



1Z0-816^{Q&As}

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.pass4itsure.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**

Which interface in the java.util.function package can return a primitive type?

- A. ToDoubleFunction
- B. Supplier
- C. BiFunction
- D. LongConsumer

Correct Answer: A

Reference: <http://java.boot.by/ocpjp8-upgrade-guide/ch02s07.html>

QUESTION 2

Given the code fragment:

```
Path currentFile = Paths.get("/scratch/exam/temp.txt"); Path outputFile = Paths.get("/scratch/exam/new.txt"); Path  
directory = Paths.get("/scratch/");
```

```
Files.copy(currentFile, outputFile); Files.copy(outputFile, directory); Files.delete (outputFile);
```

The /scratch/exam/temp.txt file exists. The /scratch/exam/new.txt and /scratch/new.txt files do not exist.

What is the result?

- A. /scratch/exam/new.txt and /scratch/new.txt are deleted.
- B. The program throws a FileAlreadyExistsException.
- C. The program throws a NoSuchFileException.
- D. A copy of /scratch/exam/new.txt exists in the /scratch directory and /scratch/exam/new.txt is deleted.

Correct Answer: C

Explanation:

```
27 public class Main {  
28     public static void main(String[] args) {  
29         Path currentFile = Paths.get("/scratch/exam/temp.txt");  
30         Path outputFile = Paths.get("/scratch/exam/new.txt");  
31         Path directory = Paths.get("/scratch/");  
32  
33         Files.copy(currentFile, outputFile);  
34         Files.copy(outputFile, directory);  
35         Files.delete (outputFile);  
36     }  
37 }  
38
```

**QUESTION 3**

Given:

```
var data = new ArrayList();
```

```
data.add("Peter");
```

```
data.add(30);
```

```
data.add("Market Road");
```

```
data.set(1, 25);
```

```
data.remove(2);
```

```
data.set(3, 1000L);
```

```
System.out.print(data);
```

What is the output?

- A. [Market Road, 1000]
- B. [Peter, 30, Market Road]
- C. [Peter, 25, null, 1000]
- D. An exception is thrown at run time.

Correct Answer: D

```
Console 1
Exception in thread "main" java.lang.IndexOutOfBoundsException: Index 3 out of bounds for length 2
    at java.base/jdk.internal.util.Preconditions.outOfBounds(Preconditions.java:64)
    at java.base/jdk.internal.util.Preconditions.outOfBoundsCheckIndex(Preconditions.java:70)
    at java.base/jdk.internal.util.Preconditions.checkIndex(Preconditions.java:249)
    at java.base/java.util.Objects.checkIndex(Objects.java:372)
    at java.base/java.util.ArrayList.set(ArrayList.java:472)
    at abc.main(abc.java:13)
Completed with exit code: 1
```

QUESTION 4

A company has an existing sales application using a Java 8 jar file containing packages:

```
com.company.customer; com.company.customer.orders; com.company.customer.info; com.company.sales;
com.company.sales.leads; com.company.sales.closed; com.company.orders; com.company.orders.pending;
com.company.orders.shipped.
```

To modularize this jar file into three modules, customer, sales, and orders, which module-info.java would be correct?



- A.

```
module com.company.customer {
    opens com.company.customer;
}
module com.company.sales{
    opens com.company.sales;
}
module com.company.orders {
    opens com.company.orders;
}
```
- B.

```
module com.company.customer {
    exports com.company.customer;
}
module com.company.sales{
    exports com.company.sales;
}
module com.company.orders{
    exports com.company.orders;
}
```
- C.

```
module com.company.customer {
    requires com.company.customer;
}
module com.company.sales{
    requires com.company.sales;
}
module com.company.orders {
    requires com.company.orders;
}
```
- D.

```
module com.company.customer {
    provides com.company.customer;
}
module com.company.sales{
    provides com.company.sales;
}
module com.company.orders {
    provides com.company.orders;
}
```

- A. Option A
- B. Option B
- C. Option C
- D. Option D



Correct Answer: C

Reference: <https://developer.ibm.com/tutorials/java-modularity-3/>

QUESTION 5

A bookstore's sales are represented by a list of Sale objects populated with the name of the customer and the books they purchased.

```
public class Sale {  
  
    private String customer;  
  
    private List items;  
  
    // constructor, setters and getters not shown  
  
}  
  
public class Book {  
  
    private String name;  
  
    private double price;  
  
    // constructor, setters and getters not shown  
  
}
```

Given a list of Sale objects, tList, which code fragment creates a list of total sales for each customer in ascending order?



```
A. List<String> totalByUser = tList.stream()
    .collect(flatMapping(t -> t.getItems().stream(),
        groupingBy(Sale::getCustomer,
            summingDouble(Book::getPrice))))
    .entrySet().stream()
    .sorted(Comparator.comparing(Entry::getValue))
    .collect(mapping(e -> e.getKey() + ":" + e.getValue(),toList()));

B. List<String> totalByUser = tList.stream()
    .collect(groupingBy(Sale::getCustomer,
        flatMapping(t -> t.getItems().stream(),
            summingDouble(Book::getPrice))))
    .sorted(Comparator.comparing(Entry::getValue))
    .collect(mapping(e -> e.getKey() + ":" + e.getValue(),toList()));

C. List<String> totalByUser = tList.stream()
    .collect(groupingBy(Sale::getCustomer,
        flatMapping(t -> t.getItems().stream(),
            summingDouble(Book::getPrice))))
    .entrySet().stream()
    .sorted(Comparator.comparing(Entry::getValue))
    .collect(mapping(e -> e.getKey() + ":" + e.getValue(),toList()));

D. List<String> totalByUser = tList.stream()
    .collect(flatMapping(t -> t.getItems().stream(),
        groupingBy(Sale::getCustomer,
            summingDouble(Book::getPrice))))
    .sorted(Comparator.comparing(Entry::getValue))
    .collect(mapping(e -> e.getKey() + ":" + e.getValue(),toList()));
```

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: C

[Latest 1Z0-816 Dumps](#)

[1Z0-816 Study Guide](#)

[1Z0-816 Exam Questions](#)