

# Exam Questions 1Z0-809

Java SE 8 Programmer II

<https://www.2passeasy.com/dumps/1Z0-809/>



### 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:

```
class Sum extends RecursiveAction { //line n1 static final int THRESHOLD_SIZE = 3;
int stIndex, lstIndex; int [ ] data;
public Sum (int [ ]data, int start, int end) { this.data = data;
this stIndex = start; this. lstIndex = end;
}
protected void compute ( ) { int sum = 0;
if (lstIndex – stIndex <= THRESHOLD_SIZE) { for (int i = stIndex; i < lstIndex; i++) {
sum += data [i];
}
System.out.println(sum);
} else {
new Sum (data, stIndex + THRESHOLD_SIZE, lstIndex).fork( ); new Sum (data, stIndex,
Math.min (lstIndex, stIndex + THRESHOLD_SIZE)
).compute ();
}
}
}
```

and the code fragment:

```
ForkJoinPool fjPool = new ForkJoinPool ( ); int data [ ] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
fjPool.invoke (new Sum (data, 0, data.length));
and given that the sum of all integers from 1 to 10 is 55. Which statement is true?
```

- A. The program prints several values that total 55.
- B. The program prints 55.
- C. A compilation error occurs at line n1.
- D. The program prints several values whose sum exceeds 55.

**Answer: A**

#### NEW QUESTION 4

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 5

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 6

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 7

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 8

Given the code fragment:

```
Stream<Path> files = Files.walk(Paths.get(System.getProperty("user.home"))); files.forEach (fName -> { //line n1
try {
Path aPath = fName.toAbsolutePath(); //line n2 System.out.println(fName + ":"
+ Files.readAttributes(aPath, Basic.File.Attributes.class).creationTime ());
} catch (IOException ex) { ex.printStackTrace();
});
What is the result?
```

- A. All files and directories under the home directory are listed along with their attributes.
- B. A compilation error occurs at line n1.
- C. The files in the home directory are listed along with their attributes.
- D. A compilation error occurs at line n2.

**Answer:** A

#### NEW QUESTION 9

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 10

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 10

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 14

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 17

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 19

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 21

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 25

Which class definition compiles?

```

A. class Vehicle {
    int id;
    public void start() {
        public class Engine { int eNo = id; }
    }
}

B. class Computer {
    private Card sCard = new SoundCard();
    private abstract class Card { }
    private class SoundCard extends Card { }
}

C. class Block {
    int bno;
    static class Counter {
        int locator;
        Counter() { locator = bno; }
    }
}

D. class Product {
    interface Moveable { void move(); }
    Moveable mProduct = new Moveable() {
        void move() { }
    };
}

```

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

**Answer:** A

#### NEW QUESTION 29

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 31

Given the code fragment:

```
Path file = Paths.get("courses.txt");
// line n1
```

Assume the courses.txt is accessible.

Which code fragment can be inserted at line n1 to enable the code to print the content of the courses.txt file?

- A. List<String> fc = Files.list(file); fc.stream().forEach(s -> System.out.println(s));
- B. Stream<String> fc = Files.readAllLines(file); fc.forEach(s -> System.out.println(s));
- C. List<String> fc = readAllLines(file); fc.stream().forEach(s -> System.out.println(s));
- D. Stream<String> fc = Files.lines(file); fc.forEach(s -> System.out.println(s));

**Answer:** D

#### NEW QUESTION 32

Given:



```
public final class IceCream { public void prepare() {}  
}  
public class Cake {  
public final void bake(int min, int temp) {} public void mix() {}  
}  
public class Shop {  
private Cake c = new Cake (); private final double discount = 0.25;  
public void makeReady () { c.bake(10, 120); }  
}  
public class Bread extends Cake {  
public void bake(int minutes, int temperature) {} public void addToppings() {}  
}
```

Which statement is true?

- A. A compilation error occurs in IceCream.
- B. A compilation error occurs in Cake.
- C. A compilation error occurs in Shop.
- D. A compilation error occurs in Bread
- E. All classes compile successfully.

**Answer:** D

#### NEW QUESTION 36

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 41

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 42**

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 43**

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 47**

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 50**

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 52

Which two statements are true about localizing an application? (Choose two.)

- A. Support for new regional languages does not require recompilation of the code.
- B. Textual elements (messages and GUI labels) are hard-coded in the code.
- C. Language and region-specific programs are created using localized data.
- D. Resource bundle files include data and currency information.
- E. Language codes use lowercase letters and region codes use uppercase letters.

**Answer:** AE

#### NEW QUESTION 56

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 61

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 62

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 66

Given the records from the Employee table:

eid	ename
111	Tom
112	Jerry
113	Donald

and given the code fragment: try {  
 Connection conn = DriverManager.getConnection (URL, userName, passWord); Statement st = conn.createStatement(ResultSet.TYPE\_SCROLL\_INSENSITIVE, ResultSet.CONCUR\_UPDATABLE);  
 st.execute("SELECT\*FROM Employee"); ResultSet rs = st.getResultSet();  
 while (rs.next()) {  
 if (rs.getInt(1) ==112) { rs.updateString(2, "Jack");  
 }  
 }  
 rs.absolute(2);  
 System.out.println(rs.getInt(1) + " " + rs.getString(2));  
 } catch (SQLException ex) { System.out.println("Exception is raised");  
 }  
 Assume that:  
 The required database driver is configured in the classpath.  
 The appropriate database accessible with the URL, userName, and passWord exists. What is the result?

- A. The Employee table is updated with the row: 112 Jackand the program prints: 112 Jerry
- B. The Employee table is updated with the row: 112 Jackand the program prints: 112 Jack
- C. The Employee table is not updated and the program prints: 112 Jerry
- D. The program prints Exception is raised.

**Answer: A**

#### NEW QUESTION 71

Given the content of /resourses/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 74

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 75



```
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 79

Given the code fragments:

```
class MyThread implements Runnable {
private static AtomicInteger count = new AtomicInteger (0); public void run ( ) {
int x = count.incrementAndGet(); System.out.print (x+" ");
}
}
```

and

```
Thread thread1 = new Thread(new MyThread()); Thread thread2 = new Thread(new MyThread()); Thread thread3 = new Thread(new MyThread()); Thread [] ta =
{thread1, thread2, thread3};
for (int x= 0; x < 3; x++) { ta[x].start();
}
}
```

Which statement is true?

- A. The program prints 1 2 3 and the order is unpredictable.
- B. The program prints 1 2 3.
- C. The program prints 1 1 1.
- D. A compilation error occurs.

**Answer:** A

#### NEW QUESTION 84

You want to create a singleton class by using the Singleton design pattern. Which two statements enforce the singleton nature of the design? (Choose two.)

- A. Make the class static.
- B. Make the constructor private.
- C. Override equals() and hashCode() methods of the java.lang.Object class.
- D. Use a static reference to point to the single instance.
- E. Implement the Serializable interface.

**Answer:** BD

#### NEW QUESTION 88

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 89

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 94

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 95

Given the code fragments:

```
public class Book implements Comparator<Book> { String name;
double price; public Book () {}
public Book(String name, double price) { this.name = name;
this.price = price;
}
public int compare(Book b1, Book b2) { return b1.name.compareTo(b2.name);
}
public String toString() { return name + ":" + price;
}
}
and
List<Book>books = Arrays.asList (new Book ("Beginning with Java", 2), new book ("A
Guide to Java Tour", 3));
Collections.sort(books, new Book()); System.out.print(books);
What is the result?
```

- A. [A Guide to Java Tour:3.0, Beginning with Java:2.0]
- B. [Beginning with Java:2, A Guide to Java Tour:3]
- C. A compilation error occurs because the Book class does not override the abstract method compareTo().
- D. An Exception is thrown at run time.

**Answer: A**

#### NEW QUESTION 96

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 98

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 102

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 103

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 107

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 112

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.writeIn();
- C. br.flush();
- D. bw.flush();

**Answer: D**

#### NEW QUESTION 117

Given the code fragment:

```
String str = "Java is a programming language";  
ToIntFunction<String> indexVal = str::indexOf; //line n1  
int x = indexVal.applyAsInt("Java"); //line n2  
System.out.println(x);
```

What is the result?

- A. 1
- B. A compilation error occurs at line n1.
- C. A compilation error occurs at line n2.

**Answer: A**

#### NEW QUESTION 121

Given the code fragment:

```
Path source = Paths.get("/data/december/log.txt");  
Path destination = Paths.get("/data");
```

```
Files.copy(source, destination);
```

and assuming that the file /data/december/log.txt is accessible and contains: 10-Dec-2014 – Executed successfully

What is the result?

- A. A file with the name log.txt is created in the /data directory and the content of the /data/december/ log.txt file is copied to it.
- B. The program executes successfully and does NOT change the file system.
- C. A FileNotFoundException is thrown at run time.
- D. A FileAlreadyExistsException is thrown at run time.

**Answer: D**

#### NEW QUESTION 123

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 124

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 126

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 131

Given:

```
public class Canvas implements Drawable { public void draw () { }  
}  
public abstract class Board extends Canvas { }  
public class Paper extends Canvas { protected void draw (int color) { }  
}  
public class Frame extends Canvas implements Drawable { public void resize () { }  
}  
public interface Drawable { public abstract void draw ();  
}
```

Which statement is true?

- A. Board does not compile.
- B. Paper does not compile.
- C. Frame does not compile.
- D. Drawable does not compile.
- E. All classes compile successfully.

**Answer:** E

#### NEW QUESTION 135

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()  
.filter(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 140

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 143

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 throw
- C. 2016-02-29
- D. A compilation error occurs at line n1.

**Answer:** A

#### NEW QUESTION 147

Given the code fragment:

```
List<Integer> nums = Arrays.asList (10, 20, 8); System.out.println (  
//line n1  
);
```

Which code fragment must be inserted at line n1 to enable the code to print the maximum number in the nums list?

- A. nums.stream().max(Comparator.comparing(a -> a)).get()
- B. nums.stream().max(Integer : : max).get()
- C. nums.stream().max()
- D. nums.stream().map(a -> a).max()

**Answer:** A

#### NEW QUESTION 152

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 ()
.c 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 154

Given the code fragment:

```
List<Integer> prices = Arrays.asList(3, 4, 5);
prices.stream()
    .filter(e -> e > 4)
    .peek(e -> System.out.print("Price " + e))           // line n1
    .map(n -> n - 1)                                     // line n2
    .peek(n -> System.out.println(" New Price " + n));   // line n3
```

Which modification enables the code to print Price 5 New Price 4?

- A. Replace line n2 with .map (n -> System.out.println ("New Price" + n –1)) and remove line n3
- B. Replace line n2 with .mapToInt (n -> n – 1);
- C. Replace line n1 with .forEach (e -> System.out.print ("Price" + e))
- D. Replace line n3 with .forEach (n -> System.out.println ("New Price" + n));

**Answer: A**

#### NEW QUESTION 156

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 157

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 161

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 166

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 170

Assume customers.txt is accessible and contains multiple lines. Which code fragment prints the contents of the customers.txt file?

- A. Stream<String> stream = Files.find (Paths.get ("customers.txt")); stream.forEach((String c) -> System.out.println(c));
- B. Stream<Path> stream = Files.find (Paths.get ("customers.txt")); stream.forEach( c) -> System.out.println(c));
- C. Stream<Path> stream = Files.list (Paths.get ("customers.txt")); stream.forEach( c) -> System.out.println(c));
- D. Stream<String> lines = Files.lines (Paths.get ("customers.txt")); lines.forEach( c) -> System.out.println(c));

**Answer: A**

#### NEW QUESTION 175

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 177

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 180

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 184

Given:

```
class Vehicle { int vno;
String name;
public Vehicle (int vno, String name) { this.vno = vno,;
```

```
this.name = name;
}
public String toString () { return vno + ":" + name;
}
}
```

and this code fragment:

```
Set<Vehicle> vehicles = new TreeSet <> (); vehicles.add(new Vehicle (10123, "Ford")); vehicles.add(new Vehicle (10124, "BMW")); System.out.println(vehicles);
What is the result?
```

- A. 10123 Ford10124 BMW
- B. 10124 BMW10123 Ford
- C. A compilation error occurs.
- D. A ClassCastException is thrown at run time.

**Answer:** D

#### NEW QUESTION 187

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 191

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 195

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 200

Given the code fragments:

```
public class Video {
    public void play() throws IOException {
        System.out.print("Video played.");
    }
}

public class Game extends Video {
    public void play() throws Exception {
        super.play();
        System.out.print("Game played.");
    }
}
```

and

```
try {
    new Game().play();
} catch (Exception e) {
    System.out.print(e.getClass());
}
```

What is the result?

- A. Video played.Game played.
- B. A compilation error occurs.
- C. class java.lang.Exception
- D. class java.io.IOException

**Answer:** C

#### NEW QUESTION 204

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 206

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 209

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 210

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 214

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 217

Given the code fragment:  
9. `Connection conn = DriverManager.getConnection(dbURL, userName, passWord);`  
10. `String query = "SELECT id FROM Employee";`  
11. `try (Statement stmt = conn.createStatement()) {`  
12. `ResultSet rs = stmt.executeQuery(query);`  
13. `stmt.executeQuery("SELECT id FROM Customer");`  
14. `while (rs.next()) {`  
15. `//process the results`  
16. `System.out.println("Employee ID: "+ rs.getInt("id"));`  
17. `}`  
18. `} catch (Exception e) {`  
19. `System.out.println ("Error");`  
20. `}`

Assume that:

The required database driver is configured in the classpath.

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

The Employee and Customer tables are available and each table has id column with a few records and the SQL queries are valid.

What is the result of compiling and executing this code fragment?

- A. The program prints employee IDs.
- B. The program prints customer IDs.
- C. The program prints Error.
- D. compilation fails on line 13.

**Answer:** C

#### NEW QUESTION 218

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)  
.ifPresent(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 219

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 222

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()
.f orEach(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 225

Given the content:

```
MessagesBundle.properties file:

username = Enter User Name
password = Enter Password

MessagesBundle_fr_FR.properties file:

username = Entrez le nom d'utilisateur
password = Entrez le mot de passe
```

and the code fragment:

```
Locale currentLocale = new Locale.Builder().setRegion("FR").setLanguage("fr").build();
ResourceBundle messages = ResourceBundle.getBundle("MessagesBundle", currentLocale);
Enumeration<String> names = messages.getKeys();
while (names.hasMoreElements()) {
    String key = names.nextElement();
    String name = messages.getString(key);
    System.out.println(key + " = " + name);
}
```

What is the result?

- A. username = Entrez le nom d'utilisateur password = Entrez le mot de passe
- B. username = Enter User Name password = Enter Password
- C. A compilation error occurs.
- D. The program prints nothing.

**Answer:** A

#### NEW QUESTION 230

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 231

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 233

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 236



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 239

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 243

Given:

```
class UserException extends Exception { }
```

```
class AgeOutOfLimitException extends UserException { } and the code fragment:
```

```
class App {
public void doRegister(String name, int age) throws UserException, AgeOutOfLimitException { if (name.length () < 6) {
throw new UserException ();
} else if (age >= 60) {
throw new AgeOutOfLimitException ();
} else {
System.out.println("User is registered.");
}
}
}
public static void main(String[] args) throws UserException { App t = new App ();
```

- A. t.d oRegister("Mathew", 60);}}What is the result?
- B. User is registered.
- C. An AgeOutOfLimitException is thrown.
- D. A UserException is thrown.
- E. A compilation error occurs in the main method.

**Answer:** B

#### NEW QUESTION 244

Which statement is true about the single abstract method of the java.util.function.Function interface?

- A. It accepts one argument and returns void.
- B. It accepts one argument and returns boolean.
- C. It accepts one argument and always produces a result of the same type as the argument.
- D. It accepts an argument and produces a result of any data type.

Answer: D

#### NEW QUESTION 248

Which two code blocks correctly initialize a Locale variable? (Choose two.)

- A. Locale loc1 = "UK";
- B. Locale loc2 = Locale.getInstance("ru");
- C. Locale loc3 = Locale.getLocaleFactory("RU");
- D. Locale loc4 = Locale.UK;
- E. Locale loc5 = new Locale ("ru", "RU");

Answer: DE

#### NEW QUESTION 253

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 258

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 261

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 265

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 267

Given:

```
public class Emp { String fName; String lName;
    public Emp (String fn, String ln) { fName = fn;
    lName = ln;
    }
    public String getfName() { return fName; } public String getlName() { return lName; }
}
```

and the code fragment: List<Emp> emp = Arrays.asList ( new Emp ("John", "Smith"),

```
new Emp ("Peter", "Sam"),
new Emp ("Thomas", "Wale")); emp.stream()
//line n1
.collect(Collectors.toList());
```

Which code fragment, when inserted at line n1, sorts the employees list in descending order of fName and then ascending order of lName?

- A. `.sorted (Comparator.comparing(Emp::getfName).reserved().thenComparing(Emp::getlName))`
- B. `.sorted (Comparator.comparing(Emp::getfName).thenComparing(Emp::getlName))`
- C. `.map(Emp::getfName).sorted(Comparator.reserveOrder())`
- D. `.map(Emp::getfName).sorted(Comparator.reserveOrder()).map (Emp::getlName).reserved`

**Answer:** A

#### NEW QUESTION 271

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 276

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.nextlnt();`
- D. `int GDP = Integer.parseInt (br.next());`

**Answer:** A

#### NEW QUESTION 279

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 283

```
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> addr1 = Optional.ofNullable (address);
```



Employee e1 = new Employee (addr1);  
String eAddress = (addr1.isPresent()) ? addr1.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 288

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 290

Which action can be used to load a database driver by using JDBC3.0?

- A. Add the driver class to the META-INF/services folder of the JAR file.
- B. Include the JDBC driver class in a jdbc.properties file.
- C. Use the java.lang.Class.forName method to load the driver class.
- D. Use the DriverManager.getDriver method to load the driver class.

**Answer:** C

#### NEW QUESTION 294

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 295

Given the Greetings.properties file, containing:

```
HELLO_MSG = Hello, everyone!  
GOODBYE_MSG = Goodbye everyone!
```

and given:

```
import java.util.Enumeration;  
import java.util.Locale;  
import java.util.ResourceBundle;  
  
public class ResourcesApp {  
    public void loadResourceBundle() {  
        ResourceBundle resource = ResourceBundle.getBundle("Greetings", Locale.US);  
        System.out.println(resource.getObject(1));  
    }  
    public static void main(String[] args) {  
        new ResourcesApp().loadResourceBundle();  
    }  
}
```

What is the result?

- A. Compilation fails.
- B. GOODBYE\_MSG
- C. Hello, everyone!
- D. Goodbye everyone!
- E. HELLO\_MSG

**Answer:** A

#### NEW QUESTION 299

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 301

.....

## THANKS FOR TRYING THE DEMO OF OUR PRODUCT

Visit Our Site to Purchase the Full Set of Actual 1Z0-809 Exam Questions With Answers.

We Also Provide Practice Exam Software That Simulates Real Exam Environment And Has Many Self-Assessment Features. Order the 1Z0-809 Product From:

<https://www.2passeasy.com/dumps/1Z0-809/>

## Money Back Guarantee

### 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 FirstTry
- \* 1Z0-809 Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year