LESSON 30

Swing Force Logger

40 minutes | Introduction to Python | Outcome 3.11 | Outcome 3.12 | Outcome 3.13 | Outcome 3.14 | Outcome 2.6 | Outcome 2.7 | Outcome 2.18

WHAT THIS LESSON DOES

In this lesson, you'll create a project to measure swing force using your Micro:bit's accelerometer. Follow step-by-step instructions to build a logger, store data, calculate key stats, and display results, enhancing your coding skills for real-world use.

CURRICULUM OUTCOMES

Outcome	What the specification asks for
3.11	use and control digital inputs and outputs within an embedded system
3.12	measure and store data returned from an analogue input
3.13	develop a program that utilises digital and analogue inputs
3.14	design automated applications using embedded systems
2.6	construct algorithms using appropriate sequences, selections/conditionals, loops and operators to solve a range of problems, to fulfil a specific requirement
2.7	implement algorithms using a programming language to solve a range of problems
2.18	collect, store and sort both continuous and discrete data

WHAT STUDENTS SHOULD BE ABLE TO DO

- Understand how to use an accelerometer to measure acceleration as a proxy for force in physical movements.
- Develop skills in data logging and storage using lists for real-world applications.
- Apply control structures and variables to detect and manage event thresholds.
- Create and utilise functions to calculate statistical data such as maximum and average values.
- Display computed results effectively using string formatting on hardware interfaces.

BEFORE THE LESSON

- Run the finished program yourself in the editor first so you know what a magnitude above the 2000 threshold looks like on the simulator versus a real device.
- If using physical micro:bits, have them cabled up and confirm each appears as a MICROBIT drive; the log download step needs that drive.
- Have the code from each step saved somewhere the class can reach in case a student's build breaks and they need a clean version to continue.

Swing Force Logger

HOW THE LESSON RUNS

- Introduction to the Swing Force Logger Project
- Import Modules and Basic Setup
- Read Accelerometer Data
- Set Threshold and Detect Swing
- Log the Data
- Calculate Max and Average with Functions
- Download your Code
- Download the Log Data
- Wrap Up and Extensions

TEACHING MOVES

- 1. Introduction to the Swing Force Logger Project. Frame the whole build up front: this one program pulls together thresholds, detection, lists and stats functions. Name the sports proxy so students see why acceleration stands in for force.
- 2. Import Modules and Basic Setup. Point out that running this does nothing visible, and that is correct. Confirm no error appears before anyone moves on, so an import typo does not surface three steps later.
- 3. Read Accelerometer Data. Have students shake the device and watch the scrolled number change. Stress that magnitude sums the absolute values so direction does not cancel out, which is why abs() is used.
- 4. Set Threshold and Detect Swing. Get students to test what value a genuine swing produces before trusting 2000. Have them adjust the threshold and re-test, so the number is calibrated, not copied.
- 5. Log the Data. Emphasise set_labels goes outside the loop and runs once. Walk through why readings are taken every 50ms for a short burst rather than continuously.
- 6. Calculate Max and Average with Functions. Trace calculate_max by hand with three values. Ask why the empty-list check returns 0, so students see defensive code protecting against a swing that logged nothing.
- 7. Download your Code. For physical devices, confirm each student sees the transfer complete before they test real swings. Keep a spare cable for the one that will not connect.
- 8. Download the Log Data. Show students the MY_DATA.html table opened in a browser. Connect the columns back to the code that wrote them, so the file is not magic.
- 9. Wrap Up and Extensions. Ask each student which of the four takeaways their own extension would exercise before they start it. Have them save as .hex so work reloads next period.

WHAT TO WATCH FOR

- Students think magnitude could be negative or that a single axis reading is the force. They read only get_x() and miss diagonal swings. Have them shake along different axes and watch magnitude respond regardless of direction. Point back to abs() summing all three.
- Students put set_labels inside the while loop, resetting the log on every pass. Ask what runs once versus what repeats. Show that setup belongs above the loop and only detection and logging belong inside.
- Students assume any threshold value will work because the code runs without error. Have them read the actual magnitude of a gentle nudge versus a real swing, then set the threshold between the two observed values.

DIFFERENTIATION

- Emerging: Give the working simulator so a student without hardware still sees detection fire, and pair them with someone who has a cabled device. Let them build only to the detection step (happy face on swing) as a complete, satisfying result before attempting logging.