Oracle

Exam Questions 1Z0-819

Java SE 11 Developer

NEW QUESTION 1
Assuming the Widget class has a getPrice method, this code does not compile:

```
List widgets = List.of(new Widget("Basic Widget", 19.55), // line 1
                       new Widget("Enhanced Widget", 35.00),
                       new Widget("Luxury Edition Widget", 55.45));
Stream widgetStream = widgets.stream();                    // line 4
widgetStream.filter(a -> a.getPrice() > 20.00)             // line 5
            .forEach(System.out::println);
```

Which two statements, independently, would allow this code to compile? (Choose two.)

A. Replace line 5 with widgetStream.filter(a > ((Widget)a).getPrice() > 20.00).
B. Replace line 1 with List<Widget> widgetStream = widgets.stream();.
C. Replace line 5 with widgetStream.filter((Widget a) > a.getPrice() > 20.00).
D. Replace line 4 with Stream<Widget> widgetStream = widgets.stream();.

Answer: AD

NEW QUESTION 2
Which interface in the java.util.function package will return a void return type?

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

Answer: D

NEW QUESTION 3
A bookstore's sales are represented by a list of Sale objects populated with the name of the customer and the books they purchased.
public class Sale { private String customer;
private List<Book> items;
// constructor, setters and getters not shown
}
public class Book { private String name; private double price;
// constructor, setters and getters not shown
}
Given a list of Sale objects, tList, which code fragment creates a list of total sales for each customer in ascending order?

A.
```
List<String> totalByUser = tList.stream()
      .collect(flatMapping(t -> t.getItems().stream(),
               groupingBy(Sale::getCustomer,
               summingDouble(Book::getPrice))))
      .entrySet().stream()
      .sorted(Comparator.comparing(Entry::getValue))
      .collect(mapping(e -> e.getKey() + ":" + e.getValue(),toList()));
```

B.
```
List<String> totalByUser = tList.stream()
     .collect(groupingBy(Sale::getCustomer,
              flatMapping(t -> t.getItems().stream(),
              summingDouble(Book::getPrice))))
     .sorted(Comparator. comparing (Entry::getValue))
     .collect(mapping(e -> e.getKey() + ":" + e.getValue(),toList()));
```

C.
```
List<String> totalByUser = tList.stream()
     .collect(groupingBy(Sale::getCustomer,
              flatMapping(t -> t.getItems().stream(),
              summingDouble(Book::getPrice))))
     .entrySet().stream()
     .sorted(Comparator.comparing(Entry::getValue))
     .collect(mapping(e -> e.getKey() + ":" + e.getValue(),toList()));
```

D.
```
List<String> totalByUser = tList.stream()
     .collect(flatMapping(t -> t.getItems().stream(),
              groupingBy(Sale::getCustomer,
              summingDouble(Book::getPrice))))
     .sorted(Comparator.comparing (Entry::getValue))
     .collect(mapping(e -> e.getKey() + ":" + e.getValue(),toList()));
```

A. Option A
B. Option B
C. Option C

D. Option D

Answer: C

NEW QUESTION 4
Given:

```
public class A {
  private boolean checkValue(int val) {
    return true;
  }
}
```

and

```
public class B extends A {
  public int modifyVal(int val) {
    if(checkValue(val)) {
      return val;
    } else {
      return 0;
    }
  }
  public static void Main(String[] args) {
    B b = new B();
    System.out.println(b.modifyVal(10));
  }
}
```

What is the result?

A. nothing
B. It fails to compile.
C. A java.lang.IllegalArgumentException is thrown.
D. 10

Answer: B

Explanation:

```
1  public class A {
2      private boolean checkValue(int val) {
3          return true;
4      }
5  }
6  and
7  public class B extends A {
8      public int modifyVal(int val) {
9          if(checkValue(val)) {
10             return val;
11         } else {
12             return 0;
13         }
14     }
15     public static void Main(String[] args) {
16         B b = new B();
17         system.out.println(b.modfiyVal (10));
18     }
19 }
```

Execute Mode, Version, Inputs & Arguments

JDK 11.0.4

CommandLine Arguments

Result

CPU Time: sec(s), Memory: kilobyte(s)

```
/A.java:6: error: class, interface, or enum expected
and
^
1 error
```

NEW QUESTION 5
Given the code fragment:
Path currentFile = Paths.get("/scratch/exam/temp.txt"); Path outputFile = Paths get("/scratch/exam/new.txt"); Path directory = Paths.get("/scratch/");
Files.copy(currentFile, outputFile); Files.copy(outputFile, directory);
Files.delete (outputFile);
The /scratch/exam/temp.txt file exists. The /scratch/exam/new.txt and /scratch/new.txt files do not exist. What is the result?

A. /scratch/exam/new.txt and /scratch/new.txt are deleted.
B. The program throws a FileaAlreadyExistsException.
C. The program throws a NoSuchFileException.
D. A copy of /scratch/exam/new.txt exists in the /scratch directory and /scratch/exam/new.txt is deleted.

Answer: C

Explanation:

```
27  public class Main {
28    public static void main(String[] args) {
29     Path currentFile = Paths.get("/scratch/exam/temp.txt");
30     Path outputFile = Paths.get("/scratch/exam/new.txt");
31     Path directory = Paths.get("/scratch/");
32
33     Files.copy(currentFile, outputFile);
34     Files.copy(outputFile, directory);
35     Files.delete (outputFile);
36    }
37    }
38
```

NEW QUESTION 6
Given:

```
int arr[][] = {{5,10},{8,12},{9,3}};
long count = Stream.of(arr)
                   .flatMapToInt(IntStream::of)
                   .map(n -> n + 1)
                   .filter(n -> (n % 2 == 0))
                   .peek(System.out::print)
                   .count();
System.out.println(" " + count);
```

What is the result?

A. 6910 3
B. 10126 3
C. 3
D. 6104 3

Answer: D

Explanation:

```
1  import java.util.*;
2  import java.io.*;
3  import java.lang.Thread;
4  import java.util.ArrayList;
5  import java.util.LinkedList;
6  import java.util.List;
7  import java.util.function.Consumer;
8  import java.util.stream.Stream;
9  import java.util.stream.IntStream;
10
11
12 public class Main {
13
14   public static void main(String[] args) {
15     int arr[][] = {{5,10}, {8,12}, {9,3}};
16     long count = Stream.of(arr)
17         .flatMapToInt(IntStream::of)
18         .map (n -> n + 1)
19         .filter(n -> (n % 2 == 0))
20         .peek(System.out::print)
21         .count();
22     System.out.println(" " + count);
23   }
24 }
```

Execute Mode, Version, Inputs & Arguments

JDK 11.0.4

CommandLine Arguments

Result

CPU Time: 0.32 sec(s), Memory: 34220 kilobyte(s)

```
6104 3
```

NEW QUESTION 7
Given the formula to calculate a monthly mortgage payment:

$$M = P\frac{r(1+r)^n}{(1+r)^n-1}$$

and these declarations:

```
double m;             //monthly payment
double r = 0.05/12; //monthly interest rate
int p = 100_000;     //principal
int n = 180;          //number of payments
```

How can you code the formula?

A. m = p * (r * Math.pow(1 + r, n) / (Math.pow(1 + r, n) - 1));
B. m = p * ((r * Math.pow(1 + r, n) / (Math.pow(1 + r, n)) - 1));
C. m = p * r * Math.pow(1 + r, n) / Math.pow(1 + r, n) - 1;
D. m = p * (r * Math.pow(1 + r, n) / Math.pow(1 + r, n) - 1);

Answer: A

NEW QUESTION 8
Given the code fragment:

```
public static void main(String[] args) {
   List<Integer> even = List.of();
   even.add(0, -1);
   even.add(0, -2);
   even.add(0, -3);
   System.out.println(even);
}
```

What is the output?

A. The compilation fail
B. [-1, -2, -3]
C. [-3, -2, -1]
D. A runtime exception is thrown.

Answer: D

NEW QUESTION 9
Consider this method declaration:

```
void setSessionUser(Connection conn, String user) throws SQLException {
     Statement stmt = conn.createStatement();
     String sql = <EXPRESSION>;
     stmt .execute();
}
```

A) “SET SESSION AUTHORIZATION “ + user
B) “SET SESSION AUTHORIZATION “ + stmt.enquoteIdentifier(user) Is A or B the correct replacement for <EXPRESSION> and why?

A. A, because it sends exactly the value of user provided by the calling code.
B. B, because enquoting values provided by the calling code prevents SQL injection.
C. A and B are functionally equivalent.
D. A, because it is unnecessary to enclose identifiers in quotes.
E. B, because all values provided by the calling code should be enquoted.

Answer: A

NEW QUESTION 10
Given:

```
public class Person {
   private String name;
   public void setName(String name) {
      String title = "Dr. ";
      name = title+name;
   }
   public String toString() {
      return name;
   }
}
```

and

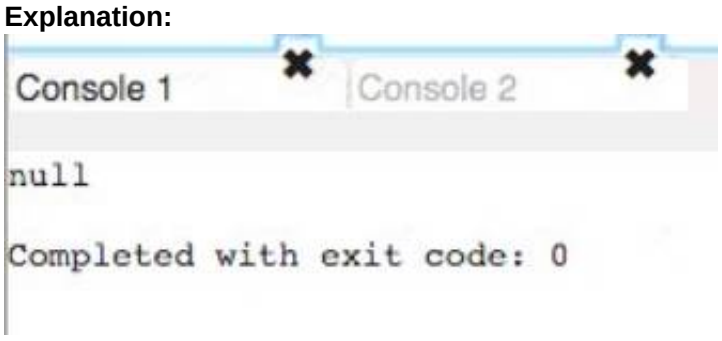
```
public class Test {
   public static void main(String args[]) {
      Person p = new Person();
      p.setName("Who");
      System.out.println(p);
   }
}
```

What is the result?

A. D
B. Who
C. D
D. Null
E. An exception is thrown at runtime.
F. null

Answer: D

Explanation:



NEW QUESTION 10
Given:

```
1.    public class Secret {
2.         String[] names;
3.         public Secret(String[] names) {
4.              this.names = names;
5.         }
6.         public String[] getNames() {
7.              return names;
8.         }
9.    }
```

Which three actions implement Java SE security guidelines? (Choose three.)

A. Change line 7 to return names.clone();.
B. Change line 4 to this.names = names.clone();.
C. Change the getNames() method name to get$Names().
D. Change line 6 to public synchronized String[] getNames() {.
E. Change line 2 to private final String[] names;.
F. Change line 3 to private Secret(String[] names) {.
G. Change line 2 to protected volatile String[] names;.

Answer: EFG

NEW QUESTION 12
Given:

```
public class DNASynth {
    int aCount;
    int tCount;
    int cCount;
    int gCount;

    DNASynth(int a, int tCount, int c, int g){
        // line 1
    }
    int setCCount(int c){
        return c;
    }
    void setGCount(int gCount){
        this.gCount = gCount;
    }
}
```

Which two lines of code when inserted in line 1 correctly modifies instance variables? (Choose two.)

A. setCCount(c) = cCount;
B. tCount = tCount;
C. setGCount(g);
D. cCount = setCCount(c);
E. aCount = a;

Answer: BE

NEW QUESTION 16
Given the code fragment:

```
int x = 0;
do {
   x++;
   if (x == 1) {
      continue;
   }
   System.out.println(x);
} while(x < 1);
```

What is the result?

A. 01
B. 1
C. The program prints nothing.
D. It prints 1 in the infinite loop.

Answer: D

NEW QUESTION 19
Which statement about a functional interface is true?

A. It must be defined with the public access modifier.
B. It must be annotated with @FunctionalInterface.
C. It is declared with a single abstract method.
D. It is declared with a single default method.
E. It cannot have any private methods and static methods.

Answer: C

NEW QUESTION 22
Given:

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

What is the result?

A. 5
B. 3
C. 23
D. 25
E. 11

Answer: E

NEW QUESTION 23
Which two statements are correct about modules in Java? (Choose two.)

A. java.base exports all of the Java platforms core packages.
B. module-info.java can be placed in any folder inside module-path.
C. A module must be declared in module-info.java file.
D. module-info.java cannot be empty.
E. By default, modules can access each other as long as they run in the same folder.

Answer: AC

NEW QUESTION 25
Given:
String originalPath = “data\\projects\\a-project\\..\\..\\another-project”; Path path = Paths.get(originalPath); System.out.print(path.normalize());
What is the result?

A. data\another-project
B. data\projects\a-project\another-project
C. data\\projects\\a-project\\..\\..\\another-project
D. data\projects\a-project\..\..\another-project

Answer: D

Explanation:

```
1  import java.util.*;
2  import java.io.*;
3  import java.nio.file.*;
4
5  public class Test {
6
7    public static void main(String[] args) {
8      String originalPath = "data\\projects\\a-project\\..\\..\\another-project";
9  Path path = Paths.get(originalPath);
10 System.out.print(path.normalize());
11   }
12 }
```

Execute Mode, Version, Inputs & Arguments

JDK 11.0.4 Interactive Stdin Inp

CommandLine Arguments

Execute

Result

CPU Time: 0.19 sec(s), Memory: 31984 kilobyte(s)

```
data\projects\a-project\..\..\another-project
```

NEW QUESTION 28
Assume ds is a DataSource and the EMP table is defined appropriately.

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

What does executing this code fragment do?

A. inserts two rows (101, 'SMITH', 'HR') and (102, 'JONES', NULL)
B. inserts two rows (101, 'SMITH', 'HR') and (102, 'JONES', 'HR')
C. inserts one row (101, 'SMITH', 'HR')
D. throws a SQLException

Answer: C

NEW QUESTION 32
Which two are successful examples of autoboxing? (Choose two.)

A. String a = “A”;
B. Integer e = 5;
C. Float g = Float.valueOf(null);
D. Double d = 4;
E. Long c = 23L;
F. Float f = 6.0;

Answer: AB

NEW QUESTION 37
Which three initialization statements are correct? (Choose three.)

A. int x = 12_34;
B. short sh = (short)‘A’;
C. String contact# = “(+2) (999) (232)”;
D. boolean true = (4 == 4);
E. float x = 1.99;
F. int[][] e = {{1,1},{2,2}};
G. byte b = 10;char c = b;

Answer: ABF

NEW QUESTION 42
Given:

```
public class Main {
   public static void main(String[] args) {
      Consumer consumer = msg -> System.out::print; // line 1
      consumer.accept("Hello Lambda !");
   }
}
```

This code results in a compilation error.
Which code should be inserted on line 1 for a successful compilation?

A. Consumer consumer = msg -> { return System.out.print(msg); };
B. Consumer consumer = var arg > {System.out.print(arg);};
C. Consumer consumer = (String args) > System.out.print(args);
D. Consumer consumer = System.out::print;

Answer: D

Explanation:

```
1  import java.util.*;
2  import java.io.*;
3  import java.nio.file.*;
4  import java.util.List;
5  import java.util.function.Consumer;
6
7  public class Main {
8
9    public static void main(String[] args) {
10     Consumer consumer = System.out::print;
11     consumer.accept("Hello Lambda !");
12   }
13 }
```

Execute Mode, Version, Inputs & Arguments

JDK 11.0.4

CommandLine Arguments

Result

CPU Time: 0.16 sec(s), Memory: 32896 kilobyte(s)

```
Hello Lambda !
```

NEW QUESTION 43
Which two are functional interfaces? (Choose two.)

A.
```
@FunctionalInterface
interface MyRunnable {
   public void run();
}
```

B.
```
@FunctionalInterface
interface MyRunnable {
   public void run();
   public void call();
}
```

C.
```
interface MyRunnable {
   public default void run() {}
   public void run(String s);
}
```

D.
```
@FunctionalInterface
interface MyRunnable {
}
```

E.
```
interface MyRunnable {
   @FunctionalInterface
   public void run();
}
```

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

Answer: CE

NEW QUESTION 46
Given:

```
package test;
import java.time.*;
public class Diary {
   private LocalDate now = LocalDate.now();
   public LocalDate getDate() {
      return now;
   }
}
```

and

```
package test;
public class Tester {
   public static void main(String[] args) {
      Diary d = new Diary();
      System.out.println(d.getDate());
   }
}
```

Which statement is true?

A. Class Tester does not need to import java.time.LocalDate because it is already visible to members of the package test.
B. All classes from the package java.tim
C. are loaded for the class Diary.
D. Only LocalDate class from java.time package is loaded.
E. Tester must import java.time.LocalDate in order to compile.

Answer: A

NEW QUESTION 49
Given:

```
public class Main {
  public static void main(String[] args) {
    try (BufferedReader br = new BufferedReader(new InputStreamReader(System.in));) {
      String input = br.readLine();
      System.out.println ("Input String was: " + input);
    } catch (IOException e) {
      e.printStackTrace();
    }
  }
}
```

Which is true?

A. System.out is the standard output strea
B. The stream is open only when System.out is called.
C. System.in cannot reassign the other stream.
D. System.out is an instance of java.io.OutputStream by default.
E. System.in is the standard input strea
F. The stream is already open.

Answer: D

NEW QUESTION 53
Given:

```
public class Main {
   public static void main(String[] args) {
      Thread t1 = new Thread(new MyThread());
      Thread t2 = new Thread(new MyThread());
      Thread t3 = new Thread(new MyThread());

      t1.start();
      t2.run();
      t3.start();

      t1.start();
   }
}
class MyThread implements Runnable {
   public void run() {
      System.out.println("Running.");
   }
}
```

Which one is correct?

A. An IllegalThreadStateException is thrown at run time.
B. Three threads are created.
C. The compilation fails.
D. Four threads are created.

Answer: A

Explanation:

```
Running.
Running.
Running.

Exception in thread "main" java.lang.IllegalThreadStateException
    at java.base/java.lang.Thread.start(Thread.java:794)
    at Main.main(Main.java:12)
```

NEW QUESTION 57
Given:

```
public class Foo {
    private void print() {
        System.out.println("Bonjour le monde!");
    }
    public void foo() {
        print();
    }
}

public class Bar extends Foo {
    private void print() {
        System.out.println("Hello world!");
    }
    public void bar() {
        print();
    }
    public static void main(String... args) {
        Bar b = new Bar();
        b.foo();
        b.bar();
    }
}
```

What is the output?

A. Hello world!Bonjour le monde!
B. Hello world!Hello world!
C. Bonjour le monde!Hello world!

D. Bonjour le monde!Bonjour le monde!

Answer: C

Explanation:

NEW QUESTION 58
Given:

```
public interface EulerInterface {
    double getEulerValue();
}

public class EulerLambda {
    public static void main(String[] args) {
        EulerInterface  myEulerInterface;
        myEulerInterface = () -> "2.71828";
        System.out.println("Value of Euler = " + myEulerInterface.getEulerValue());
    }
}
```

What is the result?

A. It throws a runtime exception.
B. Value of Euler = 2.71828
C. The code does not compile.
D. Value of Euler = “2.71828”

Answer: C

NEW QUESTION 59
Given:

```
public class Main {
   public static void main(String[] args) {
      List 1 = new ArrayList();
      l.add(“hello”);
      l.add(“world”);
      print(1);
   }
   private static void print(List<String>... args) {
        for (List<String> str : args) {
          System.out.println (str);
        }
   }
}
```

Which annotation should be used to remove warnings from compilation?

A. @SuppressWarnings on the main and print methods
B. @SuppressWarnings(“unchecked”) on main and @SafeVarargs on the print method
C. @SuppressWarnings(“rawtypes”) on main and @SafeVarargs on the print method
D. @SuppressWarnings(“all”) on the main and print methods

Answer: B

Explanation:

```
13  @SuppressWarnings("unchecked")
14  public class Main {
15
16      public static void main(String[] args) {
17
18          List l = new ArrayList();
19          l.add("Hello");
20          l.add("world");
21          print(1);
22
23      }
24
25      private static void print(List<String>... args) {
26          for (List<String> str : args) {
27              System.out.println (str);
28
29          }
30      }
31      @SafeVarargs
32  }
```

NEW QUESTION 62
Given:

```
public class Employee {
   private String name;
   private LocalDate birthday;
   // the constructors, getters, and setters methods go here
}
```

and

```
List<Employee> roster = new ArrayList<>();
// ...
Predicate<Employee> y = (Employee e) -> e.getBirthday()
    .isBefore(IsoChronology.INSTANCE.date(1989, 1, 1));
Set<String> s1 = roster.stream()
// Line 1
```

Which code fragment on line 1 makes the s1 set contain the names of all employees born before January 1, 1989?

A.
```
.collect(Collectors.partitioningBy(y))
.get(true)
.stream()
.map(Employee::getName)
.collect(Collectors.toCollection(TreeSet::new));
```

B.
```
.collect(Collectors.partitioningBy(y))
.get(true)
.map(Employee::getName)
.collect(Collectors.toSet());
```

C.
```
.collect(Collectors.partitioningBy(y, Collectors.mapping(
    Employee::getName, Collectors.toSet())));
```

D.
```
.collect(Collectors.partitioningBy(y, Collectors.groupingBy(
    Employee::getName, Collectors.toCollection(TreeSet::new))));
```

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

Answer: B

NEW QUESTION 63
Given:

```
public class Main {
   public static void main(String[] args) {
      var numbers = List.of(1,2,3,4,5,6,7,8,9,10);
      Optional<Integer> result = numbers.stream().filter(x -> x % 3 != 0).reduce((i, j)
-> i + j);
      result.ifPresent(System.out::print); // line 1
```

Which is true about line 1?

A. If the value is not present, a NoSuchElementException is thrown at run time.
B. It always executes the System.out::print statement.
C. If the value is not present, a NullPointerException is thrown at run time.
D. If the value is not present, nothing is done.

Answer: D

Explanation:

```
1  import java.util.*;
2  import java.io.*;
3  import java.lang.Thread;
4  import java.util.ArrayList;
5  import java.util.LinkedList;
6  import java.util.List;
7  import java.util.function.Consumer;
8  import java.util.stream.Stream;
9  import java.util.stream.IntStream;
10 import java.util.Optional;
11
12
13 public class Main {
14     public static void main(String[] args) {
15         var numbers = List.of(1,2,3,4,5,6,7,8,9,10);
16         Optional<Integer> result = numbers.stream().filter (x -> x % 3 != 0).reduce( (i, j) -> i + j);
17
18     }
19 }
```

Result

CPU Time: 0.18 sec(s), Memory: 33380 kilobyte(s)

```
Doodle in Action.... Running the program...
```

NEW QUESTION 64
Given:

```
List<String> list = ... ;
list.forEach( x -> { System.out.println(x); } );
```

What is the type of x?

A. char
B. List<Character>
C. String
D. List<String>

Answer: C

NEW QUESTION 66
Which two modules include APIs in the Java SE Specification? (Choose two.)

A. java.logging
B. java.desktop
C. javafx
D. jdk.httpserver
E. jdk.jartool

Answer: AD

NEW QUESTION 68
Given:

```
public class Sportscar extends Automobile{
  private float turbo;
    ....
  public void setTurbo (float turbo){
     this.turbo = turbo;
  }
}
```

What is known about the Sportscar class?

A. The Sportscar class is a subclass of Automobile and inherits its methods.
B. The Sportscar subclass cannot override setTurbo method from the superclass Automobile.
C. The Sportscar class is a superclass that has more functionality than the Automobile class.
D. The Sportscar class inherits the setTurbo method from the superclass Automobile.

Answer: A

NEW QUESTION 70
Given:

```
List<Reader> dataFiles = new ArrayList<>();
File indexFile = new File("MyIndex.idx");
try (BufferedReader indexReader =
     new BufferedReader(new FileReader(indexFile))) {
   for(String file = indexReader.readbine(); file != null;
      file = indexReader.readLine()) {
    BufferedReader dataReader = new BufferedReader (
      new FileReader(new File(file))); // Line 1
    dataFiles.add(dataReader); // Line 2
    processData(dataReader); // Line 3
   }
}  catch (IOException ex) {
   ...
}  finally {
   for(Reader r : dataFiles) {
      try {
         r.close();
      } catch (IOException ex) (
        ...
      } // Line 4
   }
}
```

What will secure this code from a potential Denial of Service condition?

A. After Line 4, add indexReader.close().
B. On Line 3, enclose processData(dataReader) with try with resources.
C. After Line 3, add dataReader.close().
D. On Line 1, use try with resources when opening each dataReader.
E. Before Line 1, check the size of dataFiles to make sure it does not exceed a threshold.

Answer: B

NEW QUESTION 73
Given:

```
for(var i = 0; i < 10; i++) {
  switch(i%5) {
    case 2:
      i *= i;
      break;
    case 3:
      i++;
      break;
    case 1:
    case 4:
      i++;
      continue;
    default:
      break;
  }
  System.out.print(i + " ");
  i++;
}
```

What is the result?

A. nothing
B. 10
C. 0 4 9

Answer: A

NEW QUESTION 78
.....

Thank You for Trying Our Product

We offer two products:

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

2nd - Questons and Answers in PDF Format

1Z0-819 Practice Exam Features:

* 1Z0-819 Questions and Answers Updated Frequently

* 1Z0-819 Practice Questions Verified by Expert Senior Certified Staff

* 1Z0-819 Most Realistic Questions that Guarantee you a Pass on Your FirstTry

* 1Z0-819 Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year