

1Z0-819 Dumps

Java SE 11 Developer

<https://www.certleader.com/1Z0-819-dumps.html>



NEW QUESTION 1

Given:

```
public class Tester {
    public static void main(String[] args) {
        StringBuilder sb = new StringBuilder(5);
        sb.append("HOWDY");
        sb.insert(0, ' ');
        sb.replace(3, 5, "LL");
        sb.insert(6, "COW");
        sb.delete(2, 7);
        System.out.println(sb.length());
    }
}
```

What is the result?

- A. 4
- B. 3
- C. An exception is thrown at runtime.
- D. 5

Answer: D

Explanation:

```
6 public class Tester {
7     public static void main(String[] args) {
8         StringBuilder sb = new StringBuilder (5);
9         sb.append ("HOWDY") ;
10        sb.insert (0, ' ');
11        sb.replace(3, 5, "LL");
12        sb.insert (6, "COW");
13        sb.delete(2, 7);
14        System.out.println(sb.length());
15    }
16 }
```

(command line arguments)

COMPILE & EXECUTE

PASTE SOURCE

Successfully compiled /tmp/java_82Tlan/Tester.java <-- main method

5

NEW QUESTION 2

Given:

```
package b;
public class Person {
    protected Person() {                //line 1
    }
}
```

and

```
package a;
import b.Person;
public class Main {                    //line 2
    public static void main(String[] args) {
        Person person = new Person(); //line 3
    }
}
```

Which two allow a.Main to allocate a new Person? (Choose two.)

- A. In Line 1, change the access modifier to privateprivate Person() {
- B. In Line 1, change the access modifier to publicpublic Person() {
- C. In Line 2, add extends Person to the Main classpublic class Main extends Person {and change Line 3 to create a new Main objectPerson person = new Main();
- D. In Line 2, change the access modifier to protectedprotected class Main {
- E. In Line 1, remove the access modifierPerson() {

Answer: BC

NEW QUESTION 3

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 4

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 5

Which two commands are used to identify class and module dependencies? (Choose two.)

- A. jmod describe
- B. java Hello.java
- C. jdeps --list-deps
- D. jar --show-module-resolution
- E. java --show-module-resolution

Answer: CE

NEW QUESTION 6

Given:

```
public interface API {    //line 1
    public void checkValue(Object value)
        throws IllegalArgumentException; //line 2
    public boolean isValueANumber(Object val) {
        if(val instanceof Number) {
            return true;
        }else {
            try {
                Double.parseDouble(val.toString());
                return true;
            }catch (NumberFormatException ex) {
                return false;
            }
        }
    }
}
```

Which two changes need to be made to make this class compile? (Choose two.)

- A. Change Line 1 to an abstract class:public abstract class API {
- B. Change Line 2 access modifier to protected:protected void checkValue(Object value)throws IllegalArgumentException;
- C. Change Line 1 to a class:public class API {
- D. Change Line 1 to extend java.lang.AutoCloseable:public interface API extends AutoCloseable {
- E. Change Line 2 to an abstract method:public abstract void checkValue(Object value)throws IllegalArgumentException;

Answer: CE

NEW QUESTION 7

Given:

```
1. {
2.     Iterator iter = List.of(1,2,3).iterator();
3.     while (iter.hasNext()) {
4.         foo(iter.next());
5.     }
6.     Iterator iter2 = List.of(1,2,3).iterator();
7.     while (iter.hasNext()) {
8.         bar(iter2.next());
9.     }
10. }
11. for (Iterator iter = List.of(1,2,3).iterator(); iter.hasNext(); ) {
12.     foo(iter.next());
13. }
14. for (Iterator iter2 = List.of(1,2,3).iterator(); iter.hasNext(); ) {
15.     bar(iter2.next());
16. }
```

Which loop incurs a compile time error?

- A. the loop starting line 11
- B. the loop starting line 7
- C. the loop starting line 14
- D. the loop starting line 3

Answer: C

NEW QUESTION 8

Given this enum declaration:

```
1. enum Letter {
2.     ALPHA(100), BETA(200), GAMMA(300);
3.     int v;
4.     Letter(int v) { this.v = v; }
5.     /* Insert code here */
6. }
```

Examine this code: System.out.println(Letter.values()[1]);

What code should be written at line 5 for this code to print 200?

- A. public String toString() { return String.valueOf(ALPHA.v); }
- B. public String toString() { return String.valueOf(Letter.values()[1]); }
- C. public String toString() { return String.valueOf(v); }
- D. String toString() { return "200"; }

Answer: C

Explanation:

```

13 public class Main {
14     enum Letter {
15         ALPHA(100), BETA(200), GAMMA(300);
16         int v;
17         Letter(int v) { this.v = v; }
18         public String toString() { return String.valueOf(v); }
19     }
20
21
22 }
23 public static void main (String[] args) {
24     System.out.println(Letter.values() [1]);
25 }
26 }
27
28

```

Result

compiled and executed in 1.099 sec(s)

200

NEW QUESTION 9

Given an application with a main module that has this module-info.java file:

```

module main {
    exports country;
    uses country.CountryDetails;
}

```

Which two are true? (Choose two.)

- A. A module providing an implementation of country.CountryDetails can be compiled and added without recompiling the main module.
- B. A module providing an implementation of country.CountryDetails must have a requires main; directive in its module-info.java file.
- C. An implementation of country.countryDetails can be added to the main module.
- D. To compile without an error, the application must have at least one module in the module source path that provides an implementation of country.CountryDetails.
- E. To run without an error, the application must have at least one module in the module path that provides an implementation of country.CountryDetails.

Answer: BD

NEW QUESTION 10

Given:

```

public class MethodTest {
    // line 1
}

```

Which two method implementations are correct, when inserted independently in line 1? (Choose two.)

A.

```
public boolean methodD(int x) {  
    return x > 0;  
}
```

B.

```
public String methodB() {  
    System.out.println("methodB");  
}
```

C.

```
public char methodE (String msg) {  
    return msg;  
}
```

D.

```
public void methodC(int x) {  
    return ++x;  
}
```

E.

```
public void methodA() {  
    System.out.println("methodA");  
}
```

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

Answer: AE

NEW QUESTION 10

Given:

```
1. interface Pastry {  
2.     void getIngredients();  
3. }  
4. abstract class Cookie implements Pastry {}  
5.  
6. class ChocolateCookie implements Cookie {  
7.     public void getIngredients() {}  
8. }  
9. class CoconutChocolateCookie extends ChocolateCookie {  
10.     void getIngredients(int x) {}  
11. }
```

Which is true?

- A. The compilation fails due to an error in line 6.
- B. The compilation succeeds.
- C. The compilation fails due to an error in line 4.
- D. The compilation fails due to an error in line 10.
- E. The compilation fails due to an error in line 7.
- F. The compilation fails due to an error in line 9.
- G. The compilation fails due to an error in line 2.

Answer: A

NEW QUESTION 11

Which two statements set the default locale used for formatting numbers, currency, and percentages? (Choose two.)

- A. `Locale.setDefault(Locale.Category.FORMAT, "zh-CN");`
- B. `Locale.setDefault(Locale.Category.FORMAT, Locale.CANADA_FRENCH);`
- C. `Locale.setDefault(Locale.SIMPLIFIED_CHINESE);`
- D. `Locale.setDefault("en_CA");`
- E. `Locale.setDefault("es", Locale.US);`

Answer: BD

NEW QUESTION 12

Given:

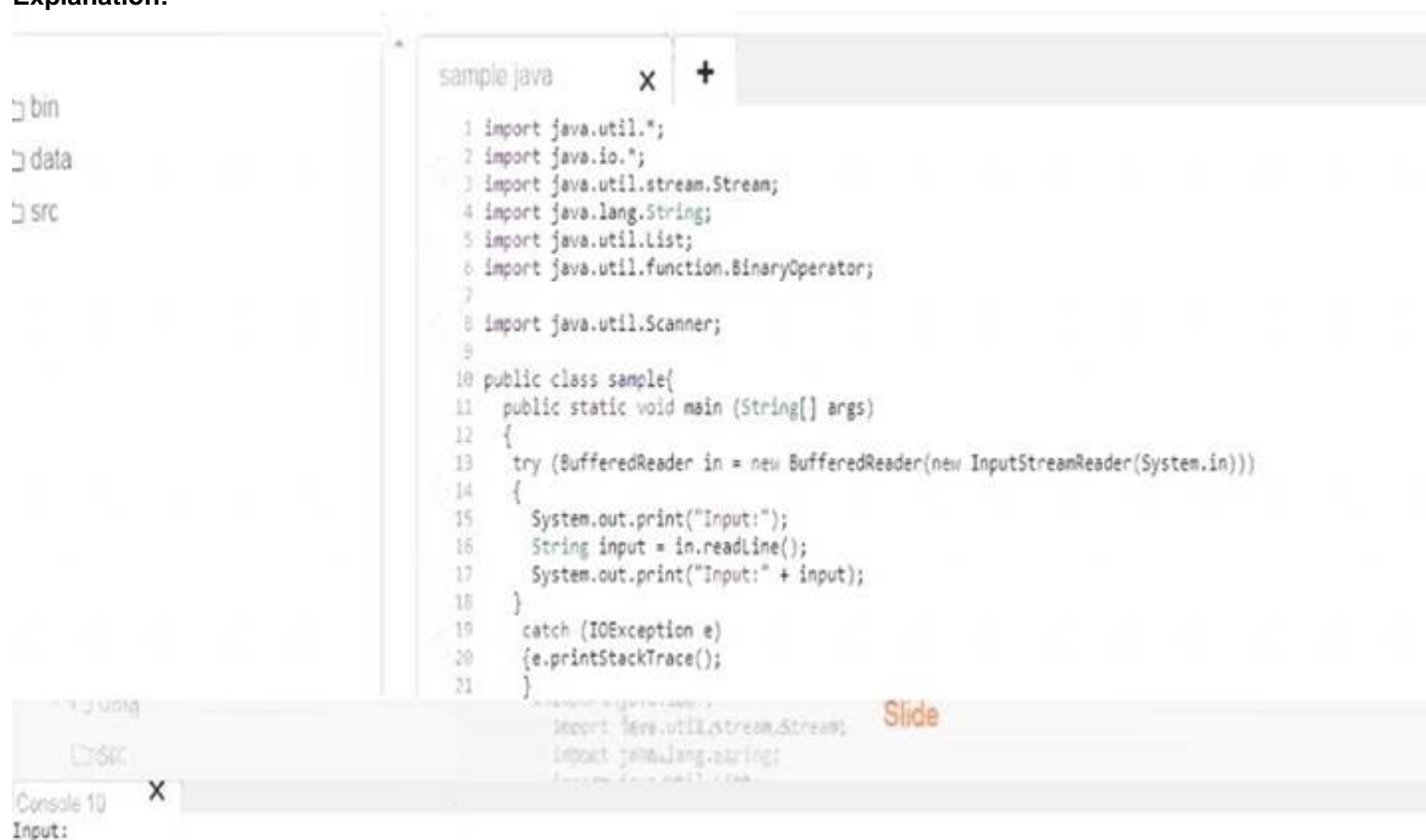
```
public class Main {  
    public static void main(String[] args) {  
        try(BufferedReader in = new BufferedReader(new InputStreamReader(System.in))) {  
            System.out.print("Input: ");  
            String input = in.readLine();  
            System.out.println("Echo: " + input); 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

And the command: java Main Helloworld What is the result ?

- A. Input: Echo:
- B. Input: Helloworld Echo: Helloworld
- C. Input:Then block until any input comes from System.in.
- D. Input:Echo: Helloworld
- E. A NullPointerException is thrown at run time.

Answer: C

Explanation:



NEW QUESTION 15

Given:

```
public class SerializedMessage implements Serializable {
    String message;
    LocalDateTime createdAt;
    transient LocalDateTime updatedAt;
    SerializedMessage(String message) {
        this.message = message;
        this.createdAt = LocalDateTime.now();
    }
    private void readObject (ObjectInputStream in) {
        try {
            in.defaultReadObject();
            this.updatedAt = LocalDateTime.now();
        } catch (IOException | ClassNotFoundException e) {
            e.printStackTrace();
        }
    }
}
```

When is the readObject method called?

- A. before this object is deserialized
- B. after this object is deserialized
- C. before this object is serialized
- D. The method is never called.
- E. after this object is serialized

Answer: B

NEW QUESTION 16

Given:

```
class Mycar {
}
```

and

```
javac C:\workspace4\Mycar.java
```

What is the expected result of javac?

- A. javac fails to compile the class and prints the error message, C:\workspace4\Mycar.java:1:error: packagejava does not exist
- B. javac compiles Mycar.java without errors or warnings.
- C. javac fails to compile the class and prints the error message, C:\workspace4\Mycar.java:1:error: expected import java.lang
- D. javac fails to compile the class and prints the error message, Error: Could not find or load main class Mycar.class

Answer: B

NEW QUESTION 17

Given:

```
public class Main {

    public static void checkConfiguration(String filename) {
        File file = new File(filename);
        if(!file.exists()) {
            throw new Error("Fatal Error: Configuration File, "
                + filename + ", is missing.");
        }
    }

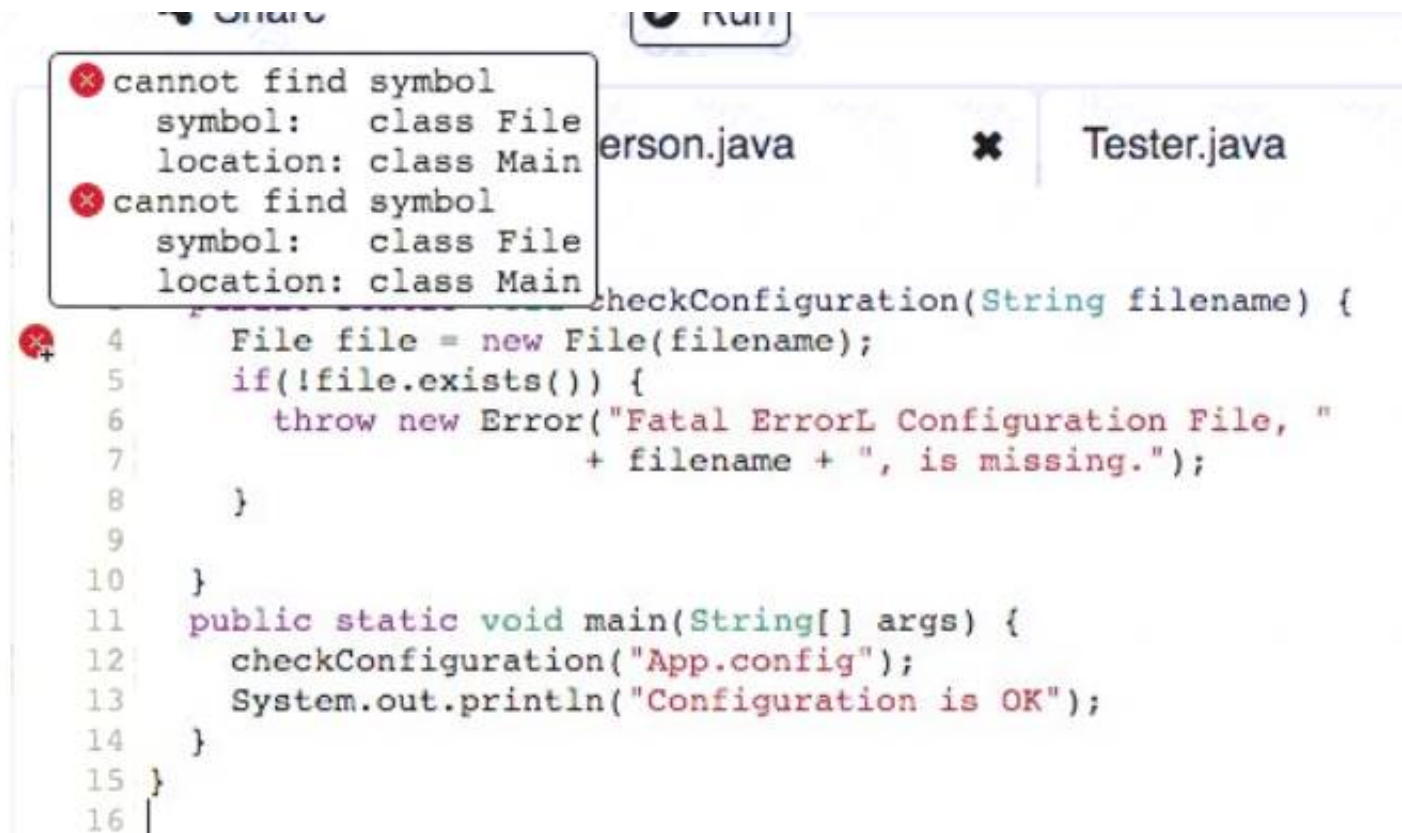
    public static void main(String[] args) {
        checkConfiguration("App.config");
        System.out.println("Configuration is OK");
    }
}
```

If file "App.config" is not found, what is the result?

- A. Configuration is OK
- B. The compilation fails.
- C. Exception in thread "main" java.lang.Error:Fatal Error: Configuration File, App.config, is missing.
- D. nothing

Answer: B

Explanation:



NEW QUESTION 21

Given:

```

public interface A {
    abstract void x();
}

```

and

```

public abstract class B /* position 1 */ {
    /* position 2 */
    public void x() { }
    public abstract void z();
}

```

and

```

public class C extends B implements A {
    /* position 3 */
}

```

Which code, when inserted at one or more marked positions, would allow classes B and C to compile?

- A. @Override // position 3 void x () {} // position 3 @Override // position 3 public void z() {} // position 3
- B. @Override // position 2 public void z() {} // position 3
- C. implements A // position 1 @Override // position 2
- D. public void z() {} // position 3

Answer: B

NEW QUESTION 25

Given:

```

public class Tester {
    public static void main(String[] args) {
        int x = 4;
        int y = 2;
        System.out.println(x+y+"=(x+y)="+x+y);
    }
}

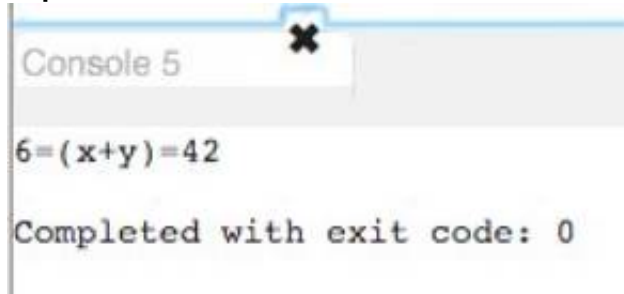
```

What is the result?

- A. An exception is thrown at runtime
- B. 42=(x+y)=42
- C. 42=(x+y)=6
- D. 6=(x+y)=42
- E. 6=(x+y)=6

Answer: D

Explanation:



NEW QUESTION 26

Which describes a characteristic of setting up the Java development environment?

- A. Setting up the Java development environment requires that you also install the JRE.
- B. The Java development environment is set up for all operating systems by default.
- C. You set up the Java development environment for a specific operating system when you install the JDK.
- D. Setting up the Java development environment occurs when you install an IDE before the JDK.

Answer: D

NEW QUESTION 30

Given:

```
class Myclass {
public static void main(String [] args) {
    System.out.println(arg[1] + "--" + arg[3] + "--" + arg[0]);
}
}
```

executed using this command: java Myclass My Car is red What is the output of this class?

- A. Car--red--My
- B. My--Car--is
- C. My--is--java
- D. java--Myclass--My
- E. Myclass--Car--red

Answer: A

NEW QUESTION 31

What makes Java dynamic?

- A. At runtime, classes are loaded as needed, and new code modules can be loaded on demand.
- B. The runtime can process machine language sources as well as executables from different language compilers.
- C. The Java compiler uses reflection to test if class methods are supported by resources of a target platform.
- D. The Java compiler preprocesses classes to run on specific target platforms.

Answer: A

NEW QUESTION 32

Given:

```
public class Test{
    private int num = 1;
    private int div = 0;

    public void divide() {
        try {
            num = num / div;
            System.out.print("Exception");
        }
        catch(ArithmeticException ae) { num = 100; }
        catch(Exception e) { num = 200; }
        finally { num = 300; }
        System.out.print(num);
    }
    public static void main(String args[])
    {
        Test test = new Test();
        test.divide();
    }
}
```

What is the output?

- A. 300
- B. Exception
- C. 200
- D. 100

Answer: A

Explanation:

```
1 public class Test{
2     private int num = 1;
3     private int div = 0;
4
5     public void divide() {
6         try {
7             num = num / div;
8             System.out.print("Exception");
9         }
10        catch(ArithmeticException ae) { num = 100; }
11        catch(Exception e) { num = 200; }
12        finally { num = 300; }
13        System.out.print(num);
14    }
15    public static void main(String args[])
16    {
17        Test test = new Test();
18        test.divide();
19    }
20 }
```

Execute Mode, Version, Inputs & Arguments

JDK 11.0.4



In

CommandLine Arguments

Result

CPU Time: 0.15 sec(s), Memory: 32484 kilobyte(s)

300

NEW QUESTION 34

Given:


```
public class Test {
    public static void doThings() throws GeneralException {
        try {
            throw new RuntimeException("Someting happened");
        } catch (Exception e) {
            throw new SpecificException(e.getMessage());
        }
    }
    public static void main(String args[]) {
        try{
            Test.doThings();
        } catch (Exception e) {
            System.out.println(e.getMessage());
        }
    }
}
class GeneralException /* line 1 */ {
    public GeneralException(String s) { super(s); }
}
class SpecificException /* line 2 */ {
    public SpecificException(String s) { super(s); }
}
```

Which option should you choose to enable the code to print Something happened?

- A. Add extends GeneralException on line 1.Add extends Exception on line 2.
- B. Add extends SpecificException on line 1.Add extends GeneralException on line 2.
- C. Add extends Exception on line 1.Add extends Exception on line 2.
- D. Add extends Exception on line 1.Add extends GeneralException on line 2.

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
8  public class Test {
9
10     public static void doThings() throws GeneralException {
11         try{
12             throw new RuntimeException("Something happened");
13         } catch (Exception e) {
14             throw new SpecificException (e.getMessage());
15         }
16     }
17 }
18
19     public static void main(String args[]) {
20         try{
21             Test.doThings();
22         }catch (Exception e) {
23             System.out.println(e.getMessage());
24         }
25     }
26     class GeneralException extends Exception {
27         public GeneralException(String s) { super(s); }
28     }
29     class SpecificException extends GeneralException {
30         public SpecificException(String s) { super(s);}
31     }
32 }
```

NEW QUESTION 38

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 41

Given:

```
public class X {
    private Collection collection;
    public void set(Collection collection) {
        this.collection = collection;
    }
}
```

and

```
public class Y extends X {
    public void set(Map<String,String> map) {
        super.set(map); // line 1
    }
}
```

Which two lines can replace line 1 so that the Y class compiles? (Choose two.)

- A. map.forEach((k, v) -> set(v));
- B. set(map.values());
- C. super.set(List<String> map)
- D. super.set(map.values());
- E. set(map)

Answer: BD

NEW QUESTION 43

Which is the correct order of possible statements in the structure of a Java class file?

- A. class, package, import
- B. package, import, class
- C. import, package, class
- D. package, class, import
- E. import, class, package

Answer: B

NEW QUESTION 45

Given:

```
public class FunctionalInterfaceTest {
    public static void main(String[] args) {
        List fruits = Arrays.asList("apple", "orange", "banana");
        Consumer<String> c = System.out::print;
        Consumer<String> output = c.andThen(x -> System.out.println(":" + x.toUpperCase
    ));
        fruits.forEach(output);
    }
}
```

What is the output?

- A. :APPLE:ORANGE:BANANAappleorangebanana

- B. :APPLE:ORANGE:BANANA
C. APPLE:apple ORANGE:orange BANANA:banana
D. appleorangebanana:APPLE:ORANGE:BANANA
E. apple:APPLE orange:ORANGE banana:BANANA

Answer: E

Explanation:

The screenshot shows a Java code editor with the following code:

```

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
9  public class FunctionalInterfaceTest {
10     public static void main (String[] args) {
11         List fruits = Arrays.asList("apple", "orange", "banana");
12         Consumer<String> c = System.out::print;
13         Consumer<String> output = c.andThen(x -> System.out.println(": " + x.toUpperCase()));
14
15         fruits.forEach(output);
16
17     }
18 }

```

Below the code editor, there is a section titled "Execute Mode, Version, Inputs & Arguments". It shows "JDK 11.0.4" and an "Interactive" checkbox. There is a "Stdin Inputs" field and a "CommandLine Arguments" field. An "Execute" button is visible.

The "Result" section shows the output of the program:

```

apple:APPLE
orange:ORANGE
banana:BANANA

```

NEW QUESTION 47

You are working on a functional bug in a tool used by your development organization. In your investigation, you find that the tool is executed with a security policy file containing this grant.

```

grant codebase "file:${klib.home}/j2se/home/klib.jar" {
    permission java.security.AllPermission;
};

```

What action should you take?

- A. Nothing, because it is an internal tool and not exposed to the public.
B. Remove the grant because it is excessive.
C. Nothing, because it is not related to the bug you are investigating.
D. File a security bug against the tool referencing the excessive permission granted.
E. Nothing, because listing just the required permissions would be an ongoing maintenance challenge.

Answer: D

NEW QUESTION 50

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 53

Given:

```
public interface TestInterface {  
    default void samplingProbeProcedure() {  
        probeProcedure();  
        System.out.println("Collect Sample");  
        System.out.println("Leave Asteroid");  
        System.out.println("Dock with Main Craft");  
    }  
    default void explosionProbeProcedure() {  
        probeProcedure();  
        System.out.println("Explode")  
    }  
}
```

Examine these requirements:

- Eliminate code duplication.
- Keep constant the number of methods other classes may implement from this interface. Which method can be added to meet these requirements?

- A.

```
private default void probeProcedure() {  
    System.out.println("Launch Probe");  
    System.out.println("Land on Asteroid");  
}
```
- B.

```
static void probeProcedure() {  
    System.out.println("Launch Probe");  
    System.out.println("Land on Asteroid");  
}
```
- C.

```
private void probeProcedure() {  
    System.out.println("Launch Probe");  
    System.out.println("Land on Asteroid");  
}
```
- D.

```
default void probeProcedure() {  
    System.out.println("Launch Probe");  
    System.out.println("Land on Asteroid");  
}
```

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

Answer: B

NEW QUESTION 55

Given:

```
void myLambda() {  
    int i = 25;  
    Supplier<Integer> foo = () -> i;  
    i++;  
    System.out.println(foo.get());  
}
```

Which is true?

- A. The code compiles but does not print any result.
- B. The code prints 25.
- C. The code does not compile.
- D. The code throws an exception at runtime.

Answer: C

NEW QUESTION 57

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 58**

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 60**

Given:

```
enum QUALITY {
    A(100), B(75), C(50);
    int percent;
    private QUALITY(int percent) {
        this.percent = percent;
    }
}
```

and checkQuality(QUALITY.A); and

```
void checkQuality(QUALITY q) {
    switch (q) {
        case /* Insert code here */ :
            System.out.println("Best");
            break;
        default :
            System.out.println("Not best");
            break;
    }
}
```

Which code fragment can be inserted into the switch statement to print Best?

- A. QUALITY.A.ValueOf()
- B. A
- C. A.toString()
- D. QUALITY.A

Answer: B**NEW QUESTION 63**

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:**

The screenshot shows a Java IDE interface. At the top, there is a code editor with the following Java code:

```

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 }

```

Below the code editor, there is a section titled "Execute Mode, Version, Inputs & Arguments". It includes a dropdown menu for the JDK version (JDK 11.0.4), a toggle switch for "Interactive" (which is currently turned off), and a text input field for "Stdin Input". Below these, there is a section for "CommandLine Arguments" with an empty text input field. A blue "Execute" button with a play icon is located to the right of the command line arguments field.

Below the execution controls, there is a "Result" section. It displays the CPU time as "0.19 sec(s)" and memory usage as "31984 kilobyte(s)". The output of the program is shown in a black box with white text: "data\\projects\\a-project\\..\\..\\another-project".

NEW QUESTION 68

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 72

Given:

```

public class Main {
    public static void main(String[] args) {
        int i = 1;
        for(String s : args) {
            System.out.println((i++) + ") " + s);
        }
    }
}

```

executed with this command: java Main one two three

What is the output of this class?

- A. The compilation fails.
- B. 1) one2) two3) three
- C. A java.lang.ArrayIndexOutOfBoundsException is thrown.
- D. 1) one

E. nothing

Answer: B

NEW QUESTION 76

Given:

```
public class Person {  
    private String name = "Joe Bloggs";  
    public Person(String name) {  
        this.name = name;  
    }  
    public String toString() {  
        return name;  
    }  
}
```

and

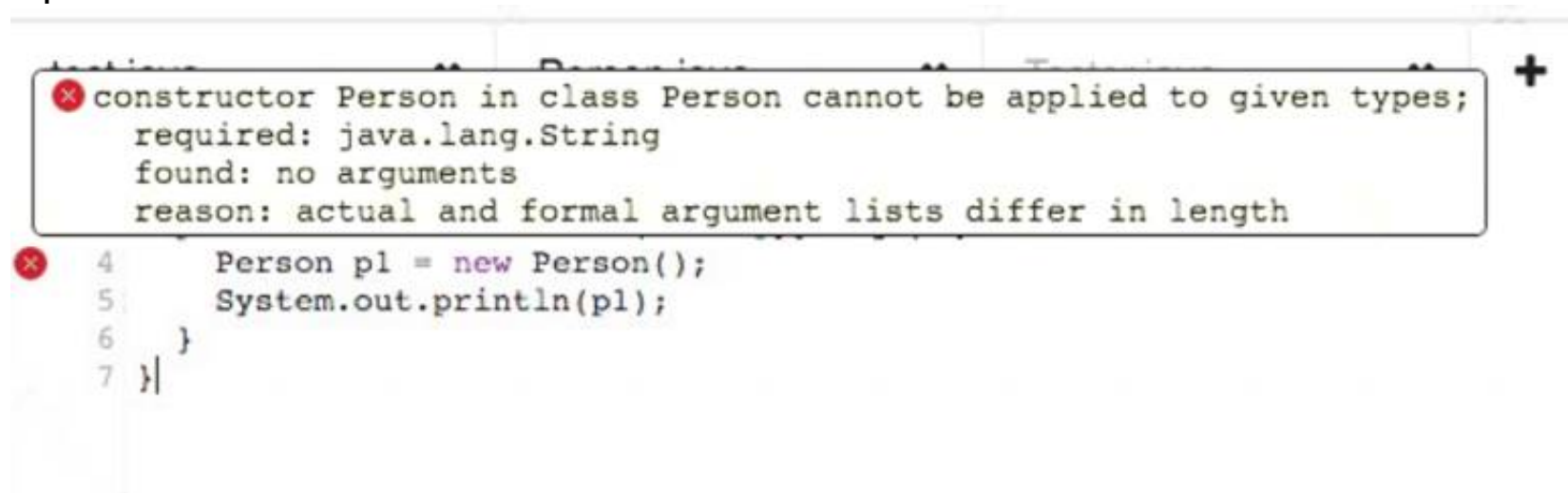
```
public class Tester {  
    public static void main(String[] args) {  
        Person p1 = new Person(); // line 1  
        System.out.println(p1);  
    }  
}
```

What is the result?

- A. null
- B. Joe Bloggs
- C. The compilation fails due to an error in line 1.
- D. p1

Answer: C

Explanation:



NEW QUESTION 80

Given:

```
public class Test {  
    public static void main(String[] args) {  
        int x;  
        int y = 5;  
        if (y > 2) {  
            x = ++y;  
            y = x + 7;  
        } else {  
            y++;  
        }  
        System.out.print(x + " " + y);  
    }  
}
```

What is the result?

- A. compilation error
- B. 0 5
- C. 6 13
- D. 5 12

Answer: A

Explanation:

```
1 public class Test {
2     public static void main (String[] args) {
3         int x;
4         int y = 5;
5         if (y > 2) {
6             x = ++y;
7             y = x + 7;
8         } else {
9             y++;
10        }
11        System.out.print(x + " " + y);
12    }
13 }
```

✖ variable x might not have been initialized

✖

NEW QUESTION 85

Which describes an aspect of Java that contributes to high performance?

- A. Java prioritizes garbage collection.
- B. Java has a library of built-in functions that can be used to enable pipeline burst execution.
- C. Java monitors and optimizes code that is frequently executed.
- D. Java automatically parallelizes code execution.

Answer: C

NEW QUESTION 89

Given:

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

    int getACount(int aCount) {
        return aCount;
    }
    int getTCount(int tCount) {
        return this.tCount;
    }
    int getCCount() {
        return getTotalCount() - this.aCount - getTCount(0) - gCount;
    }
    int getGCount() {
        return getGCount();
    }
    int getTotalCount() {
        return aCount + getTCount(0) + this.cCount + this.gCount;
    }
}
```

Which two methods facilitate valid ways to read instance fields? (Choose two.)

- A. getTCount
- B. getACount
- C. getTotalCount
- D. getCCount
- E. getGCount

Answer: CD

NEW QUESTION 91

Given:

```
import java.util.function.BiFunction;
public class Pair<T> {
    final BiFunction<T, T, Boolean> validator;
    T left = null;
    T right = null;
    private Pair() {
        validator=null;
    }
    Pair(BiFunction<T, T, Boolean> v, T x, T y) {
        validator = v;
        set(x, y);
    }
    void set(T x, T y) {
        if (!validator.apply(x, y)) throw new IllegalArgumentException();
        setLeft(x);
        setRight(y);
    }
    void setLeft(T x) {
        left = x;
    }
    void setRight(T y) {
        right = y;
    }
    final boolean isValid() {
        return validator.apply(left, right);
    }
}
```

It is required that if p instanceof Pair then p.isValid() returns true.
Which is the smallest set of visibility changes to insure this requirement is met?

- A. setLeft and setRight must be protected.
- B. left and right must be private.
- C. isValid must be public.
- D. left, right, setLeft, and setRight must be private.

Answer: B

NEW QUESTION 94

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

- A. Modular jars loaded from --module-path are automatic modules.
- B. Any named module can directly access all classes in an automatic module.
- C. Classes found in -classpath are part of an unnamed module.
- D. Modular jars loaded from -classpath are automatic modules.
- E. If a package is defined in both the named module and the unnamed module, then the package in the unnamed module is ignored.

Answer: AC

NEW QUESTION 98

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.time
- 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 99

Given:

```
public method foo() throws FooException {  
    ...  
}
```

and omitting the throws FooException clause results in a compilation error. Which statement is true about FooException?

- A. FooException is a subclass of RuntimeException.
- B. FooException is unchecked.
- C. The body of foo can only throw FooException.
- D. The body of foo can throw FooException or one of its subclasses.

Answer: D

NEW QUESTION 100

Given:

```
public class Test {  
    public static void main(String[] args) {  
        AnotherClass ac = new AnotherClass();  
        SomeClass sc = new AnotherClass();  
        ac = sc;  
        sc.methodA();  
        ac.methodA();  
    }  
}  
class SomeClass {  
    public void methodA() {  
        System.out.println("SomeClass#methodA()");  
    }  
}  
class AnotherClass extends SomeClass {  
    public void methodA() {  
        System.out.println("AnotherClass#methodA()");  
    }  
}
```

What is the result?

- A. A ClassCastException is thrown at runtime.
- B. AnotherClass#methodA()AnotherClass#methodA()
- C. The compilation fails.
- D. SomeClass#methodA()AnotherClass#methodA()
- E. AnotherClass#methodA()SomeClass#methodA()
- F. SomeClass#methodA()SomeClass#methodA()

Answer: C

Explanation:

```

1 public class Test {
2     public static void main (String[] args) {
3         AnotherClass ac = new AnotherClass();
4
5         ac = sc;
6         sc.methodA();
7         ac.methodA();
8     }
9 }
10 class SomeClass {
11     public void methodA() {
12         System.out.println("SomeClass#methodA()");
13     }
14 }
15 }
16 class AnotherClass extends SomeClass {
17     public void methodA() {
18         System.out.println("AnotherClass#methodA()");
19     }
20 }

```

✖ incompatible types: SomeClass cannot be converted to AnotherClass

NEW QUESTION 102

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 103

Given:

```

var data = new ArrayList<>(); data.add("Peter");
data.add(30); data.add("Market Road"); data.set(1, 25); data.remove(2); data.set(3, 1000L); System.out.print(data);

```

What is the output?

- A. [Market Road, 1000]
- B. [Peter, 30, Market Road]
- C. [Peter, 25, null, 1000]
- D. An exception is thrown at run time.

Answer: D

Explanation:

```
Console 1 ✕
Exception in thread "main" java.lang.IndexOutOfBoundsException: Index 3 out of bounds for length 2
    at java.base/jdk.internal.util.Preconditions.outOfBounds(Preconditions.java:64)
    at java.base/jdk.internal.util.Preconditions.outOfBoundsCheckIndex(Preconditions.java:70)
    at java.base/jdk.internal.util.Preconditions.checkIndex(Preconditions.java:248)
    at java.base/java.util.Objects.checkIndex(Objects.java:372)
    at java.base/java.util.ArrayList.set(ArrayList.java:472)
    at abc.main(abc.java:13)

Completed with exit code: 1
```

NEW QUESTION 104

Given:

```
import java.util.List;
import java.util.function.BinaryOperator;
public class Main {
    public static void main(String... args) {
        List<Employee> list = List.of(new Employee("John", 80000.0), new Employee("Scott",
90000.0));
        double starts = 0.0;
        double ratio = 1.0;
        BinaryOperator<Double> bo = (a, b) -> a + b;
        double totalSalary = list.stream().map(e -> e.getSalary() * ratio).reduce(starts, bo);
        // line 1
        System.out.println("Total salary = " + totalSalary);
    }
}

class Employee {
    String name;
    double salary;
    public Employee(String name, double salary) {
        this.name = name;
        this.salary = salary;
    }
    public String getName() { return name; }
    public double getSalary() { return salary; }
}
```

Which statement is equivalent to line 1?

- A. `double totalSalary = list.stream().map(e -> e.getSalary() * ratio).reduce(bo).ifPresent(p -> p.doubleValue());`
- B. `double totalSalary = list.stream().mapToDouble(e -> e.getSalary() * ratio).sum;`
- C. `double totalSalary = list.stream().map(Employee::getSalary * ratio).reduce(bo).orElse(0.0);`
- D. `double totalSalary = list.stream().mapToDouble(e -> e.getSalary() * ratio).reduce(starts, bo);`

Answer: C

Explanation:


```

Employee.java x Main.java x +
1 import java.util.List;
2 Employee.java:8:util.function.BinaryOperator;
3
4 public class Main {
5     public static void main (String... args) {
6         List<Employee> list = List.of(new Employee("John", 80000.0), new Employee("Scott", 90000.0));
7         double starts = 0.0;
8         double ratio = 1.0;
9         BinaryOperator<Double> bo = (a, b) -> a + b;
10        double totalSalary = list.stream().map(e -> e.getSalary() * ratio).reduce(starts, bo);
11        //line 1
12        System.out.println("Total salary = " + totalSalary);
13    }
14
15 }
16

Console 1 x
Total salary = 170000.0
Completed with exit code: 0

```

NEW QUESTION 109

Given:

```

public class Person {
    private String name;
    public Person(String name) {
        this.name = name;
    }
    public String toString() {
        return name;
    }
}

```

and

```

public class Tester {
    public static void main(String[] args) {
        Person p = null;
        checkPerson(p);
        System.out.println(p);
        p = new Person("Mary");
        checkPerson(p);
        System.out.println(p);
    }
    public static Person checkPerson(Person p) {
        if (p == null) {
            p = new Person("Joe");
        }else{
            p = null;
        }
        return p;
    }
}

```

What is the result?

- A. JoeMarry
- B. Joenull
- C. nullnull
- D. nullMary

Answer: D

Explanation:

```

Console 1 x Console 2 x Console 3 x
null
Mary
Completed with exit code: 0

```

NEW QUESTION 113

Given:

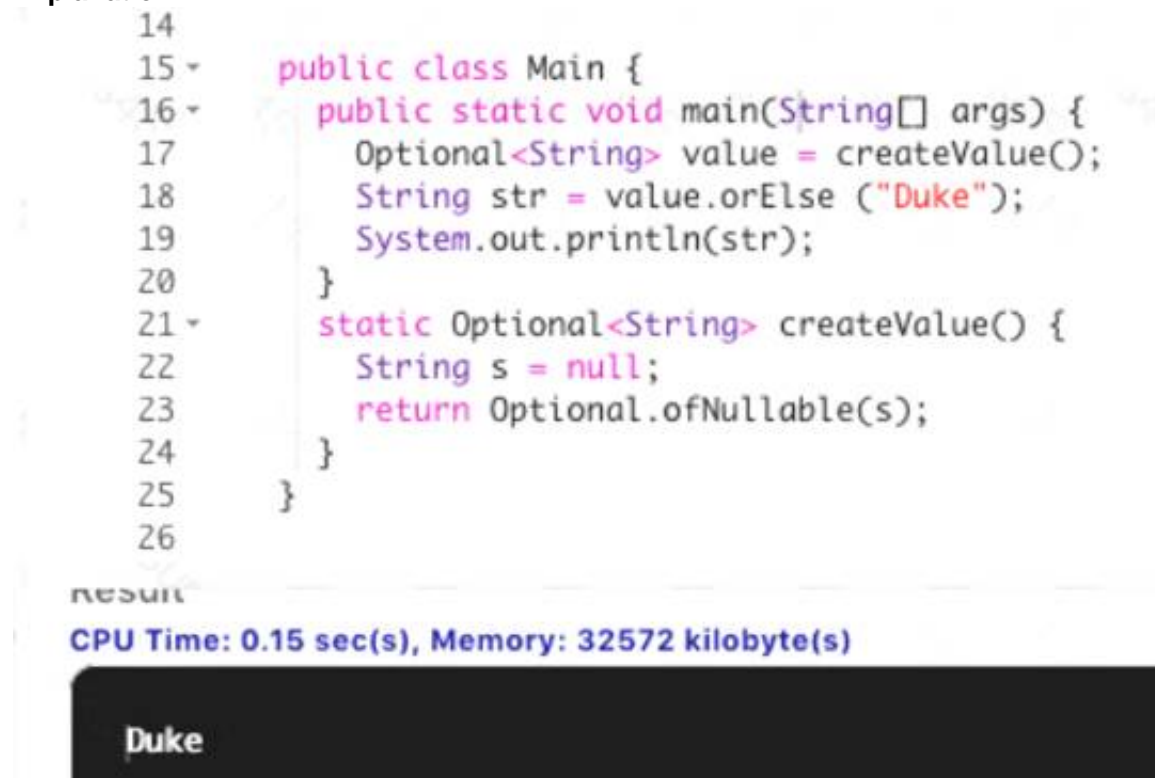

```
public class Main {  
    public static void main(String[] args) {  
        Optional<String> value = createValue();  
        String str = value.orElse ("Duke");  
        System.out.println(str);  
    }  
    static Optional<String> createValue() {  
        String s = null;  
        return Optional.ofNullable(s);  
    }  
}
```

What is the output?

- A. null
- B. A NoSuchElementException is thrown at run time.
- C. Duke
- D. A NullPointerException is thrown at run time.

Answer: C

Explanation:



NEW QUESTION 115

Which is a proper JDBC URL?

- A. jdbe.mysql.com://localhost:3306/database
- B. http://localhost.mysql.com:3306/database
- C. http://localhostmysql.jdbc:3306/database
- D. jdbc:mysql://localhost:3306/database

Answer: D

NEW QUESTION 119

Given:

```
public class X {  
}
```

and

```
public final class Y extends X {  
}
```

What is the result of compiling these two classes?

- A. The compilation fails because there is no zero args constructor defined in class X.
- B. The compilation fails because either class X or class Y needs to implement the toString() method.
- C. The compilation fails because a final class cannot extend another class.
- D. The compilation succeeds.

Answer: B

Explanation:

```

13
14 public class Main {
15     public static void main (String[] args) {
16         public class X {
17
18         }
19
20     public final class Y extends X {
21
22     }
23 }
24

```

NEW QUESTION 124

Given:

```

public class Main {
    public static void main(String[] args) {
        try {
            Path path = Paths.get("/u01/work/filestore.txt");
            boolean result = Files.deleteIfExists(path);
            if(result) System.out.println(path + "is deleted.");
            else System.out.println(path + "is not deleted.");
        } catch(IOException e) {
            System.out.println("Exception");
        }
    }
}

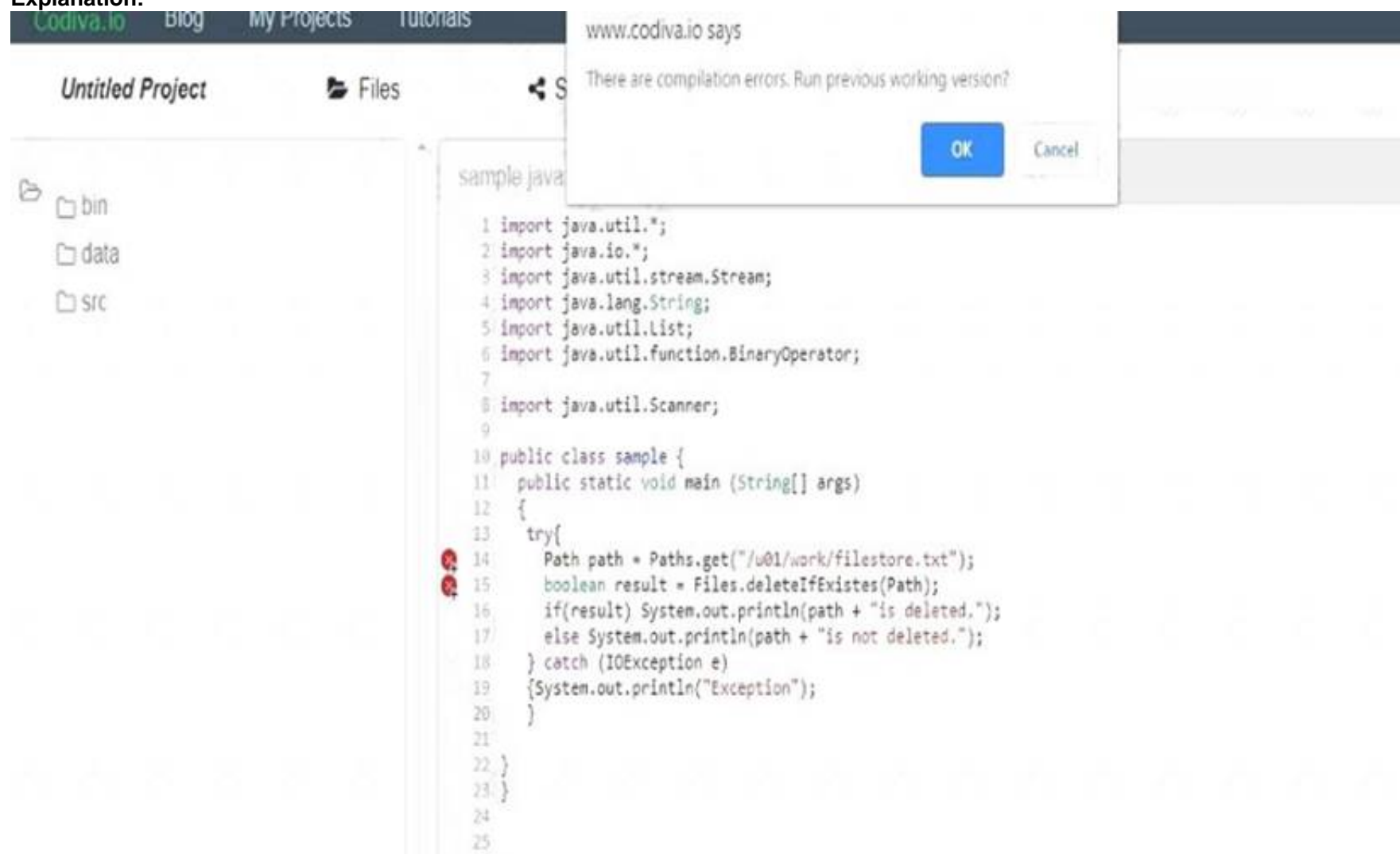
```

Assume the file on path does not exist. What is the result?

- A. The compilation fails.
- B. /u01/work/filestore.txt is not deleted.
- C. Exception
- D. /u01/work/filestore.txt is deleted.

Answer: A

Explanation:



NEW QUESTION 126

Given:

```
enum Color implements Serializable {  
    R(1), G(2), B(3);  
    int c;  
    public Color(int c) {  
        this.c = c;  
    }  
}
```

What action ensures successful compilation?

- A. Replace public Color(int c) with private Color(int c).
- B. Replace int c; with private int c;.
- C. Replace int c; with private final int c;.
- D. Replace enum Color implements Serializable with public enum Color.
- E. Replace enum Color with public enum Color.

Answer: A

Explanation:

```
1  
2 import java.io.*;  
3 import java.util.*;  
4 class Hello {  
5  
6  
7     enum Color implements Serializable {  
8         R(1), G(2), B(3);  
9         int c;  
10        private Color (int c) {  
11            this.c = c;  
12        }  
13    }  
14 }
```

NEW QUESTION 131

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 136

Given:

```
package test.t1;
public class A {
    public int x = 42;
    protected A() {}           // line 1
}
```

and

```
package test.t2;
import test.t1.*;
public class B extends A {
    int x = 17;                 // line 2
    public B() { super(); }     // line 3
}
```

and

```
package test;
import test.t1.*;
import test.t2.*;
public class Tester {
    public static void main(String[] args) {
        A obj = new B();        // line 4
        System.out.println(obj.x); // line 5
    }
}
```

What is the result?

- A. 42
- B. The compilation fails due to an error in line 4.
- C. 17
- D. The compilation fails due to an error in line 3.
- E. The compilation fails due to an error in line 2.
- F. The compilation fails due to an error in line 1.
- G. The compilation fails due to an error in line 5.

Answer: A

NEW QUESTION 138

Given:

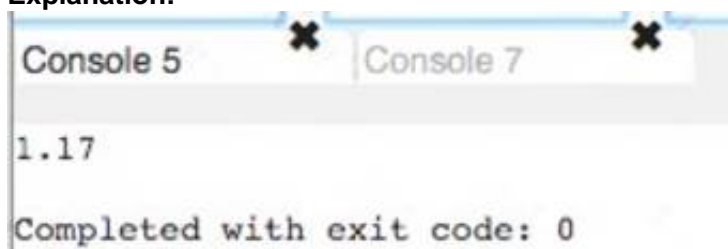
```
public class Tester {
    public static void main(String[] args) {
        byte x = 7, y = 6;
        // line 1
        System.out.println(z);
    }
}
```

Which expression when added at line 1 will produce the output of 1.17?

- A. float z = (float)(Math.round((float)x/y*100)/100);
- B. float z = Math.round((int)(x/y),2);
- C. float z = Math.round((float)x/y,2);
- D. float z = Math.round((float)x/y*100)/(float)100;

Answer: D

Explanation:



NEW QUESTION 140

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 143

Given:

```
List<Reader> dataFiles = new ArrayList<>();
File indexFile = new File("MyIndex.idx");
try (BufferedReader indexReader =
    new BufferedReader(new FileReader(indexFile))) {
    for(String file = indexReader.readLine(); 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 144

Given:

```
public class Main {
    public static void main(String[] args) {
        for(int i = 0; i < args.length; i++) {
            System.out.println(i + "). " + args[i]);
            switch(args[i]) {
                case "one":
                    continue;
                case "two":
                    i--;
                    continue;
                default:
                    break;
            }
        }
    }
}
```

executed with this command: java Main one two three What is the result?

- A. 0). one
- B. 0). one1). two2). three
- C. The compilation fails.
- D. It creates an infinite loop printing:0). one1). two1). two...
- E. A java.lang.NullPointerException is thrown.

Answer: D

NEW QUESTION 146

Given:

```
public class Price {
    private final double value;
    public Price(String value) {
        this(Double.parseDouble(value));
    }
    public Price(double value) {
        this.value = value;
    }
    public Price () {}
    public double getValue() { return value; }
    public static void main(String[] args) {
        Price p1 = new Price("1.99");
        Price p2 = new Price(2.99);
        Price p3 = new Price();
        System.out.println(p1.getValue()+" "+p2.getValue()+" "+p3.getValue());
    }
}
```

What is the result?

- A. The compilation fail
- B. 1.99,2.99,0
- C. 1.99,2.99,0.0
- D. 1.99,2.99

Answer: A

Explanation:

```
1
2 public class Price {
3     private final double value;
4     public Price(String value) {
5         this(Double.parseDouble (value));
6     }
7     public Price(double value) {
8         this.value = value;
9     }
10    public Price (){}
11    public double getValue() { return value; }
12    public static void main (String[] args) {
13        Price p1 = new Price("1.99");
14        Price p2 = new Price("2.99");
15        Price p3 = new Price();
16        System.out.println(p1.getValue()+" "+p2.getValue()+" "+p3.getValue());
17    }
18 }
```

✖ variable value might not have been initialized

NEW QUESTION 150

Given:

```
import java.util.*;

public class Main {
    static Map<String, String> map = new HashMap<>();
    static List<String> keys =
        new ArrayList<>(List.of("A", "B", "C", "D"));
    static String[] values =
        {"one", "two", "three", "four" };

    static {
        for(var i = 0; i < keys.size(); i++) {
            map.put(keys.get(i), values[i]);
        }
    }

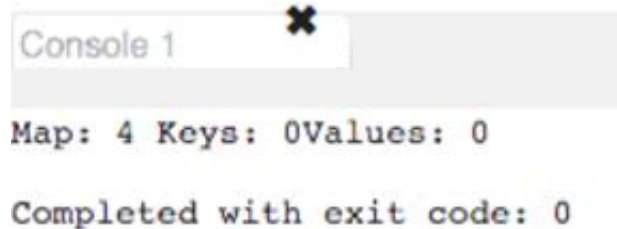
    public static void main(String[] args) {
        keys.clear();
        values = new String[0];
        System.out.println("Map: " + map.size() +
            " Keys: " + keys.size() +
            " Values: " + values.length);
    }
}
```

What is the result?

- A. Map: 0 Keys: 0 Values: 0
- B. The compilation fails.
- C. Map: 4 Keys: 4 Values: 4
- D. Map: 4 Keys: 0 Values: 0
- E. Map: 0 Keys: 4 Values: 4

Answer: D

Explanation:



The screenshot shows a console window titled 'Console 1' with the following output:

```
Map: 4 Keys: 0Values: 0

Completed with exit code: 0
```

NEW QUESTION 155

Given:

```
public class Foo {
    public static void main(String... args) {
        for (var x : args) {
            System.out.println(x);
        }
    }
}
```

What is the type of the local variable x?

- A. Character
- B. char
- C. String[]
- D. String

Answer: D

NEW QUESTION 156

.....

Thank You for Trying Our Product

* 100% Pass or Money Back

All our products come with a 90-day Money Back Guarantee.

* One year free update

You can enjoy free update one year. 24x7 online support.

* Trusted by Millions

We currently serve more than 30,000,000 customers.

* Shop Securely

All transactions are protected by VeriSign!

100% Pass Your 1Z0-819 Exam with Our Prep Materials Via below:

<https://www.certleader.com/1Z0-819-dumps.html>