Lesson 7: Selection challenge

Introduction

This lesson is an extension of Lesson 8, where learners completed a PRIMM activity that introduced them to selection. This is the ‘make’ part of PRIMM, learners will complete a pair programming activity to create a joke machine. This will allow learners to apply their knowledge to a new, but similar scenario.

Learning objectives

- Use selection statements in a program
- Identify when selection statements should be used in programs
- Write and use expressions that use comparison operators (equal to, not equal to, less than, greater than, less than or equal to, greater than or equal to)

Key vocabulary

Comparison operators, expression, condition, selection

Preparation

Subject knowledge:

This lesson is designed to allow learners to apply the knowledge gained from the last lesson to their own programs. There is no new subject knowledge, however learners will need support while applying the construct in their programs.

You will need:

- Slides
- The joke machine: worksheet

You may also need:

- Example joke machine solution

Assessment opportunities

The starter activity will allow you to assess what learners can remember about comparison operators.

You can also use the activity time to circulate the room and formatively assess the progress of your learners through questioning and observation.

Outline plan

Please note that the slide deck labels the activities in the top right-hand corner to help you navigate the lesson.

**Timings are rough guides*

Starter activity (Slides 2–4) 5 mins	**What does each comparison operator do?** Give learners a few minutes to look at the list of comparison operators and decide the function of each one. They will already be familiar with the top three, as they have used those already. However, the last three will be new to them. Hopefully they should be able to make a prediction based on what they know from their math lessons. Slide 3 reveals the answers one by one through animations to allow you to question learners before revealing the answer. On slide 4 there is a brief method for remembering which symbol is less than when programming which may be useful for learners.

Activity 1 (Slides 6–8) 40 mins	**Use pair programming to create a joke machine program** This lesson uses pair programming, which involves placing the learners into suitable pairings and assigning the roles of driver and navigator. The driver's role is to control the keyboard and mouse, and type the code into the correct places. The navigator's role is to help the driver by watching for any mistakes, reading the instructions to the driver, and seeking support if needed. Both partners must contribute equally. You can find out more about pair programming from this Quick Read (ncce.io/ks4-qr-pp). A worksheet has been provided to guide learners through the creation of the program. It has been decomposed for them and offers the learners tips and sample code to help them find the solution. There is also an explorer task for learners that would like to improve their joke machines once they have handled the basics. Support learners by circulating the room, providing feedback and guidance as needed.

Plenary (Slides 9–11) 5–7 mins	**Solve the anagrams** Give learners some time to study the anagrams and figure out which programming keywords they are. If there is time, they should also attempt the explorer task, which asks them to write a definition for the keyword. They could use mini whiteboards for this, or a blank slide on their computer screens to allow you to check work quickly. Slide 10 reveals the answers to the anagrams. Definitions can be found on the vocabulary sheets if needed.
Homework	N/A