



**Oracle**

## **Exam Questions 1Z0-809**

Java SE 8 Programmer II

### NEW QUESTION 1

Given:

```
class Book { int id;
String name;
public Book (int id, String name) { this.id = id;
this.name = name;
}
public boolean equals (Object obj) { //line n1 boolean output = false;
Book b = (Book) obj;
if (this.name.equals(b.name)) output = true;
}
return output;
}
}
```

and the code fragment:

Book b1 = new Book (101, "Java Programing"); Book b2 = new Book (102, "Java Programing"); System.out.println (b1.equals(b2)); //line n2 Which statement is true?

- A. The program prints true.
- B. The program prints false.
- C. A compilation error occur
- D. To ensure successful compilation, replace line n1 with: boolean equals (Book obj) {
- E. A compilation error occur
- F. To ensure successful compilation, replace line n2 with: System.out.println (b1.equals((Object) b2));

**Answer:** A

### NEW QUESTION 2

Given the code fragment:

```
public static void main (String[] args) throws IOException { BufferedReader brCopy = null;
try (BufferedReader br = new BufferedReader (new FileReader("employee.txt")))
{ // line n1
br.lines().forEach(c -> System.out.println(c)); brCopy = br; //line n2
}
brCopy.ready(); //line n3;
}
```

Assume that the ready method of the BufferedReader, when called on a closed BufferedReader, throws an exception, and employee.txt is accessible and contains valid text.

What is the result?

- A. A compilation error occurs at line n3.
- B. A compilation error occurs at line n1.
- C. A compilation error occurs at line n2.
- D. The code prints the content of the employee.txt file and throws an exception at line n3.

**Answer:** D

### NEW QUESTION 3

Given the content of the employee.txt file: Every worker is a master.

Given that the employee.txt file is accessible and the file allemp.txt does NOT exist, and the code fragment:

```
try {
    List<String> content = Files.readAllLines (Paths.get ("employee.txt"));
    content.stream().forEach(line -> {
        try {
            Files.write(
                Paths.get("allemp.txt"),
                line.getBytes(),
                StandardOpenOption.APPEND
            );
        } catch (IOException e) { System.out.println("Exception 1"); }
    });
} catch (IOException e) { System.out.println("Exception 2"); }
```

What is the result?

- A. Exception 1
- B. Exception 2
- C. The program executes, does NOT affect the system, and produces NO output.
- D. allemp.txt is created and the content of employee.txt is copied to it.

**Answer:** A

### NEW QUESTION 4

Which code fragment is required to load a JDBC 3.0 driver?

- A. Connection con = Connection.getDriver ("jdbc:xyzdata://localhost:3306/EmployeeDB");
- B. Class.forName("org.xyzdata.jdbc.NetworkDriver");
- C. Connection con = DriverManager.getConnection ("jdbc:xyzdata://localhost:3306/EmployeeDB");
- D. DriverManager.loadDriver ("org.xyzdata.jdbc.NetworkDriver");

**Answer:** B

#### NEW QUESTION 5

Which two statements are true about the Fork/Join Framework? (Choose two.)

- A. The RecursiveTask subclass is used when a task does not need to return a result.
- B. The Fork/Join framework can help you take advantage of multicore hardware.
- C. The Fork/Join framework implements a work-stealing algorithm.
- D. The Fork/Join solution when run on multicore hardware always performs faster than standard sequential solution.

**Answer:** AC

#### 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> words = Arrays.asList("win", "try", "best", "luck", "do");
Predicate<String> test1 = w -> {
    System.out.println("Checking...");
    return w.equals("do"); // line n1
};
Predicate test2 = (String w) -> w.length() > 3; // line n2
words.stream()
    .filter(test2)
    .filter(test1)
    .count();
```

What is the result?

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

**Answer:** A

#### NEW QUESTION 8

Given the code fragment:

```
ProductCode<Number, Integer> c1 = new ProductCode<Number, Integer>(); /* c1
instantiation */
ProductCode<Number, String> c2 = new ProductCode<Number, String>(); /* c2
instantiation */
```

You have been asked to define the ProductCode class. The definition of the ProductCode class must allow c1 instantiation to succeed and cause a compilation error on c2 instantiation.

Which definition of ProductCode meets the requirement?

```
A. class ProductCode<T, S<Integer>> {
    T c1;
    S c2;
}

B. class ProductCode<T, S extends T> {
    T c1;
    S c2;
}

C. class ProductCode<T, S> {
    T c1;
    S c2;
}

D. class ProductCode<T, S super T> {
    T c1;
    S c2;
}
```

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

**Answer:** B

#### NEW QUESTION 9

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 10

Given the code fragment:

```
Path p1 = Paths.get("/Pics/MyPic.jpeg"); System.out.println (p1.getNameCount() + ":" + p1.getName(1) +
":" + p1.getFileName());
```

Assume that the Pics directory does NOT exist.

What is the result?

- A. An exception is thrown at run time.
- B. 2:MyPic.jpeg: MyPic.jpeg
- C. 1:Pics:/Pics/ MyPic.jpeg
- D. 2:Pics: MyPic.jpeg

**Answer:** B

#### NEW QUESTION 10

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 13

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 17

Given:

```
public class Customer { private String fName; private String lName; private static int count;
public customer (String first, String last) {fName = first, lName = last;
++count;}
static { count = 0; }
public static int getCount() {return count; }
}
public class App {
public static void main (String [] args) { Customer c1 = new Customer("Larry", "Smith");
Customer c2 = new Customer("Pedro", "Gonzales"); Customer c3 = new Customer("Penny", "Jones"); Customer c4 = new Customer("Lars", "Svenson"); c4 =
null;
c3 = c2;
System.out.println (Customer.getCount());
}
}
```

What is the result?

- A. 2
- B. 3
- C. 4
- D. 5

**Answer:** D

#### NEW QUESTION 18

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 22

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 26

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 29

What is true about the java.sql.Statement interface?

- A. It provides a session with the database.
- B. It is used to get an instance of a Connection object by using JDBC drivers.
- C. It provides a cursor to fetch the resulting data.
- D. It provides a class for executing SQL statements and returning the results.

**Answer:** D

#### NEW QUESTION 33

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 36

Given the code fragment:

```
Connection con = null;
try {
    // line n1
    if(con != null){
        System.out.print("Connection Established.");
    }

} catch (Exception e) {
    System.out.print(e);
}
```

Assume that dbURL, userName, and password are valid.

Which code fragment can be inserted at line n1 to enable the code to print Connection Established?

- A. Properties prop = new Properties(); prop.put ("user", userName); prop.put ("password", password);con = DriverManager.getConnection (dbURL, prop);
- B. con = DriverManager.getConnection (userName, password, dbURL);
- C. Properties prop = new Properties(); prop.put ("userid", userName); prop.put ("password", password); prop.put("url", dbURL);con = DriverManager.getConnection (prop);
- D. con = DriverManager.getConnection (dbURL); con.setClientInfo ("user", userName); con.setClientInfo ("password", password);

**Answer:** A

#### NEW QUESTION 37

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

ID	NAME
102	Edwin
103	Edward
103	Edwin

Given the code fragment:

```
Connection conn = DriverManager.getConnection(dbURL, userName, passWord);
Statement st = conn.createStatement();
String query = "DELETE FROM Student WHERE id = 103";
System.out.println("Status: " + st.execute(query));
```

Assume that:

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. The program prints Status: true and two records are deleted from the Student table.
- B. The program prints Status: false and two records are deleted from the Student table.
- C. A SQLException is thrown at runtime.
- D. The program prints Status: false but the records from the Student table are not deleted.

**Answer: B**

#### NEW QUESTION 38

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 41

Given the code fragments:

```
interface CourseFilter extends Predicate<String> { public default boolean test (String str) {
    return str.equals ("Java");
}
}
and
List<String> strs = Arrays.asList("Java", "Java EE", "Java ME"); Predicate<String> cf1 = s -> s.length() > 3;
Predicate cf2 = new CourseFilter() { //line n1 public boolean test (String s) {
    return s.contains ("Java");
}
};
long c = strs.stream()
    .filter(cf1)
    .f ilter(cf2 //line n2
    .count(); System.out.println(c); What is the result?
```

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

**Answer: B**

#### NEW QUESTION 44

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 47

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 48

Given:

```
public class Vehicle {
    int vId;
    String vName;
    public Vehicle(int vIdArg, String vNameArg) {
        this.vId = vIdArg;
        this.vName = vNameArg;
    }
    public int getVId() { return vId; }
    public String getVName() { return vName; }
    public String toString() {
        return vName;
    }
}
```

and the code fragment:

```
List<Vehicle> vehicle = Arrays.asList(
    new Vehicle(2, "Car"),
    new Vehicle(3, "Bike"),
    new Vehicle(1, "Truck"));
vehicle.stream()
    // line n1
    .forEach(System.out::print);
```

Which two code fragments, when inserted at line n1 independently, enable the code to print TruckCarBike?

- A. `.sorted ((v1, v2) -> v1.getVId() < v2.getVId())`
- B. `.sorted (Comparable.comparing (Vehicle: :getVName)).reversed ()`
- C. `.map (v -> v.getVid()).sorted ()`
- D. `.sorted((v1, v2) -> Integer.compare(v1.getVId(), v2.getVid()))`
- E. `.sorted(Comparator.comparing ((Vehicle v) -> v.getVId()))`

**Answer:** B

#### NEW QUESTION 49

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 54

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 58

Given the code fragment:

```
final String str1 = "Java";  
StringBuffer strBuf = new StringBuffer ("Course");  
UnaryOperator<String> u = (str2) -> str1.concat(str2); // line n1  
UnaryOperator<String> c = (str3) -> str3.toLowerCase();  
System.out.println(u.apply(c.apply(strBuf))); // line n2
```

What is the result?

- A. A compilation error occurs at line n1.
- B. `courseJava`
- C. `Javacourse`
- D. A compilation error occurs at line n2.

**Answer:** A

**NEW QUESTION 61**

Given the code fragment:

```
Stream<List<String>> strs = Stream.of(
    Arrays.asList("text1", "text2"),
    Arrays.asList("text2", "text3"));
Stream<String> bs2 = strs
    .filter(b -> b.contains("text1"))
    .flatMap(rs -> rs.stream());
bs2.forEach(b -> System.out.print(b));
```

What is the result?

- A. text1text2
- B. text1text2text2text3
- C. text1
- D. [text1, text2]

**Answer:** A**NEW QUESTION 62**

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 65**

Given the code fragment: Stream<List<String>> iStr= Stream.of ( Arrays.asList ("1", "John"), Arrays.asList ("2", null)0;  
Stream<<String> nInSt = iStr.flatMapToInt ((x) -> x.stream ()); nInSt.forEach (System.out :: print);  
What is the result?

- A. 1John2null
- B. 12
- C. A NullPointerException is thrown at run time.
- D. A compilation error occurs.

**Answer:** D**NEW QUESTION 70**

Given the code fragment:

```
List<String> colors = Arrays.asList("red", "green", "yellow"); Predicate<String> test = n -> { System.out.println("Searching...");
return n.contains("red");
};
colors.stream()
.f ilter(c -> c.length() > 3)
.allMatch(test); What is the result?
```

- A. Searching...
- B. Searching...Searching...

C. Searching...Searching... Searching...  
D. A compilation error occurs.

**Answer:** A

#### NEW QUESTION 71

Given the code fragment:

```
List<Integer> codes = Arrays.asList (10, 20); UnaryOperator<Double> uo = s -> s +10.0; codes.replaceAll(uo);  
codes.forEach(c -> System.out.println(c));
```

 What is the result?

- A. 20.030.0
- B. 1020
- C. A compilation error occurs.
- D. A NumberFormatException is thrown at run time.

**Answer:** C

#### NEW QUESTION 76

Given:

```
public enum USCurrency { PENNY (1),  
NICKLE(5), DIME (10), QUARTER(25);  
private int value;  
public USCurrency(int value) { this.value = value;  
}  
public int getValue() {return value;}  
}  
public class Coin {  
public static void main (String[] args) { USCurrency usCoin =new USCurrency.DIME; System.out.println(usCoin.getValue());  
}  
}
```

Which two modifications enable the given code to compile? (Choose two.)

- A. Nest the USCurrency enumeration declaration within the Coin class.
- B. Make the USCurrency enumeration constructor private.
- C. Remove the new keyword from the instantiation of usCoin.
- D. Make the getter method of value as a static method.
- E. Add the final keyword in the declaration of value.

**Answer:** BC

#### NEW QUESTION 77

Given the definition of the Country class:

```
public class country {  
public enum Continent {ASIA, EUROPE} String name;  
Continent region;  
public Country (String na, Continent reg) { name = na, region = reg;  
}  
public String getName () {return name;} public Continent getRegion () {return region;}  
}
```

and the code fragment:

```
List<Country> couList = Arrays.asList (  
new Country ("Japan", Country.Continent.ASIA), new Country ("Italy", Country.Continent.EUROPE),  
new Country ("Germany", Country.Continent.EUROPE)); Map<Country.Continent, List<String>> regionNames = couList.stream ()  
.collect(Collectors.groupingBy (Country ::getRegion, Collectors.mapping(Country::getName, Collectors.toList()))); System.out.println(regionNames);
```

- A. {EUROPE = [Italy, Germany], ASIA = [Japan]}
- B. {ASIA = [Japan], EUROPE = [Italy, Germany]}
- C. {EUROPE = [Germany, Italy], ASIA = [Japan]}
- D. {EUROPE = [Germany], EUROPE = [Italy], ASIA = [Japan]}

**Answer:** B

#### NEW QUESTION 82

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 87**

Given:

```
class Person {
    private String firstName;
    private int salary;
    public Person(String fN, int sal) {
        this.firstName = fN;
        this.salary = sal;
    }
    public int getSalary() { return salary; }
    public String getFirstName() { return firstName; }
}
```

and the code fragment:

```
List<Person> prog = Arrays.asList(
    new Person("Smith", 1500),
    new Person("John", 2000),
    new Person("Joe", 1000));
double dVal = prog.stream()
    .filter(s -> s.getFirstName().startsWith("J"))
    .mapToInt(Person::getSalary)
    .average()
    .getAsDouble();
System.out.print(dVal);
```

What is the result?

- A. 0.0
- B. 1500.0
- C. A compilation error occur
- D. 2000.0

**Answer: D****NEW QUESTION 89**

Given:

```
class RateOfInterest {
    public static void main (String[] args) { int rateOfInterest = 0;
    String accountType = "LOAN"; switch (accountType) {
    case "RD"; rateOfInterest = 5; break;
    case "FD"; rateOfInterest = 10; break;
    default:
    assert false: "No interest for this account"; //line n1
    }
    System.out.println ("Rate of interest:" + rateOfInterest);
    }
}
```

and the command:

java -ea RateOfInterest What is the result?

- A. Rate of interest: 0
- B. An AssertionError is thrown.
- C. No interest for this account
- D. A compilation error occurs at line n1.

**Answer: B****NEW QUESTION 92**

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 95

You have been asked to create a ResourceBundle which uses a properties file to localize an application. Which code example specifies valid keys of menu1 and menu2 with values of File Menu and View Menu?

- A. <key name = 'menu1'>File Menu</key><key name = 'menu2'>View Menu</key>
- B. <key>menu1</key><value>File Menu</value><key>menu2</key><value>View Menu</value>
- C. menu1, File Menu, menu2, View Menu Menu
- D. menu1 = File Menu menu2 = View Menu

**Answer:** D

#### NEW QUESTION 100

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 105

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 107

The data.doc, data.txt and data.xml files are accessible and contain text. Given the code fragment:

```
Stream<Path> paths = Stream.of (Paths. get("data.doc"),
Paths. get("data.txt"),
Paths. get("data.xml"));
paths.filter(s-> s.toString().endsWith("txt")).forEach( s -> {
try { Files.readAllLines(s)
.stream()
.f orEach(System.out::println); //line n1
} catch (IOException e) { System.out.println("Exception");
}
});
```

What is the result?

- A. The program prints the content of data.txt file.
- B. The program prints: Exception<<The content of the data.txt file>> Exception
- C. A compilation error occurs at line n1.
- D. The program prints the content of the three files.

**Answer:** A

#### NEW QUESTION 109

Given the definition of the Vehicle class:

```
Class Vehicle {
int distance; //line n1 Vehicle (int x) {
this distance = x;
}
public void increSpeed(int time) { //line n2 int timeTravel = time; //line n3
class Car { int value = 0;
public void speed () {
value = distance /timeTravel;
System.out.println ("Velocity with new speed"+value+"kmph");
}
}
new Car().speed();
}
```

```
}
```

and this code fragment: `Vehicle v = new Vehicle (100); v.increSpeed(60);`  
What is the result?

- A. Velocity with new speed
- B. A compilation error occurs at line n1.
- C. A compilation error occurs at line n2.
- D. A compilation error occurs at line n3.

**Answer:** A

#### 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 117

Given:  
`interface Doable {`  
`public void doSomething (String s);`  
`}`

Which two class definitions compile? (Choose two.)

- A. `public abstract class Task implements Doable { public void doSomethingElse(String s) { }}`
- B. `public abstract class Work implements Doable { public abstract void doSomething(String s) { } public void doYourThing(Boolean b) { }}`
- C. `public class Job implements Doable { public void doSomething(Integer i) { }}`
- D. `public class Action implements Doable { public void doSomething(Integer i) { } public String doThis(Integer j) { }}`
- E. `public class Do implements Doable { public void doSomething(Integer i) { } public void doSomething(String s) { } public void doThat (String s) { }}`

**Answer:** AE

#### NEW QUESTION 119

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 120

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 124

For which three objects must a vendor provide implementations in its JDBC driver? (Choose three.)

- A. Time
- B. Date
- C. Statement
- D. ResultSet
- E. Connection
- F. SQLException
- G. DriverManager

**Answer:** CDE

#### Explanation:

Database vendors support JDBC through the JDBC driver interface or through the ODBC connection. Each driver must provide implementations of java.sql.Connection, java.sql.Statement, java.sql.PreparedStatement, java.sql.CallableStatement, and java.sql.ResultSet. They must also implement the java.sql.Driver interface for use by the generic java.sql.DriverManager interface.

#### 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 129

Given the definition of the Vehicle class: class Vehicle {  
String name;  
void setName (String name) { this.name = name;  
}  
String getName() { return name;  
}  
}  
Which action encapsulates the Vehicle class?

- A. Make the Vehicle class public.
- B. Make the name variable public.
- C. Make the setName method public.
- D. Make the name variable private.
- E. Make the setName method private.
- F. Make the getName method private.

**Answer:** D

#### NEW QUESTION 130

Given the code fragments: class TechName {  
String techName;  
TechName (String techName) { this.techName=techName;  
}  
}  
and  
List<TechName> tech = Arrays.asList ( new TechName("Java-"),  
new TechName("Oracle DB-"), new TechName("J2EE-")  
);  
Stream<TechName> stre = tech.stream();  
//line n1  
Which should be inserted at line n1 to print Java-Oracle DB-J2EE-?

- A. stre.forEach(System.out::print);
- B. stre.map(a-> a.techName).forEach(System.out::print);
- C. stre.map(a-> a).forEachOrdered(System.out::print);
- D. stre.forEachOrdered(System.out::print);

**Answer:** B

#### NEW QUESTION 133

Which statement is true about java.util.stream.Stream?

- A. A stream cannot be consumed more than once.
- B. The execution mode of streams can be changed during processing.
- C. Streams are intended to modify the source data.
- D. A parallel stream is always faster than an equivalent sequential stream.

**Answer:** B

#### NEW QUESTION 136

Given the code fragment:

```
List<String> empDetails = Arrays.asList("100, Robin, HR", "200, Mary, AdminServices",  
"101, Peter, HR");  
empDetails.stream()  
.filter(s-> s.contains("1"))  
.sorted()  
.forEach(System.out::println); //line n1
```

What is the result?

- A. 100, Robin, HR101, Peter, HR
- B. A compilation error occurs at line n1.
- C. 100, Robin, HR101, Peter, HR200, Mary, AdminServices
- D. 100, Robin, HR200, Mary, AdminServices101, Peter, HR

**Answer:** A

#### NEW QUESTION 139

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 143

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 145

Given the code fragments:

```
class Caller implements Callable<String> { String str;
public Caller (String s) {this.str=s;}
public String call()throws Exception { return str.concat ("Caller");}
}
class Runner implements Runnable { String str;
public Runner (String s) {this.str=s;}
public void run () { System.out.println (str.concat ("Runner"));}
}
and
public static void main (String[] args) InterruptedException, ExecutionException
{
    ExecutorService es = Executors.newFixedThreadPool(2); Future f1 = es.submit (new Caller ("Call"));
    Future f2 = es.submit (new Runner ("Run")); String str1 = (String) f1.get();
    String str2 = (String) f2.get(); //line n1 System.out.println(str1+ ":" + str2);
}
```

What is the result?

- A. The program prints: Run RunnerCall Caller : nullAnd the program does not terminate.
- B. The program terminates after printing: Run RunnerCall Caller : Run
- C. A compilation error occurs at line n1.
- D. An Execution is thrown at run time.

**Answer:** A

#### NEW QUESTION 148

Given the code fragment:

```
Deque<Integer> nums = new ArrayDeque<>();
nums.add(1000);
nums.push(2000);
nums.add(3000);
nums.push(4000);
Integer i1 = nums.remove();
Integer i2 = nums.pop();
System.out.println(i1 + " : " + i2);
```

What is the result?

- A. 4000 : 2000
- B. 4000 : 1000
- C. 1000 : 4000
- D. 1000 : 2000

**Answer:** B

#### NEW QUESTION 150

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 153

Given the code fragments:

```
class Person // line n1
{
    String name;
    Person(String name) {
        this.name = name;
    }
    // line n2
}
```

and

```
List<Person> emps = new ArrayList<>();
/* code that adds objects of the Person class to the emps list goes here */
Collections.sort(emps);
```

Which two modifications enable to sort the elements of the emps list? (Choose two.)

- A. Replace line n1 with `class Person extends Comparator<Person>`
- B. At line n2 insert `public int compareTo (Person p) { return this.name.compareTo (p.name);}`
- C. Replace line n1 with `class Person implements Comparable<Person>`
- D. At line n2 insert `public int compare (Person p1, Person p2) { return p1.name.compareTo (p2.name);}`
- E. At line n2 insert: `public int compareTo (Person p, Person p2) { return p1.name.compareTo (p2.name);}`
- F. Replace line n1 with `class Person implements Comparator<Person>`

**Answer:** CE

#### NEW QUESTION 158

Given the code fragment:

```
List<String> str = Arrays.asList ("my", "pen", "is", "your", "pen"); Predicate<String> test = s -> {
int i = 0;
boolean result = s.contains ("pen");
System.out.print(i++) + ":"; return result;
};
str.stream()
.filter(test)
.findFirst()
.ifPresent(System.out ::print); What is the result?
```

- A. 0 : 0 : pen
- B. 0 : 1 : pen
- C. 0 : 0 : 0 : 0 : 0 : pen
- D. 0 : 1 : 2 : 3 : 4 :
- E. A compilation error occurs.

**Answer:** A

#### NEW QUESTION 162

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 164

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 167

Given:

```
class ImageScanner implements AutoCloseable { public void close () throws Exception { System.out.print ("Scanner closed.");
```

```
}
```

```
public void scanImage () throws Exception { System.out.print ("Scan.");
```

```
throw new Exception("Unable to scan.");
```

```
}
```

```
}
```

```
class ImagePrinter implements AutoCloseable { public void close () throws Exception { System.out.print ("Printer closed.");
```

```
}
```

```
public void printImage () {System.out.print("Print."); }
```

```
}
```

and this code fragment:

```
try (ImageScanner ir = new ImageScanner(); ImagePrinter iw = new ImagePrinter()) { ir.scanImage();
```

```
iw.printImage();
```

```
} catch (Exception e) { System.out.print(e.getMessage());
```

```
}
```

What is the result?

- A. Scan.Printer close
- B. Scanner close
- C. Unable to scan.
- D. Scan.Scanner close
- E. Unable to scan.
- F. Sca
- G. Unable to scan.
- H. Sca
- I. Unable to sca
- J. Printer closed.

**Answer: A**

#### 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 177

Given the code fragments:

```
class R implements Runnable {  
    public void run() { System.out.println("Run..."); }  
}  
  
class C implements Callable<String> {  
    public String call() throws Exception { return "Call..."; }  
}
```

and

```
ExecutorService es = Executors.newSingleThreadExecutor();  
es.execute(new R()); // line n1  
Future<String> f1 = es.submit(new C()); // line n2  
System.out.println(f1.get());  
es.shutdown();
```

What is the result?

- A. The program prints Run... and throws an exception.
- B. A compilation error occurs at line n1.
- C. Run...Call...
- D. A compilation error occurs at line n2.

**Answer:** B

#### NEW QUESTION 178

Given:

```
class Person {  
    String name;  
    int age;  
    public Person(String name, int age) {  
        this.name = name;  
        this.age = age;  
    }  
    public String getName(){ return name; }  
    public int getAge(){ return age; }  
}
```

and the code fragment:

```
List<Person> sts = Arrays.asList(  
    new Person("Jack", 30),  
    new Person("Mike Hill", 21),  
    new Person("Thomas Hill", 24));  
Stream<Person> resList = sts.stream().filter(s -> s.getAge() >= 25);    // line n1  
long count = resList.filter(s -> s.getName().contains("Hill")).count();  
System.out.print(count);
```

What is the result?

- A. A compilation error occurs at line n1.
- B. An Exception is thrown at run time.
- C. 2

**Answer:** B

#### NEW QUESTION 183

.....

## 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](#)**