YEAR 4

	EYFS	YEAR 1	YEAR 2	YEAR 3	YEAR 4	YEAR 5	YEAR 6
Programming – Repetition in games	**The three Prime ELGS of Communication and Language, PSED and Physical Development provide the foundations of which all other learning is built upon.** **No Specific ELG links.**	To be introduced to on-screen programming. To explore sprites and backgrounds, using programming blocks to use, modify and create programs. To be introduced to program design through algorithms.	To understand that sequences of commands have an outcome. To use and modify designs to create and evaluate a project.	To explore the links between events and actions. To move sprites in four directions. To introduce programming extensions, changing the size and colour. To design and code their own program.	To explore the concept of repetition in programming. To look at similarities and differences between count controlled and infinite loops. To modify, design and create a game using repetition, applying stages of programming design throughout.	To develop knowledge of selection by revisiting how conditions can be used in programming. To create algorithms and programs using this understanding. To design, write and evaluate a quiz program.	To use the four programming constructs (sequence, repetition, selection and variables) whilst using a physical device. To design, build and test a simple program and transfer to device.

COMPOSITES

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

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

COMPONENTS

	1	2	3	4	5	6	End Point
	How can I develop the use of count-controlled loops in a different programming environment?	Can I explain that in programming there are infinite loops and count-controlled loops?	How can I develop a design that includes two or more loops which run at the same time?	Can I modify an infinite loop in a given program?	Can I design a project that includes repetition?	Can I create a project that includes repetition?	Prior experience of programming. The KS1 NCCE units cover floor robots and ScratchJr, and Scratch is introduced in the Year 3 programming units. However, experience of other languages or environments may also be useful.

CONCEPTS **Link to concept map**	Programming – Repetition in games Information Technology Computer Science	Programming – Repetition in games Information Technology Computer Science	Programming – Repetition in games Information Technology Computer Science	Programming – Repetition in games Information Technology Computer Science	Programming – Repetition in games Information Technology Computer Science	Programming – Repetition in games Information Technology Computer Science	Programming – Repetition in games Information Technology Computer Science
SKILLS	I can list an everyday task as a set of instructions including repetition I can predict the outcome of a snippet of code I can modify a snippet of code to create a given outcome	I can modify loops to produce a given outcome I can choose when to use a count-controlled and an infinite loop	I can choose which action will be repeated for each object I can evaluate the effectiveness of the repeated sequences used in my program	I can identify which parts of a loop can be changed I can re-use existing code snippets on new sprites	I can evaluate the use of repetition in a project I can select key parts of a given project to use in my own design I can develop my own design explaining what my project will do	I can refine the algorithm in my design I can build a program that follows my design I can evaluate the steps I followed when building my project	Children will explore the concept of repetition in programming using the Scratch environment. The unit begins with a Scratch activity similar to that carried out in Logo in Programming unit A, where children can discover similarities between two environments. Children look at the difference between count-controlled and infinite loops. They will modify existing animations and games using repetition. Their final project is to design and create a game which uses repetition, applying stages of programming design throughout.
KNOWLEDGE	I know what a snippet is.	I can recognise that some programming languages enable more than one process to be run at once	I can explain what the outcome of the repeated action should be	I can explain the effect of my changes	I can use my knowledge from prior lessons to create a plan.	I know how to build a program.	Children will explore the concept of repetition in programming using the Scratch environment. Children will use their knowledge of programming to develop a game.

LESSON LINK	Programming B – Repetition in games	Programming B – Repetition in games	Programming B – Repetition in games	Programming B – Repetition in games	Programming B – Repetition in games	Programming B – Repetition in games	Programming B – Repetition in games
PROGRESSIVE VOCABULARY	Scratch, programming, sprite, blocks, code, loop, repeat, value	block, repeat, forever, infinite loop, count-controlled loop, costume	repetition, forever, infinite loop, count-controlled loop, animate, costume, event block, duplicate	block, repeat, forever, infinite loop, modify, design	infinite loop, count-controlled loop, repetition, design, sprite, algorithm	repetition, design, algorithm, duplicate, debug, refine, evaluate	
CURRICULUM EXPERIENCES	Use real life experiences.				Design and create a game which uses repetition, applying stages of programming design throughout.		
END POINT	This lesson starts by looking at repetition within real-life examples, where children identify which parts of a set of instructions are repeated. Children then use Scratch, a block-based programming environment, to create shapes using count-controlled loops. They consider what the different values in each loop signify, then use existing code	In this lesson, children look at different types of loops: infinite loops and count-controlled loops. They practise using these within Scratch and think about which might be more suitable for different purposes	In this lesson, children create designs for an animation of the letters in their names. The animation uses repetition to change the costume (appearance) of the sprite. The letter sprites will all animate together when the **event** block (**green flag**) is clicked. When they have designed their animations, the	In this lesson, children look at an existing game and match parts of the game with the design. They make changes to a sprite in the existing game to match the design. They then look at a completed design, and implement the remaining changes in the Scratch game. They add a sprite, re-use	In this lesson, children look at a model project that uses repetition. They then design their own games based on the model project, producing designs and algorithms for sprites in the game. They share these designs with a partner and have time to make any changes to	In this lesson, children build their games, using the designs they created in Lesson 5. They follow their algorithms, fix mistakes, and refine designs in their work as they build. They evaluate their work once it is completed, and showcase their games at the end.	

	to modify and create new code, and work on reading code and predicting what the output will be once the code is run.		children will program them in Scratch. After programming, children then evaluate their work, considering how effectively they used repetition in their code.	and modify code blocks within loops, and explain the changes made.	their design as required.		