

LESSON 28

Emojis

30 minutes

Microbit Basics

WHAT THIS LESSON DOES

Embark on a fun journey of programming your Microbit to display various emojis. Start a new project, create different faces such as asleep, happy, sad, and meh. Test your code in the simulator, add an angry face for when you shake your Microbit, design your own face, and finally, test your code on your Microbit.

WHAT STUDENTS SHOULD BE ABLE TO DO

- Understand and utilise the Microbit project platform.
- Programme Microbit to display various emoji faces.
- Implement button-press functions to change displayed emojis.
- Use Microbit's accelerometer sensor to trigger emoji changes.
- Design and implement a custom emoji using the 'show leds' block.

BEFORE THE LESSON

- Open makecode.com on pupil devices and confirm the site loads through the school filter.
- Devices ready, one per pupil or pair, charged and logged in as needed.
- If using physical Microbits for the final step, have boards, USB cables and a download route ready. Simulator-only is fine if boards are not available.

Emojis

HOW THE LESSON RUNS

- Create a new Microbit project
- Show the asleep face
- Make it happy
- Make it sad
- Make it meh
- Test your code in the simulator
- Make it angry
- Design your own face
- Test your code on your Microbit

TEACHING MOVES

- 1. Create a new Microbit project. Get everyone to makecode.com together and start a new Microbit project so the whole class shares a blank workspace before any blocks go in.
- 2. Show the asleep face. Have pupils place show icon on start and set it to Asleep. Confirm the simulator already shows the sleeping face before adding inputs.
- 3. Make it happy. Show how to drag on button A pressed and nest Happy inside it. Stress that the face only changes when A is pressed, not on start.
- 4. Make it sad. Repeat for button B with the Sad icon. Spot-check that nobody has left both faces on A.
- 5. Make it meh. Explain A+B means both buttons at once. Have them add the Meh icon on that event and try pressing A and B together.
- 6. Test your code in the simulator. Run the four checks from the step aloud: asleep on start, A happy, B sad, A+B meh. Fix any that fail before adding sensors.
- 7. Make it angry. Name the accelerometer in plain terms: it notices shakes and tilts. Add on shake and show Angry, then shake the simulator to prove it.
- 8. Design your own face. Show the show leds grid: hash is lit, dot is off. Let pupils redraw the pattern, then trigger it with screen up on the simulator.
- 9. Test your code on your Microbit. If boards are present, help download and transfer. If not, stay on the simulator and treat that as a full finish.

WHAT TO WATCH FOR

- Pupils press A then B in sequence and expect Meh, instead of holding A and B together. Model one clear simultaneous press on the simulator and have them watch for the Meh face only on A+B.
- Shake or screen-up code is put under a button event, so gestures never fire. Point to the Input gesture blocks and ask which event should own Angry and the custom face.
- In show leds, pupils mix up lit and unlit cells and the face looks wrong. Remind them hash means on and dot means off, then rebuild one row together before they edit freely.

DIFFERENTIATION

- Emerging: Sit them with a partner and get only the asleep face plus button A working before adding more events. Stay on the simulator and complete the ready-made icons first, leaving custom leds for later.
- Developing: After each new event, have them run the step's test list so buttons are solid before gestures. Ask them to say which input controls which face before they add the next block.
- Proficient: Invite them to redesign several show leds faces and try other gestures already in the Input menu. Have them explain their full input-to-face map to a peer using the simulator.