



# 1Z0-816<sup>Q&As</sup>

Java SE 11 Programmer II

**Pass Oracle 1Z0-816 Exam with 100% Guarantee**

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.passapply.com/1z0-816.html>

100% Passing Guarantee  
100% Money Back Assurance

Following Questions and Answers are all new published by Oracle  
Official Exam Center

-  **Instant Download** After Purchase
-  **100% Money Back** Guarantee
-  **365 Days** Free Update
-  **800,000+** Satisfied Customers





### QUESTION 1

The screenshot shows an IDE with two tabs: Employee.java and Main.java. The Main.java tab is active and contains the following code:

```
1 import java.util.List;
2 import java.util.function.BinaryOperator;
3
4 public class Main {
5     public static void main (String... args) {
6         List<Employee> list = List.of(new Employee("John", 80000.0), new Employee("Scott", 90000.0));
7         double starts = 0.0;
8         double ratio = 1.0;
9         BinaryOperator<Double> bo = (a, b) -> a + b;
10        double totalSalary = list.stream().map(e -> e.getSalary() * ratio).reduce(starts, bo);
11        //line 1
12        System.out.println("Total salary = " + totalSalary);
13    }
14 }
15
16
```

The console output shows:

```
Console 1
Total salary = 170000.0
Completed with exit code: 0
```

Which interface in the java.util.function package will return a void return type?

- A. Supplier
- B. Predicate
- C. Function
- D. Consumer

Correct Answer: D

Reference: <https://www.geeksforgeeks.org/java-8-consumer-interface-in-java-with-examples/>

### QUESTION 2

Which two statements correctly describe capabilities of interfaces and abstract classes? (Choose two.)

- A. Interfaces cannot have protected methods but abstract classes can.
- B. Both interfaces and abstract classes can have final methods.
- C. Interfaces cannot have instance fields but abstract classes can.
- D. Interfaces cannot have static methods but abstract classes can.
- E. Interfaces cannot have methods with bodies but abstract classes can.

Correct Answer: AC

Reference: <https://www.guru99.com/interface-vs-abstract-class-java.html>

### QUESTION 3

Given:



```
1. public class Test {
2.     private static class Greet {
3.         private void print() {
4.             System.out.println("Hello World");
5.         }
6.     }
7.     public static void main(String[] args) {
8.         Test.Greet i = new Greet();
9.         i.print();
10.    }
11. }
```

What is the result?

- A. The compilation fails at line 9.
- B. The compilation fails at line 2.
- C. Hello World
- D. The compilation fails at line 8.

Correct Answer: C

#### QUESTION 4

Given the declaration:

```
@interface Resource {
    String name();
    int priority() default 0;
}
```

Examine this code fragment:

```
/* Loc1 */ class ProcessOrders { ... }
```

Which two annotations may be applied at Loc1 in the code fragment? (Choose two.)

- A. @Resource(priority=100)
- B. @Resource(priority=0)
- C. @Resource(name="Customer1", priority=100)
- D. @Resource(name="Customer1")
- E. @Resource



Correct Answer: AB

### QUESTION 5

Given:

```
// line 1
List<String> fruits = new ArrayList<>(List.of("apple", "orange", "banana"));
fruits.replaceAll(function);
```

Which statement on line 1 enables this code fragment to compile?

- A. Function function = String::toUpperCase;
- B. UnaryOperator function = s -> s.toUpperCase();
- C. UnaryOperator function = String::toUpperCase;
- D. Function function = m -> m.toUpperCase();

Correct Answer: C

```
1
2 import java.io.*;
3 import java.util.*;
4 import java.util.stream.Stream;
5 import java.util.function.Function;
6 import java.util.function.UnaryOperator;
7
8 class Hello {
9     public static void main(String[] args) {
10
11         UnaryOperator<String> function = String::toUpperCase;
12         List<String> fruits = new ArrayList<>(List.of("apple", "orange", "banana"));
13         fruits.replaceAll(function);
14
15     }
16 }
17
```

[Latest 1Z0-816 Dumps](#)

[1Z0-816 PDF Dumps](#)

[1Z0-816 Braindumps](#)