

Oracle

Exam Questions 1z0-808

Java SE 8 Programmer I



NEW QUESTION 1

Given:

```
public static void main(String[] args) {  
    String ta = "A ";  
    ta = ta.concat("B ");  
    String tb = "C ";  
    ta = ta.concat(tb);  
    ta.replace('C', 'D');  
    ta = ta.concat(tb);  
    System.out.println(ta);  
}
```

What is the result?

- A. A B C D
- B. A C D
- C. A C D D
- D. A B D
- E. A B D C

Answer: C**NEW QUESTION 2**

Given the content of three files:

A.java:

```
public class A {  
    public void a() {}  
    int a;  
}
```

B.java:

```
public class B {  
    private int doStuff() {  
        private int x = 100;  
        return x++;  
    }  
}
```

C.java:

```
import java.io.*;  
package p1;  
class A {  
    public void main(String fileName) throws IOException { }  
}
```

Which statement is true?

- A. Only the A.Java file compiles successfully.
- B. Only the B.java file compiles successfully.
- C. Only the C.java file compiles successfully.
- D. The A.Java and B.java files compile successfully.
- E. The B.java and C.java files compile successfully.
- F. The A.Java and C.java files compile successfully.

Answer: A**NEW QUESTION 3**

Given the following main method:

```
public static void main(String[] args) {  
    int num = 5;  
    do {  
        System.out.print(num-- + " ");  
    } while(num == 0);  
}
```

What is the result?

- A. 5 4 3 2 1 0
- B. 5 4 3 2 1
- C. 4 2 1

- D. 5
- E. Nothing is printed

Answer: D

NEW QUESTION 4

You are asked to develop a program for a shopping application, and you are given this information:

- The application must contain the classes Toy, EduToy, and ConsToy. The Toy class is the superclass of the other two classes.
- The int calculatePrice (Toy t) method calculates the price of a toy.
- The void printToy (Toy t) method prints the details of a toy.

Which definition of the Toy class adds a valid layer of abstraction to the class hierarchy?

- A**
- ```
public abstract class Toy{
 public abstract int calculatePrice(Toy t);
 public void printToy(Toy t) { /* code goes here */ }
}
```
- B**
- ```
public abstract class Toy {
    public int calculatePrice(Toy t) ;
    public void printToy(Toy t) ;
}
```
- C**
- ```
public abstract class Toy {
 public int calculatePrice(Toy t);
 public final void printToy(Toy t){ /* code goes here */ }
}
```
- D**
- ```
public abstract class Toy {
    public abstract int calculatePrice(Toy t) { /* code goes here */ }
    public abstract void printToy(Toy t) { /* code goes here */ }
}
```

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

Answer: A

NEW QUESTION 5

Given the definitions of the MyString class and the Test class:

```
package p1;
class MyString {
    String msg;
    MyString(String msg) {
        this.msg = msg;
    }
}
```

Test.java:

```
package p1;
public class Test {
    public static void main(String[] args) {
        System.out.println("Hello " + new StringBuilder("Java SE 8"));
        System.out.println("Hello " + new MyString("Java SE 8").msg);
    }
}
```

What is the result?

- A
- ```
Hello Java SE 8
Hello Java SE 8
```
- B
- ```
Hello java.lang.StringBuilder@<<hashCode1>>
Hello pl.MyString@<<hashCode2>>
```
- C
- ```
Hello Java SE 8
Hello pl.MyString@<<hashCode>>
```
- D Compilation fails at the Test class

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

**Answer: D**

#### NEW QUESTION 6

Given the code fragment:

```
public static void main(String[] args) {
 short s1 = 200;
 Integer s2 = 400;
 Long s3 = (long) s1 + s2; //line n1
 String s4 = (String) (s3 * s2); //line n2
 System.out.println("Sum is " + s4);
}
```

What is the result?

- A. Sum is 600  
B. Compilation fails at line n1.  
C. Compilation fails at line n2.  
D. A ClassCastException is thrown at line n1.  
E. A ClassCastException is thrown at line n2.

**Answer: C**

#### NEW QUESTION 7

Given:

```
public class App {
 int count;
 public static void displayMsg () {
 count++; // line n1
 System.out.println ("Welcome "+"Visit Count: "+count); // line n2
 }
 public static void main (String [] args) {
 App.displayMsg (); // line n3
 App.displayMsg (); // line n4
 }
}
```

What is the result?

- A. Compilation fails at line n3 and line n4.  
B. Compilation fails at line n1 and line n2.  
C. Welcome Visit Count:1Welcome Visit Count: 1  
D. Welcome Visit Count:1Welcome Visit Count: 2

**Answer: B**

#### NEW QUESTION 8

Which two are benefits of polymorphism? (Choose two.)

- A. Faster code at runtime  
B. More efficient code at runtime

- C. More dynamic code at runtime
- D. More flexible and reusable code
- E. Code that is protected from extension by other classes

**Answer:** BD

#### NEW QUESTION 9

Given the code fragment:

```
public static void main(String[] args) {
 int ii = 0;
 int jj = 7;
 for (ii = 0; ii < jj - 1; ii = ii + 2) {
 System.out.print(ii + " ");
 }
}
```

What is the result?

- A. 2 4
- B. 0 2 4 6
- C. 0 2 4
- D. Compilation fails

**Answer:** C

#### NEW QUESTION 10

Which two statements are true about Java byte code? (Choose two.)

- A. It can be serialized across network.
- B. It can run on any platform that has a Java compiler.
- C. It can run on any platform.
- D. It has ".java" extension.
- E. It can run on any platform that has the Java Runtime Environment.

**Answer:** AE

#### NEW QUESTION 10

This grid shows the state of a 2D array:

|   |   |   |
|---|---|---|
| 0 | 0 |   |
|   | X | 0 |
| X |   | X |

The grid is created with this code:

```
char[][] grid = new char[3][3];
grid[1][1] = 'X';
grid[0][0] = '0';
grid[2][0] = 'X';
grid[0][1] = '0';
grid[2][2] = 'X';
grid[1][2] = '0';
//line n1
```

Which line of code, when inserted in place of //line n1, adds an X into the grid so that the grid contains three consecutive Xs?

- A. grid[2][1] = 'X';
- B. grid[3][2] = 'X';
- C. grid[3][1] = 'X';
- D. grid[2][3] = 'X';

**Answer:** D

#### NEW QUESTION 12

Which three statements are true about exception handling? (Choose three.)

- A. Only unchecked exceptions can be rethrown.
- B. All subclasses of the RuntimeException class are not recoverable.
- C. The parameter in a catch block is of Throwable type.
- D. All subclasses of the RuntimeException class must be caught or declared to be thrown.
- E. All subclasses of the RuntimeException class are unchecked exceptions.
- F. All subclasses of the Error class are not recoverable.

**Answer:** BCD

#### NEW QUESTION 15

Given the code fragment:

```
abstract class Toy {
 int price;
 // line n1
}
```

Which three code fragments are valid at line n1?

A

```
public static void insertToy() {
 /* code goes here */
}
```

B

```
final Toy getToy() {
 return new Toy();
}
```

C

```
public void printToy();
```

D

```
public int calculatePrice() {
 return price;
}
```

E

```
public abstract int computeDiscount();
```

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

**Answer:** CDE

#### NEW QUESTION 18

Given the code fragment:

```
public static void main(String[] args) {
 LocalDate date = LocalDate.of(2012, 01, 32);
 date.plusDays(10);
 System.out.println(date);
}
```

What is the result?

- A. 2012-02-10
- B. 2012-02-11
- C. Compilation fails
- D. A DateTimeException is thrown at runtime.

**Answer:** D

#### NEW QUESTION 23

Given the code fragment:



```
abstract class Planet {
 protected void revolve() { //line n1
 }

 abstract void rotate(); //line n2
}

class Earth extends Planet {
 void revolve() { //line n3
 }

 protected void rotate() { //line n4
 }
}
```

Which two modifications, made independently, enable the code to compile? (Choose two.)

- A. Make the method at line n1 public.
- B. Make the method at line n2 public.
- C. Make the method at line n3 public.
- D. Make the method at line n3 protected.
- E. Make the method at line n4 public.

**Answer:** CD

#### NEW QUESTION 27

Given the code fragment:

```
public static void main(String[] args) {
 String myStr = "Hello World ";
 myStr.trim();
 int i1 = myStr.indexOf(" ");
 System.out.println(i1);
}
```

What is the result?

- A. An exception is thrown at runtime.
- B. -1
- C. 5
- D. 10

**Answer:** A

#### NEW QUESTION 32

Given:

```
public class Test {
 int x, y;

 public Test(int x, int y) {
 initialize(x, y);
 }

 public void initialize(int x, int y) {
 this.x = x * x;
 this.y = y * y;
 }

 public static void main(String[] args) {
 int x = 3, y = 5;
 Test obj = new Test(x, y);
 System.out.println(x + " " + y);
 }
}
```

What is the result?

- A. Compilation fails.
- B. 3 5
- C. 0 0
- D. 9 25

**Answer:** B

#### NEW QUESTION 35

Given:

```
public class Test {
 public static void main(String[] args) {
 Test ts = new Test();
 System.out.print(isAvailable + " ");
 isAvailable= ts.doStuff();
 System.out.println(isAvailable);
 }
 public static boolean doStuff() {
 return !isAvailable;
 }
 static boolean isAvailable = false;
}
```

What is the result?

- A. Compilation fails.
- B. false true
- C. true false
- D. true true
- E. false false

**Answer: B**

#### NEW QUESTION 38

Given:

```
class Student {
 String name;
 public Student(String name) {
 this.name = name;
 }
}

public class Test {
 public static void main(String[] args) {
 Student[] students = new Student[3];
 students[1] = new Student("Richard");
 students[2] = new Student("Donald");
 for (Student s : students) {
 System.out.println("" + s.name);
 }
 }
}
```

What is the result?

- A. nullRichardDonald
- B. RichardDonald
- C. Compilation fails.
- D. An ArrayIndexOutOfBoundsException is thrown at runtime.
- E. A NullPointerException is thrown at runtime.

**Answer: E**

#### NEW QUESTION 40

Given:



```
class Caller {
 private void init () {
 System.out.println("Initialized");
 }

 private void start () {
 init();
 System.out.println("Started");
 }
}

public class TestCall {
 public static void main(String[] args) {
 Caller c = new Caller();
 c.start();
 c.init();
 }
}
```

What is the result?

- A. An exception is thrown at runtime.
- B. InitializedStartedInitialized
- C. InitializedStarted
- D. Compilation fails.

**Answer: D**

#### NEW QUESTION 43

Given the code fragment:

```
3. public static void main(String[] args) {
4. int x = 6;
5. while (isAvailable(x)) {
6. System.out.print(x);
7.
8. }
9. }
10.
11. public static boolean isAvailable(int x) {
12. return --x > 0 ? true : false;
13. }
```

Which modification enables the code to print 54321?

- A. Replace line 6 with System.out.print (--x);
- B. At line 7, insert x --;
- C. Replace line 5 with while (is Available(--x)) {
- D. Replace line 12 with return (x > 0) ? false : true;

**Answer: C**

#### NEW QUESTION 47

Which statement best describes encapsulation?

- A. Encapsulation ensures that classes can be designed so that only certain fields and methods of an object are accessible from other objects.
- B. Encapsulation ensures that classes can be designed so that their methods are inheritable.
- C. Encapsulation ensures that classes can be designed with some fields and methods declared as abstract.
- D. Encapsulation ensures that classes can be designed so that if a method has an argument MyType x, any subclass of MyType can be passed to that method.

**Answer: A**

#### NEW QUESTION 48

Given the code fragment:

```
LocalDate date1 = LocalDate.now();
LocalDate date2 = LocalDate.of(6, 20, 2014);
LocalDate date3 = LocalDate.parse("2014-06-20", DateTimeFormatter.ISO_DATE);
System.out.println("date1 = " + date1);
System.out.println("date2 = " + date2);
System.out.println("date3 = " + date3);
```

Assume that the system date is June 20, 2014. What is the result?

A

```
date1 = 2014-06-20
date2 = 2014-06-20
date3 = 2014-06-20
```

B

```
date1 = 06/20/2014
date2 = 2014-06-20
date3 = Jun 20, 2014
```

C Compilation fails.

D An exception is thrown at runtime.

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

**Answer:** A

#### NEW QUESTION 53

Given:

```
class Vehicle {
 int x;
 Vehicle() {
 this(10); // line n1
 }
 Vehicle(int x) {
 this.x = x;
 }
}

class Car extends Vehicle {
 int y;
 Car() {
 super();
 this(20); // line n2
 }
 Car(int y) {
 this.y = y;
 }
 public String toString() {
 return super.x + ":" + this.y;
 }
}
```

And given the code fragment:

And given the code fragment:

```
Vehicle y = new Car();
System.out.println(y);
```

What is the result?

- A. 10:20
- B. 0:20
- C. Compilation fails at line n1
- D. Compilation fails at line n2

**Answer:** D

#### NEW QUESTION 57

Given the code fragment:

```
public static void main(String[] args) {
 int[][] arr = new int [2] [4];
 arr[0] = new int []{1, 3, 5, 7};
 arr[1] = new int []{1, 3};
 for (int[] a : arr) {
 for (int i : a) {
 System.out.print(i+ " ");
 }
 System.out.println();
 }
}
```

What is the result?

- A Compilation fails.
- B
- ```
1 3  
1 3
```
- C
- ```
1 3
followed by an ArrayIndexOutOfBoundsException
```
- D
- ```
1 3  
1 3 0 0
```
- E
- ```
1 3 5 7
1 3
```

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

**Answer:** E

**Explanation:**

```
Your Code ...
1- public class MyClass {
2- public static void main (String [] args) {
3- int [][] arr = new int [2] [4];
4- arr[0] = new int [] {1, 3, 5, 7};
5- arr[1] = new int [] {1, 3};
6- for (int [] a : arr) {
7- for (int i : a) {
8- System.out.print(i+ " ");
9- }
10- System.out.println ();
11- }
12- }
13- }
14-

External Libraries ... Add External Library \(from Maven Repo\)

CommandLine Arguments ...

Interactive mode : ☐ OFF Version: JDK 9.0.1

Stdin Inputs...

[Execute] [Save] [My Projects] [Recent] [Collaborate] [More Options v]

Result...
CPU Time: 0.13 sec(s), Memory: 30680 kilobyte(s) compiled and executed in 0.705 sec(s)
1 3 5 7
1 3
```

**NEW QUESTION 60**

• • • • •

## Thank You for Trying Our Product

### We offer two products:

1st - We have Practice Tests Software with Actual Exam Questions

2nd - Questions and Answers in PDF Format

### 1z0-808 Practice Exam Features:

- \* 1z0-808 Questions and Answers Updated Frequently
- \* 1z0-808 Practice Questions Verified by Expert Senior Certified Staff
- \* 1z0-808 Most Realistic Questions that Guarantee you a Pass on Your FirstTry
- \* 1z0-808 Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year

**100% Actual & Verified — Instant Download, Please Click**  
**[Order The 1z0-808 Practice Test Here](#)**