Exercise walk-through

1.9
Method Signatures

AP Computer Science A

Documentation with Comments

Comments are ignored by the compiler but explain code to people.

Three comment styles

```
// a single-line comment
/* a block comment that
   may span several lines */
/** Returns the larger of a and b. */
int max(int a, int b)
```

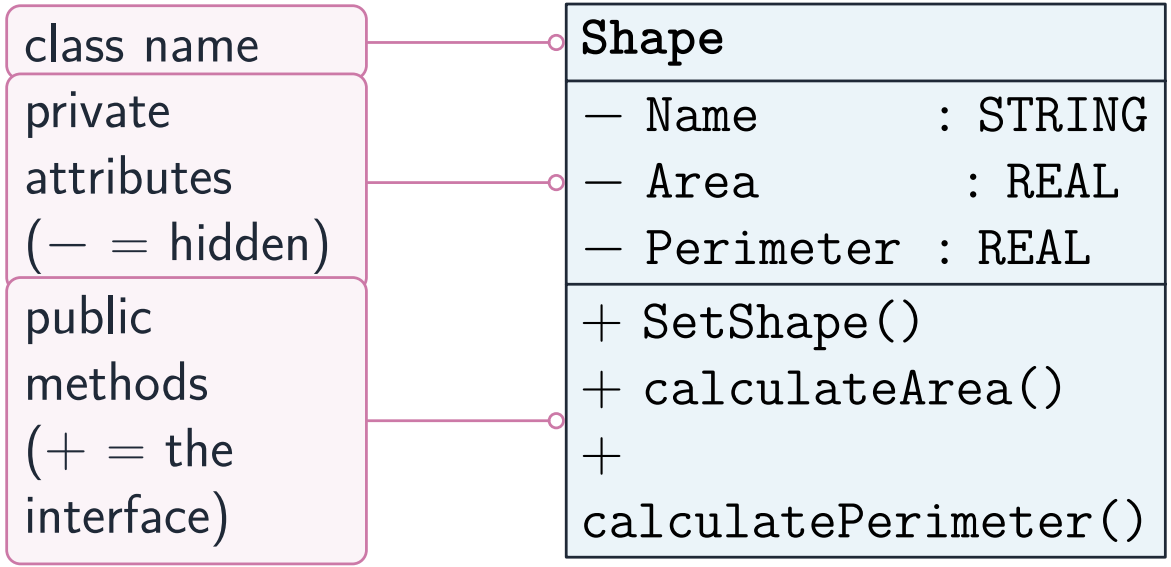
A `/** ... */` **Javadoc** comment documents a method's purpose, parameters and return value, including its preconditions and postconditions.

Class methods, instance methods, and where code comes from

class (static) method	belongs to the class itself, so you call it on the class name: `Math.abs(x)`. No object needed
instance method	belongs to an object, so you call it on a reference: `s.length()`
a library	pre-written classes you can use; Java's is the API
compiled	Java source is compiled once to bytecode, then run – so syntax errors surface before the program starts
abstraction	using a method by its name and contract, without knowing how it works inside

> If you find yourself creating an object just to call a method on it, check whether the method is `static` – `Math.sqrt` on a `new Math()` does not compile.

Methods, the API & the Math class, in detail



A class diagram: private attributes and public methods

Exercises 2.1·2.3

2.1 For `public static boolean inRange(int n, int low, int high)`, state the meth
name, return type and number of parameters. [2]

2.2 Write a call to `inRange` that stores in `ok` whether 15 is between 10 and 20. [1]

2.3 Explain why `int r = twice(2.5);` does not compile. [2]

Exercises 2.1·2.3

2.1 For `public static boolean inRange(int n, int low, int high)`, state the meth name, return type and number of parameters. [2]

2.1

- name `inRange`, return type `boolean`
- **3** parameters

2.2 Write a call to `inRange` that stores in `ok` whether 15 is between 10 and 20. [1]

2.2

- `boolean ok = inRange(15, 10, 20);`

2.3 Explain why `int r = twice(2.5);` does not compile. [2]

2.3

- 2.5 runs the `double` version, which returns a `double`
- a `double` cannot be stored in an `int`

Exercises 3.1·3.2

3.1 `public static void greet(String name, int times)`. Which call compiles? [1]

- [ ] **A** `greet(3, "Ana");`
- [ ] **B** `String g = greet("Ana", 3);`
- [ ] **C** `greet("Ana", 3);`
- [ ] **D** `greet("Ana");`

3.2 Which pair of headers can **not** both be in one class? [1]

- [ ] **A** `void f(int a)` and `void f(double a)`
- [ ] **B** `void f(int a)` and `void f(int a, int b)`
- [ ] **C** `int f(int a)` and `double f(int a)`
- [ ] **D** `void f(String s)` and `void f(int a)`

Exercises 3.1·3.2

3.1 `public static void greet(String name, int times)`. Which call compiles? [1]

- [ ] **A** `greet(3, "Ana");`
- [ ] **B** `String g = greet("Ana", 3);`
- [ ] **C** `greet("Ana", 3);`
- [ ] **D** `greet("Ana");`

3.1 C

- right number, order and types; B stores a `void` result

3.2 Which pair of headers can **not** both be in one class? [1]

- [ ] **A** `void f(int a)` and `void f(double a)`
- [ ] **B** `void f(int a)` and `void f(int a, int b)`
- [ ] **C** `int f(int a)` and `double f(int a)`
- [ ] **D** `void f(String s)` and `void f(int a)`

3.2 C

- same parameter list; the return type alone does not tell them apart

Exercise 3.3

3.3 What is printed by `System.out.println(twice(twice(3)));`? [1]

- [ ] **A** `6`
- [ ] **B** `12`
- [ ] **C** `33`
- [ ] **D** `12.0`

Exercise 3.3

3.3 What is printed by `System.out.println(twice(twice(3)));`? [1]

- ☐ **A** `6`
- **B** `12`
- **C** `33`
- **D** `12.0`

3.3 B

- `twice(3)` is 6; `twice(6)` is 12, an `int`

Exercise 3.4 (a)

3.4 A class has these methods:

```
public static double volume(double r)                  // a sphere
public static double volume(double r, double h)        // a cylinder
public static void report(String shape, double v)
```

(a) Write a statement that stores the volume of a cylinder of radius 2 and height 5 in `v`. [1]

Exercise 3.4 (a)

3.4 A class has these methods:

```
public static double volume(double r)                  // a sphere
public static double volume(double r, double h)        // a cylinder
public static void report(String shape, double v)
```

(a) Write a statement that stores the volume of a cylinder of radius 2 and height 5 in `v`. [1]

(a)

- `double v = volume(2.0, 5.0);`

Exercise 3.4 (b)

3.4 A class has these methods:

```
public static double volume(double r)                  // a sphere
public static double volume(double r, double h)        // a cylinder
public static void report(String shape, double v)
```

(b) Write a statement that calls `report` for a sphere of radius 3, using `volume` as an argument. [2]

Exercise 3.4 (b)

3.4 A class has these methods:

```
public static double volume(double r)                  // a sphere
public static double volume(double r, double h)        // a cylinder
public static void report(String shape, double v)
```

(b) Write a statement that calls `report` for a sphere of radius 3, using `volume` as an argument. [2]

(b)

- `report("sphere", volume(3.0));`

Exercise 3.4 (c)

3.4 A class has these methods:

```
public static double volume(double r)                 // a sphere
public static double volume(double r, double h)       // a cylinder
public static void report(String shape, double v)
```

(c) Explain how Java decides which `volume` method to run for each call. [2]

Exercise 3.4 (c)

3.4 A class has these methods:

```
public static double volume(double r)                 // a sphere
public static double volume(double r, double h)       // a cylinder
public static void report(String shape, double v)
```

(c) Explain how Java decides which `volume` method to run for each call. [2]

(c)

- by the number of arguments (and their types)
- one argument matches `volume(double)`; two match `volume(double, double)`

Exercise 3.4 (d)

3.4 A class has these methods:

```
public static double volume(double r)                  // a sphere
public static double volume(double r, double h)        // a cylinder
public static void report(String shape, double v)
```

(d) Explain why `double x = report("cube", 8.0);` does not compile. [1]

Exercise 3.4 (d)

3.4 A class has these methods:

```
public static double volume(double r)                 // a sphere
public static double volume(double r, double h)       // a cylinder
public static void report(String shape, double v)
```

(d) Explain why `double x = report("cube", 8.0);` does not compile. [1]

(d)

- `report` is `void`: it returns no value to store