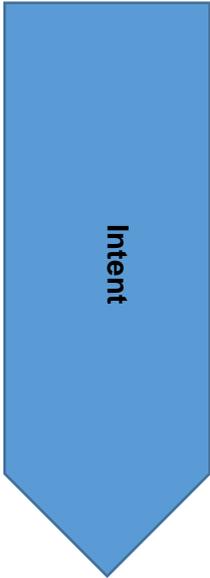




St Thomas' Catholic Primary School

Computing Intent and Progression Map



Information Communication Technology is rapidly developing and required in the modern world, and it is important to St Thomas' that our children go forward with an understanding of how to safely consume and produce ICT in everyday life and beyond. Throughout the years, children will learn and progress in their skills of safe information retrieval and storage, producing and presenting work using various programs (such as those used for word processing, coding, movie making etc.), self-evaluating what they have produced, and exchanging information in a safe manner.

ICT is a creative subject, with children able to produce incredible pieces of work with pride. At St Thomas', we promote and encourage this by allowing children to experiment and problem solve in a safe environment. ICT is often used across the curriculum, and through engaging teaching and curiosity, children gain confidence and enthusiasm in their learning.

Progression in Computing	EYFS	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Digital Literacy (e safety)	<ul style="list-style-type: none">• 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.						

<p>Computer Science</p> <p>Hardware</p>	<ul style="list-style-type: none"> Learn how to use the basic parts of hardware to find out how it works - i.e. power button for computer and monitor, the home key on an iPad. 	<ul style="list-style-type: none"> Learn how to explore & tinker with hardware to find out how it works. Understand that computers & devices around us use inputs & outputs, identifying some of these. Learn where keys are located on the keyboard. Learn how to operate a camera. 	<ul style="list-style-type: none"> Understand what a computer is & that it is made up of different components. Recognise that buttons cause effects and that technology follows instructions. Learn how we know that technology is doing what we want it to via its output. Use greater control when taking photos with tablets or computers. Develop confidence with the keyboard and the basics of touch & typing. 	<ul style="list-style-type: none"> Understand what the different components of a computer do & how they work together. Draw comparisons across different types of computers. Learn what a server does. 	<ul style="list-style-type: none"> Learn about the purpose of routers. 	<ul style="list-style-type: none"> Learn the difference between ROM and RAM. Recognise how the size of RAM affects the processing of data. Understand the fetch, decode, execute cycle. 	<ul style="list-style-type: none"> Learn about the history of computers & how they have evolved over time. Use the understanding of historic computers to design a computer of the future. Learn how barcodes, QR codes and RFID work. Learn about some of the methods which cause data corruption. Learn that external devices can be programmed by a separate computer.
<p>CS</p> <p>Networks & Data Representation</p>				<ul style="list-style-type: none"> Learn what a network is & its purpose. Identify the key components within a network, including whether they are wired or wireless. Recognise links between networks and the internet. Learn how data is transferred. 	<ul style="list-style-type: none"> Consolidate understanding of the key components of a network. Understand that websites & videos are files that are shared from one computer to another. Learn about the role of packets. Understand that computer networks provide multiple services, such as the WWW, & opportunities for communication & collaboration 	<ul style="list-style-type: none"> Learn the vocabulary associated with data: data & transmit. Learn how the data for digital images can be compressed. Recognise that computers transfer data in binary & understand simple binary addition. Relate binary signals (Boolean) to the simple character-based language, ASCII. Learn that messages can be sent by binary code, reading binary up to 8 characters & carrying out binary calculations. Understand how bit patterns represent images as pixels. 	<ul style="list-style-type: none"> Understand that computer networks provide multiple services.

<p>CS Computational thinking</p>		<ul style="list-style-type: none"> • Learn that decomposition means breaking a problem down into smaller parts. • Use decomposition to solve unplugged challenges. • Use logical reasoning to predict the behavior of simple programs. • Develop the skills associated with sequencing in unplugged activities. • Learn that an algorithm is a set of step by step instructions used to carry out a task, in a specific order. • Follow a basic set of instructions • Assemble instructions into a simple algorithm. 	<ul style="list-style-type: none"> • Articulate what decomposition is. • Decompose a game to predict the algorithms used to create it. • Use decomposition to decompose a story into smaller parts. • Learn what abstraction is. • Learn that there are different levels of abstraction. • Explain what an algorithm is. • Follow an algorithm. • Create a clear and precise algorithm. • Learn that computers use algorithms to make predictions. • Learn that programs execute by following precise instructions. • Incorporate loops with algorithms. 	<ul style="list-style-type: none"> • Use decomposition to explain the parts of a laptop computer. • Use decomposition to explore the code behind an animation. • Use repetition in programs. • Understand that computer follow instructions. • Use an algorithm to explain the roles of different parts of a computer. • Use logical reasoning to explain how simple algorithms work. • Explain the purpose of an algorithm. • Form algorithms independently. 	<ul style="list-style-type: none"> • Solve unplugged problems by decomposing them into smaller parts. • Use decomposition to understand the purpose of a script of code. • Use decomposition to help solve problems. • Identify patterns through unplugged activities. • Use past experiences to help solve new problems. • Use abstraction to identify the important parts when completing both plugged and unplugged activities. • Create algorithms for a specific purpose. 	<ul style="list-style-type: none"> • Decompose animations into a series of images. • Decompose a program without support. • Decompose a story to be able to plan a program to tell a story. • Predict how software will work based on previous experiences. • Write more complex algorithms for a purpose. 	<ul style="list-style-type: none"> • Decompose a program into an algorithm. • Use past experiences to help solve new problems. • Write increasingly complex algorithms for a purpose.
--	--	---	---	---	---	---	--

<p>CS Programming</p>	<ul style="list-style-type: none"> • Program a BeeBot to follow a planned route. 	<ul style="list-style-type: none"> • Program a BeeBot to follow a planned route. • Learn to debug instructions when things go wrong. • Develop a 'how to' video to explain how the BeeBot works. • Learn to debug an algorithm in an unplugged scenario. 	<ul style="list-style-type: none"> • Use logical thinking to explore software predicting, testing & explaining what it does. • Use an algorithm to write a basic computer program. • Learn what loops are. • Incorporate loops to make code more efficient. 	<ul style="list-style-type: none"> • Use logical thinking to explore more complex software; predicting & testing & explaining what it does. • Incorporate loops to make code more efficient. • Remix existing code. • Use a more systematic approach to debugging code, justifying what is wrong & how it can be corrected. 	<ul style="list-style-type: none"> • Understand that website can be altered by exploring the code beneath the site. • Code a simple game. • Use abstraction & pattern recognition to modify code. 	<ul style="list-style-type: none"> • Program an animation, altering & developing their programming as they work. • Begin to use nested loops (loops within loops) • Debug their own code. • Write code to create a desired effect. • Use a range of programming commands. • Use repetition within a program. • Amend code with a live scenario. • Program using the language Python. 	<ul style="list-style-type: none"> • Debug quickly & effectively to make a program more efficient. • Remix existing code to explore a problem. • Use & adapt nested loops. • Program using the language HTML. • Change a program to personalise it. • Evaluate code to understand its purpose. • Predict code & adapt it to a chosen purpose. • Alter a website's code to create changes.
------------------------------	---	--	---	---	--	--	---

<p>Information Technology</p> <p>Using software</p>	<ul style="list-style-type: none"> • Take and edit photographs using an ipad. 	<ul style="list-style-type: none"> • Use a basic range of tools with graphic editing software. • Take and edit photographs. • Understand how to create digital art using an online paint tool. • Develop control of the mouse through dragging, clicking & resizing of images to create different effects. • Develop understanding of different software tools. • Develop word processing skills - open, save and write short sentence in Word 	<ul style="list-style-type: none"> • Develop word processing skills, including altering text, copying and pasting and using keyboard shortcuts. • Use word processing software to type and reformat text. • Use software to create story animations. • Create and label images. 	<ul style="list-style-type: none"> • Take photographs & record video to tell a story. • Use software to edit and enhance video adding music, sounds & text on screen with transitions. 	<ul style="list-style-type: none"> • Build a web page and create content for it. • Design and create a webpage for a given purpose. • Use Google online software for documents, presentations, forms & spreadsheets. • Work collaboratively with others. <p>Does this include a blog?</p>	<ul style="list-style-type: none"> • Use logical thinking to explore software more independently, make predictions based on their previous experience. • Use software programme Audacity to create music. • Plan, record and edit a radio play. • Create & edit sound recordings for a specific purpose. • Create & edit videos, adding multiple elements; music, voiceover, sound, text & transitions to create a video advert. • Independently learn how to use 3D design software package TinkerCAD. 	<ul style="list-style-type: none"> • Use logical thinking to explore software independently, altering ideas & testing continuously. • Use search & word processing skills to create a presentation. • Use design software TinkerCAD to design a product. • Create a website with embedded links & multiple pages. • Use animation software: Stop Motion to create video animation. • Identify ways to improve & edit final products.
<p>IT</p> <p>Using e mail & internet</p>		<ul style="list-style-type: none"> • Search & download images from the internet safely. 	<ul style="list-style-type: none"> • Using the internet safely - using a search engine on an educational website (e.g. espresso) 	<ul style="list-style-type: none"> • Learn to log in and out of an email account. • Write an email including subject 'to' & 'from'. • Send an email with an attachment. • Reply to an email. 	<ul style="list-style-type: none"> • Search the internet safely using appropriate search engines. 	<ul style="list-style-type: none"> • Develop searching skills to help find relevant information on the internet. • Learn how use search engines effectively to find information, focusing on keyword searches & evaluating search returns. 	<ul style="list-style-type: none"> • Understand how search engines work.

IT Using Data		<ul style="list-style-type: none"> • Introduction to spreadsheets. • Represent data in tables, charts & pictograms. • Sort data & create branching databases. • Identify where digital content can have advantages over paper when storing & manipulating data. 	<ul style="list-style-type: none"> • Collect & input data into a spreadsheet. • Interpret data. 	<ul style="list-style-type: none"> • Understand the vocabulary associated with database: field, record, data. • Learn about the pros and cons of digital versus paper databases. • Sort & filter databases to easily retrieve information. • Create a interpret charts & graphs to understand data. 	<ul style="list-style-type: none"> • Design a weather station which gathers and records sensor data. <p style="color: orange;">Better for Year 5??</p>	<ul style="list-style-type: none"> • Understand how data is collected. 	<ul style="list-style-type: none"> • Understand how barcodes, QR codes and RFID work. • Gather and analyse data in real time. • Create formulas and sort data within spreadsheets.
IT Wider use of technology		<ul style="list-style-type: none"> • Recognise common uses of information technology including beyond school. 	<ul style="list-style-type: none"> • Learn how computers are used in the wider world. 	<ul style="list-style-type: none"> • Understand the purpose of emails. 	<ul style="list-style-type: none"> • Understand that software can be used collaboratively online to work as a team. 	<ul style="list-style-type: none"> • Learn what a search engine is. 	<ul style="list-style-type: none"> • Learn about the Internet of Things and how it has led to 'big data'. • Learn how 'big data' can be used to solve a problem or improve efficiency.
Digital Literacy	<ul style="list-style-type: none"> • Learning what to do if they come across something that worries them or makes them feel uncomfortable. • Understand the importance of a password (using simple password such as 'cat' or '1234') 	<ul style="list-style-type: none"> • Log in and out and save work on own account. • Understand the importance of a password. • When using the internet to search for images, learning what to do if they come across something that worries them or makes them feel uncomfortable. 	<ul style="list-style-type: none"> • Understand how to stay safe when talking to people online. • Not sharing personal information & what to do if they see or hear something online that makes them feel upset or uncomfortable. 	<ul style="list-style-type: none"> • Learn to be a responsible digital citizen; understanding their responsibilities to treat others respectfully & recognizing when digital behavior is unkind. • Learn about cyberbullying. • Learn that not all emails are genuine, recognizing when an email might be fake & what to do about it. 	<ul style="list-style-type: none"> • Recognise what appropriate behavior is when collaborating with others online. • Recognise that information on the internet might not be true or correct & that some sources are more trustworthy than others. 	<ul style="list-style-type: none"> • Identify possible dangers online and learn how to stay safe. • Create a video about digital safety. • Recognise that information on the internet might not be true or correct and learn ways of checking validity. • Learn to use an online community safely. 	<ul style="list-style-type: none"> • Understand the importance of secure passwords & how to create them. • Use search engine safely and effectively. • Recognise that updated software can help to prevent data corruption & hacking.