ANSWERS

F1.5 Comments and readable code

Programming basics · GCSE · OCR J277 2.3.1, AQA 8525 3.2.2, Edexcel 1CP2 6.1.4 · about 10 min

What this lesson is about

Comments, names, blank lines: code a person can read and change.

- Comments
- What a good comment says
- Names that explain themselves
- Blank lines and indentation
- Turning a line off

Questions

6 marks in all

1. What does Python do with a line that starts with `#`? [1 mark]

- ☐ **A** Skips it completely
- ☐ **B** Prints it
- ☐ **C** Runs it twice
- ☐ **D** Treats it as an error

> **Answer:** A. A comment is there for people. Python ignores it.

2. What does this program print? [1 mark]

```
print("before")
# print("middle")
print("after")
```

> **Answer:**
>
> ```
> before
> after
> ```
>
> Putting # in front of a line switches it off, so the middle line does not run.

3. Which comment is the most useful? [1 mark]

- ☐ **A** `led("red") # warn: battery is low`
- ☐ **B** `led("red") # set the LED to red`
- ☐ **C** `led("red") # led`
- ☐ **D** `led("red") # a line of code`

> **Answer:** A. A good comment says why. Saying what the line obviously does adds nothing.

4. What do blank lines between parts of a program change about how it runs? [1 mark]

- ☐ **A** Nothing: they only make it easier to read
- ☐ **B** They make it run slower
- ☐ **C** They cause an error
- ☐ **D** They end the program

> **Answer:** A. Python ignores blank lines. Like paragraphs, they separate the steps for the reader.

5. Why do programmers comment a line out instead of deleting it? [1 mark]

- ☐ **A** To switch it off while testing, and put it back easily
- ☐ **B** To make the program faster
- ☐ **C** Because deleting is not allowed
- ☐ **D** To make Python print it

> **Answer:** A. Removing the # brings the line straight back.

6. Which of these make a program easier to maintain? [1 mark]

Tick every answer that is true.

- ☐ **A** Comments that say why
- ☐ **B** Meaningful variable names
- ☐ **C** Blank lines between steps
- ☐ **D** Putting the whole program on as few lines as possible

> **Answer:** A, B, C. Comments, meaningful names, indentation and white space all help the next person read and change the code. Squashing it up does the opposite.

The task: a tidy drive

Write a program that turns the LED on, drives forward for a second, stops, turns the LED off and prints `done`. Put a comment above each part saying what it is for, and a blank line between the parts.

```
# the two lines every program starts with: the commands, then the robot
from bugbot import *
connect()

# ...
```

The hint students can ask for: LED on, drive forward for a second, stop, LED off, print 'done'. Put a comment above each part saying what it does.

A solution

```
# the two lines every program starts with: the commands, then the robot
from bugbot import *
connect()

# Light on: the program has started
led("green")

# Drive forward for a second
forward(50)
wait(1)

# Stop before the edge of the mat
stop()

# Light off and report: the program has finished
led("off")
print("done")
```

Any program that meets the task's checks is marked correct in the simulator; this is one way, not the only way.