



F1.4 Errors: syntax, runtime and logic

Programming basics · GCSE · OCR J277 2.3.2, AQA 8525 3.2.11, Edexcel

1CP2 1.2.5 · about 15 min

What this lesson is about

Reading an error message, and the three kinds of error by their exam names.

- Read the whole message
- Syntax errors: Python cannot read it
- Runtime errors: it crashes while running
- Logic errors: no message at all
- How to fix things

Questions

6 marks in all

1. `print("hello"` is missing its closing bracket. What kind of error is this? [1 mark]

- ☐ A Syntax error
- ☐ B Runtime error
- ☐ C Logic error
- ☐ D It is not an error

Answer: A. The code breaks the rules of the language, so Python cannot run it at all.

2. A program runs with no error message, but the robot drives backwards when it should go forwards. What kind of error is this? [1 mark]

- ☐ A Logic error
- ☐ B Syntax error
- ☐ C Runtime error
- ☐ D No error: the program finished

Answer: A. The program runs to the end but does the wrong thing. Only testing finds a logic error.

3. A program prints two lines, then crashes on `print(10 / 0)`. What kind of error is this? [1 mark]

- ☐ A Runtime error
- ☐ B Syntax error
- ☐ C Logic error
- ☐ D It prints 0

Answer: A. The line follows the rules, but it cannot be carried out when the program reaches it.

4. A program has a syntax error on its last line. How many of its lines run?

[1 mark]

- ☐ A None
- ☐ B All the lines before the error
- ☐ C All of them
- ☐ D Only the last line

Answer: A. Python cannot read a program with a syntax error, so it runs none of it. A runtime error is different: the lines before it do run.

5. Which of these would stop the program with an error message?

[1 mark]

Tick every answer that is true.

- ☐ A `wait("two")`
- ☐ B `prin("hi")`
- ☐ C `forward(50, distance=30)` when you meant 20
- ☐ D `stop` with no brackets

Answer: A, B. `wait("two")` and `prin("hi")` both fail when they run. Driving 30 instead of 20 and a bare `stop` give no message: they are logic errors.

6. Put the steps for fixing an error in order.

[1 mark]

Number the lines 1 to 4 to put them in the right order.

- ___ Read the error: find the kind and the line number
- ___ Look at that line, and the line above if it looks fine
- ___ Run again
- ___ Change one thing

Answer:

Read the error: find the kind and the line number
Look at that line, and the line above if it looks fine
Change one thing
Run again

Change one thing at a time, so you know which change mattered.

The task: fix three errors

This program should turn the LED green, wait a second, drive 30 cm forward into the green zone and print `arrived` . It has one syntax error, one runtime error and one logic error. Fix them one at a time.

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

led("green"
wait("1")
backward(50, distance=30)
print("arrived")
```

The hint students can ask for: One syntax error, one runtime error, one logic error. Fix the first error you are shown, run again, and watch where the robot goes.

A solution

```
from bugbot import *
connect()
led("green")
wait(1)
forward(50, distance=30)
print("arrived")
```

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