

LESSON 27

Body Percussion and Sequencing

40 minutes

Unplugged Coding - Music

WHAT THIS LESSON DOES

In this lesson, you'll explore programming concepts through body percussion. Follow step-by-step activities to learn about commands, algorithms, debugging, and loops using actions like clapping and stomping. Engage in interactive tasks to build and perform rhythmic sequences.

WHAT STUDENTS SHOULD BE ABLE TO DO

- Understand the concept of algorithms through creating and following sequenced body percussion instructions.
- Develop skills in giving and interpreting clear, precise commands for consistent outcomes.
- Explore programming concepts like debugging and loops using physical movement and rhythm.
- Collaborate effectively to design and perform rhythmic sequences with peers.
- Recognise the importance of specificity in instructions to achieve intended results.

BEFORE THE LESSON

- This is an unplugged lesson: no pupil devices needed. Clear a safe space for movement.
- Print and cut the cut-and-stick worksheet attached to step 1; have scissors and glue ready for each pair.
- Open the interactive sequence generator on the board in step 4 and test it displays before the lesson.
- Plan an alternative for pupils who cannot snap fingers, such as a finger click.

Body Percussion and Sequencing

HOW THE LESSON RUNS

Introduction
Introduce the Robot Role
Introduce Body Percussion Commands
The Rhythm Builder
Cut and Stick Percussion Code
Run the Code and Provide Feedback
Whole class dance programme

TEACHING MOVES

- 2. Introduce the Robot Role. Ask the class to 'do something with your hands' and let them notice everyone did something different. Draw out that the instruction was vague, then give 'clap your hands once' and compare.
- 3. Introduce Body Percussion Commands. Model each of the four actions once, clearly and separately, and have the class copy. Check every pupil can perform clap, snap, stomp and pat on their own before moving on.
- 4. The Rhythm Builder. Build a four-step sequence on the board with the class, then name it: a list of instructions is an algorithm. Perform it together, saying each command aloud as you move.
- 5. Cut and Stick Percussion Code. Pair pupils and set the task clearly: each writes a four-step programme for their partner using only the four approved commands. Circulate to check sequences stick to the approved actions.
- 6. Run the Code and Provide Feedback. Cast the writer as the monitor. When the performer goes out of order, the programmer stops the action, names the error and corrects it before continuing. This is debugging.
- 7. Whole class dance programme. Brainstorm new moves like jump, wiggle or hop, then build one six-step whole-class algorithm on the whiteboard. Name a repeated set of steps a function as you sequence it.

WHAT TO WATCH FOR

- Pupils treat the order of commands as unimportant, thinking any four actions in any order is the same algorithm. Perform the same four commands in two different orders and ask if they looked the same. Reinforce that the computer follows the exact sequence written.
- Pupils, acting as programmer, fill in what their partner 'meant to do' instead of the error, letting mistakes pass. Remind them a robot cannot guess. The performer must do exactly what is written, and the programmer must stop and correct any step out of order.

DIFFERENTIATION

- Emerging: Reduce the sequence to two or three steps, and let the pupil use only clap and stomp until the pattern is secure. Pair with a confident partner and let them perform alongside before writing their own.
- Developing: Have the pupil say each command aloud as they perform it to keep sequence and action matched. Ask them to point to each written step as their partner performs, tracking the algorithm in order.
- Proficient: Challenge them to write a longer sequence that repeats a pair of actions, and introduce this as a loop. Ask them to explain their sequence back to you, naming why the order matters.