Reception to year 6

Technologies

Digital technologies

September 2022

Scope and sequence

Revised to align with the Australian Curriculum V9.0 (2022)

V2.0

Government of South Australia
Department for Education

Digital technologies: Reception to year 6

Contents

Context statement

Digital technologies is about exciting interest in computer science. It is about empowering students to understand how the devices they use to connect, learn and play actually work from the inside out. Students will develop the skills and confidence to imagine, design and create their own digital solutions. Learning in digital technologies requires that students use 3 distinct thinking strategies.

Systems thinking

Systems thinking helps people to think holistically about the interactions and interconnections that shape the behaviour of systems. It could involve:

- making distinctions – knowing what something is and isn't by looking for similarities and differences, making comparisons and evaluating against agreed criteria
- parts and wholes – understanding the way each of the parts of a system contributes to the whole and if one part breaks down this can impact on the whole system
- relationships – identifying the relationship between cause and effect by looking at the interactions between all parts of a system, understanding that systems have input and outputs
- perspectives – considering varying points of view, understanding biases and the impacts of systems on people and the planet.

Computational thinking

Computational thinking helps people to organise data logically by breaking down problems into parts; defining abstract concepts; and designing and using algorithms, patterns and models. It could involve the following:

- **Pattern recognition** – Data is all around us and is increasingly a valuable resource to indicate people's preferences. Identifying and understanding patterns, trends, similarities and differences in data is key to defining problems.
- **Decomposition** – Breaking big problems down into smaller parts to make them more manageable to solve in a logical step-by-step process.
- **Algorithmic design** – Creating and communicating a specific set of steps to solve problems using graphics, symbols, text and diagrams which support the development of efficient computer programs.
- **Abstraction** – Removing unnecessary detail to simplify the steps to a solution. In computer programming the steps are often grouped together under one function. It's similar to the instruction 'brush your teeth', which implies many steps: walk to the bathroom, pick up toothpaste, open toothpaste, pick up toothbrush, and so on. If you break down all the steps it would be a lot of instructions, but they can all be grouped under the banner 'brush your teeth'. This is an example of abstraction.
- **Modelling and simulating** – Implementing by acting out or trialling algorithms to identify and fix 'bugs' or mistakes – this is to also to make sure a solution is safe. Imagine road testing a self-driving car without modelling and simulating first.
- **Evaluating** – Once the solution has been modelled and simulated it's time to iron out the 'bugs' and test the solution with real users. This allows us to see if the solution meets design criteria and user needs.

Design thinking

Design thinking is the process of imagining, creating and realising solutions. It involves:

- empathy – **investigating** a problem by understanding a potential end user by finding out their story
- **defining** a problem by considering the end user, data and the potential impacts of a solution
- ideation – the process of **generating** and imagining multiple solution ideas, no matter how farfetched, to find innovative solutions
- prototyping – **designing**, **producing** or **implementing** a model or an iteration of a solution that is simple and inexpensive to present to an end user
- testing – understanding what works for an end user and **evaluating** whether to reiterate, refine or modify solutions to better meet design criteria.

When using these thinking strategies together, students can develop powerful digital solutions.

This document is designed to:

- provide clarity and context for teaching digital technologies in South Australia
- identify the discipline-specific knowledge, skills and understanding learners need at each year level
- guide educators to teach and model computational, systems and design thinking
- support educators with the language and processes to design, create and produce digital solutions.

Achievement standards

Reception	Years 1 to 2	Years 3 to 4	Years 5 to 6
By the end of reception, students: • **show** familiarity with digital systems and use them for a purpose • **represent** data using objects, pictures and symbols • **identify** examples of data that is owned by them.	By the end of year 2, students: • **show** how simple digital solutions meet a need for known users • **represent** and **process** data in different ways • **follow** and **describe** basic algorithms involving a sequence of steps and branching • (with assistance) **access** and **use** digital systems for a purpose • **use** the basic features of common digital tools to **create**, **locate** and **share** content • **collaborate**, following agreed procedures • **recognise** that digital tools may store their personal information online.	By the end of year 4, students: • **create** simple digital solutions • **use** provided design criteria to check if solutions meet needs • **process** and **represent** data for different purposes • **follow** and **describe** simple algorithms involving branching and iteration • **implement** algorithms as visual programs • securely **access** and **use** digital systems and their peripherals for a range of purposes including transmitting data • **use** core features of common digital tools to plan, create, locate and share content • **collaborate** following agreed behaviours • **identify** their personal data stored online and recognise the risks.	By the end of year 6, students: • **develop** and **modify** digital solutions, and define problems and evaluate solutions using user stories and design criteria • **process** data and show how digital systems represent data • **design** algorithms involving complex branching and iteration • **implement** algorithms as visual programs including variables • securely **access** and **use** multiple digital systems and describe their components and how they interact to process and transmit data • **select** and **use** appropriate digital tools to effectively plan, create, locate and share content • **collaborate**, applying agreed conventions and behaviours • **identify** their digital footprint and recognise its permanence.

Scope and sequence

Strand: Knowledge and understanding

Sub-strand	Reception	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6

Digital systems
 	
How do computers work and interact?
 	
The devices we use every day are **digital systems**. Computers, laptops, smartphones, tablets and programmable toys are examples of digital systems. Understanding the way they work requires systems thinking.
 	
Students use **systems thinking** to unpack the ways digital devices work individually and how they connect to larger networks using hardware and software.
 	
Students use **computational thinking** to break down a digital system and understand its component parts.
 	
Students use **design thinking** to imagine new possibilities for digital systems.
	Recognise and explore digital systems (hardware and software) for a purpose.
	
	Students:
	
	• **identify** a system as a group of parts that work together
	• **identify** digital systems as the devices they interact with every day at school, home or in the community, for example, laptops, smart TVs, tablets and programmable toys
	• **notice** 'hardware' as the parts in or on a digital system that we can see, touch and feel
	• **identify** 'software' as the programs or applications we use on devices and recognise their icons
	• **make** a model of a digital system, use it in a role-play scenario and describe its components, for example, keyboard, screen, icons
	• (with guidance) safely **explore** and **play** with
		Recognise that digital systems have many component parts that are used for different purposes.
		
		Students:
		
		• **identify** common digital systems such as computers, tablets, laptops, smartphones, self-serve checkouts, gaming consoles
		• **explore** and **investigate** the parts that make up a digital system, for example, identifying on/off buttons, cords and wires, keyboards, screens, mouse and speakers, and identify these as hardware
		• **describe** the purpose of common component parts, for example, screens, keyboards, controller
		• **show** curiosity about how digital devices work and connect
		• **describe** their favourite applications and recognise them as 'software'
			Identify and **explore** digital systems and their components for a purpose.
			
			Students:
			
			• **notice** that digital systems work with inputs and outputs
			• **identify** the parts of digital systems that are inputs and outputs. For example, in a computer the keyboard and mouse are used to put information in. The screen and speakers send information out of the computer.
			• **suggest** ways in which the component parts of a digital device might work together as a system
			• **identify** software as sets of instructions given to a digital system
			• **imagine** new digital systems and their purposes.
				Understand that peripheral devices can be added to a digital system so that it works with more inputs and outputs.
				
				Students:
				
				• **recognise** different peripheral devices and their input and output functions. For example, a microphone is an example of a peripheral device that allows a user to input voice and sound. Earphones are an example of a peripheral device that allows the user to receive sound as an output.
				• **experiment** with a range of peripheral devices
				• **pose** questions about how digital devices work
				• **understand** software as the instructions given to a digital system so that it will work with peripheral devices
					Explore and **describe** a range of digital systems and their peripherals for a variety of purposes.
					
					Explore transmitting different types of data between digital systems.
					
					Students:
					
					• **identify** the main component parts of a digital system including hardware, software and peripherals
					• **describe** how digital systems need software to process inputs and outputs from peripheral devices
					• **experiment** with using a range of peripheral devices for a variety of purposes
					• **suggest** and describe new possibilities for peripheral devices and their purposes.
						Understand that digital systems are made up of many component parts that support the processing of inputs and outputs.
						
						Understand that the output of a digital system can be images, sound and text that can be sent as data to other systems in a network.
						
						Students:
						
						• **describe** the main component parts of a digital system and their functions
						• **differentiate** between hardware and software
						• **recognise** data as image sound or text that can be processed by digital systems
						• **identify** a network as a collection of digital systems that are connected to allow data to be shared, for example, the internet is a large network of digital systems
						• **explore** how smart systems are the latest iterations of a long human tradition of
							Investigate the main internal components of common digital systems and their function.
							
							Examine how digital systems form networks to transmit data.
							
							Students:
							
							• **explain** how digital systems are made up of many parts that:
							  ◦ receive inputs
							  ◦ process data into information
							  ◦ store information to be retrieved and used later
							  ◦ produce an output that may be images, sound or text
							• **explain** the function of the main component parts of a digital system
							• **describe** how digital systems break down information into smaller parts (packages) and transfer these via a network to other digital systems

Sub-strand	Reception	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
	digital systems for a range of purposes.	• **safely use** digital systems for a variety of purposes		• **identify** hardware and software components in a digital system.		sending and receiving messages. For example, students explore message sticks used by some Aboriginal peoples as a sophisticated communication network.	• **investigate** how smart systems connect, send and receive messages between each other • **imagine** and **propose** new possibilities for digital systems.
Representation of data **How do computers process information?** Students use **systems thinking** to recognise data as agreed codes. Students use the **computational thinking** strategy of **pattern recognition** to identify the ways text, image and sound can be represented as agreed codes. Understanding the ways data can be represented builds student capacity, so they understand digital devices as integrated systems which deliver images, sound and text using electrical impulses. These impulses are represented using the binary number system	**Represent** data as objects, pictures and symbols. **Students:** • **experiment** with using and playing with objects, images and symbols to convey a message • **notice** and **predict** patterns when using manipulatives • **understand** that symbols can be a powerful way to represent big ideas, for example, the ways colours and shapes in the Aboriginal flag have specific meaning.	**Recognise** that digital systems send and receive data. **Identify** data can be represented using movements, symbols, numbers, images and words. **Students:** • (with support) **use** digital tools to **create** and **share** symbols that represent data, for example, showing letter/sound relationships or beats and rests in music • **explore** common logos and symbols and what they represent in and outside of digital systems • **experiment** with communicating meaning using only symbols or images**.**	**Represent** data as pictures, numbers, symbols and words. **Students:** • **identify** and **record** patterns as repeated sequences using digital tools and manipulatives • **recognise** and **describe** the ways that important ideas can be reflected in symbolic representations. For example, exploring the Kaurna seasonal calendar and identifying the symbols, images and words used to represent changing seasons and cultural practices. • **describe** data as images, sound or text that can be stored in or sent to and from digital systems	**Understand** that agreed ways of representing data help users to work with digital systems. **Students:** • **explain** how agreed codes can be learned and used to send and receive messages, for example, using Semaphore flag signalling, Braille, or Morse code • **experiment** with safely sending and receiving coded messages with and without digital tools • **identify** the way that rock paintings and other cultural expressions use images to encode Aboriginal peoples' knowledge.	**Recognise** different types of data and **explore** how the same data can be represented differently depending on the purpose. **Students:** • **explore** the ways that agreed codes are used in everyday contexts, for example, barcodes and QR codes are used to indicate or trigger information • **create** and **use** image codes to trigger information, for example, creating QR coded labels for a local nature trail that link to information about indigenous food-producing plants • **explore** the ways ciphers are used to encode and secure information.	**Understand** that images, sound and text can be represented using combinations of whole numbers and agreed codes. **Students:** • **describe** how text can be represented as whole numbers using the American Standard Code for Information Interchange (ASCII) • **understand** how images can be represented as whole numbers using the red, green and blue (RGB) values in pixels • **experiment** with the binary number system, for example, using 8-bit strings of binary digits to represent numbers and letters • **understand** that digital systems are made up of complex circuits and wires that can only exist in 2 states – on or off.	**Explain** how digital systems represent all data using whole numbers. **Explore** how data can be represented by off and on states (zeroes and ones in binary). **Students:** • **understand** that 1s and 0s are used to represent the on off states in complex electrical circuitry within digital devices • **explain** how text, images and sound are represented using whole numbers. For example, identify how sound can be represented as a wave and how the highs and lows in a sound wave can be encoded as numbers. • **identify** and **explain** how data is stored and retrieved in digital systems using memory and storage

Sub-strand	Reception	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
							• **explore** cryptography as a way to secure data • **consider** how machine learning and Artificial Intelligence (AI) systems use data to sense, reason, act and adapt.

Process and production skills

Sub-strand	Reception	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Investigating and defining **How do I identify a problem that needs solving and who will benefit from the solution?** Students use the **design thinking** strategy of **empathy** and **ideation**. They identify the needs of users through the development of 'user stories' to design suitable digital solutions. Students then use the **computational thinking** strategy of **decomposition**. They decompose (break down) a problem into more manageable steps to define successful solution criteria.		**Explore** and **evaluate** common digital systems based on personal preferences. **Students:** • **listen** and **respond** to people who use digital systems in creative ways to solve problems • **explore** safely using and playing with digital systems • **consider** how digital systems have changed over time to suit changing needs • **investigate** the ways that family, friends and people in the community select and use digital systems for different purposes. For example, easily speaking to people who live far away.	**Investigate** simple problems for known users that can be solved with digital systems. **Students:** • **investigate** problems that have been solved using digital systems • **use** information from data investigations in mathematics to define a need, opportunity or problem that could be solved using digital systems • **investigate** the ways that solutions change according to the needs of a user, for example, a hearing-impaired person may need speech-to-text software so they can read text instead of listening to phone calls.	**Identify** user needs and how they are met through effective design criteria. **Students:** • **use** a range of digital systems for a variety of purposes and **evaluate** based on personal preferences • **investigate** ways digital solutions solve problems for people with diverse needs • **consider** the design criteria needed to be inclusive of people with diverse needs. For example, using speech-to-text to support people with communication difficulties to interact with others.	**Define** problems with given design criteria and by **co-creating** user stories. **Students:** • **investigate** a range of digital systems and their uses • **use** empathy to create user stories and **investigate** what is needed from a digital solution. For example, using translation tools to communicate in different languages. • **define** a problem using the needs and opportunities revealed in information taken from data sets and user stories • **explore** and **respond** to design criteria	**Identify** the useful features of a digital system and how they solve problems for specific users. **Suggest** improvements or modifications. **Students:** • **explore** a range of digital systems and how they solve problems for diverse users, for example, how people with a visual impairment might access online gaming • **identify** the ways people are represented in existing digital solutions and create more diverse representations, for example, identify cultural and gender stereotypes in electronic games and	**Define** problems with given or co-developed design criteria and by creating user stories. **Students:** • **investigate** a range of digital systems and how they are adapted to meet the needs of diverse users • **investigate** and **disrupt** cultural and gender stereotypes in digital solutions by creating inclusive user stories • **define** the specific needs of a potential end user through data collection and user stories • **develop** design and success criteria based on identified needs or opportunities.

Sub-strand	Reception	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Students use **systems thinking** to consider different perspectives of potential users.						develop more diverse characters • **create** user stories to support the development of design criteria for a digital solution, for example, interviewing younger students about their gaming habits and preferences.	
Generating and designing **What ideas do I have about possible solutions?** Students also use the **design thinking** strategy of **ideation** to come up with solution ideas. They **evaluate** and select the best solution based on the needs of end users. Students use **systems thinking** to understand solutions as a **user interface**. A **user interface** is the way humans interact with digital systems. They could be a keyboard, buttons or touch sensors that activate apps, or the way a website is displayed for easy use.		**Generate** and **organise** solution ideas. **Identify** a preferred solution and break it down into achievable steps. **Students:** • **consider** a simple problem and possible solution ideas • **select** solution ideas based on personal preferences and experiences with trial and error • **identify** the steps and choices required to solve a problem • **generate** and **describe** the steps to complete an everyday task, for example, brushing your teeth, tying shoelaces or making a sandwich • **act out** the steps to solve a problem and	**Follow** and **describe** algorithms involving branching (decisions) and iteration (repetition). **Students:** • **consider** procedural texts and identify the step-by-step process used to achieve a task and why order is important • **identify** assumed information in a step-by-step process, for example, a recipe that instructs a baker to whisk eggs without telling them to crack the eggs into a bowl first • **describe** a step-by-step process (algorithm) that requires a decision (branching), for example, arranging images and arrows to demonstrate a choice; if it's cold, wear a jumper. If not, wear a t-shirt.	**Break** identified problems **down** into smaller, more manageable steps. **Create** algorithms (step-by-step processes) using text, symbols or pictures to show a sequence of steps which includes branching (decisions) and iteration (repeated instructions or loops). **Students:** • **investigate** existing step-by-step processes and **identify** common features such as specificity, repetitions and decision points. For example, examining the rules for outdoor games, instructions to construct an object out of plastic bricks or methods to create woven or knitted fabrics.	**Follow** and **describe** algorithms involving sequencing, comparison operators (branching) and iteration. **Generate**, **communicate** and **compare** design. **Students:** • **design** a solution to a common problem, for example creating a step-by-step process (algorithm) to achieve a task • **record** algorithms in pseudocode (writing instructions in plain English) and flowcharts, or pictorially showing a logical sequence of ordered steps • **use comparison operators** in an algorithm to show a decision based on comparing 2 values, for example, 'if score is	**Design** and **develop** algorithms which solve problems logically and include loops (iterations) and branching using conditional statements. **Identify** the useful features of a user interface and how it solves problems for users, and **suggest** improvements or modifications. **Reproduce**, **modify** or **design** a user interface that meets design criteria identified in user stories. **Students:** • **explore** and play with different user interfaces and evaluate them based on their own and others' preferences, for example, giving reasons for preferred game controller designs	**Design** algorithms involving multiple alternatives (branching) and iteration. **Design** a user interface for a digital system. **Generate**, **modify**, **communicate**, and **evaluate** designs. **Students:** • **design** a digital solution against design criteria identified in user stories, for example, designing a nightlight for a child who is afraid of the dark so that the light comes on or off in certain light levels • **modify** algorithms to include more options, for example, design a moisture sensor that sounds an alarm when soil is too dry • **design** algorithms using pseudocode or diagrams which show

Sub-strand	Reception	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Students use the **computational thinking** strategy of **algorithmic design** to define the solution as a sequence of steps. **Algorithms** are very specific, logical sequences of instructions. Algorithms can contain **branching,** which is a decision between 2 or more options. Algorithms can also include **loops or iterations**, which are repeated instructions.		see if there are any steps missing • **identify** repetitions in instructions, for example, repeated dance movements in a performance.	• **identify** and **follow** repeated steps (iteration) in an algorithm, for example, when performing a repeated phrase in music or completing a pattern in mathematics using manipulatives • **design** a step-by-step process to solve a problem, for example, photographing and displaying the steps to build a model out of plastic bricks.	• **identify** and **represent** decision points in an algorithm as conditional statements (if statements). For example, in games like Red Rover the rule is 'if tagged, a player is "out", if not, player keeps running until they reach the other side'. • **use** diagrams, flowcharts, symbols, arrows and other graphic representations to show repetition and choices between 2 or more options in an algorithm • **design** a solution to a simple problem by recording a step-by-step process, for example, creating instructions for drawing shapes or designing a poster for keyboard shortcuts to assist other learners.	>100 end game, else continue playing' • **demonstrate** iteration and branching in algorithms, for example, instructing a character to move through a maze: 'if there is a path to the left (branching) repeat moving forward 10 steps (iteration) until boundary is reached (branching)'.	• **construct** or **modify** algorithms that solve the same problem in different ways, for example, experiment with moving a character in a game using arrow keys or text commands • **construct** paper-based or digital prototypes of digital systems to evaluate their effectiveness against design criteria. For example, designing an application using paper-based icons, buttons, and user input frames. • **seek** or **suggest** modifications to algorithms to ensure the design is efficient and logical, for example, record instructions in pseudocode and ask a user to act out the steps and provide feedback about what is missing or needs changing.	branching and iterations, for example, constructing a flowchart that shows how an animal tracker will send information to scientists • **modify** existing algorithms to include more user-friendly features, for example, adding alternative buttons to achieve the same outcomes, such as adding a button to a step counter to allow the user to reset it to zero • **provide** and **receive** feedback on algorithmic design to ensure branching and iterations are correctly represented.
Producing and implementing **What are the most efficient steps to creating an effective solution?**		**Describe** the steps (algorithm) required to complete a simple task or solve a simple problem using symbols, pictures, diagrams or movements.	**Use** digital systems to demonstrate a sequence of steps to solve a simple problem or task. **Students:** • **explore** and **experiment** with a variety of digital	**Create** algorithms to solve simple problems and transfer them to a visual program to create a digital solution. **Students:** • **examine** an existing visual program to	**Implement** simple algorithms as visual programs involving control structures and input. **Students:** • **investigate** existing programs to **identify**	**Identify** how a visual program can be made more efficient by grouping instructions or iterations (repetition). **Demonstrate** branching using conditional	**Implement** algorithms as simple visual programs involving control structures, variables, and input. **Students:** • **investigate** existing visual programs to

Sub-strand	Reception	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Students use the computational thinking strategies of algorithmic design and abstraction. Students design a step-by-step process (**algorithm**) to solve a problem. They learn how to get rid of unnecessary detail (**abstraction**) to identify a more efficient set of steps. To do this they use **control structures** that group instructions together so that they are run correctly and in order. A **visual program** is a block-based coding tool that allows students to create a digital solution without needing to know text-based programming languages. Students use **design thinking** to create and **prototype** solutions that meet the needs of a defined user. Students use **systems thinking** to examine their solution from the point of view of the user.		**Students:** • **use** images or verbal instructions to describe the steps to achieve a simple task, for example, recording directions using arrow cards or keys to show how to move a programmable robot to places on a floor map • **experiment** with order and sequence to find the most efficient way to solve a problem, for example, playing a game like 'Guess who' using strategies to eliminate options to identify the correct person in the least number of guesses.	systems to achieve a simple task • **identify** the steps to achieve a simple task such as how to move an object around a screen • **experiment** with order and sequence to find more efficient ways of achieving simple tasks • **show** a decision between 2 possible solutions, for example, using the hyperlink function in a slide deck to skip slides**.**	**identify** the blocks used to trigger a sequence of events • **select** and **use** visual programming blocks to respond to user input and trigger a sequence of events, for example, creating an animated character that jumps over obstacles or draws shapes • **select** and **use visual** programming blocks to create a program that demonstrates a decision based on a comparison, for example, making an interactive quiz that provides immediate yes/no feedback to user input • **create** and **prototype** visual programs to solve simple problems and respond to user input. For example, building a program that supports a person with a disability, like a communication board for a person with a speech impairment to order at the canteen. • **identify** variables in a visual program.	the ways visual blocks represent conditional statements, for example, blocks that use 'if'; such as 'if [up arrow] pressed move north 10 steps' • **investigate** existing programs to **identify** the ways that visual programs allow comparisons, for example, using blocks with symbols such as >, <, =, + • **investigate** existing programs to identify the ways visual programs allow user input, for example, allowing a user to personalise their experience by inputting a username using blocks such as 'ask [username] and wait, then set [name] to [answer]' • **reproduce** programs that allow user input, branching and iteration • **design, create and prototype** visual programs that allow the user to make decisions and solve simple problems, for example, creating commands in a game that allows a character to acquire superpowers	statements and comparison operators. **Students:** • **investigate** and **modify** existing visual programs • **identify** and **reproduce** the ways visual blocks use conditional statements to allow interaction between objects, for example, 'if character is touching apple make chomping sound' • **investigate** and **modify** programs to **identify** how to **create** and **use** variables, for example, **identify** an aspect of a computer game that changes (like the score), and create and name a variable block as 'score' • **investigate** and **modify** existing programs to identify and **reproduce** the ways visual blocks are used to make comparisons between variables, for example, 'if score is more than 50 but less than 100 change background to level 2' • **design, create and prototype** visual programs that group instructions within control structures so that they run in order	**identify**, **modify** or **reproduce** visual blocks that connect with other devices, for example, the blocks required to make a light sensor to turn on or off in certain circumstances • **investigate** existing visual programs to **identify**, **modify** or **reproduce** visual blocks that use variable data. For example, creating a program that recognises sensor data by naming a variable as 'steps' and then applying a control structure which contains a group of instructions that allows a device to count the number of steps taken by a user. • **investigate** and **modify** existing programs to **identify** or **reproduce** the ways visual blocks are used to respond to user input, for example, using commands that control buttons such as 'if button A pressed reset to zero' • **design, create** and **prototype** visual programs that could connect with another device to solve a problem and meets

Sub-strand	Reception	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
					• **identify** and **use** variables in a visual program to allow user input and decisions.	and efficiently, for example, create a computer game that uses multiple characters and levels.	given or co-created design criteria, for example, creating a moisture sensor for a pot plant that triggers water input when the soil is dry.
Evaluating **Have I created the best solution for the end user?** Students use the **computational thinking** strategies of **modelling and simulating** to identify improvements. They **evaluate** solutions with identified users and seek feedback to improve, further debug or redefine the problem. Students use the **design thinking** strategy of **testing** to see if their prototype works and is worth investing in to create an improved version. Students use **systems thinking** to consider the intended and unintended impacts of their solutions.		**Use** agreed criteria to **evaluate** the success of simple solutions or the outcomes of tasks. **Students:** • **propose** and **test** a range of alternatives to solve a problem, for example, combining and repeating movements in a dance sequence • **observe** and **decide** whether a solution meets a defined set of criteria, such as sorting applications from favourite to least favourite • **respond** to and **seek** feedback from peers about solutions, for example, following and providing advice to a friend giving instructions to move a programmable toy around a defined space.	**Discuss** how existing digital solutions satisfy identified needs for known users. **Students:** • **describe** how friends and family use common digital systems to solve everyday problems, for example, using a smartphone to video call family and friends who may live long distances away • **discuss** how digital systems are used to store and access information that can help us learn, for example, using translation tools to learn a language other than English • **test** or **enact** step-by-step processes to find the most efficient way of solving a problem.	**Evaluate** how people use digital systems to meet common personal, school or community needs. **Students:** • **experiment** with more than one solution to the same problem and **evaluate** which one is the most efficient, for example, choosing the best software to support someone to achieve a particular task • **test** solutions against design criteria for an intended user, for example, trialling a digital communication board for a person with a speech impairment • **fix** or debug an algorithm after trialling.	**Discuss** how existing and student solutions satisfy the design criteria and user stories. **Students:** • **seek** and **respond** to feedback on a visual program they have created or co-created against given design criteria • **evaluate** and **redesign** existing digital solutions, for example, allowing users to customise an avatar to make it more appealing • **identify** inefficiencies in visual programs they have created and **improve** them with modifications.	**Explain** how testing a solution as well as user feedback contribute to the development effective digital solutions. **Students:** • **evaluate** and **modify** existing digital solutions to adapt them to user requirements • **evaluate** and **modify** their visual programs against agreed design criteria • **test** and **incorporate** user feedback into designed digital solutions • **debug** and **modify** algorithms to fix errors and make them run more efficiently.	**Evaluate** existing and student solutions against design criteria, user stories and their broader community. **Students:** • **explain** how their own and existing digital solutions meet user needs • **identify** design criteria from user stories and **evaluate** and **modify** visual programs to improve on their digital solutions • **test** existing digital solutions with intended users and seek and respond to user feedback • **consider** the functionality of existing and student-created digital solutions and how they can be improved and updated according to new technologies, design criteria or new user information.

Sub-strand	Reception	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Collaborating and managing **How do I work with others to create a solution?** Collaborating in **connected environments** include online spaces (internet) and shared digital spaces that are protected within internal digital infrastructure (intranets, shared folders). It can also mean working effectively with others in face-to-face environments. Students use **systems thinking** to develop their skills in **project management**. This means they understand all of the factors that influence a solution including how to manage people, resources and time to solve a problem and deliver an outcome. Students use the **design thinking** strategy of **empathy** to work safely with others using agreed protocols to share, plan and complete solution processes in collaborative teams.		**Work** with others to **create** and **organise** ideas and information **using** digital systems safely. **Students:** • **identify** ways to safely share information with known people in connected environments • **co-develop** or **follow** agreed protocols for using digital systems and communicating in shared digital environments • **use** familiar digital systems to collaboratively **create** and **share** content • **locate**, **generate** and **access** digital data and information.	**Use** the basic features of common digital tools to **create**, **locate** and **communicate** content. **Use** the basic features of common digital tools to **share** content and **collaborate, demonstrating** agreed behaviours, guided by trusted adults. **Students:** • independently or with others, **create** and **share** digital artefacts in safe connected environments with known people • **identify** and **apply** agreed protocols for participating independently and with others in connected environments, for example, knowing when they are participating online or offline • effectively **locate** and **select** information from a variety of sources to create digital artefacts • **support**, **share** and **cooperate** with others to achieve shared tasks using digital systems.	**Use** the core features of common digital systems to safely **create** and **share** content, both independently and with others, using agreed protocols. **Establish** and **use** practices that **demonstrate** effective file management and cyber safety when **participating** independently and with others in connected environments. **Students:** • **co-develop** or **follow** agreed practices to effectively **manage** digital content • **identify** and **use** strategies to keep themselves and others safe in connected environments • **identify** and **consult** networks of trusted people to support ways to develop and follow agreed conventions • **identify** and **consult** networks of trusted people to manage and report examples of inappropriate or harmful behaviour in connected environments.	**Use** the core features of common digital tools to **create, locate** and **communicate** content, following agreed conventions. **Use** the core features of common digital tools to **share** content, **plan** tasks and **collaborate,** following agreed behaviours supported by trusted adults. **Students:** • **work** safely with others to **develop** plans to **manage** and **complete** collaborative digital projects in online or face-to-face environments • **understand** how to efficiently **use** common digital systems to **create** shared digital content • **understand** ways to **locate**, **create** and **share** information in connected environments respectfully • **identify** and use **agreed** protocols to manage and retrieve digital content • **understand** how to **respond** to and **seek** support from trusted adults when managing	**Identify** the ways to develop positive digital identities and the ways to protect themselves and others online. **Plan**, **create** and **communicate** ideas and information independently and with others. **Apply** agreed ethical and social protocols. **Students:** • **identify** and **demonstrate** the characteristics of a responsible digital citizen • **plan** and **implement** digital projects safely in collaboration with others **using** agreed roles and responsibilities • **develop** logical content management strategies to work effectively both independently and with others in connected environments • **respect** intellectual property and ownership of content created in connected environments • **use** agreed protocols to keep themselves and others safe when	**Select** and **use** appropriate digital tools effectively to **create**, **locate** and **communicate** content, **applying** common conventions. **Select** and **use** appropriate digital tools effectively to **share** content online, **plan** tasks and **collaborate** on projects, **demonstrating** agreed behaviours. **Students:** • **understand** the responsibilities and **demonstrate** the characteristics of an ethical global digital citizen • **plan** and successfully **complete** digital projects using co-developed agreed protocols, roles and responsibilities • **develop** strategic, logical content creation and management strategies to work effectively both independently and with others in connected environments • **respect** the rights and **enact** the responsibilities of ethical content creators/users in

Sub-strand	Reception	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
					content and behaviour in connected environments.	working in connected environments.	connected environments • **understand** and apply agreed protocols to keep themselves and others safe when working in connected environments • **understand** the impact of existing and emerging digital solutions in local and global contexts.
Privacy, security and data management **How do I keep myself and other people safe in connected environments?** **How do I ensure my data and other people's data is secure?** Students use **systems thinking** by recognising that data is all around us and there are many ways to organise, use, store and secure data in connected environments. Students also use **systems thinking** to explore biases in data, ownership and attribution. Students use the **computational thinking** strategy of **evaluation** to	**Identify** some data that is personal and owned by them. **Students:** • **explore** what data is and where it is located • **identify** where their personal data may be visible in connected environments • **identify** the difference between public and private data.	**Identify** trusted networks with which to safely share data and information. **Explore** ways to organise and safely store personal data. **Students:** • **identify** trusted adults who can provide guidance about data sharing • **explore** securing digital identities, for example, through usernames, passwords, avatars, and consent to share • **consider** apps and websites where their data may be shared • **explore** how people collect, organise and share data.	**Access** their school account with a recorded username and password. **Discuss** that some websites and apps store their personal data online. **Students:** • **explore** and explain ways to build secure digital identities, for example, using safe usernames, strong passwords, avatars and other non-identifying methods • **explore** and use efficient ways to organise and store data on secure systems • **identify** the risks associated with sharing data in connected environments • **identify** how to protect their data and other	**Identify** information that is safe to share with others and information that should only be shared with trusted networks. **Explore** and **use** efficient ways of organising, storing and securing personal data in common digital systems. **Students:** • **identify** the characteristics of an ethical and global digital citizen • **develop** and **use** ways to build secure digital identities, for example, using appropriate usernames, strong passwords, avatars, sharing only with known users, and other cyber safety strategies	**Access** their school account using a memorised password and **explain** why it should be easy to remember but hard for others to guess. **Identify** what personal data is stored and shared in their online accounts and **discuss** any associated risks. **Students:** • **analyse**, **develop** and **create** strong digital habits using appropriate usernames, strong passwords, agreed protocols, reporting procedures and other cyber safe strategies • **investigate**, **collect** and **organise** data securely to identify and solve problems	**Develop** strong privacy and security habits while accessing multiple personal accounts. **Identify** their digital footprint and recognise when data is permanently available in connected environments. **Students:** • **develop** and **demonstrate** strong digital habits to keep themselves and others safe and secure online • **investigate** and **explore** the reliability of data from a variety of sources • **identify** intellectual property rights and the need to act in accordance with copyright attribution conventions	**Access** multiple personal accounts using unique passphrases and **explain** the risk of password reuse. **Explain** the creation and permanence of their digital footprint and **consider** privacy when collecting user data. **Students:** • **develop**, **demonstrate** and **promote** strong digital habits to keep themselves and others safe and secure online • **analyse** the reliability of data sources when conducting investigations • **recognise** and **respect** intellectual property rights and the need to act in accordance with copyright and

Sub-strand	Reception	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
make informed decisions and develop strategies to ensure they are aware of how their data is being used. Students understand their data and other people's data need to be respected and secured. Students also use **pattern recognition** to identify trends and problems highlighted in data collections.			people's data in connected environments.	• **access** and ethically **use** data to investigate and solve problems • **identify** who owns the data they are using and whether they have permission to use it • **understand** that consent is required to share their data and other people's data • **generate** and **store** data in organised and secure ways	• **understand** data ownership and consent to use • **participate** in online environments using protocols in line with ethical and global digital citizenship.	• **respect** the privacy and security of their data and other people's data, and demonstrate safe data collection strategies • **explore** the ways diverse groups are represented in data collections and disrupt common stereotypes with inclusive data sets • **develop** the attributes of an ethical and global digital citizen when participating in connected environments.	attribution conventions • **develop** strong practices that ensure the privacy and security of their data and other people's data • **interrogate** biases in data and ensure data collections are representative. • **identify**, **demonstrate** and **promote** the conventions of ethical and global digital citizenship.