

## LESSON 3

# Conditionals and Loops in Python

40 minutes

Python from the Start

### WHAT THIS LESSON DOES

Students use PRIMM to explore conditionals and loops in Python on the micro:bit, building from simple if statements through nested structures and combined loop-and-condition patterns, then solve two short challenges.

### WHAT STUDENTS SHOULD BE ABLE TO DO

- Students use comparison operators and Boolean combinations in if, elif and else structures, control repetition with for and while loops, explain how indentation shapes program structure, and contrast sequential Python flow with parallel micro:bit handlers.

### BEFORE THE BELL

- Students logged in to the platform with a Python micro:bit editor ready (USB or simulator as your room uses).
- micro:bits charged or simulators open; one shared board view for modelling the first steps.
- Remind students how the class saves Python work so files are available next lesson.

## Conditionals and Loops in Python

### HOW THE LESSON RUNS

|       |              |                                      |
|-------|--------------|--------------------------------------|
| 4 min | introduction | Start: What We're Building           |
| 2 min | content      | Introduction                         |
| 2 min | content      | Understanding Conditional Statements |
| 2 min | content      | Simple If Statement                  |
| 2 min | content      | If-else Statement                    |
| 2 min | content      | If-elif-else Statement               |
| 3 min | content      | Nested If Statements                 |
| 2 min | content      | Introduction to Loops                |
| 2 min | content      | For Loops                            |
| 3 min | content      | While Loops                          |
| 2 min | content      | Nested Loops                         |
| 2 min | content      | Using Loops with Conditionals        |
| 2 min | content      | Exercise: Decision Maker             |
| 2 min | content      | Exercise: Countdown Timer            |
| 2 min | content      | Wrap up                              |
| 3 min | content      | Make Sense: What Worked, What Broke  |
| 2 min | content      | Reflect                              |
| 1 min | summary      | What You Covered                     |

### TEACHING MOVES

- Model indentation on the board with a clear indent mark so students see that the block under if or for is what runs.
- When you reach while True and button checks, briefly name sequential flow in the main script and contrast it with event handlers that can run at the same time on the micro:bit.
- For the exercises, keep the challenge brief: a loop plus a compound or multi-branch condition is enough.
- Park one prediction at the start (for example what a simple if does when the condition is false) and return to it in Make sense.

### WHAT TO WATCH FOR

- Missing or uneven indentation after if, elif, else, for or while: the editor shows an IndentationError; realign the block to one consistent indent.
- Forgetting to update the loop variable in a while loop, causing an infinite loop: stop the program, then add the increment or change inside the loop.
- Comparing without converting types (for example scrolling a number without str): fix with str() on the value passed to display.scroll.

#### **DIFFERENTIATION**

- Support: Provide a skeleton with the import, one variable and the while True line already in place for the first exercise.
- Extension: Ask confident students to add a compound condition with and/or (for example both buttons) or a nested check inside the countdown.
- Extension: Invite a pair to explain sequential main-loop flow versus handler-style parallel flow using their own button program as the example.