

Oracle

Exam Questions 1Z0-819

Java SE 11 Developer



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

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 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:

```
import java.time.LocalDate;
import static java.time.DayOfWeek.*;
public class Main {
    public static void main(String[] args) {
        var today = LocalDate.now().with(TUESDAY).getDayOfWeek();
        switch(today) {
            case SUNDAY:
            case SATURDAY:
                System.out.println("Weekend");
                break;
            case MONDAY:  FRIDAY:
                System.out.println("Working");
            default:
                System.out.println("Unknown");
        }
    }
}
```

What is the result?

- A. WorkingUnknown
- B. Unknown
- C. TuesdayUnknown
- D. The compilation fails.
- E. Tuesday
- F. Working

Answer: B

Explanation:



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:

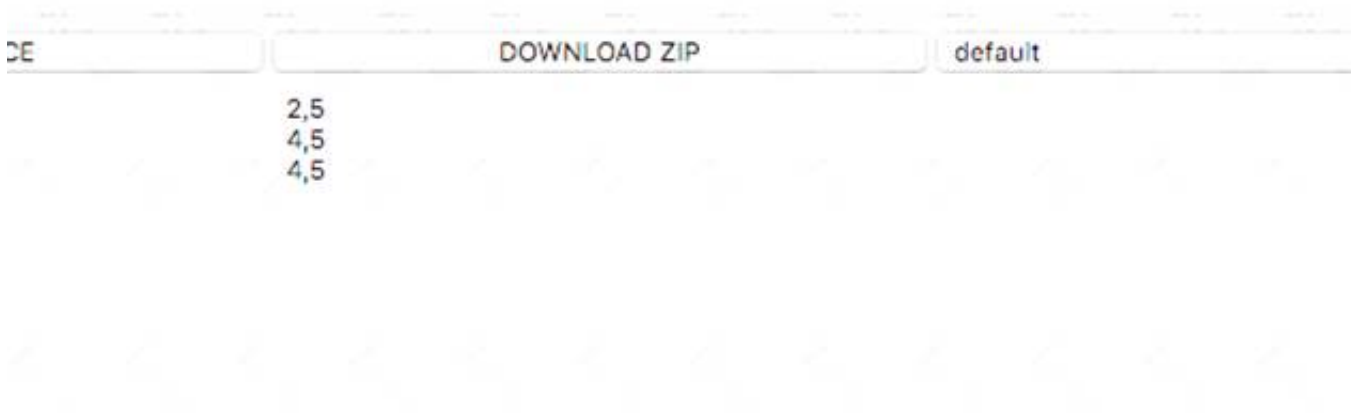
```
public class Tester {
    private int x;
    private static int y;
    public static void main(String[] args) {
        Tester t1 = new Tester();
        t1.x = 2;
        Tester.y = 3;
        Tester t2 = new Tester();
        t2.x = 4;
        t2.y = 5;
        System.out.println(t1.x+", "+t1.y);
        System.out.println(t2.x+", "+Tester.y);
        System.out.println(t2.x+", "+t1.y);
    }
}
```

What is the result?

- A. 2,34,34,5
- B. 2,34,54,5
- C. 2,54,54,5
- D. 2,34,54,3

Answer: C

Explanation:



NEW QUESTION 7

Which two statements are true about the modular JDK? (Choose two.)

- A. The foundational APIs of the Java SE Platform are found in the java.base module.
- B. An application must be structured as modules in order to run on the modular JDK.
- C. It is possible but undesirable to configure modules' exports from the command line.
- D. APIs are deprecated more aggressively because the JDK has been modularized.

Answer: AC

NEW QUESTION 8

Given:

```
public class Tester {  
    public static void main(String[] args) {  
        char letter = 'b';  
        int i = 0;  
        switch(letter) {  
            case 'a':  
                i++;  
                break;  
            case 'b':  
                i++;  
            case 'c' | 'd': // line 1  
                i++;  
            case 'e':  
                i++;  
                break;  
            case 'f':  
                i++;  
                break;  
            default:  
                System.out.print(letter);  
        }  
        System.out.println(i);  
    }  
}
```

What is the result?

- A. b1
- B. 2
- C. b2
- D. 1
- E. b3
- F. 3
- G. The compilation fails due to an error in line 1.

Answer: F

Explanation:

Result

CPU Time: 0.23 sec(s), Memory: 32708 kilobyte(s)

3

NEW QUESTION 9

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 10

Examine this excerpt from the declaration of the java.se module:

```

module java.se {
    ...
    requires transitive java.sql;
    ...
}

```

What does the transitive modifier mean?

- A. Only a module that requires the java.se module is permitted to require the java.sql module.
- B. Any module that requires the java.se module does not need to require the java.sql module.
- C. Any module that attempts to require the java.se module actually requires the java.sql module instead.
- D. Any module that requires the java.sql module does not need to require the java.se module.

Answer: A

NEW QUESTION 10

Given:

jdeps -jdkinternals C:\workspace4\SimpleSecurity\jar\classes.jar

Which describes the expected output?

- A. jdeps lists the module dependencies and the package names of all referenced JDK internal API
- B. If any are found, the suggested replacements are output in the console.
- C. jdeps outputs an error message that the -jdkinternals option requires either the -summary or the -verbose options to output to the console.
- D. The -jdkinternals option analyzes all classes in the .jar and prints all class-level dependencies.
- E. The -jdkinternals option analyzes all classes in the .jar for class-level dependencies on JDK internal API
- F. If any are found, the results with suggested replacements are output in the console.

Answer: A

Explanation:

-jdkinternals option analyzes all classes in the .jar for class-level dependencies on JDK internal APIs. If any are found, the results with suggested replacements are output in the console.

NEW QUESTION 11

Given:

```
package A;
class Test {
    String name;
    public Test(String name) {
        this.name = name;
    }
    public String toString() {
        return name;
    }
}
```

and

```
package B;
import A.Test;
public class Main {
    public static void main(String[] args) {
        Test test = new Test("Student");
        System.out.println(test);
    }
}
```

What is the result?

- A. null
- B. nothing
- C. It fails to compile.
- D. java.lang.IllegalAccessException is thrown.
- E. Student

Answer: C

NEW QUESTION 12

Given:

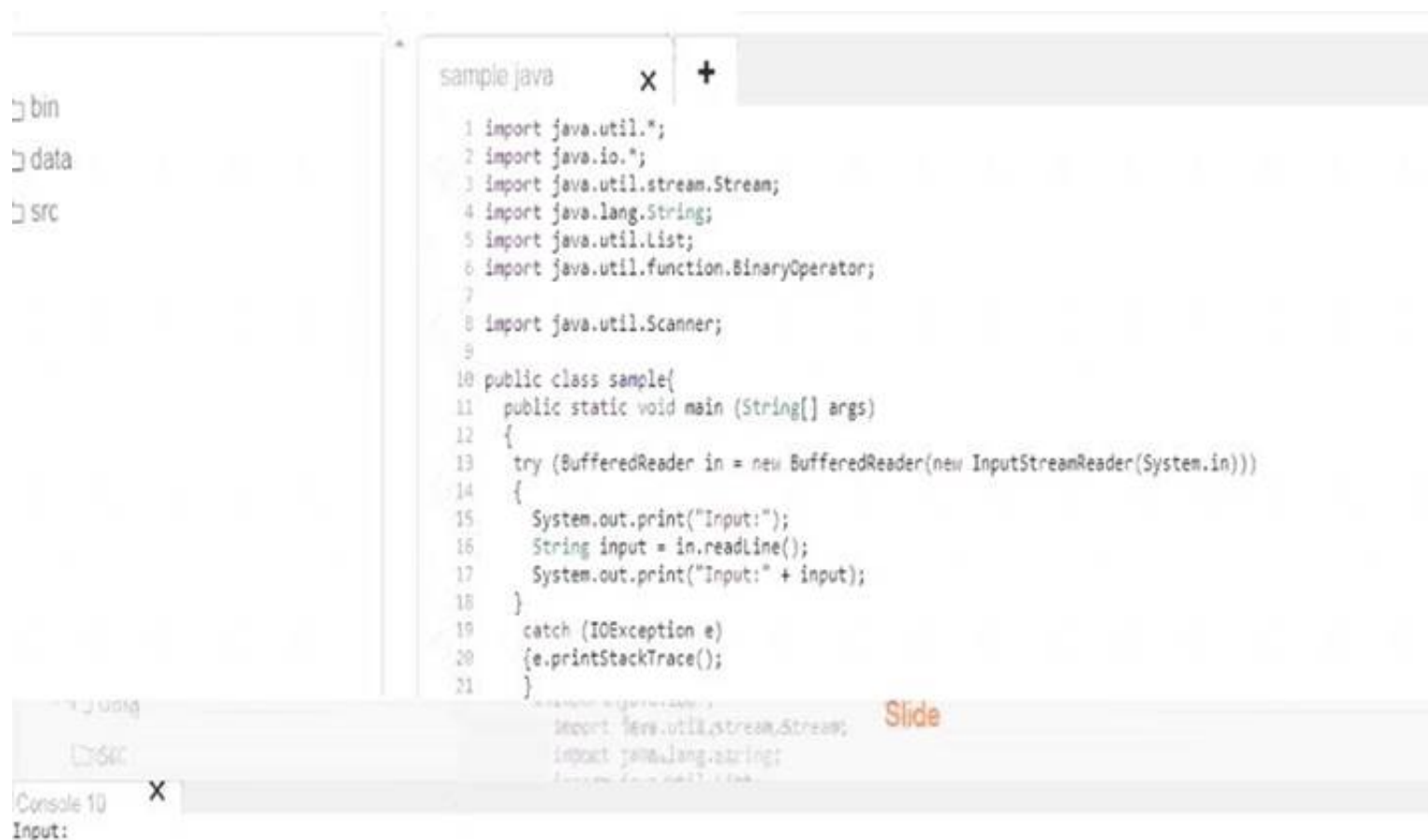
```
public class Main {
    public static void main(String[] args) {
        try(BufferedReader in = new BufferedReader(new InputStreamReader(System.in))) {
            System.out.print("Input: ");
            String input = in.readLine();
            System.out.println("Echo: " + input);
        } catch (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 13

Given:

```
public class Main {
    class Student {                                // line 1
        String classname;
        Student(String classname) {                // line 2
            this.classname = classname;
        }
    }
    public static void main(String[] args) {
        var student = new Student("Biology"); // line 3
    }
}
```

Which two independent changes will make the Main class compile? (Choose two.)

- A. Move the entire Student class declaration to a separate Java file, Student.java.
- B. Change line 2 to public Student(String classname).
- C. Change line 1 to public class Student {.
- D. Change line 3 to Student student = new Student("Biology");.
- E. Change line 1 to static class Student {.

Answer: BD

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     class Student {
15         String classname;
16         public Student (String classname) {
17             this.classname = classname;
18         }
19
20     }
21     public static void main (String[] args) {
22         var student = new Student ("Biology");
23     }
24 }
```

NEW QUESTION 16

Which command line runs the main class com.acme.Main from the module com.example?

- A. java --module-path mods com.example/com.acme.Main
- B. java -classpath com.example.jar com.acme.Main
- C. java --module-path mods -m com.example/com.acme.Main
- D. java -classpath com.example.jar -m com.example/com.acme.Main

Answer: D

NEW QUESTION 21

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 22

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.");
        }
    }

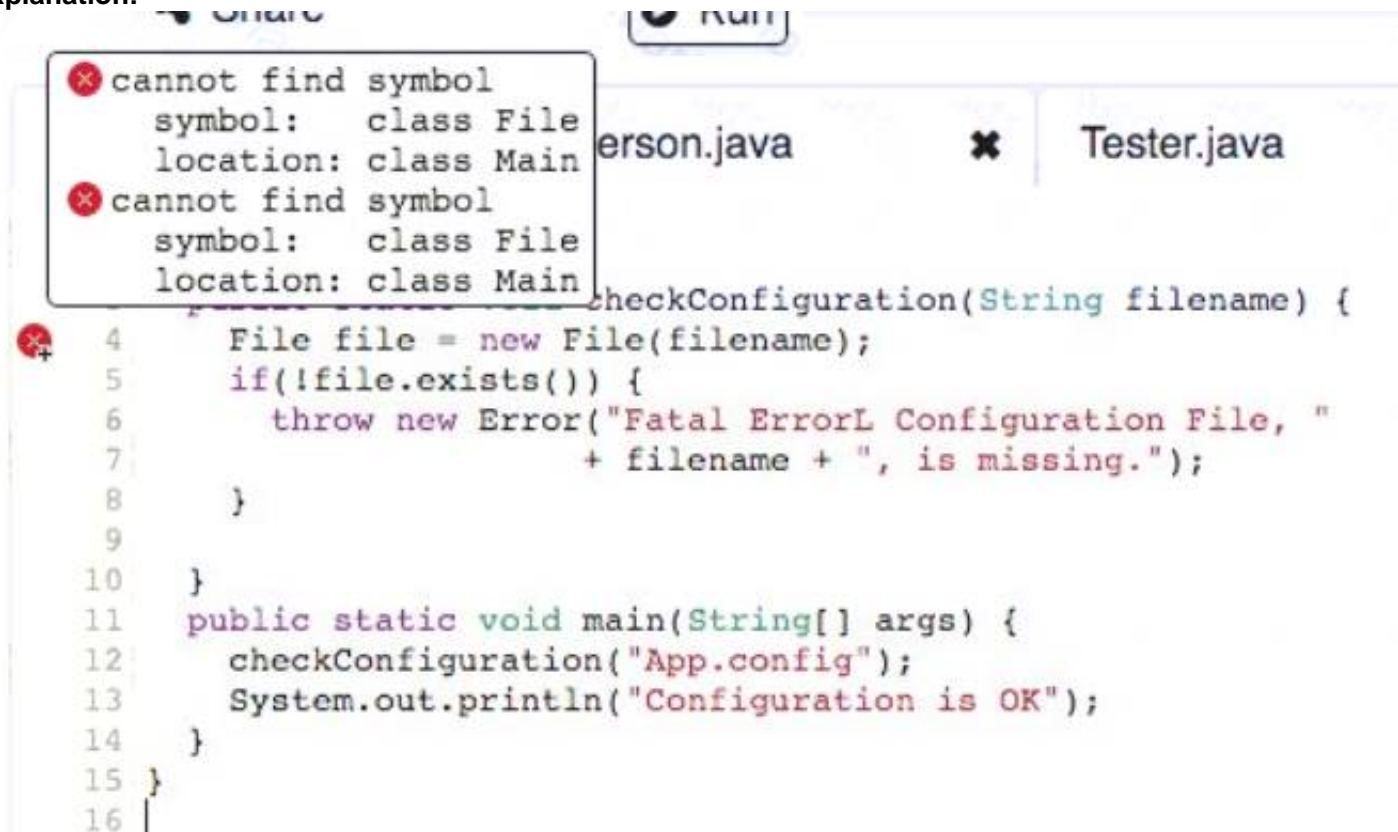
    public static void main(String[] args) {
        checkConfiguration("App.config");
        System.out.println("Configuration is OK");
    }
}
```

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 24

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 */
}
```

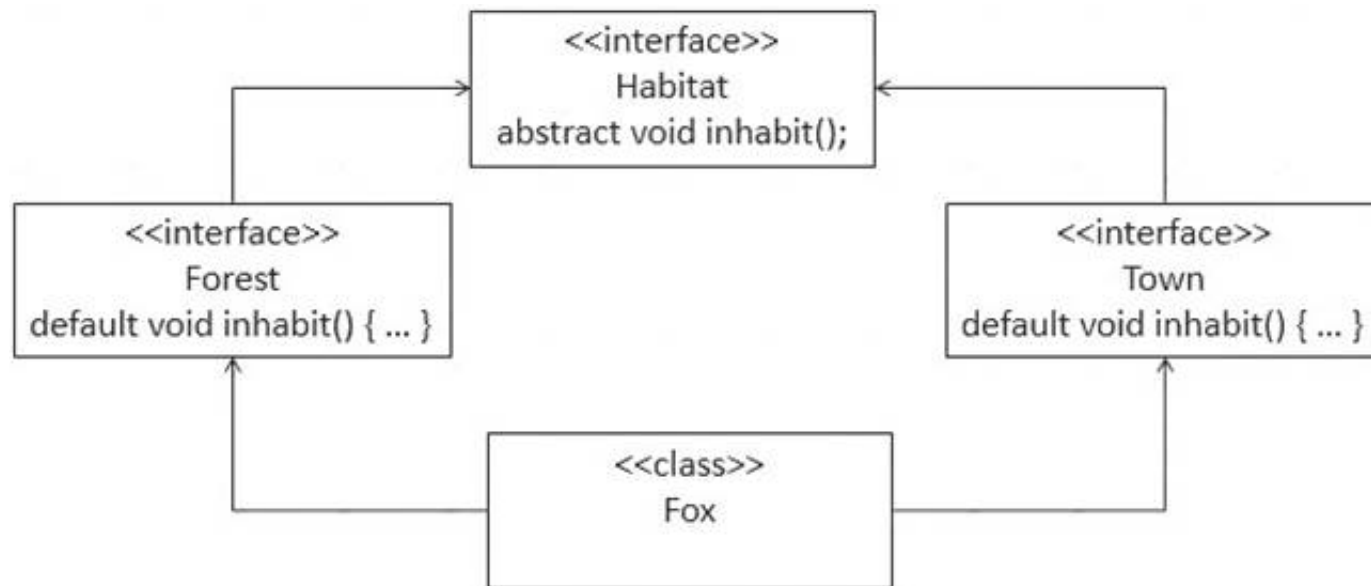
Which code, when inserted at one or more marked positions, would allow classes B and C to compile?

- 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:



Which statement is true about the Fox class?

- A. Fox class does not have to override inhabit method, so long as it does not try to call it.
- B. Fox class does not have to override the inhabit method if Forest and Town provide compatible implementations.
- C. Fox class must implement either Forest or Town interfaces, but not both.
- D. The inhabit method implementation from the first interface that Fox implements will take precedence.
- E. Fox class must provide implementation for the inhabit method.

Answer: B

NEW QUESTION 30

Given the declaration:

```

@interface Resource {
    String name();
    int priority() default 0;
}
    
```

Examine this code fragment:

```

/* Loc1 */ class ProcessOrders { ... }
    
```

Which two annotations may be applied at Loc1 in the code fragment? (Choose two.)

- A. @Resource(priority=100)
- B. @Resource(priority=0)
- C. @Resource(name="Customer1", priority=100)
- D. @Resource(name="Customer1")
- E. @Resource

Answer: AB

NEW QUESTION 31

Given:

```

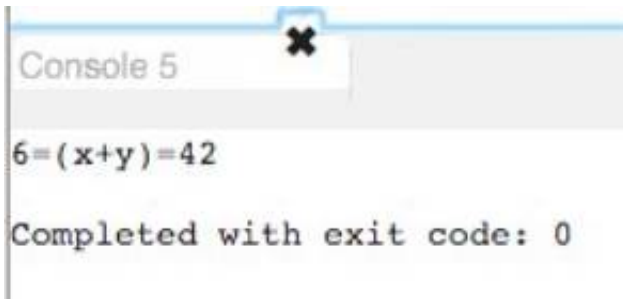
public class Tester {
    public static void main(String[] args) {
        int x = 4;
        int y = 2;
        System.out.println(x+y+"=(x+y)="+x+y);
    }
}
    
```

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 35

Which two statements are correct about try blocks? (Choose two.)

- A. A try block can have more than one catch block.
- B. A finally block in a try-with-resources statement executes before the resources declared are closed.
- C. A finally block must be immediately placed after the try or catch blocks.
- D. A try block must have a catch block and a finally block.
- E. catch blocks must be ordered from generic to specific exception types.

Answer: AC

NEW QUESTION 38

Given:

```
public static void main(String[] args) {
    try (Reader reader1 = new FileReader("File1.txt");
        Reader reader2 = new FileReader("File2.txt");
        Reader reader3 = new FileReader("File3_txt")) {

    } catch (IOException ex) {
        Logger.getLogger(Main.class.getName()).log(Level.SEVERE, null, ex);
    }
    // Line 1
    System.out.println("Done");
}
```

When run and all three files exist, what is the state of each reader on Line 1?

- A. All three readers are still open.
- B. All three readers have been closed.
- C. The compilation fails.
- D. Only reader1 has been closed.

Answer: C

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 44

Given:


```
import java.io.FileNotFoundException;
import java.io.IOException;

public class Tester {
    public static void main(String[] args) {
        try {
            doA();
        } //line 1
    }
    private static void doA() throws IOException, IndexOutOfBoundsException {
        if (false) {
            throw new FileNotFoundException();
        } else {
            throw new IndexOutOfBoundsException();
        }
    }
}
```

What must be added in line 1 to compile this class?

- A. catch(IOException e) {}
- B. catch(FileNotFoundException | IndexOutOfBoundsException e) {}
- C. catch(FileNotFoundException | IOException e) {}
- D. catch(IndexOutOfBoundsException e) {} catch(FileNotFoundException e) {}
- E. catch(FileNotFoundException e) {} catch(IndexOutOfBoundsException e) {}

Answer: A

NEW QUESTION 45

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 46

Given:

```
@Target(ElementType.METHOD)
@Retention(RetentionPolicy.RUNTIME)
public @interface AuthorInfo {
    String author() default "";
    String date();
    String[] comments() default {};
}
```

Which two are correct? (Choose two.)

- A. `@AuthorInfo(date="1-1-2020", comments={ null })`
`public class Hello {`
 `public void func() {}`
`}`
- B. `public class Hello {`
`@AuthorInfo (date="1-1-2020. comments="Hello")`
 `public void func() {}`
`}`
- C. `public class Hello {`
 `@AuthorInfo`
 `public void func() {}`
`}`
- D. `@AuthorInfo(date="1-1-2020")`
`public class Hello {`
 `public void func() {}`
`}`
- E. `public class Hello {`
 `@AuthorInfo(date="1-1-2020", author="Gandhi", comments={ "world" })`
 `public void func () {}`
`}`

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

Answer: CD

NEW QUESTION 49

Given the code fragment:

```
char[][] arrays = {{'a', 'd'}, {'b', 'e'}, {'c', 'f'}};
for (char[] xx : arrays) {
    for (char yy : xx) {
        System.out.print(yy);
    }
    System.out.print(" ");
}
```

What is the result?

- A. ab cd ef
B. An `ArrayIndexOutOfBoundsException` is thrown at runtime.
C. The compilation fails.
D. abc def
E. ad be cf

Answer: E

NEW QUESTION 54

Which two safely validate inputs? (Choose two.)

- A. Delegate numeric range checking of values to the database.
B. Accept only valid characters and input values.
C. Use trusted domain-specific libraries to validate inputs.
D. Assume inputs have already been validated.
E. Modify the input values, as needed, to pass validation.

Answer: AB

NEW QUESTION 58

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:

```


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 }

```

Execute Mode, Version, Inputs & Arguments

JDK 11.0.4 ☐ Interactive Stdin Inputs

CommandLine Arguments

Execute ... 

Result
CPU Time: 0.26 sec(s), Memory: 32984 kilobyte(s)

```

apple:APPLE
orange:ORANGE
banana:BANANA

```

NEW QUESTION 62

Given:

```

class Employee {
    String office;
}

```

and the code fragment:

```

5. public class HRApp {
6.     var employee = new ArrayList<Employee>();
7.     public var display() {
8.         var employee = new Employee();
9.         var offices = new ArrayList<>();
10.        offices.add("Chicago");
11.        offices.add("Bangalore");
12.        for (var office : offices) {
13.            System.out.print("Employee Location"+ office);
14.        }
15.    }
16. }

```

Which two lines cause compilation errors? (Choose two.)

- A. line 12
- B. line 6
- C. line 9
- D. line 8

E. line 7

Answer: BE

NEW QUESTION 66

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 69

Given:

```
String[][] arr = {  
    {"Red", "White"},  
    {"Black"},  
    {"Blue", "Yellow", "Green", "Violet"}  
};  
for(int row = 0; row < arr.length; row++) {  
    int column = 0;  
    for(; column < arr[row].length; column++) {  
        System.out.println "[" + row + ", " + column + "] = " + arr[row][column];  
    }  
}
```

What is the result?

- A. [0,0] = Red[0,1] = White[1,0] = Black[1,1] = Blue[2,0] = Yellow[2,1] = Green[3,0] = Violet
- B. [0,0] = Red[1,0] = Black[2,0] = Blue
- C. java.lang.ArrayIndexOutOfBoundsException thrown
- D. [0,0] = Red[0,1] = White[1,0] = Black[2,0] = Blue[2,1] = Yellow[2,2] = Green[2,3] = Violet

Answer: D

Explanation:



```
Console 1 Console 2 Console 3  
[0,0] =Red  
[0,1] =White  
[1,0] =Black  
[2,0] =Blue  
[2,1] =Yellow  
[2,2] =Green  
[2,3] =Violet  
Completed with exit code: 0
```

NEW QUESTION 73

Examine these module declarations:


```
module ServiceAPI {
    exports com.example.api;
}

module ServiceProvider {
    requires ServiceAPI;
    provides com.example.api with com.myimpl.Impl;
}

module Consumer {
    requires ServiceAPI;
    uses com.example.api;
}
```

Which two statements are correct? (Choose two.)

- A. The ServiceProvider module is the only module that, at run time, can provide the com.example.api API.
- B. The placement of the com.example.api API in a separate module, ServiceAPI, makes it easy to install multiple provider modules.
- C. The Consumer module should require the ServiceProvider module.
- D. The ServiceProvider module should export the com.myimpl package.
- E. The ServiceProvider module does not know the identity of a module (such as Consumer) that uses the com.example.api API.

Answer: AC

NEW QUESTION 78

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 81

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 84

Which two statements correctly describe capabilities of interfaces and abstract classes? (Choose two.)

- A. Interfaces cannot have protected methods but abstract classes can.
- B. Both interfaces and abstract classes can have final methods.
- C. Interfaces cannot have instance fields but abstract classes can.
- D. Interfaces cannot have static methods but abstract classes can.
- E. Interfaces cannot have methods with bodies but abstract classes can.

Answer: AC

NEW QUESTION 89

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 91

Given:

```
public class Hello {  
    class Greeting {  
        void sayHi() {  
            System.out.println("Hello world");  
        }  
    }  
    public static void main(String... args) {  
        // Line 1  
    }  
}
```

What code must you insert on Line 1 to enable the code to print Hello world?

- A. Hello.Greeting myG = new Hello.Greeting() myG.sayHi();
- B. Hello myH = new Hello();Hello.Greeting myG = myH.new Greeting(); myG.sayHi();
- C. Hello myH = new Hello();Hello.Greeting myG = myH.new Hello.Greeting(); myG.sayHi();
- D. Hello myH = new Hello(); Greeting myG = new Greeting(); myG.sayHi ();

Answer: B

NEW QUESTION 92

Which code fragment prints 100 random numbers?

- A.

```
var r= new Random();  
new DoubleStream(r::nextDouble).limit(100).forEach(System.out::print);
```
- B.

```
DoubleStream.generate(Random::nextDouble)  
    .limit (100).forFach(System.out::print);
```
- C.

```
Doublestream.generate(Random.nextDouble).limit(100).forEach(System.out.print);
```
- D.

```
var r = new Random(); DoubleStream.generate(r::nextDouble).limit(100).forEach(System.out::print);
```

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

Answer: D

NEW QUESTION 96

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 100

Given:

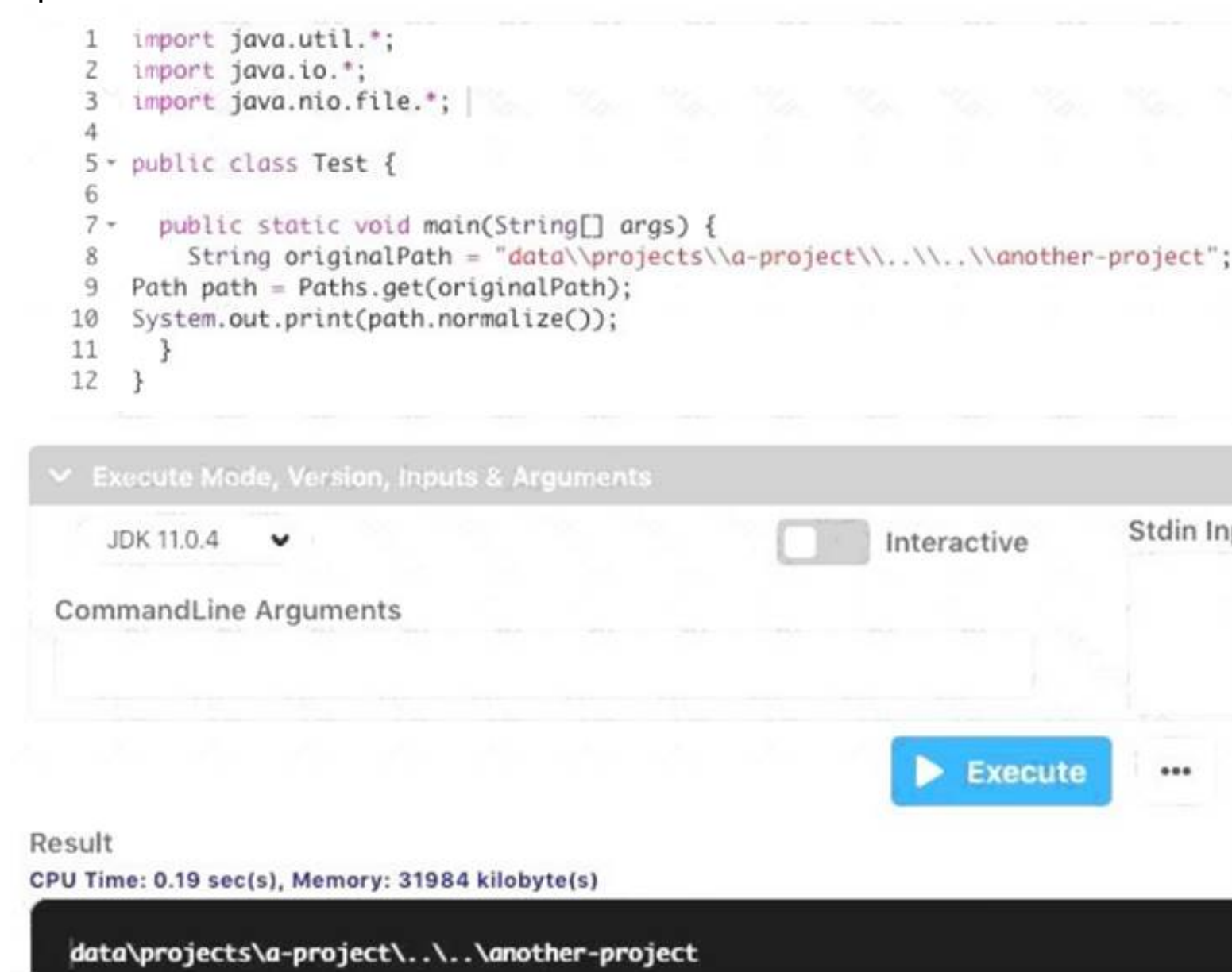
String originalPath = "data\\projects\\a-project\\..\\..\\another-project"; Path path = Paths.get(originalPath); System.out.print(path.normalize());

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 Input

CommandLine Arguments

Execute

Result

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

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

NEW QUESTION 102

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 105

Given:

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

and

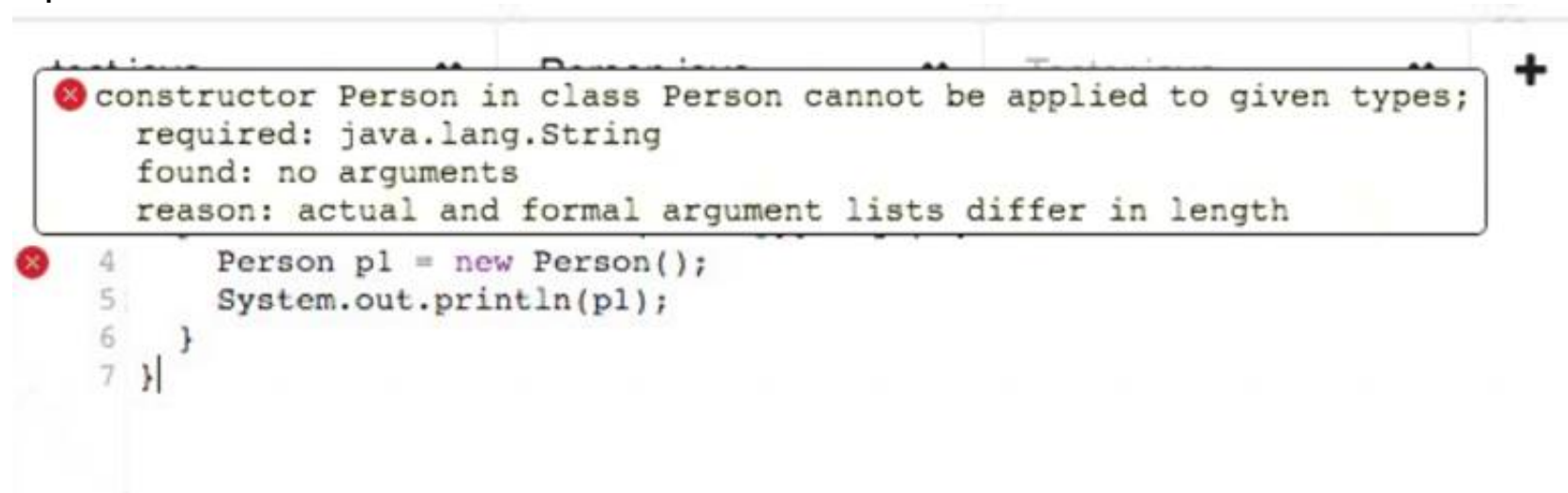
```
public class Tester {
    public static void main(String[] args) {
        Person p1 = new Person(); // line 1
        System.out.println(p1);
    }
}
```

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 107

Given these two classes:

```
public class Resource {
    public Worker owner;
    public synchronized boolean claim(Worker worker) {
        if (owner == null) {
            owner = worker;
            return true;
        }
        else return false;
    }
    public synchronized void release() {
        owner = null;
    }
}

public class Worker {
    public synchronized void work(Resource... resources) {
        for (int i = 0; i < 10; i++) {
            while (!resources[0].claim(this)) { }
            while (!resources[1].claim(this)) { }
            // do work with resource
            resources[1].release();
            resources[0].release();
        }
    }
}
```

And given this fragment:

```
Worker w1 = new Worker();
Worker w2 = new Worker();
Resource r1 = new Resource();
Resource r2 = new Resource();
new Thread( () -> {
    w1.work(r1, r2);
} ).start();
new Thread( () -> {
    w2.work(r2, r1);
} ).start();
```

Which describes the fragment?

- A. It throws IllegalMonitorStateException.
- B. It is subject to deadlock.
- C. It is subject to livelock.
- D. The code does not compile.

Answer: D

NEW QUESTION 108

Given:

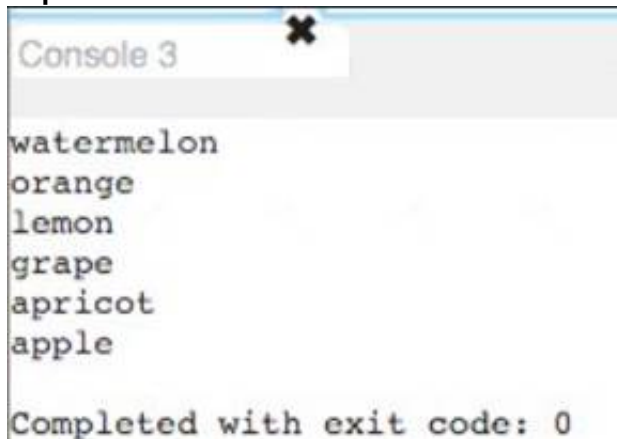
```
import java.util.ArrayList;
import java.util.Arrays;
public class NewMain {
    public static void main(String[] args) {
        String[] fruitNames = { "apple", "orange",
                                "grape", "lemon", "apricot", "watermelon" };
        var fruits = new ArrayList<>(Arrays.asList(fruitNames));
        fruits.sort((var a, var b) -> -a.compareTo(b));
        fruits.forEach(System.out::println);
    }
}
```

What is the result?

- A. watermelonorangelemongrapeapricotapple
- B. nothing
- C. appleapricotgrapelemonorangewatermelon
- D. appleorangegrapelemonapricotwatermelon

Answer: A

Explanation:



```
Console 3
watermelon
orange
lemon
grape
apricot
apple
Completed with exit code: 0
```

NEW QUESTION 112

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 114

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 119

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 120

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. getCCCount
- E. getGCount

Answer: CD

NEW QUESTION 121

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 126

Given:

```
public interface A {
    public Iterable a();
}
public interface B extends A {
    public Collection a();
}
public interface C extends A {
    public Path a();
}
public interface D extends B, C {
}
```

Why does D cause a compilation error?

- A. D inherits a() only from C.
- B. D inherits a() from B and C but the return types are incompatible.
- C. D extends more than one interface.
- D. D does not define any method.

Answer: B

NEW QUESTION 129

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 130

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 131

Given this requirement:

Module vehicle depends on module part and makes its com.vehicle package available for all other modules. Which module-info.java declaration meets the requirement?

A

```
module vehicle{
    requires part;
    exports com.vehicle;
}
```

B

```
module vehicle {
    requires part;
    uses com.vehicle;
}
```

C

```
module vehicle{
    requires part;
    exports com.vehicle to part;
}
```

D

```
module vehicle {
    requires com.vehicle;
    exports part;
}
```

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

Answer: A**NEW QUESTION 132**

Given:

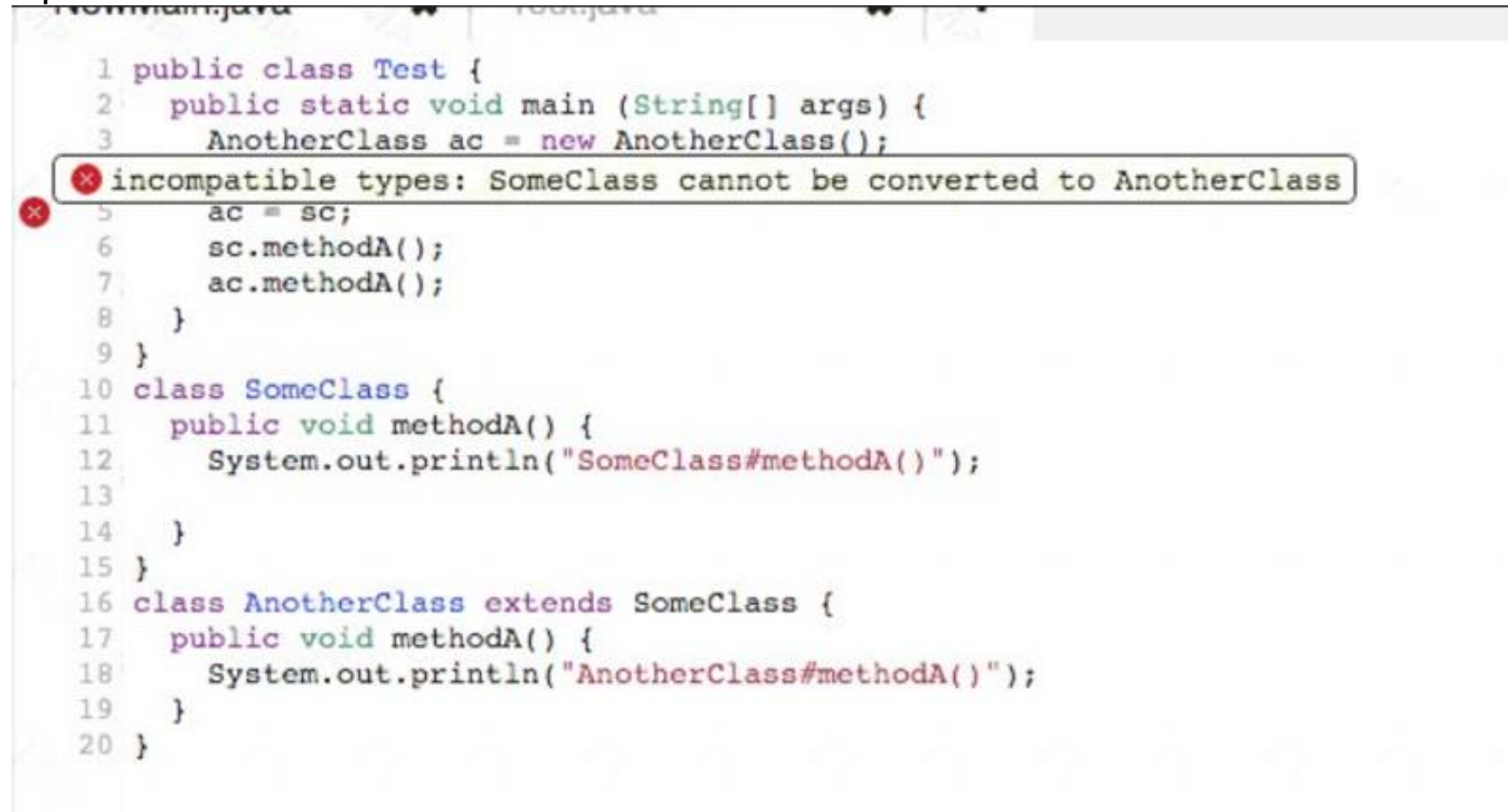
```
public class Test {
    public static void main(String[] args) {
        AnotherClass ac = new AnotherClass();
        SomeClass sc = new AnotherClass();
        ac = sc;
        sc.methodA();
        ac.methodA();
    }
}
class SomeClass {
    public void methodA() {
        System.out.println("SomeClass#methodA()");
    }
}
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         // incompatible types: SomeClass cannot be converted to AnotherClass
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 }

```

NEW QUESTION 135

Which set of commands is necessary to create and run a custom runtime image from Java source files?

- A. java, jdeps
- B. javac, jlink
- C. jar, jlink
- D. javac, jar

Answer: B

NEW QUESTION 139

Given:

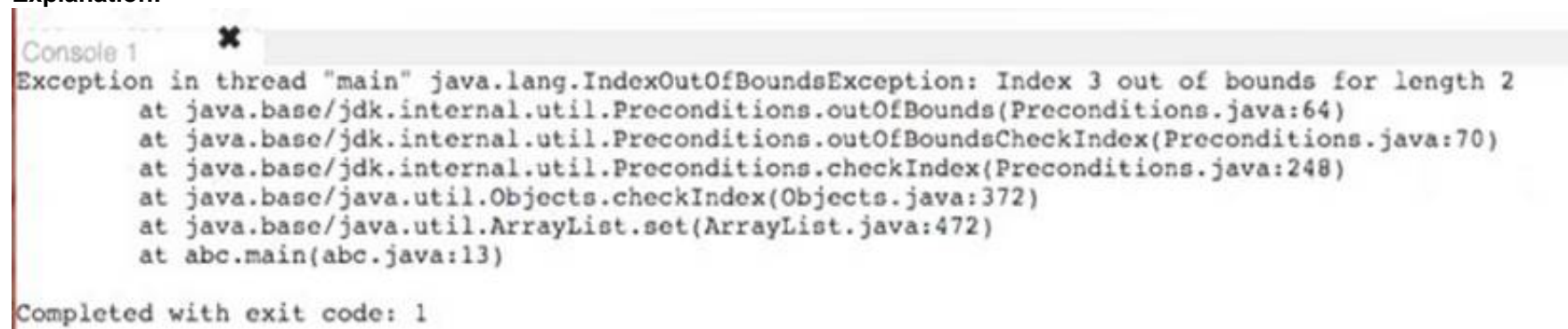
```
var data = new ArrayList<>(); data.add("Peter");
data.add(30); data.add("Market Road"); data.set(1, 25); data.remove(2); data.set(3, 1000L); System.out.print(data);
```

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 144

Given the code fragment:

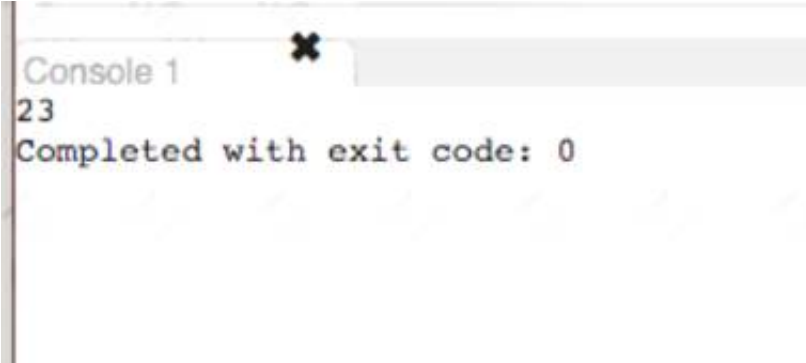

```
String s = "";
if (Double.parseDouble("11.00f") > 11) {
    s += 1;
}
if (1_7 == Integer.valueOf("17")) {
    s += 2;
}
if (1024 > 1023L) {
    s += 3;
}
System.out.print(s);
```

What is the result?

- A. 23
- B. 12
- C. 123
- D. 13

Answer: A

Explanation:



NEW QUESTION 145

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);
    }
}

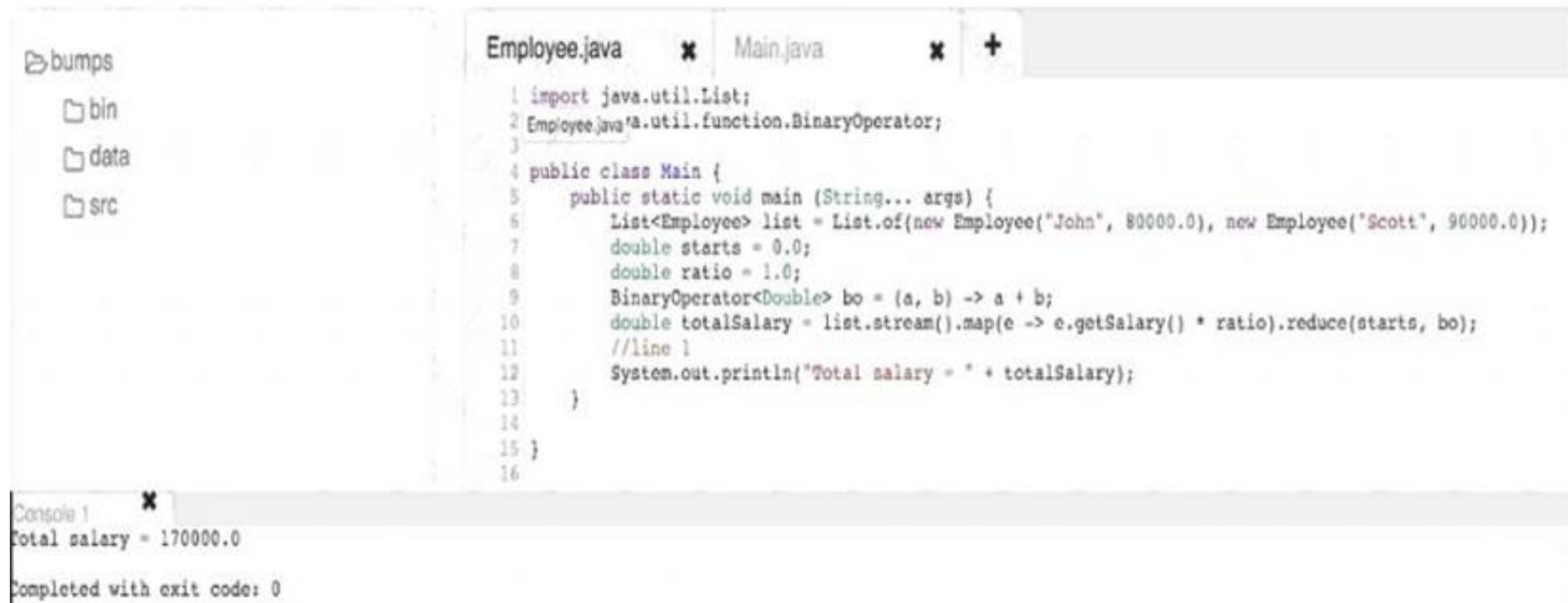
class Employee {
    String name;
    double salary;
    public Employee(String name, double salary) {
        this.name = name;
        this.salary = salary;
    }
    public String getName() { return name; }
    public double getSalary() { return salary; }
}
```

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 149

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 150

Given:

```
interface MyInterface1 {
    public int method() throws Exception;
    private void pMethod() { /* an implementation of pMethod */ }
}
interface MyInterface2 {
    public static void sMethod() { /* an implementation of sMethod */ }
    public boolean equals();
}
interface MyInterface3 {
    public void method();
    public void method(String str);
}
interface MyInterface4 {
    public void dMethod() { /* an implementation of dMethod */ }
    public void method();
}
interface MyInterface5 {
    public static void sMethod();
    public void method(String str);
}
```

Which two interfaces can be used in lambda expressions? (Choose two.)

- A. MyInterface1
- B. MyInterface3
- C. MyInterface5
- D. MyInterface2
- E. MyInterface4

Answer: CD

NEW QUESTION 155

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 157

Given:

```
public class Test {
    private String[] strings;
}
```

Which two constructors will compile and set the class field strings? (Choose two.)

A.

```
public Test(List<String> strings) {  
    this.strings = strings;  
}
```

B.

```
public Test(String... strings) {  
    strings = strings;  
}
```

C.

```
public Test(String... strings) {  
    this.strings = strings;  
}
```

D.

```
public Test(String strings) {  
    strings = strings;  
}
```

E.

```
public Test(String[] strings) {  
    this.strings = strings;  
}
```

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

Answer: CE

NEW QUESTION 160

Given: Automobile.java

```
public abstract class Automobile { //line 1  
    abstract void wheels();  
}
```

Car.java

```
public class Car extends Automobile {  
    // line 2  
    void wheels(int i) { // line 3  
        System.out.print(4);  
    }  
    public static void main(String[] args) {  
        Automobile ob = new Car(); // line 4  
        ob.wheels();  
    }  
}
```

What must you do so that the code prints 4?

- A. Remove the parameter from wheels method in line 3.
- B. Add @Override annotation in line 2.
- C. Replace the code in line 2 with Car ob = new Car();
- D. Remove abstract keyword in line 1.

Answer: B

Explanation:

```

1  Car is not abstract and does not override abstract method wheels() in
2  Automobile
3  public class Car extends Automobile {
4      void wheels(int i) {
5          System.out.print(4);
6      }
7      public static void main(String[] args) {
8          Automobile ob = new Car();
9          ob.wheels();
10     }
11 }

```

NEW QUESTION 163

Given:

/code/a/Test.java containing:

```

package a;
import b.Best;
public class Test {
    public static void main(String[] args) {
        Best b = new Best();
    }
}

```

and

/code/b/Best.java containing: package b;

```
public class Best { }
```

Which is the valid way to generate bytecode for all classes?

- A. java /code/a/Test.java
- B. javac -d /code /code/a/Test
- C. java /code/a/Test.java /code/b/Best.java
- D. java -cp /code a.Test
- E. javac -d /code /code/a/Test.java /code/b/Best.java
- F. javac -d /code /code/a/Test.java

Answer: E

NEW QUESTION 165

Given:

```

import java.util.*;
public class Foo {
    public List<Number> foo(Set<CharSequence> m) { ... }
}

```

and

```

import java.util.*;
public class Bar extends Foo {
    //line 1
}

```

Which two statements can be added at line 1 in Bar to successfully compile it? (Choose two.)

- A. public List<Integer> foo(Set<CharSequence> m) { ... }
- B. public ArrayList<Number> foo(Set<CharSequence> m) { ... }
- C. public List<Integer> foo(TreeSet<String> m) { ... }
- D. public List<Integer> foo(Set<String> m) { ... }
- E. public List<Object> foo(Set<CharSequence> m) { ... }
- F. public ArrayList<Integer> foo(Set<String> m) { ... }

Answer: BC

NEW QUESTION 167

Given:

```

public class Employee {
    private String name;
    private String locality;
    /* the constructor, getter and setter methods code goes here */
}

```

and:


```
8. List<Employee> roster = new ArrayList<>();
9. long empCount = roster.stream()
10. /* insert code here */
11. System.out.print(empCount);
```

Which code, when inserted on line 10, prints the number of unique localities from the roster list?

- A. `.map(Employee::getLocality).distinct().count();`
- B. `map(e -> e.getLocality()).count();`
- C. `.map(e -> e.getLocality()).collect(Collectors.toSet()).count();`
- D. `.filter(Employee::getLocality).distinct().count();`

Answer: D

NEW QUESTION 168

Which interface in the `java.util.function` package can return a primitive type?

- A. `ToDoubleFunction`
- B. `Supplier`
- C. `BiFunction`
- D. `LongConsumer`

Answer: A

NEW QUESTION 170

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)

JDoodle in Action.... Running the program...

NEW QUESTION 171

Given:

`List<String> longlist = List.of("Hello", "World", "Beat");` `List<String> shortlist = new ArrayList<>();`

Which code fragment correctly forms a short list of words containing the letter "e"?

```
A. longList.stream()
    .filter(w -> w.indexOf('e') != -1)
    .parallel()
    .forEach(w -> shortList.add(w));

B. longList.parallelStream()
    .filter(w -> w.indexOf('e') != -1)
    .forEach(w -> shortList.add(w));

C. shortList = longList.stream()
    .filter(w -> w.indexOf('e') != -1)
    .parallel()
    .collect(Collectors.toList());

D. longList.stream()
    .filter(w -> w.indexOf('e') != -1)
    .parallel()
    .collect(shortlist);
```

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

Answer: C

NEW QUESTION 173

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 178

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 183

var numbers = List.of(0,1,2,3,4,5,6,7,8,9);

You want to calculate the average of numbers. Which two codes will accomplish this? (Choose two.)

- A. double avg = numbers.stream().parallel().averagingDouble(a -> a);
- B. double avg = numbers.parallelStream().mapToInt(m -> m).average().getAsDouble();
- C. double avg = numbers.stream().mapToInt(i -> i).average().parallel();
- D. double avg = numbers.stream().average().getAsDouble();
- E. double avg = numbers.stream().collect(Collectors.averagingDouble(n -> n));

Answer: BD

Explanation:

```

1
2 import java.io.*;
3 import java.util.*;
4 class Hello {
5     public static void main(String[] args) {
6
7         var numbers = List.of(0,1,2,3,4,5,6,7,8,9);
8         double avg = numbers.parallelStream().mapToInt (m -> m).average().getAsDouble();
9
10    }
11 }

```

NEW QUESTION 187

Given:

```

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

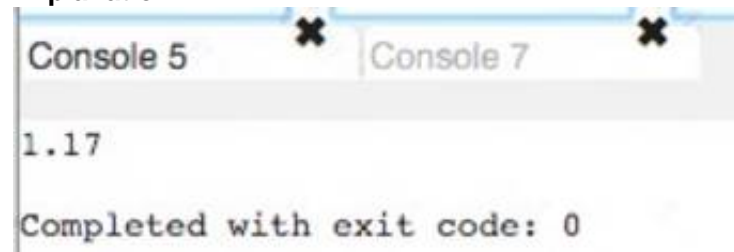
```

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 188

Given:

```

public class Hello {
    public static void main(String[] args) {
        System.out.println(args[0]+args[1]+args[2]);
    }
}

```

executed using command:

java Hello "Hello World" Hello World What is the output?

- A. An exception is thrown at runtime.
- B. Hello WorldHello World
- C. Hello World Hello World
- D. Hello WorldHelloWorld
- E. HelloHello WorldHelloWorld

Answer: C

NEW QUESTION 191

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 192

Given the contents:

MessageBundle.properties file: message=Hello MessageBundle_en.properties file: message=Hello (en) MessageBundle_US.properties file: message=Hello (US)

MessageBundle_en_US.properties file: message=Hello (en_US) MessageBundle_fr_FR.properties file: message=Bonjour

and the code fragment: Locale.setDefault(Locale.FRANCE);

Locale currentLocale = new Locale.Builder().setLanguage("en").build();

ResourceBundle messages = ResourceBundle.getBundle("MessageBundle", currentLocale); System.out. println(messages.getString("message"));

Which file will display the content on executing the code fragment?

- A. MessageBundle_en_US.properties
- B. MessageBundle_en.properties
- C. MessageBundle_fr_FR.properties
- D. MessageBundle_US.properties
- E. MessageBundle.properties

Answer: C

NEW QUESTION 193

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 198

Which code fragment compiles?

```
A. Comparator comparator = new Comparator<?>() {  
    public int compare(Integer i, Integer j) {  
        return i.compareTo(j);  
    }  
};  
  
B. var comparator = new Comparator<>() {  
    public int compare(Integer i, Integer j) {  
        return i.compareTo(j);  
    }  
};  
  
C. Comparator<> comparator = new Comparator<Integer>() {  
    public int compare(Integer i, Integer j) {  
        return i.compareTo(j);  
    }  
};  
  
D. Comparator<Integer> comparator = new Comparator<>() {  
    public int compare(Integer i, Integer j) {  
        return i.compareTo(j);  
    }  
};
```

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

Answer: D

Explanation:

```
1 import java.io.*;  
2 import java.util.*;  
3 class abc {  
4     public static void main(String[] args) {  
5  
6         Comparator<Integer> comparator = new Comparator<>() {  
7             public int compare(Integer i, Integer j) {  
8                 return i.compareTo(j);  
9             }  
10        };  
11    }  
12 }  
13 }|  
14
```

NEW QUESTION 199

Given:

```
// line 1  
List<String> fruits = new ArrayList<>(List.of("apple", "orange", "banana"));  
fruits.replaceAll(function);
```

Which statement on line 1 enables this code fragment to compile?

- A. Function function = String::toUpperCase;
- B. UnaryOperator function = s > s.toUpperCase();
- C. UnaryOperator<String> function = String::toUpperCase;
- D. Function<String> function = m > m.toUpperCase();

Answer: C

Explanation:

```

1
2 import java.io.*;
3 import java.util.*;
4 import java.util.stream.Stream;
5 import java.util.function.Function;
6 import java.util.function.UnaryOperator;
7
8 class Hello {
9     public static void main(String[] args) {
10
11         UnaryOperator<String> function = String::toUpperCase;
12         List<String> fruits = new ArrayList<>(List.of("apple", "orange", "banana"));
13         fruits.replaceAll(function);
14
15     }
16 }

```

NEW QUESTION 204

Given:

```
StringBuilder s = new StringBuilder("ABCD");
```

Which would cause s to be AQCD?

- A. s.replace(s.indexOf("A"), s.indexOf("C"), "Q");
- B. s.replace(s.indexOf("B"), s.indexOf("C"), "Q");
- C. s.replace(s.indexOf("B"), s.indexOf("B"), "Q");
- D. s.replace(s.indexOf("A"), s.indexOf("B"), "Q");

Answer: B

NEW QUESTION 209

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);
    }
}

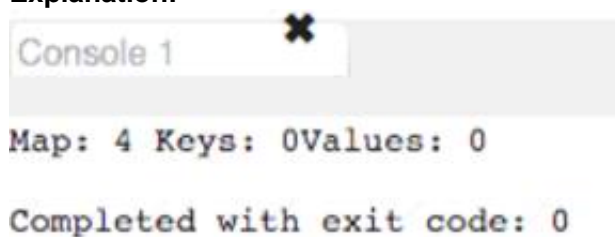
```

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:



```

Console 1
Map: 4 Keys: 0 Values: 0
Completed with exit code: 0

```

NEW QUESTION 212

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 215**

Given this enum declaration:

```
1. enum Alphabet {  
2.     A, B, C  
3.  
4. }
```

Examine this code: `System.out.println(Alphabet.getFirstLetter());`

What code should be written at line 3 to make this code print A?

- A. `final String getFirstLetter() { return A.toString(); }`
- B. `static String getFirstLetter() { return Alphabet.values()[1].toString(); }`
- C. `static String getFirstLetter() { return A.toString(); }`
- D. `String getFirstLetter() { return A.toString(); }`

Answer: C**NEW QUESTION 216**

.....

Thank You for Trying Our Product

We offer two products:

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

2nd - Questions and Answers in PDF Format

1Z0-819 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 First Try
- * 1Z0-819 Practice Test Questions in Multiple Choice Formats and Updates for 1 Year

100% Actual & Verified — Instant Download, Please Click
[Order The 1Z0-819 Practice Test Here](#)