

Exercise walk-through

String Manipulation ■ 1.15

eXercise

You must be able to

- create **String** 字符串 objects and combine them with **concatenation** 拼接 (+)
- call methods such as `length`, `substring`, and `indexOf`
- understand that a String is **immutable** 不可变
- compare strings with `equals` and `compareTo`, not `==`

Method — Strings and methods

Strings are made from literals and joined with `+`. They are **zero-indexed** (the first character is at index 0) and **immutable** (methods return **new** strings):

- `"hello".length()` is 5
- `"hello".substring(1, 3)` is `"el"` (indices 1 and 2)
- `"hello".indexOf("l")` is 2

Method — Comparing strings

Use `s.equals(t)` for equality and `s.compareTo(t)` for order — **not** `==`, which compares references.

Practice 2.1 ■ 1 mark

State the result of `"cat" + "fish"`.

Solution 2.1

Worked steps

"catfish" **B1**.

Practice 2.2 ■ 1 mark

State the value of `"data".length()`.

Solution 2.2

Worked steps

4 **B1**.

Practice 2.3 ■ 1 mark

State why you should compare strings with `equals`, not `==`.

Solution 2.3

Method: Comparing strings

Worked steps

`==` compares references (identity), while `equals` compares the actual characters **B1**.

Exam-style 3.1 ■ 1 mark

Strings in Java are

- A** mutable
- B** immutable
- C** numbers
- D** booleans

Solution 3.1

- A mutable
- B immutable
- C numbers
- D booleans



Worked steps

B.

Exam-style 3.2 ■ 1 mark

The first character of a String is at index

A 1

B 0

C -1

D its length

Solution 3.2

Method: Strings and methods

A 1

B 0



C -1

D its length

Worked steps

B.

Exam-style 3.3 ■ 1 mark

```
String s = "computer";
```

- (a) State the value of `s.length()`.
- (b) State the value of `s.substring(0, 3)`.
- (c) State the correct way to test whether `s` equals `"computer"`.

Solution 3.3

Method: Strings and methods

Worked steps

(a) 8 **B1**. (b) "com" **B1**. (c) `s.equals("computer")` **B1**.