

# Oracle

## Exam Questions 1Z0-809

Java SE 8 Programmer II



**NEW QUESTION 1**

What is the result?

```
7. BiPredicate<String, String> bp = (String s1, String s2) -> s1.contains("SG") &&
   s2.contains("Java");
8. BiFunction<String, String, Integer> bf = (String s1, String s2) -> {
9.     int fee = 0;
10.    if (bp.test(s1, s2)) {
11.        fee = 100;
12.    }
13.    return fee;
14. };
15. int fee1 = bf.apply("D101SG", "Java Programming");
16. System.out.println(fee1);
```

- A. A compilation error occurs at line 7.
- B. 100
- C. A compilation error occurs at line 8.
- D. A compilation error occurs at line 15.

**Answer:** A

**NEW QUESTION 2**

Given the code fragment:

```
public class FileThread implements Runnable { String fName;
public FileThread(String fName) { this.fName = fName; } public void run () System.out.println(fName);}
public static void main (String[] args) throws IOException, InterruptedException {
ExecutorService executor = Executors.newCachedThreadPool(); Stream<Path> listOfFiles = Files.walk(Paths.get("Java Projects")); listOfFiles.forEach(line -> {
executor.execute(new FileThread(line.getFileName().toString ())); //
line n1
});
executor.shutdown(); executor.awaitTermination(5, TimeUnit.DAYS); // line n2
}
}
```

The Java Projects directory exists and contains a list of files. What is the result?

- A. The program throws a runtime exception at line n2.
- B. The program prints files names concurrently.
- C. The program prints files names sequentially.
- D. A compilation error occurs at line n1.

**Answer:** B

**NEW QUESTION 3**

Given the code fragment:

```
for (Course a : Course.values()) {
    System.out.print(a + " Fees " + a.getCost()+" " );
}
```

Which is the valid definition of the Course enum?

```
A. enum Course { JAVA(100), J2ME(150);
    private int cost;
    public Course(int c) {
        this.cost = c;
    }
    int getCost() {
        return cost;
    }
}

B. enum Course { JAVA(100), J2ME(150);
    private static int cost;
    private Course(int c) {
        this.cost = c;
    }
    static int getCost() {
        return cost;
    }
}

C. final enum Course { JAVA(100), J2ME(150);
    private int cost;
    public Course(int c) {
        this.cost = c;
    }
    int getCost() {
        return cost;
    }
    void setCost(int c) {
        this.cost = c;
    }
}

D. enum Course { JAVA(100), J2ME(150);
    private int cost;
    Course(int c) {
        this.cost = c;
    }
    int getCost() {
        return cost;
    }
}
```

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

Answer: A

#### NEW QUESTION 4

Which statement is true about java.time.Duration?

- A. It tracks time zones.
- B. It preserves daylight saving time.
- C. It defines time-based values.
- D. It defines date-based values.

**Answer:** C

#### NEW QUESTION 5

Given:

```
public class Product {
    public double applyDiscount(double price) {
        assert (price > 0); // line n1
        return price * 0.50;
    }
    public static void main(String[] args) {
        Product p = new Product();
        double newPrice =
            p.applyDiscount(Double.parseDouble(args[0]));
        System.out.println("New Price: " + newPrice);
    }
}
```

and the command: java Product 0 What is the result?

- A. An AssertionError is thrown.
- B. A compilation error occurs at line n1.
- C. New Price: 0.0
- D. A NumberFormatException is thrown at run time.

**Answer:** D

#### NEW QUESTION 6

Given:

```
class Resource implements AutoCloseable {
    public void close() throws Exception {
        System.out.print("Close-");
    }
    public void open() {
        System.out.print("Open-");
    }
}
```

and this code fragment:

```
Resource res1 = new Resource();
try {
    res1.open();
    res1.close();
} catch (Exception e) {
    System.out.println("Exception - 1");
}
try (res1 = new Resource()) { // line n1
    res1.open();
} catch (Exception e) {
    System.out.println("Exception - 2");
}
```

What is the result?

- A. Open-Close- Exception – 1 Open-Close-
- B. Open-Close-Open-Close-
- C. A compilation error occurs at line n1.
- D. Open-Close-Open-

**Answer:** C

#### NEW QUESTION 7

Given the code fragment:

```
List<String> codes = Arrays.asList ("DOC", "MPEG", "JPEG"); codes.forEach (c -> System.out.print(c + " "));  
String fmt = codes.stream()  
.filter (s-> s.contains ("PEG"))  
.reduce((s, t) -> s + t).get(); System.out.println("\n" + fmt); What is the result?
```

- A. DOC MPEG JPEG MPEGJPEG
- B. DOC MPEG MPEGJPEG MPEGMPEGJPEG
- C. MPEGJPEG MPEGJPEG
- D. The order of the output is unpredictable.

**Answer:** A

#### NEW QUESTION 8

Given:

```
public class Job {  
    String name;  
    Integer cost;  
    Job(String name, Integer cost) {  
        this.name = name;  
        this.cost = cost;  
    }  
    String getName() { return name; }  
    int getCost() { return cost; }  
    public static void main(String[] args) {  
        Job j1 = new Job("IT", null);  
        DoubleSupplier jS1 = j1::getCost;  
        System.out.println(j1.getName() + ":" + jS1.getAsDouble());  
    }  
}
```

What is the result?

- A. IT:null
- B. A NullPointerException is thrown at run time.
- C. A compilation error occurs.
- D. IT:0.0

**Answer:** D

#### NEW QUESTION 9

Given:



```
public class Foo {
    public void methodB(String s) { System.out.println("Foo " + s ); }
}

public class Bar extends Foo {
    public void methodB(String s) { System.out.println("Bar " + s); }
}

public class Baz extends Bar {
    public void methodB(String s) { System.out.println("Baz " + s); }
}

public class Daze extends Baz{
    private Bar bb = new Bar();
    public void methodB(String s) {
        bb.methodB(s);
        super.methodB(s);
    }
}

public class TestClass {
    public static void main(String[] args) {
        Baz d = new Daze();
        d.methodB("Hello");
    }
}
```

What is the result?

- A. Bar Hello Foo Hello
- B. Bar Hello Baz Hello
- C. Baz Hello
- D. A compilation error occurs in the Daze class.

**Answer: C**

#### NEW QUESTION 10

Which two statements are true about synchronization and locks? (Choose two.)

- A. A thread automatically acquires the intrinsic lock on a synchronized statement when executed.
- B. The intrinsic lock will be retained by a thread if return from a synchronized method is caused by an uncaught exception.
- C. A thread exclusively owns the intrinsic lock of an object between the time it acquires the lock and the time it releases it.
- D. A thread automatically acquires the intrinsic lock on a synchronized method's object when entering that method.
- E. Threads cannot acquire intrinsic locks on classes.

**Answer: AB**

#### NEW QUESTION 10

Given the code fragment:

```
List<String> valList = Arrays.asList("", "George", "", "John", "Jim");
Long newVal = valList.stream()           // line n1
    .filter(x -> !x.isEmpty())
    .count();                           // line n2
System.out.print(newVal);
```

What is the result?

- A. A compilation error occurs at line n2.
- B. 3
- C. 2
- D. A compilation error occurs at line n1.

**Answer: A**

#### NEW QUESTION 13

Given the code fragment:

```
List<String> listVal = Arrays.asList("Joe", "Paul", "Alice", "Tom"); System.out.println (
```

```
// line n1
```

```
);
```

Which code fragment, when inserted at line n1, enables the code to print the count of string elements whose length is greater than three?

- A. `listVal.stream().filter(x -> x.length()>3).count()`
- B. `listVal.stream().map(x -> x.length()>3).count()`
- C. `listVal.stream().peek(x -> x.length()>3).count().get()`
- D. `listVal.stream().filter(x -> x.length()>3).mapToInt(x -> x).count()`

**Answer:** A

#### NEW QUESTION 16

Given the content:

```
MessagesBundle.properties file:

inquiry = How are you?

MessagesBundle_de_DE.properties file:

inquiry = Wie geht's?
```

and given the code fragment:

```
Locale currentLocale;
// line 1
ResourceBundle messages = ResourceBundle.getBundle("MessagesBundle", currentLocale);
System.out.println(messages.getString("inquiry"));
```

Which two code fragments, when inserted at line 1 independently, enable the code to print "Wie geht's?"

- A. `currentLocale = new Locale ("de", "DE");`
- B. `currentLocale = new Locale.Builder ().setLanguage ("de").setRegion ("DE").build ();`
- C. `currentLocale = Locale.GERMAN;`
- D. `currentLocale = new Locale(); currentLocale.setLanguage ("de"); currentLocale.setRegion ("DE");`
- E. `currentLocale = Locale.getInstance(Locale.GERMAN,Locale.GERMANY);`

**Answer:** B

#### NEW QUESTION 19

Given:

```
public class Foo<K, V> {
    private K key;
    private V value;

    public Foo(K key, V value) { this.key = key; this.value = value; }

    public static <T> Foo<T, T> twice(T value) { return new Foo<T, T>(value, value); }

    public K getKey() { return key; }
    public V getValue() { return value; }
}
```

Which option fails?

- A. `Foo<String, Integer> mark = new Foo<String, Integer> ("Steve", 100);`
- B. `Foo<String, String> pair = Foo.<String>twice ("Hello World!");`
- C. `Foo<Object, Object> percentage = new Foo<String, Integer>("Steve", 100);`
- D. `Foo<String, String> grade = new Foo <> ("John", "A");`

**Answer:** A

#### NEW QUESTION 21

Given the code fragment:

```
public static void main(String[] args) {
    Stream.of("Java", "Unix", "Linux")
        .filter(s -> s.contains("n"))
        .peek(s -> System.out.println("PEEK: " + s))
        // line n1
}
```

Which two code fragments, when inserted at line n1 independently, result in the output PEEK: Unix?

- A. `.anyMatch ();`
- B. `.allMatch ();`
- C. `.findAny ();`
- D. `.noneMatch ();`
- E. `.findFirst ();`

**Answer:** E

#### NEW QUESTION 23

Given:

```
IntStream stream = IntStream.of (1,2,3); IntFunction<Integer> inFu= x -> y -> x*y; //line n1
```

```
IntStream newStream = stream.map(inFu.apply(10)); //line n2 newStream.forEach(System.out::print);
```

Which modification enables the code fragment to compile?

- A. Replace line n1 with: `IntFunction<UnaryOperator> inFu = x -> y -> x*y;`
- B. Replace line n1 with: `IntFunction<IntUnaryOperator> inFu = x -> y -> x*y;`
- C. Replace line n1 with: `BiFunction<IntUnaryOperator> inFu = x -> y -> x*y;`
- D. Replace line n2 with: `IntStream newStream = stream.map(inFu.applyAsInt (10));`

**Answer:** B

#### NEW QUESTION 27

Given:

```
class Bird {
    public void fly () { System.out.print("Can fly"); }
}
```

```
class Penguin extends Bird {
    public void fly () { System.out.print("Cannot fly"); }
}
```

```
and the code fragment: class Birdie {
    public static void main (String [ ] args) { fly( ( ) -> new Bird ( ));
    fly (Penguin : : new);
    }
/* line n1 */
}
```

Which code fragment, when inserted at line n1, enables the Birdie class to compile?

- A. `static void fly (Consumer<Bird> bird) { bird :: fly ();}`
- B. `static void fly (Consumer<? extends Bird> bird) {bird.accept( ) fly ();}`
- C. `static void fly (Supplier<Bird> bird) { bird.get( ) fly ();}`
- D. `static void fly (Supplier<? extends Bird> bird) { LOST`

**Answer:** C

#### NEW QUESTION 32

Given:

```
public class Counter {
    public static void main (String[ ] args) { int a = 10;
    int b = -1;
    assert (b >=1) : "Invalid Denominator"; int = a / b;
    System.out.println (c);
    }
}
```

What is the result of running the code with the `-ea` option?

- A. -10
- B. An `AssertionError` is thrown.
- C. A compilation error occurs.

**Answer:** C

#### NEW QUESTION 34

Given the code fragment:

```
Path path1 = Paths.get("/app/./sys/"); Path res1 = path1.resolve("log");
```

```
Path path2 = Paths.get("/server/exe/"); Path res1 = path1.resolve("/readme/"); System.out.println(res1); System.out.println(res2);
```

What is the result?



- A. /app/sys/log/readme/server/exe
- B. /app/log/sys/server/exe/readme
- C. /app/./sys/log/readme
- D. /app/./sys/log/server/exe/readme

**Answer:** C

#### NEW QUESTION 36

Given the code fragment:

```
public void recDelete (String dirName) throws IOException { File [ ] listOfFiles = new File (dirName) .listFiles();
if (listOfFiles != null && listOfFiles.length >0) {
for (File aFile : listOfFiles) { if (aFile.isDirectory ()) {
recDelete (aFile.getAbsolutePath ());
} else {
if (aFile.getName ().endsWith (".class")) aFile.delete ();
}
}
}
}
```

Assume that Projects contains subdirectories that contain .class files and is passed as an argument to the recDelete () method when it is invoked. What is the result?

- A. The method deletes all the .class files in the Projects directory and its subdirectories.
- B. The method deletes the .class files of the Projects directory only.
- C. The method executes and does not make any changes to the Projects directory.
- D. The method throws an IOException.

**Answer:** A

#### NEW QUESTION 38

Given the definition of the Emp class: public class Emp

```
private String eName; private Integer eAge;
Emp(String eN, Integer eA) { this.eName = eN;
this.eAge = eA;
}
public Integer getEAge () {return eAge;} public String getENAME () {return eName;}
}
```

and code fragment:

```
List<Emp>li = Arrays.asList(new Emp("Sam", 20), New Emp("John", 60), New Emp ("Jim", 51));
Predicate<Emp> agVal = s -> s.getEAge() > 50; //line n1 li = li.stream().filter(agVal).collect(Collectors.toList());
Stream<String> names = li.stream().map.(Emp::getENAME); //line n2 names.forEach(n -> System.out.print(n + " "));
What is the result?
```

- A. Sam John Jim
- B. John Jim
- C. A compilation error occurs at line n1.
- D. A compilation error occurs at line n2.

**Answer:** B

#### NEW QUESTION 42

Given the content of /resources/Message.properties: welcome1="Good day!"

and given the code fragment: Properties prop = new Properties ();

FileInputStream fis = new FileInputStream ("/resources/Message.properties"); prop.load(fis);

System.out.println(prop.getProperty("welcome1")); System.out.println(prop.getProperty("welcome2", "Test")); //line n1

System.out.println(prop.getProperty("welcome3"));

What is the result?

- A. Good day!Testfollowed by an Exception stack trace
- B. Good day!followed by an Exception stack trace
- C. Good day!Test null
- D. A compilation error occurs at line n1.

**Answer:** C

#### NEW QUESTION 43

Given the code fragment: public class Foo {

```
public static void main (String [ ] args) {
```

```
Map<Integer, String> unsortMap = new HashMap< > ( ); unsortMap.put (10, "z");
```

```
unsortMap.put (5, "b");
```

```
unsortMap.put (1, "d");
```

```
unsortMap.put (7, "e");
```

```
unsortMap.put (50, "j");
```

```
Map<Integer, String> treeMap = new TreeMap <Integer, String> (new Comparator<Integer> ( ) {
```

```
@Override public int compare (Integer o1, Integer o2) {return o2.compareTo
```

```
(o1); } } );
```

```
treeMap.putAll (unsortMap);
```

```
for (Map.Entry<Integer, String> entry : treeMap.entrySet ( ) ) { System.out.print (entry.getValue ( ) + " " );
```

```
}
```

```
}  
}
```

What is the result?

- A. A compilation error occurs.
- B. d b e z j
- C. j z e b d
- D. z b d e j

**Answer: C**

#### NEW QUESTION 46

Given the code fragments:

```
4. void doStuff() throws ArithmeticException, NumberFormatException, Exception  
{  
5. if (Math.random() > .1 throw new Exception ("Try again"); 6. }  
and  
24. try {  
25. doStuff ( ):  
26. } catch (ArithmeticException | NumberFormatException | Exception e) {  
27. System.out.println (e.getMessage()); }  
28. catch (Exception e) {  
29. System.out.println (e.getMessage()); }  
30. }
```

Which modification enables the code to print Try again?

- A. Comment the lines 28, 29 and 30.
- B. Replace line 26 with: } catch (Exception | ArithmeticException | NumberFormatException e) {
- C. Replace line 26 with: } catch (ArithmeticException | NumberFormatException e) {
- D. Replace line 27 with: throw e;

**Answer: C**

#### NEW QUESTION 50

Given:

```
class FuelNotAvailException extends Exception { }  
class Vehicle {  
void ride() throws FuelNotAvailException { //line n1 System.out.println("Happy Journey!");  
}  
}  
class SolarVehicle extends Vehicle {  
public void ride () throws Exception { //line n2 super ride ();  
}  
}
```

and the code fragment:

```
public static void main (String[] args) throws FuelNotAvailException, Exception  
{  
Vehicle v = new SolarVehicle (); v.ride();  
}
```

Which modification enables the code fragment to print Happy Journey!?

- A. Replace line n1 with public void ride() throws FuelNotAvailException {
- B. Replace line n1 with protected void ride() throws Exception {
- C. Replace line n2 with void ride() throws Exception {
- D. Replace line n2 with private void ride() throws FuelNotAvailException {

**Answer: B**

#### NEW QUESTION 51

Given the structure of the STUDENT table: Student (id INTEGER, name VARCHAR) Given:

```
public class Test {  
static Connection newConnection =null;  
public static Connection get DBConnection () throws SQLException { try (Connection con = DriverManager.getConnection(URL, username, password)) {  
newConnection = con;  
}  
return newConnection;  
}  
public static void main (String [] args) throws SQLException { get DBConnection ();  
Statement st = newConnection.createStatement(); st.executeUpdate("INSERT INTO student VALUES (102, 'Kelvin')");  
}  
}
```

Assume that:

The required database driver is configured in the classpath.

The appropriate database is accessible with the URL, userName, and passWord exists. The SQL query is valid.

What is the result?

- A. The program executes successfully and the STUDENT table is updated with one record.
- B. The program executes successfully and the STUDENT table is NOT updated with any record.
- C. A SQLException is thrown as runtime.
- D. A NullPointerException is thrown as runtime.

**Answer:** C

#### NEW QUESTION 54

Which statement is true about the DriverManager class?

- A. It returns an instance of Connection.
- B. it executes SQL statements against the database.
- C. It only queries metadata of the database.
- D. it is written by different vendors for their specific database.

**Answer:** A

#### Explanation:

The DriverManager returns an instance of Doctrine\DBAL\Connection which is a wrapper around the underlying driver connection (which is often a PDO instance).

#### NEW QUESTION 58

Given that course.txt is accessible and contains:

Course : : Java

and given the code fragment:

```
public static void main (String[ ] args) { int i;  
char c;  
try (FileInputStream fis = new FileInputStream ("course.txt"); InputStreamReader isr = new InputStreamReader(fis);) { while (isr.ready()) { //line n1  
isr.skip(2);  
i = isr.read (); c = (char) i;  
System.out.print(c);  
}  
} catch (Exception e) { e.printStackTrace();  
}  
}
```

What is the result?

- A. ur :: va
- B. ueJa
- C. The program prints nothing.
- D. A compilation error occurs at line n1.

**Answer:** B

#### NEW QUESTION 63

Given:

```
interface P { public void method1(); }  
  
interface Q extends P { public void method1(); }  
  
interface R extends P { public void method2(); }  
  
interface S { public default void method() { } }  
  
interface T { public void method1(); public void method2(); }  
  
interface U { public void method1(); public abstract void method2(); }
```

Which two interfaces can you use to create lambda expressions? (Choose two.)

- A. T
- B. R
- C. P
- D. S
- E. Q
- F. U

**Answer:** AF

#### NEW QUESTION 64

Given:

```
class Counter extends Thread {
    int i = 10;
    public synchronized void display(Counter obj) {
        try {
            Thread.sleep(5);
            obj.increment(this);
            System.out.println(i);
        } catch (InterruptedException ex) { }
    }
    public synchronized void increment (Counter obj) {
        i++;
    }
}

public class Test {
    public static void main(String[] args) {
        final Counter obj1 = new Counter();
        final Counter obj2 = new Counter();
        new Thread(new Runnable() {
            public void run() {obj1.display(obj2);
            }
        }).start();
        new Thread(new Runnable() {
            public void run() { obj2.display(obj1); }
        }).start();
    }
}
```

From what threading problem does the program suffer?

- A. race condition
- B. deadlock
- C. starvation
- D. livelock

**Answer: B**

#### NEW QUESTION 69

In 2015, daylight saving time in New York, USA, begins on March 8th at 2:00 AM. As a result, 2:00 AM becomes 3:00 AM. Given the code fragment:

```
ZoneId zone = ZoneId.of("America/New_York");
ZonedDateTime dt = ZonedDateTime.of(LocalDate.of(2015, 3, 8), LocalTime.of(1, 0),
zone);
ZonedDateTime dt2 = dt.plusHours(2);
System.out.print(DateTimeFormatter.ofPattern("H:mm - ").format(dt2));
System.out.println("difference: " + ChronoUnit.HOURS.between(dt, dt2));
```

Which is the result?

- A. 3:00 – difference: 2
- B. 2:00 – difference: 1
- C. 4:00 – difference: 3
- D. 4:00 – difference: 2

**Answer: B**

#### NEW QUESTION 73

Given the code fragment:

ZonedDateTime depart = ZonedDateTime.of(2015, 1, 15, 3, 0, 0, 0, ZoneID.of("UTC-



```
7"));
ZonedDateTime arrive = ZonedDateTime.of(2015, 1, 15, 9, 0, 0, 0, ZoneID.of("UTC-
5"));
long hrs = ChronoUnit.HOURS.between(depart, arrive); //line n1 System.out.println("Travel time is" + hrs + "hours");
What is the result?
```

- A. Travel time is 4 hours
- B. Travel time is 6 hours
- C. Travel time is 8 hours
- D. An exception is thrown at line n1.

**Answer:** A

#### NEW QUESTION 77

Given that data.txt and alldata.txt are accessible, and the code fragment:

```
public void writeFiles() throws IOException {
    BufferedReader br = new BufferedReader(new FileReader("data.txt"));
    BufferedWriter bw = new BufferedWriter(new FileWriter("alldata.txt"));
    String line = null;
    while ((line = br.readLine()) != null) {
        bw.append(line + "\n");
    }
    // line n1
}
```

What is required at line n1 to enable the code to overwrite alldata.txt with data.txt?

- A. br.close();
- B. bw.writeLn();
- C. br.flush();
- D. bw.flush();

**Answer:** D

#### NEW QUESTION 82

Given:

```
interface Interfacel {
    public default void sayHi() {
        System.out.println("Hi Interface-1");
    }
}

interface Interface2 {
    public default void sayHi() {
        System.out.println("Hi Interface-2");
    }
}

public class MyClass implements Interfacel, Interface2 {
    public static void main(String[] args) {
        Interfacel obj = new MyClass();
        obj.sayHi();
    }
    public void sayHi() {
        System.out.println("Hi MyClass");
    }
}
```

What is the result?

- A. Hi Interface-2
- B. A compilation error occurs.
- C. Hi Interface-1
- D. Hi MyClass

**Answer:** D

#### NEW QUESTION 83

Given:



```
class Student {
    String course, name, city;
    public Student(String name, String course, String city) {
        this.course = course; this.name = name; this.city = city;
    }
    public String toString() {
        return course + ":" + name + ":" + city;
    }
    public String getCourse() { return course; }
    public String getName() { return name; }
    public String getCity() { return city; }
}
```

and the code fragment:

```
List<Student> stds = Arrays.asList(
    new Student ("Jessy", "Java ME", "Chicago"),
    new Student ("Helen", "Java EE", "Houston"),
    new Student ("Mark", "Java ME", "Chicago"));
stds.stream()
    .collect(Collectors.groupingBy(Student::getCourse))
    .forEach(src, res) -> System.out.println(src));
```

What is the result?

- A. [Java EE: Helen:Houston][Java ME: Jessy:Chicago, Java ME: Mark:Chicago]
- B. Java EEJava ME
- C. [Java ME: Jessy:Chicago, Java ME: Mark:Chicago] [Java EE: Helen:Houston]
- D. A compilation error occurs.

**Answer:** D

#### NEW QUESTION 86

Given:

```
public interface LengthValidator {
    public boolean checkLength(String str);
}
```

and

```
public class Txt {
    public static void main(String[] args) {
        boolean res = new LengthValidator() {
            public boolean checkLength(String str) {
                return str.length() > 5 && str.length() < 10;
            }
        }.checkLength("Hello");
    }
}
```

Which interface from the java.util.function package should you use to refactor the class Txt?

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

**Answer:** C

#### NEW QUESTION 91

Given:

```
class Student {  
    String course, name, city;  
    public Student (String name, String course, String city) { this.course = course; this.name = name; this.city = city;  
    }  
    public String toString() {  
        return course + ":" + name + ":" + city;  
    }  
}
```

and the code fragment: `List<Student> stds = Arrays.asList(  
 new Student ("Jessy", "Java ME", "Chicago"), new Student ("Helen", "Java EE", "Houston"), new Student ("Mark", "Java ME", "Chicago")); stds.stream()  
 .collect(Collectors.groupingBy(Student::getCourse))  
 .forEach(src, res) -> System.out.println(src));` What is the result?

- A. [Java EE: Helen:Houston][Java ME: Jessy:Chicago, Java ME: Mark:Chicago]
- B. Java EEJava ME
- C. [Java ME: Jessy:Chicago, Java ME: Mark:Chicago] [Java EE: Helen:Houston]
- D. A compilation error occurs.

**Answer: B**

#### NEW QUESTION 92

Given the code fragment:

```
LocalDate valentinesDay = LocalDate.of(2015, Month.FEBRUARY, 14);  
LocalDate nextYear = valentinesDay.plusYears(1);  
nextYear.plusDays(15); //line n1  
System.out.println(nextYear);
```

 What is the result?

- A. 2016-02-14
- B. A `DateTimeException` is thrown
- C. 2016-02-29
- D. A compilation error occurs at line n1.

**Answer: A**

#### NEW QUESTION 96

Given the code fragment:

```
//line n1  
Double d = str.average().getAsDouble();  
System.out.println("Average = " + d);
```

Which should be inserted into line n1 to print `Average = 2.5`?

- A. `IntStream str = Stream.of (1, 2, 3, 4);`
- B. `IntStream str = IntStream.of (1, 2, 3, 4);`
- C. `DoubleStream str = Stream.of (1.0, 2.0, 3.0, 4.0);`
- D. `Stream str = Stream.of (1, 2, 3, 4);`

**Answer: C**

#### NEW QUESTION 98

Given the code fragment:

```
public static void main(String[] args) {  
    Console console = System.console();  
    char[] pass = console.readPassword("Enter password:"); // line n1  
    String password = new String(pass); // line n2  
}
```

What is the result?

- A. A compilation error occurs at line n1.
- B. A compilation error occurs at line n2.
- C. The code reads the password without echoing characters on the console.
- D. A compilation error occurs because the `IOException` isn't declared to be thrown or caught?

**Answer: D**

#### NEW QUESTION 102

Given:

Item table

- ID, INTEGER: PK
- DESCRIP, VARCHAR(100)
- PRICE, REAL
- QUANTITY< INTEGER

And given the code fragment:

```
9. try {  
10. Connection conn = DriverManager.getConnection(dbURL, username, password);  
11. String query = "Select * FROM Item WHERE ID = 110";  
12. Statement stmt = conn.createStatement();
```

```
13. ResultSet rs = stmt.executeQuery(query);
14. while(rs.next()) {
15. System.out.println("ID: " + rs.getInt("Id"));
16. System.out.println("Description: " + rs.getString("Descrip"));
17. System.out.println("Price: " + rs.getDouble("Price"));
18. System.out.println("Quantity: " + rs.getInt("Quantity"));
19. }
20. } catch (SQLException se) {
21. System.out.println("Error");
22. }
```

Assume that:

The required database driver is configured in the classpath.

The appropriate database is accessible with the dbURL, userName, and passWord exists. The SQL query is valid.

What is the result?

- A. An exception is thrown at runtime.
- B. Compilation fails.
- C. The code prints Error.
- D. The code prints information about Item 110.

**Answer: D**

#### NEW QUESTION 104

Given the code fragment:

```
Deque<String> queue = new ArrayDeque<>();
queue.add("Susan");
queue.add("Allen");
queue.add("David");
System.out.println(queue.pop());
System.out.println(queue.remove());
System.out.println(queue);
```

What is the result?

- A. DavidDavid[Susan, Allen]
- B. SusanSusan[Susan, Allen]
- C. SusanAllen [David]
- D. DavidAllen [Susan]
- E. SusanAllen[Susan, David]

**Answer: C**

#### NEW QUESTION 107

Given the code fragment:

```
// Login time:2015-01-12T21:58:18.817Z
Instant loginTime = Instant.now();
Thread.sleep(1000);

// Logout time:2015-01-12T21:58:19.880Z
Instant logoutTime = Instant.now();

loginTime = loginTime.truncatedTo(ChronoUnit.MINUTES); // line n1
logoutTime = logoutTime.truncatedTo(ChronoUnit.MINUTES);

if (logoutTime.isAfter(loginTime))
    System.out.println("Logged out at:"+logoutTime);
else
    System.out.println("Can't logout");
```

What is the result?

- A. A compilation error occurs at line n1.
- B. Logged out at: 2015-01-12T21:58:19.880Z
- C. Can't logout
- D. Logged out at: 2015-01-12T21:58:00Z

**Answer: D**

#### NEW QUESTION 108



Given:

```
class Block {
    String color;
    int size;
    Block(int size, String color) {
        this.size = size;
        this.color = color;
    }
}
```

and the code fragment:

```
List<Block> blocks = new ArrayList<>();
blocks.add(new Block(10, "Green"));
blocks.add(new Block(7, "Red"));
blocks.add(new Block(12, "Blue"));
Collections.sort(blocks, new ColorSorter());
```

Which definition of the ColorSorter class sorts the blocks list?

- A. 

```
class ColorSorter implements Comparable<Block> {
    public boolean compare(Block o1, Block o2) {
        return o1.color.equals(o2.color);
    }
}
```
- B. 

```
class ColorSorter implements Comparable<Block> {
    public int compareTo(Block o1, Block o2) {
        return o1.color.compareTo(o2.color);
    }
}
```
- C. 

```
class ColorSorter implements Comparator<Block> {
    public int compare(Block o1, Block o2) {
        return o1.color.compareTo(o2.color);
    }
}
```
- D. 

```
class ColorSorter implements Comparator<Block> {
    public boolean compare(Block o1, Block o2) {
        return o1.color.compareTo(o2.color);
    }
}
```

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

**Answer:** B

#### NEW QUESTION 109

Given the code fragment:

```
Map<Integer, Integer> mVal = new HashMap<>();
mVal.put(1, 10);
mVal.put(2, 20);
//line n1
c.accept(1, 2);
mVal.forEach(c);
```

Which statement can be inserted into line n1 to print 1,2; 1,10; 2,20;?

- A. BiConsumer<Integer,Integer> c = (i, j) -> {System.out.print (i + "," + j+ "; ");};
- B. BiFunction<Integer, Integer, String> c = (i, j) -> {System.out.print (i + "," + j+ "; ");};
- C. BiConsumer<Integer, Integer, String> c = (i, j) -> {System.out.print (i + "," + j+ "; ");};
- D. BiConsumer<Integer, Integer, Integer> c = (i, j) -> {System.out.print (i + "," + j+ "; ");};

**Answer: B**

#### NEW QUESTION 112

Given:

```
public class StrMan {
    public static void doStuff(String s) {
        try {
            if (s == null) {
                throw new NullPointerException();
            }
        } finally {
            System.out.println("-finally-");
        }
        System.out.println("-doStuff-");
    }
    public static void main (String[] args) {
        try {
            doStuff(null);
        } catch (NullPointerException npe) {
            System.out.println("-catch-");
        }
    }
}
```

What is the result?

- A. -catch--finally--dostuff-
- B. -catch-
- C. -finally--catch-
- D. -finally-dostuff--catch-

**Answer: C**

#### NEW QUESTION 113

Given the code fragment:

```
Map<Integer, String> books = new TreeMap<>(); books.put (1007, "A");
books.put (1002, "C");
books.put (1001, "B");
books.put (1003, "B"); System.out.println (books); What is the result?
```

- A. {1007 = A, 1002 = C, 1001 = B, 1003 = B}
- B. {1001 = B, 1002 = C, 1003 = B, 1007 = A}
- C. {1002 = C, 1003 = B, 1007 = A}
- D. {1007 = A, 1001 = B, 1003 = B, 1002 = C}

**Answer: B**

#### NEW QUESTION 115

Given the records from the STUDENT table:



sid	sname	semail
111	James	james@uni.com
112	Jane	jane@uni.com
114	John	john@uni.com

Given the code fragment:

```
public static void main(String[] args) throws SQLException {
    //code to load and register valid jdbc driver go here
    Connection con = DriverManager.getConnection(URL, username, password);
    Statement st = con.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,
                                       ResultSet.CONCUR_UPDATABLE);

    st.execute("SELECT * FROM student");
    ResultSet rs = st.getResultSet();
    rs.absolute(3);
    rs.moveToInsertRow();
    rs.updateInt(1, 113);
    rs.updateString(2, "Jannet");
    rs.updateString(3, "jannet@uni.com");
    rs.updateRow();
    rs.refreshRow();
    System.out.println(rs.getInt(1) + " : " + rs.getString(2) + " : " + rs.getString
(3));
}
```

Assume that the URL, username, and password are valid. What is the result?

- A. The STUDENT table is not updated and the program prints: 114 : John : john@uni.com
- B. The STUDENT table is updated with the record: 113 : Jannet : jannet@uni.com and the program prints: 114 : John : john@uni.com
- C. The STUDENT table is updated with the record: 113 : Jannet : jannet@uni.com and the program prints: 113 : Jannet : jannet@uni.com
- D. A SQLException is thrown at run time.

**Answer: A**

#### NEW QUESTION 117

Given:

```
class Engine {
    double fuelLevel;
    Engine(int fuelLevel) { this.fuelLevel = fuelLevel; }
    public void start() {
        // line n1
        System.out.println("Started");
    }
    public void stop() { System.out.println("Stopped"); }
}
```

Your design requires that:

- ☒ fuelLevel of Engine must be greater than zero when the start() method is invoked.
- ☒ The code must terminate if fuelLevel of Engine is less than or equal to zero.

Which code fragment should be added at line n1 to express this invariant condition?

- A. `assert (fuelLevel) : "Terminating...";`
- B. `assert (fuelLevel > 0) : System.out.println ("Impossible fuel");`
- C. `assert fuelLevel < 0: System.exit(0);`
- D. `assert fuelLevel > 0: "Impossible fuel" ;`

**Answer: C**

#### NEW QUESTION 121

Given the code fragment:

```
List<Integer> list1 = Arrays.asList(10, 20); List<Integer> list2 = Arrays.asList(15, 30);
//line n1
```

Which code fragment, when inserted at line n1, prints 10 20 15 30?

- A. `Stream.of(list1, list2).flatMap(list -> list.stream()).forEach(s -> System.out.print(s + " "));`

B. `Stream.of(list1, list2).flatMap(list -> list.intStream()).forEach(s -> System.out.print(s + " "));`  
C. `list1.stream().flatMap(list2.stream().flatMap(e1 -> e1.stream()).forEach(s -> System.out.println(s + " "));`  
D. `Stream.of(list1, list2).flatMapToInt(list -> list.stream()).forEach(s -> System.out.print(s + " "));`

**Answer:** A

#### NEW QUESTION 126

Given:

```
final class Folder { //line n1
//line n2
public void open () { System.out.print("Open");
}
}
public class Test {
public static void main (String [] args) throws Exception { try (Folder f = new Folder()) {
```

A. `f.open();}}` Which two modifications enable the code to print Open Close? (Choose two.)  
B. Replace line n1 with: `class Folder implements AutoCloseable {`  
C. Replace line n1 with: `class Folder extends Closeable {`  
D. Replace line n1 with: `class Folder extends Exception {`  
E. At line n2, insert: `final void close () {System.out.print("Close");}`  
F. At line n2, insert: `public void close () throws IOException { System.out.print("Close");}`

**Answer:** AE

#### NEW QUESTION 131

Given the code fragment: `UnaryOperator<Integer> uo1 = s -> s*2; line n1`  
`List<Double> loanValues = Arrays.asList(1000.0, 2000.0); loanValues.stream()`  
`.filter(lv -> lv >= 1500)`  
`.map(lv -> uo1.apply(lv))`  
`.forEach(s -> System.out.print(s + " "));` What is the result?

A. 4000.0  
B. 4000  
C. A compilation error occurs at line n1.  
D. A compilation error occurs at line n2.

**Answer:** D

#### NEW QUESTION 135

Given:

```
class Product {
    String name;
    int qty;
    public String toString(){
        return name;
    }
    public Product(String name, int qty) {
        this.name = name;
        this.qty = qty;
    }
    static class ProductFilter {
        public boolean isAvailable(Product p) {    // line n1
            return p.qty >= 10;
        }
    }
}
```

and the code fragment:

```
List<Product> products = Arrays.asList(
    new Product("MotherBoard", 5),
    new Product("Speaker", 20));
products.stream()
    .filter(Product.ProductFilter::isAvailable) // line n2
    .forEach(p -> System.out.println(p));
```

Which modification enables the code fragment to print Speaker?

- A. Implement Predicate in the Product.ProductFilter class and replace line n2 with .filter (p-> p.ProductFilter.test (p))
- B. Replace line n1 with:public static boolean isAvailable (Product p) {
- C. Replace line n2 with:.filter (p -> p.ProductFilter: :isAvailable (p))
- D. Replace line n2 with:.filter (p -> Product: :ProductFilter: :isAvailable ())



**Answer: B**

#### NEW QUESTION 136

Given the code fragment:

```
try {
    Properties prop = new Properties();
    prop.put("user", userName);
    prop.put("password", passWord);
    Connection conn = DriverManager.getConnection(dbURL, prop);
    if(conn != null){
        System.out.print("Connection Established");
    }
} catch (Exception e) {
    System.out.print(e);
}
```

and the information:

-  The required database driver is configured in the classpath.
-  The appropriate database is accessible with the dbURL, username, and passWord exists. What is the result?

- A. A ClassNotFoundException is thrown at runtime.
- B. The program prints nothing.
- C. The program prints Connection Established.
- D. A SQLException is thrown at runtime.

**Answer: C**

#### NEW QUESTION 140

Given:

```
public class product { int id; int price;
public Product (int id, int price) { this.id = id;
this.price = price;
}
public String toString() { return id + ":" + price; }
}
```

and the code fragment:

```
List<Product> products = Arrays.asList(new Product(1, 10), new Product (2, 30),
new Product (2, 30));
Product p = products.stream().reduce(new Product (4, 0), (p1, p2) -> { p1.price+=p2.price;
return new Product (p1.id, p1.price);}); products.add(p); products.stream().parallel()
.reduce((p1, p2) -> p1.price > p2.price ? p1 : p2)
.i fPresent(System.out: :println); What is the result?
```

- A. 2 : 30
- B. 4 : 0
- C. 4 : 60
- D. 4 : 602 : 303 : 201 : 10
- E. The program prints nothing.

**Answer: C**

#### NEW QUESTION 144

Given the code fragment:

```
List<String> nums = Arrays.asList("EE", "SE");
String ans = nums
    .parallelStream()
    .reduce("Java ", (a, b) -> a.concat(b));
System.out.print(ans);
```

What is the result?

- A. Java EEJava EESE
- B. Java EESE
- C. The program prints either:Java EEJava SE orJava SEJava EE



D. Java EEJava SE

**Answer:** D

#### NEW QUESTION 147

Given the content of Operator.java, EngineOperator.java, and Engine.java files:

```
Operator.java:
public abstract class Operator {
    protected void turnON();
    protected void turnOFF();
}

EngineOperator.java:
public class EngineOperator extends Operator{
    public final void turnON() { System.out.print("ON "); }
    public final void turnOFF() { System.out.println("OFF"); }
}

Engine.java:
public class Engine{
    Operator m = new EngineOperator();
    public void operate() {
        m.turnON();
        m.turnOFF();
    }
}
```

and the code fragment:

```
Engine carEngine = new Engine();
carEngine.operate();
```

What is the result?

- A. The Engine.java file fails to compile.
- B. The EngineOperator.java file fails to compile.
- C. The Operator.java file fails to compile.
- D. ON OFF

**Answer:** A

#### NEW QUESTION 150

Given that version.txt is accessible and contains: 1234567890

and given the code fragment:

```
try (FileInputStream fis = new FileInputStream("version.txt");
     InputStreamReader isr = new InputStreamReader(fis);
     BufferedReader br = new BufferedReader(isr);) {
    if (br.markSupported()) {
        System.out.print((char) br.read());
        br.mark(2);
        System.out.print((char) br.read());
        br.reset();
        System.out.print((char) br.read());
    }
} catch (Exception e) {
    e.printStackTrace();
}
```

What is the result?

- A. 121
- B. 122

- C. 135
- D. The program prints nothing.

**Answer:** B

#### NEW QUESTION 154

Given the code fragments :

```
public class Product {
    String name;
    Integer price;
    Product(String name, Integer price) {
        this.name = name;
        this.price = price;
    }
    public void printVal(){ System.out.print(name + " Price:" + price + " "); }
    public void setPrice(int price) { this.price = price; }
    public Integer getPrice() { return price; }
}
```

and

```
List<Product> li = Arrays.asList(new Product("TV", 1000), new Product("Refrigerator",
2000));
Consumer<Product> raise = e -> e.setPrice(e.getPrice() + 100);
li.forEach(raise);
li.stream().forEach(Product::printVal);
```

What is the result?

- A. TV Price :110 Refrigerator Price :2100
- B. A compilation error occurs.
- C. TV Price :1000 Refrigerator Price :2000
- D. The program prints nothing.

**Answer:** C

#### NEW QUESTION 158

Given the code fragment:

```
class CallerThread implements Callable<String> { String str;
public CallerThread(String s) {this.str=s;} public String call() throws Exception { return str.concat("Call");
}
}
```

and

```
public static void main (String[] args) throws InterruptedException, ExecutionException
{
    ExecutorService es = Executors.newFixedThreadPool(4); //line n1 Future f1 = es.submit (newCallerThread("Call"));
    String str = f1.get().toString(); System.out.println(str);
}
```

Which statement is true?

- A. The program prints Call Call and terminates.
- B. The program prints Call Call and does not terminate.
- C. A compilation error occurs at line n1.
- D. An ExecutionException is thrown at run time.

**Answer:** B

#### NEW QUESTION 160

Given the definition of the Employee class:



```
class Employee {
    String dept, name;
    public Employee(String d, String n) {
        dept = d;
        name = n;
    }
    public String toString() {
        return getDept() + ":" + getName();
    }
    public String getDept() { return dept; }
    public String getName() { return name; }
}
```

and this code fragment:

```
List<Employee> emps = Arrays.asList(new Employee("sales", "Ada"),
    new Employee("sales", "Bob"),
    new Employee("hr", "Bob"),
    new Employee("hr", "Eva"));
Stream<Employee> s = emps.stream()
    .sorted(Comparator.comparing((Employee e) -> e.getDept())
        .thenComparing((Employee e) -> e.getName()));
List<Employee> eSorted = s.collect(Collectors.toList());
System.out.println(eSorted);
```

What is the result?

- A. [sales:Ada, hr:Bob, sales:Bob, hr:Eva]
- B. [Ada:sales, Bob:sales, Bob:hr, Eva:hr]
- C. [hr:Eva, hr:Bob, sales:Bob, sales:Ada]
- D. [hr:Bob, hr:Eva, sales:Ada, sales:Bob]

**Answer: A**

#### NEW QUESTION 162

Given: Book.java:

```
public class Book {
    private String read(String bname) { return "Read" + bname }
}
```

EBook.java:

```
public class EBook extends Book {
    public String read (String url) { return "View" + url }
}
```

Test.java:

```
public class Test {
    public static void main (String[] args) { Book b1 = new Book();
        b1.read("Java Programing"); Book b2 = new EBook();
        b2.read("http://ebook.com/ebook");
    }
}
```

What is the result?

- A. Read Java Programming View http:// ebook.com/ebook
- B. Read Java Programming Read http:// ebook.com/ebook
- C. The EBook.java file fails to compile.
- D. The Test.java file fails to compile.

**Answer: D**

#### NEW QUESTION 163

Which two methods from the java.util.stream.Stream interface perform a reduction operation? (Choose two.)

- A. count ()
- B. collect ()
- C. distinct ()
- D. peek ()

E. filter ()

**Answer:** AB

#### NEW QUESTION 164

Given the code fragment:

```
public static void main (String [ ] args) throws IOException {  
    BufferedReader br = new BufferedReader (new InputStremReader (System.in)); System.out.print ("Enter GDP: ");  
    //line 1  
}
```

Which code fragment, when inserted at line 1, enables the code to read the GDP from the user?

- A. int GDP = Integer.parseInt (br.readline());
- B. int GDP = br.read();
- C. int GDP = br.nextInt();
- D. int GDP = Integer.parseInt (br.next());

**Answer:** A

#### NEW QUESTION 166

Given:

```
class Worker extends Thread { CyclicBarrier cb;  
    public Worker(CyclicBarrier cb) { this.cb = cb; } public void run () {  
        try { cb.await();  
            System.out.println("Worker...");  
        } catch (Exception ex) { }  
    }  
}  
class Master implements Runnable { //line n1 public void run () { System.out.println("Master...");  
    }  
}
```

and the code fragment:

```
Master master = new Master();  
//line n2  
Worker worker = new Worker(cb); worker.start();
```

You have been asked to ensure that the run methods of both the Worker and Master classes are executed. Which modification meets the requirement?

- A. At line n2, insert CyclicBarrier cb = new CyclicBarrier(2, master);
- B. Replace line n1 with class Master extends Thread {
- C. At line n2, insert CyclicBarrier cb = new CyclicBarrier(1, master);
- D. At line n2, insert CyclicBarrier cb = new CyclicBarrier(master);

**Answer:** C

#### NEW QUESTION 171

Given the code fragments:

```
class Employee { Optional<Address> address;  
    Employee (Optional<Address> address) { this.address = address;  
    }  
    public Optional<Address> getAddress() { return address; }  
}  
class Address {  
    String city = "New York";  
    public String getCity { return city; } public String toString() {  
        return city;  
    }  
}  
and  
Address address = null;  
Optional<Address> addrs1 = Optional.ofNullable (address);  
Employee e1 = new Employee (addrs1);  
String eAddress = (addrs1.isPresent()) ? addrs1.get().getCity() : "City Not available";  
What is the result?
```

- A. New York
- B. City Not available
- C. null
- D. A NoSuchElementException is thrown at run time.

**Answer:** B

#### NEW QUESTION 172

Given:

```
class DataConverter {  
    public void copyFlatFilesToTables() { }  
    public void close() throws Exception {  
        throw new RuntimeException(); // line n1  
    }  
}
```

and the code fragment:

```
public static void main(String[] args) throws Exception {  
    try (DataConverter dc = new DataConverter()) // line n2  
    { dc.copyFlatFilesToTables(); }  
}
```

What is the result?

- A. A compilation error occurs at line n2.
- B. A compilation error occurs because the try block doesn't have a catch or finally block.
- C. A compilation error occurs at line n1.
- D. The program compiles successfully.

**Answer:** B

#### NEW QUESTION 174

Given the code fragments:

```
public static Optional<String> getCountry(String loc) {  
    Optional<String> couName = Optional.empty();  
    if ("Paris".equals(loc))  
        couName = Optional.of("France");  
    else if ("Mumbai".equals(loc))  
        couName = Optional.of("India");  
    return couName;  
}
```

and

```
Optional<String> city1 = getCountry("Paris");  
Optional<String> city2 = getCountry("Las Vegas");  
System.out.println(city1.orElse("Not Found"));  
if (city2.isPresent())  
    city2.ifPresent(x -> System.out.println(x));  
else  
    System.out.println(city2.orElse("Not Found"));
```

What is the result?

- A. FranceOptional[NotFound]
- B. Optional [France] Optional [NotFound]
- C. Optional[France] Not Found
- D. FranceNot Found

**Answer:** D

#### NEW QUESTION 179

Given the code fragment:

```
List<Integer> li = Arrays.asList(10, 20, 30);  
Function<Integer, Integer> fn = f1 -> f1 + f1;  
Consumer<Integer> conVal = s -> System.out.print("Val:" + s + " ");  
li.stream().map(fn).forEach(conVal);
```

What is the result?

- A. Val:20 Val:40 Val:60
- B. Val:10 Val:20 Val:30

C. A compilation error occurs.  
D. Val: Val: Val:

**Answer:** B

**NEW QUESTION 180**

.....

## 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

### 1Z0-809 Practice Exam Features:

- \* 1Z0-809 Questions and Answers Updated Frequently
- \* 1Z0-809 Practice Questions Verified by Expert Senior Certified Staff
- \* 1Z0-809 Most Realistic Questions that Guarantee you a Pass on Your First Try
- \* 1Z0-809 Practice Test Questions in Multiple Choice Formats and Updates for 1 Year

**100% Actual & Verified — Instant Download, Please Click**  
**[Order The 1Z0-809 Practice Test Here](#)**