LESSON 3

Reaction Timer

60 minutes (2 periods) | Introduction to Micro:bit Basics

WHAT THIS LESSON DOES

Follow this guide to build a fun project on your Micro:bit. Start by creating a new project, add a welcome message, set up a countdown, introduce a random delay, create variables, and record your reaction time.

WHAT STUDENTS SHOULD BE ABLE TO DO

- Develop skills in creating and managing a new project on the Micro:bit platform.
- Acquire knowledge on how to create and display messages using code.
- Understand and apply the concept of countdowns and delays in programming.
- Learn to create and utilise variables for storing time stamps.
- Gain proficiency in recording and displaying user interactions in real-time.

BEFORE THE BELL

- Count out one micro:bit per student or pair, plus USB cables. Check batteries or charge the day before.
- Open makecode.microbit.org on the school network and confirm students can create a project and download to a micro:bit.

Reaction Timer

HOW THE LESSON RUNS

Create a New Project
Creating a Welcome Message
Creating a Countdown
Adding a Random Delay
Create Variables
Recording Player Reaction

TEACHING MOVES

- 1. Create a New Project. Send everyone to the MakeCode micro:bit site. Have them click New Project and name it Reaction Timer so they can find it again after a break.
- 2. Creating a Welcome Message. Get them to add showString Get Ready and run it on the simulator or device before adding anything else. Confirm the scroll works.
- 3. Creating a Countdown. Build 3, pause 1000, 2, pause 1000, 1, pause 1000, then the centre line LEDs. Ask why each pause is needed. Flag the optional countdown function as a stretch.
- 4. Adding a Random Delay. Place the random pause after the line and before the full-screen LEDs. Stress that this gap is what stops players predicting the signal.
- 5. Create Variables. Create startTime and reactionTime together. Set startTime to runningTime at the exact moment the full LED screen appears, not earlier.
- 6. Recording Player Reaction. Add the A-button handler: store runningTime in reactionTime, show reactionTime minus startTime. Test in pairs and compare scores.

WHAT TO WATCH FOR

- Students set startTime at the start of the program or during the countdown, so every score is huge or meaningless. Trace the flow on the board. startTime must be assigned in the same breath as the full-screen go signal.
- Students press A during the random delay and think the timer should still work. Remind them the handler only becomes fair after the full LEDs light. Early presses are false starts. Have them replay once they wait for the signal.
- Students treat pause values as the reaction score, or show runningTime on its own without subtracting. Ask what each variable holds. The score is always reactionTime minus startTime, in milliseconds.

DIFFERENTIATION

- Emerging: Let them stabilise Get Ready and the 3-2-1 countdown on the simulator before touching variables or the button handler. Pair them with someone whose download already works so they can copy the project flow, not the answers.
- Developing: Have them point to where startTime is set and where the subtract happens before they download again. If the score looks wrong, get them to rebuild only the A-button block rather than rewriting the whole project.
- Proficient: Challenge them to wrap the 3-2-1 sequence in a countdown function as the step suggests. Have them run three trials, note the scores, and explain to a partner what a fair reaction time looks like.