according the field selected and the value order (increase/decrease). • Use the chart tool to select specific fields from a database for the x and y values on a graph. • Select the most appropriate graph type to display selected data so it makes sense in relation to a query given. • Identify differences in data held in individual tables that make a complete database. • Explain why the data can't be all held on one table. • Link tables together by using a common field. • Produce queries that query data from linked tables. • Create multiple condition and can join those using the AND/OR operator. • Use the AND/OR operator correctly and recognise the difference between the two and the impact this has on results when run. • Use brackets to contain multiple conditions and apply a general condition that applies to all conditions made.	database, table, record, field, query, filter, sort, chart, graph, data type, condition, criteria, calculation, grouping, AND operator, OR operator, brackets, linked tables, primary key, secondary key
Coding	• Cloning is a process where additional objects can be created while the program is running.	• Write code to clone objects and loop through the clones to change their attributes. • Make objects draggable in a program.	algorithm, program, clone, hotspot, variable, data type, input, output, selection, iteration, loop, function,

Core Values: Resilience, Respect, Team Work, Aspiration, Kindness, Curiosity
Curriculum Intent: RAISE

Our Mission: To drive personal and academic excellence; everyone, every day.

	• Objects can be draggable in 2Code to provide another way for the user to interact with them. • Hotspots can be created in 2Code. These are a type of object that can be used to make a specific area of the screen react to actions such as objects entering or leaving the area. • Using a combination of coding structures and objects allows the design of interactive and creative programs. • Variables can have different designated datatypes. The datatype will impact how the value of the variable can be used. • There are a variety of ways of receiving input from users and then using this input. • A flowchart is a diagram that uses specifically shaped, labelled boxes and arrows to represent an algorithm as a diagram. • Using functions helps with making programs more efficient. Instead of writing the same sequence of code repeatedly • Design documentation is vital when producing more complex programs. The process of designing supports abstraction and decomposition of the task at hand. • Text-based adventures are computer games that have been created using text instead of graphics. Players use text commands to control characters and influence the environment.	• Create hotspots and write code that controls the flow of the program when objects enter and leave hotspots. • Use code to create effects when combining the use of draggable objects, hotspots and cloning in conjunction with sequences, repetition and selection. • Make conscious choices about datatype when creating variables. • Anticipate effects such as calculating using variable values, concatenation of string variables and bugs that could occur due to datatypes when user input is used. • Use 'Prompt for input' and 'get input' commands to receive input from the user of a program and write code using variables to work with user input. • Use if statements that include checking of a value of a user input and changing a variable if user input has met the if statement condition. • Interpret flowcharts and use these as a basis to begin coding or debugging a program. • Create flowcharts to represent algorithms. • Include sub-routines within flowcharts to help with more complex program ideas. • Create functions and write code for function calls. • Interpret programs that include functions to explain the purpose of the function. • Interpret design documentation, commenting on the level of abstraction and the way that a task has been decomposed. • Use user interface designs to create the look of a program. • Use flowcharts to decide upon the structure of a program and what functions might be required. • Review the code of a text-based adventure game and relate this to game play when the code is running. • Identify how variables, functions, selection and repeat loops are used to make the program function. • Adapt the code maintaining the structure but changing the game play.	subroutine, flowchart, debugging, decomposition, abstraction, sequence, condition
3D modelling	• Designers use CAD (Computer Aided Design) to create accurate drawings or 3D Models on a screen. • CAD allows designers to create accurate and flexible designs, allowing them to test out ideas and spot problems before something is built in real life. • 2Design and Make is the CAD software on Purple Mash which allows users to develop a range of 3D models. • There are three viewpoints on 2Design and Make; Points view, Net view, and 3D view.	• Define what a 3D model is. • Identify different industries where CAD is used. • Identify the benefits of using CAD. • Explain how CAD is used in a variety of industries such as car manufacturing, architects and creating packaging. • Use 2Design and Make accurately, select models which fit their design best and adapt them where necessary. • Spot problems in their own 3D models on 2Design and Make and make adaptations where necessary.	CAD, model, prototype, design brief, net view, 3D view, points, texture, pattern fill, STL file, printer, 3D printer, refine, manipulate, packaging, industry, design slider

Core Values: Resilience, Respect, Team Work, Aspiration, Kindness, Curiosity
Curriculum Intent: RAISE

Our Mission: To drive personal and academic excellence; everyone, every day.

	- The pattern fill tool allows you to design a repeated pattern to add to your model. - Moving, adding and removing points on a model allows the designer to manipulate the shape to meet the desired design. - A prototype is a first version of a design which is designed and made to test out ideas. - CAD can be used to create a prototype for packaging for a product. - Packaging is important to protect a product, give information, promote the product, attract customers and make the product more convenient to carry, use and store. - A design brief is a set of instructions or a challenge that explains what needs to be designed. - Models need to be refined before they are printed out using a standard printer or 3D printer. - A 3D printer can print a physical object from a 3D digital model, typically by laying down many thin layers of material in succession. It requires a STL file.	- Add textures to the model using the painting tools. - Use the design slider to alter the width of the model. - View a model from each viewpoint. - Use each view to better their models, making use of the editing tools within each viewpoint. - Create repeated patterns that suit the model. - Add patterns into small sections of their model design. - Change the location of the points to alter the appearance of the model. - Add and remove points to manipulate the shape of their model to meet a purpose. - Explain why a prototype is used in a range of industries before a real product is created. - Explain how creating a packaging prototype is useful. •Recognise a range of different packaging designs and shapes and think about how effective they are. - Discuss how packaging is used in a variety of ways. - Explain why design briefs are important. - Plan a design for a model to meet a design brief – a piece of packaging for a healthy snack. - Refining a model is important prior to the final printing process. - Print their model onto paper/card or via a 3D printer. - Construct their 3D model if appropriate. - Identify the differences between a standard printer and a 3D printer. - Identify a STL file. - Discuss the use of 3D printing in different industries. - Discuss the types of materials that 3D printers can use to create a physical object.	
Python	- Python is a text-based coding language. - Python has in-built functions that can be used. - A library of functions can be imported into a program to use. - A sleep function can be used to control how a program is executed. - There are two main types of loop in Python and that these can be used to improve coding efficiency. - Range is a function that lets a coder in Python get a series of numbers or specifies how many times a loop is to run. - Python can be used to perform mathematical calculations. • Sprites can be programmed to move to positions on the screen.	- Use language syntax correctly for simple programs which includes the use of brackets, quotes, indents and correct case letters. - Use the print function to output strings, values of variables or a combination of both. - Import the sleep function from the time module within Python's library. - Use the sleep function to control the speed of execution of lines of code when printing outputs to the screen. - Use a 'for loop' to repeat a block of code for a specified time. - Explain the difference between a 'for loop' and a 'while loop'. - Use the range function in a 'for loop' to specify how many times that loop is to run.	Python, syntax, print, variable, string, integer, float, library, import, function, sleep, loop, for loop, while loop, range, operator, coordinate, sprite, output, code block

Core Values: Resilience, Respect, Team Work, Aspiration, Kindness, Curiosity
Curriculum Intent: RAISE

- | | | | |
|--|--|---|--|
| | | <ul style="list-style-type: none">• Use the range function to start from a given number and end on a given number.• Use mathematical operators with numbers to print answers to calculations to the output window.• Use design mode to select sprites to place on the screen.• Use coordinates to identify the position of a sprite.• Program a sprite to move to different places on a screen by using co-ordinates to identify current and intended location. | |
|--|--|---|--|