Exam4Tests



What Clients Say About Us

Disclaimer Policy: The site does not guarantee the content of the comments. Because of the different time and the changes in the scope of the exam, it can produce different effect. Before you purchase the dump, please carefully read the product introduction from the page. In addition, please be advised the site will not be responsible for the content of the comments and contradictions between users.

❝ My friend introduces me this exam4tests. He passed M2110-670. And then are ready for C2090-555. Now I pass M2110-670 too. It is really help. It makes me half the work, double the results. They do not lie to me. Very thanks. It is worthy. ❞



Matthew

❝ Can not believe most test questions are coming from this practice file. It is very useful and helps me get a high score. Can not believe! It saves me a lot of time and mondy. Good value for money! ❞



Winfred

Our Clients

comcast · Bank of America · TIME WARNER CABLE · verizon · vodafone · xfinity · EarthLink

http://www.exam4tests.com

Latest exam questions & answers help you to pass IT exam test easily

Exam : **1z0-819**

Title : Java SE 11 Developer

Vendor : Oracle

Version : DEMO

NO.1 Which two are successful examples of autoboxing? (Choose two.)
A. Float g = Float.valueOf(null);
B. Double d = 4;
C. Integer e = 5;
D. Float f = 6.0;
E. Long c = 23L;
F. String a = "A";
Answer: C,F

NO.2 Given:

```
public class Hello {
   public static void main(String[] args) {
      System.out.println(args[0]+args[1]+args[2]);
   }
}
```

executed using command:
java Hello "Hello World" Hello World
What is the output?
A. Hello WorldHello World
B. Hello World Hello World
C. HelloHello WorldHelloWorld
D. Hello WorldHelloWorld
E. An exception is thrown at runtime.
Answer: B

NO.3 Given:

```
var i = 10;
var j = 5;
i += (j * 5 + j) / i - 2;
System.out.println(i);
```

What is the result?
A. 3
B. 25
C. 5
D. 11
E. 23
Answer: D

NO.4 Given:

```
public class Tester {
   static class Person implements /* line 1 */ {
      private String name;
      Person(String name) { this.name = name; }
      /* line 2 */
   }
   public static void main(String[] args) {
      Person[] people = {new Person("Joe"),
                         new Person("Jane"),
                         new Person("John")};
      Arrays-sort(people);
      for(Person person: people) {
         System.out.println(person.name);
      }
   }
}
```

You want the code to produce this output:
John
Joe
Jane
Which code fragment should be inserted on line 1 and line 2 to produce the output?

A. Insert Comparator<Person> on line 1.
Insert
public int compare(Person person) {
return person.name.compare(this.name);
}
on line 2.

B. Insert Comparable<Person> on line 1.
Insert
public int compare(Person p1, Person p2) {
return p1.name.compare(p2.name);
}
on line 2.

C. Insert Comparator<Person> on line 1.
Insert
public int compareTo(Person person) {
return person.name.compareTo(this.name);
}
on line 2.

D. Insert Comparator<Person> on line 1.
Insert
public int compare(Person p1, Person p2) {

return p1.name.compare(p2.name);
}
on line 2.
Answer: C

NO.5 Given:

```
public static void main(String[] args) {
    final List<String> fruits =
       List.of("Orange", "Apple", "Lemmon", "Raspberry");
    final List<String> types =
       List.of("Juice", "Pie", "Ice", "Tart");
    final var stream =
       IntStream.range(0, Math.min(fruits.size(), types.size()))
           .mapToObj((i) -> fruits.get(i) + " " + types.get(i) );
    stream. forEach(System.out::println);
}
```

What is the result?
A. The compilation fails.
B. Orange Juice
C. The program prints nothing.
D. Orange Juice Apple Pie Lemmon Ice Raspberry Tart
Answer: D

```
12 public class Person {
13     public static void main (String[] args) {
14         final List<String> fruits =
15         List.of("Orange", "Apple", "Lemmon", "raspberry");
16         final List<String> types =
17         List.of("Juice", "Pie", "Ice", "Tart");
18         final var stream =
19         IntStream.range(0, Math.min(fruits.size(), types.size()))
20         .mapToObj ((i) -> fruits.get(i) + " " + types.get(i) );
21         stream. forEach(System.out::println);
22     }
23
24 }
```

Result

compiled and executed in 1.227 sec(s)

```
Orange Juice
Apple Pie
Lemmon Ice
raspberry Tart
```

NO.6 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. The program throws a NoSuchFileException.

B. A copy of /scratch/exam/new.txt exists in the /scratch directory and /scratch/exam/new.txt is deleted.

C. /scratch/exam/new.txt and /scratch/new.txt are deleted.

D. The program throws a FileaAlreadyExistsException.

Answer: A

```
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
```

NO.7 Assume ds is a DataSource and the EMP table is defined appropriately.

```
try (Connection conn = ds.getConnection();
     PreparedStatement ps = conn.prepareStatement("INSERT INTO EMP VALUES(?, ?, ?)")) {
     ps.setObject(1, 101, JDBCType.INTEGER);
     ps.setObject(2, "SMITH", JDBCType.VARCHAR);
     ps.setObject(3, "HR", JDBCType.VARCHAR);
     ps.executeUpdate();
     ps.setInt(1, 102);
     ps.setString(2, "JONES");
     ps.executeUpdate();
}
```

What does executing this code fragment do?

A. inserts two rows (101, 'SMITH', 'HR') and (102, 'JONES', NULL)

B. inserts one row (101, 'SMITH', 'HR')

C. inserts two rows (101, 'SMITH', 'HR') and (102, 'JONES', 'HR')

D. throws a SQLException

Answer: B

NO.8 Given:

```
public interface InterfaceOne {
    void printOne();
}
```

Which three classes successfully override printOne()? (Choose three.)

A.
```
public abstract class TestClass implements InterfaceOne {
   public abstract void printOne();
}
```

B.
```
public class TestClass implements InterfaceOne {
   private void printOne(){
      System.out.println("one");
   }
}
```

C.
```
public class TestClass implements InterfaceOne {
   public void printOne(){
      System.out.println("one");
   }
}
```

D.
```
public abstract class TestClass implements InterfaceOne {
   public void printOne(){
      System.out.println("one");
   }
}
```

E.
```
public abstract class TestClass implements InterfaceOne {
   public String printOne(){
      return "one";
   }
}
```

F.
```
public class TestClass{
   public void printOne(){
      System.out.println("one");
   }
}
```

A. Option F
B. Option C
C. Option D
D. Option B
E. Option E
F. Option A
Answer: B,C,F

NO.9 Given:

```
public method foo() throws FooException {
     ...
}
```

and omitting the throws FooException clause results in a compilation error.
Which statement is true about FooException?
A. FooException is unchecked.
B. The body of foo can only throw FooException.
C. FooException is a subclass of RuntimeError.
D. The body of foo can throw FooException or one of its subclasses.
Answer: D

NO.10 Given:

```
enum QUALITY {
     A(100), B(75), C(50);
     int percent;
     private QUALITY(int percent) {
        this.percent = percent;
     }
}
```

and
checkQuality(QUALITY.A);
and

```
void checkQuality(QUALITY q) {
    switch (q) {
        case /* Insert code here */ :
            System.out.println("Best");
            break;
        default :
            System.out.println("Not best");
            break;
    }
}
```

Which code fragment can be inserted into the switch statement to print Best?

A. A

B. QUALITY.A

C. QUALITY.ValueOf()

D. toString()

Answer: A