

YEAR 6

	EFYS	YEAR 1	YEAR 2	YEAR 3	YEAR 4	YEAR 5	YEAR 6
Programming A – Variables in Games	<p>The three Prime ELGS of Communication and Language, PSED and Physical Development provide the foundations of which all other learning is built upon.</p> <p>No Specific ELG links.</p>	To introduce early programming concepts, exploring commands and algorithms.	To develop understanding of instructions in sequences, using commands in different orders to investigate how this affects the outcome. To design, test and debug algorithms.	To explore the concept of sequencing, using motion, sound and event blocks to create their own programs with sequences. To begin to apply stages of program design.	To create programs by planning, modifying and testing commands. To understand repetition and loops within programming.	To explore the concept of selection in programming. To apply knowledge of repetition and conditions to the concept of selection and write algorithms and programs using this concept. To apply stages of programming design.	To explore the concept of variables in programming. To design, modify and improve their own project.

Programming A – Variables in Games

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
	What is a 'variable'?	Why is a variable is used in a program?	How can I improve a game by using variables?	Can I design a project that builds on a given example?	Can I use my design to create a project?	Can I evaluate my project?	This unit explores the concept of variables in programming through games in Scratch. First, Pupils find out what variables are and relate them to real-world examples of values that can be set and changed. Then they use variables to create a simulation of a scoreboard. In Lessons 2, 3, and 5, which follow the Use-Modify-Create model, Pupils experiment with variables in an existing project, then modify them, before they create their own project. In Lesson 4, Pupils focus on design. Finally, in Lesson 6, Pupils apply

							their knowledge of variables and design to improve their games in Scratch.
CONCEPTS Link to concept map	Programming A Selection in Physical Computing Information Technology Computer Science	Programming A Selection in Physical Computing Information Technology Computer Science	Programming A Selection in Physical Computing Information Technology Computer Science	Programming A Selection in Physical Computing Information Technology Computer Science	Programming A Selection in Physical Computing Information Technology Computer Science	Programming A Selection in Physical Computing Information Technology Computer Science	
SKILLS	Design and make their own project that includes variables.	Identify a program variable as a placeholder in memory for a single value Recognise that the value of a variable can be changed	Make use of an event in a program to set a variable Recognise that the value of a variable can be used by a program	Choose the artwork for my project Create algorithms for my project	Create the artwork for my project Choose a name that identifies the role of a variable Test the code that I have written	Use variables to extend my game Share my game with others	Pupils apply their knowledge of variables and design to improve their games in scratch.
KNOWLEDGE	Identify examples of information that is variable Explain that the way a variable changes can be defined Identify that variables can hold numbers or letters	Explain that a variable has a name and a value	Decide where in a program to change a variable	Explain my design choices	Use knowledge to test the written code	Identify ways that my game could be improved	Pupils find out what variables are and relate them to real-world examples of values that can be set and changed.
LESSON LINK	5 - Programming A – Variables in games	5 - Programming A – Variables in games	5 - Programming A – Variables in games	5 - Programming A – Variables in games	5 - Programming A – Variables in games	5 - Programming A – Variables in games	
PROGRESSIVE VOCABULARY	variable, change, name, value	variable, name, value, set, change	variable, set, change, design, event	design, algorithm, code	task, algorithm, design, artwork, program,	improve, evaluate, share	Children will understand, articulate and use the vocabulary

					project, code, test, debug		
CURRICULUM EXPERIENCES					Design own Scratch project		
END POINT	<p>Pupils are introduced to variables. They see examples of real-world variables (score and time in a football match) before they explore them in a Scratch project. Pupils then design and make their own project that includes variables. Finally, Pupils identify that variables are named and that they can be letters (strings) as well as numbers.</p>	<p>Pupils understand that variables are used in programs, and that they can only hold a single value at a time. They complete an unplugged task that demonstrates the process of changing variables. Then, Pupils explore why it is important to name variables and apply their learning in a Scratch project in which they make, name, and update variables.</p>	<p>Pupils apply the concept of variables to enhance an existing game in Scratch. They predict the outcome of changing the same change score block in different parts of a program, then they test their predictions in Scratch. Pupils also experiment with using different values in variables, and with using a variable elsewhere in a program. Finally, they add comments to their project to explain how they have met the objectives of the lesson.</p>	<p>In this lesson, Pupils work at the 'design' level of abstraction, where they create their artwork and algorithms. Pupils first design the sprites and backgrounds for their project, then they design their algorithms to create their program flow.</p>	<p>In this lesson, Pupils implement the algorithms that they created in Lesson 4. In doing this, they identify variables in an unfamiliar project and learn the importance of naming variables. They also have the opportunity to add another variable to enhance their project.</p>	<p>In this lesson, Pupils build on the project that they created in Lesson 5. They consider how they could improve their own projects and make small changes to achieve this. Pupils then have the opportunity to add a variable independently. Finally, Pupils evaluate each other's projects; they identify features that they liked and features that could be improved.</p>	