

Exercise walk-through

Expressions and Output ■ 1.3

eXercise

You must be able to

- evaluate arithmetic **expressions** 表达式 using +, -, *, /, and %
- understand **integer division** 整数除法 and the remainder operator %
- apply **operator precedence** 运算符优先级
- use `System.out.print` and `System.out.println`

Method — Arithmetic and integer division

With two `int` operands, `/` is **integer division** (the fraction is dropped) and `%` gives the remainder: `7 / 2` is 3 and `7 % 2` is 1.

Method — Precedence and mixing types

`*`, `/`, `%` are done before `+`, `-`; use parentheses to change the order. Mixing an `int` and a `double` gives a `double`: `5 / 2.0` is `2.5`.

Method — Output

`System.out.print` writes without a newline; `System.out.println` adds a newline at the end.

Practice 2.1 ■ 1 mark

Evaluate $7 / 2$ in Java (integer division).

Solution 2.1

Method: Arithmetic and integer division

Worked steps

3 **B1**.

Practice 2.2 ■ 1 mark

Evaluate $7 \% 2$.

Solution 2.2

Worked steps

1 **B1**.

Practice 2.3 ■ 1 mark

State the difference between `print` and `println`.

Solution 2.3

Method: Output

Worked steps

`println` moves to a new line after printing; `print` does not **B1**.

Exam-style 3.1 ■ 1 mark

In Java, $9 / 4$ evaluates to

A 2.25

B 2

C 3

D 1

Solution 3.1

A 2.25

B 2



C 3

D 1

Worked steps

B.

Exam-style 3.2 ■ 1 mark

`System.out.println` differs from `print` by

- A** printing nothing
- B** adding a newline at the end
- C** rounding the value
- D** casting the value

Solution 3.2

Method: Output

- A printing nothing
- B adding a newline at the end 
- C rounding the value
- D casting the value

Worked steps

B.

Exam-style 3.3 ■ 1 mark

Evaluate the following.

(a) $10 \% 3$.

(b) $10 / 3$.

(c) State the value **and type** of $10 / 3.0$.

Solution 3.3

Worked steps

(a) 1 **B1**. (b) 3 **B1**. (c) about 3.33, a double **B1**.