

LESSON 39

Functions in Python

40 minutes

Strand 3: Coding at the next level

Built in Python

Outcome 3.1

Outcome 3.3

WHAT THIS LESSON DOES

Students write and call procedures and functions in Python on the micro:bit, then build a simple weather station that chooses a cold, moderate or hot symbol from a temperature value.

CURRICULUM OUTCOMES

Outcome	What the specification asks for
3.1	Creatively design and write code for short programming tasks, demonstrating operators for assignment, arithmetic, comparison and Boolean combinations.
3.3	Use functions or procedures.

WHAT STUDENTS SHOULD BE ABLE TO DO

- Write and call a function (LO 3.1, LO 3.3)
- Explain how functions help you reuse and organise code (LO 3.3)

BEFORE THE BELL

- Students logged in to the platform on devices with the Python micro:bit editor ready
- micro:bit hardware connected where available, or the on-screen simulator open
- Have a short example of a named block of code ready to show on the board before students start

WHAT YOU NEED

Item	Qty	Per	If you do not have it
tablet or laptop with Python	1	pair	run the lesson on the IWB with students taking turns

Functions in Python

HOW THE LESSON RUNS

4 min	introduction	Start: what we're building
4 min	content	Why give code a name?
4 min	content	Procedures
4 min	content	Functions
4 min	content	Functions with Parameters
3 min	content	Project - Micro:bit Weather Station
3 min	content	Temperature Display Procedures
3 min	content	Weather Function
3 min	content	Test Your Weather Station
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 the first `def` on the board and say the name aloud before students type, so the idea of a reusable block is clear
- Keep the PRIMM beat tight: predict what a call will do, run it, then investigate the return value or display
- When the weather station appears, name debugging as a skill: if the wrong symbol shows, check the temperature thresholds and the order of the conditions
- Link reuse to organisation: three small display procedures plus one decision function is easier to read than one long script

WHAT TO WATCH FOR

- Forgetting the colon after the function header or mismatching indentation under `def`: fix on the spot by lining up the body one indent in
- Calling the function without brackets, or with the wrong number of arguments: point at the call line and match it to the parameter list
- Mixing up the temperature ranges so the wrong symbol appears: walk through one test value against each condition in order

DIFFERENTIATION

- Support: Provide the procedure names and have students fill only the display lines, then call each one once
- Extension: Add another temperature range and symbol, or ask for a second parameter such as a wind flag
- Extension: Refactor repeated display code from an earlier program into a named function