

Exercise walk-through

Documentation with Comments ■ 1.8

eXercise

You must be able to

- write single-line **comments** 注释 (//) and block comments (/* */)
- understand that comments are ignored by the compiler
- state a **precondition** 前置条件 and a **postcondition** 后置条件
- use comments to explain intent

Method — Comments

```
// this is a single-line comment and /* this is a block comment */.
```

Comments are **ignored by the compiler** but help humans read the code.

Method — Pre- and postconditions

- **Precondition** — what must be true **before** the code runs.
- **Postcondition** — what will be true **after** it runs.

Good comments explain **intent**, not obvious code.

Practice 2.1 ■ 2 marks

State the two ways to write comments in Java.

Solution 2.1

Worked steps

// for a single line and /* */ for a block **B1** **B1**.

Practice 2.2 ■ 1 mark

State whether comments affect how the program runs.

Solution 2.2

Method: Pre- and postconditions

Worked steps

no — they are ignored by the compiler **B1**.

Practice 2.3 ■ 1 mark

Define a precondition.

Solution 2.3

Worked steps

something that must be true before the code runs **B1**.

Exam-style 3.1 ■ 1 mark

A single-line comment in Java starts with

A `/*`

B `//`

C `#`

D `<!--`

Solution 3.1

Method: Comments

A /*

B //



C #

D <!--

Worked steps

B.

Exam-style 3.2 ■ 1 mark

Comments are

- A** run by the compiler
- B** ignored by the compiler
- C** errors
- D** program output

Solution 3.2

- A run by the compiler
- B ignored by the compiler
- C errors
- D program output



Worked steps

B.

Exam-style 3.3 ■ 1 mark

A method assumes its input number is positive.

- (a) Name this kind of assumption.
- (b) Name what is stated to be true after the method finishes.
- (c) State one good purpose of a comment.

Solution 3.3

Method: Pre- and postconditions

Worked steps

(a) a precondition **B1**. (b) a postcondition **B1**. (c) to explain the intent / aid readability **B1**.