Lesson Plan: Loop Dance

Year Group: 7 | **Duration:** 50 minutes | **Topic:** Loops

1. Overview

Core Concept: Loops — repeating a sequence a fixed number of times or while a condition holds.

Learning Objectives:

- Identify repeated patterns in a sequence and express them as a loop
- Write REPEAT loop notation with a count and a body
- Understand nested loops (a loop inside a loop)
- Distinguish between count-controlled and condition-controlled loops

Key Vocabulary:

Term	Definition
Loop	A block of code that repeats
Iteration	One execution of the loop body
Repeat	Do something again
Count-controlled	A loop that runs a fixed number of times
Condition-controlled	A loop that runs until a condition becomes false
Body	The instructions inside the loop
Nested loop	A loop placed inside another loop

2. Before the Lesson

Print:

- ☐ `worksheet-dance-designer.md` — 1 copy per student

Room Setup:

- Move desks to the walls to create a large open floor space
- Alternatively, book the gym or a hall for this lesson
- No special equipment needed

3. Timed Lesson Flow

0–5 min — Hook: Spot the Pattern

1. Write on the board: `Clap, Clap, Stamp, Clap, Clap, Stamp, Clap, Clap, Stamp, Clap, Clap, Stamp`
2. Ask: *"What's the pattern? How could we write this more simply?"*
3. Elicit: it repeats 4 times. Guide students toward: `REPEAT 4 TIMES { Clap, Clap, Stamp }`
4. Ask: *"How many individual actions does that represent?"* (4 × 3 = 12)
5. Key point: **loops let us write a pattern once and say how many times to do it.**

5–10 min — Introduce REPEAT Notation

1. Write the notation on the board:

   ```
   REPEAT 4 TIMES {
       Clap
       Clap
       Stamp
   }
   ```

2. Explain: the number says how many times, the curly braces hold the body.
3. Perform the dance as a class — follow the algorithm, don't anticipate.

10–20 min — Class Dance Design

1. As a class, vote on 3 moves (e.g., Jump, Spin, Clap).
2. Write a simple REPEAT loop together on the board.
3. Perform it together 3 times.
4. Ask: *"What if we want to do the whole thing twice? How do we write that?"*
5. Introduce the nested loop concept — a loop inside a loop.

20–35 min — Group Design Challenge

1. Distribute `worksheet-dance-designer.md`.
2. Groups of 3–4 design their own dance using REPEAT notation.
3. Must include: at least one loop AND at least one nested loop.
4. The written algorithm must be specific enough for another group to perform without seeing the designers perform it first.

35–45 min — Perform From the Algorithm

1. Groups swap worksheets with another group.
2. The receiving group performs the algorithm as written — without any help from the authors.
3. Authors watch and note: does the performance match what they intended?

45–50 min — Debrief

- *"Did the performance match your intention? If not, what was ambiguous in your notation?"*
- *"What's the difference between REPEAT 4 TIMES and REPEAT UNTIL the music stops?"*
- *"What would happen if a loop never stopped? That's called an infinite loop — it crashes the program."*

---

4. Teacher Facilitation Notes

What to look for:

- Students who write out the full sequence instead of using a loop — ask: *"Can you spot the repeating pattern and write it once?"*
- Nested loops confuse many students — act it out slowly. The inner loop must complete ENTIRELY before the outer loop does its next iteration.
- Groups whose written algorithm is ambiguous — this is valuable! It mirrors Lesson 1 (Robot Teacher) — precision matters in loops too.

Common misconceptions:

- The inner loop only runs once per full performance — NO. The inner loop runs fully EACH TIME the outer loop iterates.
- REPEAT UNTIL runs the body until the body is done — NO. UNTIL refers to an external condition.

Managing the room:

- This lesson is deliberately physical and noisy — that's fine
- Insist students write their algorithm BEFORE performing (not after)

---

5. Extension / Early Finisher Tasks

1. **Condition-controlled version:** Rewrite your dance using `REPEAT UNTIL { music stops }` instead of a count. Discuss: what's the difference?
2. **Infinite loop:** Can you write a loop that would never stop? What condition could you add to ensure it eventually stops?
3. **Optimise:** Take the 12-action sequence from the hook. Can you express it with fewer characters of notation than `REPEAT 4 TIMES { Clap, Clap, Stamp }`?

---

6. Key Takeaway

> **Loops avoid repetition — write the pattern once and tell the computer how many times to run it. Nested loops multiply the repetitions.**