

Microbit Zombies

60 minutes

Microbit Innovators

WHAT THIS LESSON DOES

In this step-by-step lesson, you'll create a thrilling multiplayer game using microbits. You'll start by setting up a new project, creating variables for health and infection status, and setting up radio groups for communication. You'll then code your microbit to display health status and infection, lose health when near an infected microbit, and eventually become infected. The game begins with a random infection and continues until all players are infected. Enjoy the game!

WHAT STUDENTS SHOULD BE ABLE TO DO

- Develop a basic understanding of microbit project creation and the use of variables.
- Understand and implement the use of radio groups in multiplayer games.
- Learn to display different icons and messages based on variable conditions.
- Understand and apply the concept of signal strength in determining proximity in the game.
- Develop skills in creating interactive multiplayer games using randomisation and conditional statements.

BEFORE THE LESSON

- One micro:bit per player, charged, plus a computer each (or shared) with the MakeCode micro:bit editor and a way to transfer the .HEX file.
- Agree one radio group number for the whole class before coding starts.
- Clear floor space so pupils can move apart and test proximity safely.
- If possible, trial the signal-strength threshold in your room so infection distance feels fair.

Microbit Zombies

HOW THE LESSON RUNS

- Create a new microbit project
- Create two variables
- Set the radio group
- Display if you're infected or not
- Lose health
- Get infected
- Randomly infect a microbit
- Play the game!

TEACHING MOVES

1. Create a new microbit project. Open a new micro:bit project and retell the game in one minute: health starts at 9, infected starts false, one random zombie, closeness drains health until you become a zombie too.
2. Create two variables. Have every pupil create health and infected, then set health to 9 and infected to false before any other blocks.
3. Set the radio group. Add set radio group and check aloud that every board uses the same number, or the radios will not talk to each other.
4. Display if you're infected or not. Build the forever branch together: if infected, skull and send "zombie"; else happy face and show the health number.
5. Lose health. On receiving "zombie", use signal strength greater than -60 to mean too close: subtract 1 from health and flash the angry icon. Stress that -60 is closer than -90.
6. Get infected. At the bottom of forever, add if health is 0 or less then set infected to true so the board switches to zombie behaviour.
7. Randomly infect a microbit. Add the random 1-to-100 check inside forever so each board has a small ongoing chance to become the first zombie and start the spread.
8. Play the game! Download and flash the HEX. Start all boards together. If infection is too fast or slow, widen the random range or ease the signal threshold and reflash.

WHAT TO WATCH FOR

- Pupils think a more negative signal strength means closer, so they set the threshold the wrong way. Say: values nearer zero (for example -40) mean closer; -60 is a nearness cut-off, not a far one. Test two boards at arm's length.
- Boards on different radio groups never infect each other and the game seems broken. Before play, walk the room and confirm every project shows the same group number.
- Pupils expect exactly one zombie at the start and are confused when none or several appear. Explain the 1-in-100 roll runs again and again in forever. For a bigger group, raise the range (for example 1 to 200) so the first infection is rarer.

DIFFERENTIATION

- Emerging: Pair them with a partner who places blocks while they say what each variable and icon should do. Sit with them to set health, infected and the shared radio group before they build the forever logic.
- Developing: Ask them to point to where health falls, where infected flips to true, and what the board shows in each state. If their infection distance feels wrong, have them change only the signal threshold and retest with one partner.
- Proficient: Challenge them to tune random range and signal strength for the class size and room, then justify the numbers after a short playtest. Have them explain the full infection chain (broadcast, receive, health drop, flip to infected) to a pair who are still building.