Introduction to Coding — Scotland

Level	Scottish Curriculum for Excellence – the technology strand	Rapid Router coding vocabulary	Progression through teaching resources
Early Level *(Key stage equivalent: KS1)*	• I can explore computational thinking processes involved in a variety of everyday tasks and can identify patterns in objects or information. TCH 0-13a • I understand that sequences of instructions are used to control computing technology. TCH 0-14a • I can develop a sequence of instructions and run them using programmable devices or equivalent. TCH 0-15a	• Algorithm • Program • Debug **Sequence instructions:** • Move forwards • Turn left • Turn right	**Rapid Router levels 1 to16:** • Creating algorithms for physical movement • Controlling van on screen app using movement commands • Working out the shortest route to a destination
First Level *(Key stage equivalent: KS2)*	• I can explore and comment on processes in the world around me making use of core computational thinking concepts TCH 1-13a • I understand the instructions of a visual programming language and can predict the outcome of a program written using the language. TCH 1-14a • I can demonstrate a range of basic problem-solving skills by building simple programs to carry out a given task, using an appropriate language. TCH 1-15a	• Algorithm • Program • Debug **Sequence instructions:** • Move forwards • Turn left • Turn right **Repetition:** • Repeat x times	**Rapid Router levels 17 to 28:** • Understanding the repeat function • Creating and evaluating their own challenges and programs using the code skills learnt
Second Level *(Key stage equivalent: KS3)*	• I can explain core programming language concepts in appropriate technical language. TCH 2-14a • I can create, develop and evaluate computing solutions in response to a design challenge. TCH 2-15a	• Algorithm • Program • Debug **Sequence instructions:** • Move forwards • Turn left • Turn right • Wait **Repetition:** • Repeat x times • Repeat until **Selection:** • If... do... • If... else if...	**Rapid Router levels 19 to 28 (recap):** • Understanding the repeat function (recap) • Creating and evaluating their own challenges and programs using the code skills learnt **Rapid Router levels 29 to 43:** • Use the core programming commands appropriately in a visual language • Understand the repeat while command • Decompose the programming task into smaller parts

Introduction to Coding — Scotland

Level	Scottish Curriculum for Excellence – the technology strand	Rapid Router Python coding vocabulary	Progression through teaching resources
Third/Fourth Levels *(Key stage equivalent: KS3)*	• I understand language constructs for representing structured information TCH 3-14a • I can informally compare algorithms for correctness and efficiency. TCH 4-13b	• Algorithm • Program • Debug **Sequence instructions:** • Move forwards • Turn left • Turn right • Wait **Repetition:** • Repeat x times • Repeat until **Selection:** • If... do... • If... else if... **Variables:** • Traffic lights are red/green	**Rapid Router Levels 44–67:** • Use the core programming commands appropriately in a visual language • Understand the repeat while command • Decompose the programming task into smaller parts • Identify sections of code which can be used several times and write a procedure for that section • Use repeat loops within procedure

Introduction to Coding — Scotland

Key Stage	Scottish Curriculum for Excellence – the technology strand	Rapid Router Python coding vocabulary	Progression through teaching resources
Third/Fourth Levels *(Key stage equivalent: KS3)*	• I understand language constructs for representing structured information TCH 3-14a • I can informally compare algorithms for correctness and efficiency. TCH 4-13b • I understand constructs and data structures in a textual programming language TCH 4-14a	• Algorithm • Program • Debug **Sequence instructions:** • v.move_forwards() • v.turn_left () • v.turn_right () • v.wait () **Repetition:** • for count in range (3) Selection: • If ….. (): • elif …… (): • else : **Procedures:** • Define – def procname (): • Call – procname() **Variables:** • length = 10 • length = length +5	**Rapid Router Levels 68–109:** • Develop an initial understanding of Python as a text based language • Understand that Python has precise syntax • Identify characteristics of Python and compare this with Blockly • Use and understand the movement instructions in Python code • Use and understand repeat loops in Python (for count in range (n)) • Create the core program in visual Blockly and understand it in Python code • Understand how the syntax of selection statements works in Python • Understand Python while, if , elif , else commands • Analyse how procedures work in Python (extension) • Write code in Python without the support of Blockly • Write simple programs in Python using code for simple movement e.g. v.move_forwards() • Use the print command in Python (not available in Blockly) • Debug their Python programs, demonstrating an understanding of the appropriate syntax • Use indents correctly in Python • Use the Repeat loop … for count in range (n): • Design and write programs independently in Python using repetition and selection: • for count in range (n): and while, if, elif, else