



Oracle

Exam Questions 1Z0-819

Java SE 11 Developer

About ExamBible

Your Partner of IT Exam

Found in 1998

ExamBible is a company specialized on providing high quality IT exam practice study materials, especially Cisco CCNA, CCDA, CCNP, CCIE, Checkpoint CCSE, CompTIA A+, Network+ certification practice exams and so on. We guarantee that the candidates will not only pass any IT exam at the first attempt but also get profound understanding about the certificates they have got. There are so many alike companies in this industry, however, ExamBible has its unique advantages that other companies could not achieve.

Our Advances

* 99.9% Uptime

All examinations will be up to date.

* 24/7 Quality Support

We will provide service round the clock.

* 100% Pass Rate

Our guarantee that you will pass the exam.

* Unique Gurantee

If you do not pass the exam at the first time, we will not only arrange FULL REFUND for you, but also provide you another exam of your claim, ABSOLUTELY FREE!

NEW QUESTION 1

Given:

```

1. public class Test {
2.     private static class Greet {
3.         private void print() {
4.             System.out.println("Hello World");
5.         }
6.     }
7.     public static void main(String[] args) {
8.         Test.Greet i = new Greet();
9.         i.print();
10.    }
11. }

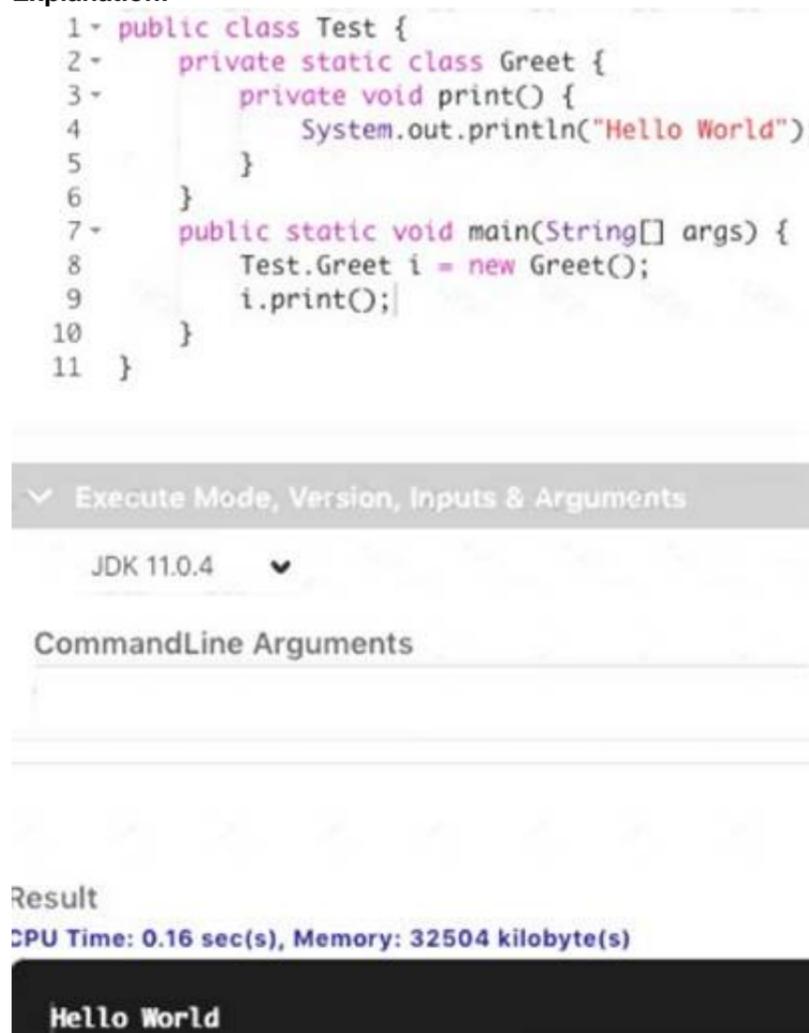
```

What is the result?

- A. The compilation fails at line 9.
- B. The compilation fails at line 2.
- C. Hello World
- D. The compilation fails at line 8.

Answer: C

Explanation:



```

1- public class Test {
2-     private static class Greet {
3-         private void print() {
4-             System.out.println("Hello World");
5-         }
6-     }
7-     public static void main(String[] args) {
8-         Test.Greet i = new Greet();
9-         i.print();
10-    }
11- }

```

Execute Mode, Version, Inputs & Arguments

JDK 11.0.4

CommandLine Arguments

Result

CPU Time: 0.16 sec(s), Memory: 32504 kilobyte(s)

Hello World

NEW QUESTION 2

Given:

```

public interface Builder {
    public A build(String str);
}

```

and

```

public class BuilderImpl implements Builder {
    @Override
    public B build(String str) {
        return new B(str);
    }
}

```

Assuming that this code compiles correctly, which three statements are true? (Choose three.)

- A. B cannot be abstract.
- B. B is a subtype of A.
- C. A cannot be abstract.
- D. A cannot be final.
- E. B cannot be final.
- F. A is a subtype of B.

Answer: ABD

NEW QUESTION 3

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 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

Given:

```
package a;
public abstract class Animal {
    protected abstract void walk();
}
package b;
public abstract class Human extends Animal {
    // line 1
}
```

Which two lines inserted in line 1 will allow this code to compile? (Choose two.)

- A. `protected void walk(){}`
 B. `void walk(){}`
 C. `abstract void walk();`
 D. `private void walk(){}`
 E. `public abstract void walk();`

Answer: AE

NEW QUESTION 6

Which code fragment does a service use to load the service provider with a Print interface?

- A. `private Print print = com.service.Provider.getInstance();`
 B. `private java.util.ServiceLoader<Print> loader = ServiceLoader.load(Print.class);`
 C. `private java.util.ServiceLoader<Print> loader = new java.util.ServiceLoader<>();`
 D. `private Print print = new com.service.Provider.PrintImpl();`

Answer: B

NEW QUESTION 7

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 8

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 9

Given:

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

and

```
public class Bar extends Foo {  
    public void foo(Collection arg) {  
        System.out.println("Hello world!");  
    }  
    public void foo(List arg) {  
        System.out.println("Olá Mundo!");  
    }  
}
```

and

```
Foo f1 = new Foo();  
Foo f2 = new Bar();  
Bar b1 = new Bar();  
Collection<String> c = new ArrayList<>();
```

Which three are true? (Choose three.)

- A. b1.foo(c) prints Bonjour le monde!
- B. f1.foo(c) prints Hello world!
- C. f1.foo(c) prints Olá Mundo!
- D. b1.foo(c) prints Hello world!
- E. f2.foo(c) prints Olá Mundo!
- F. b1.foo(c) prints Olá Mundo!
- G. f2.foo(c) prints Bonjour le monde!
- H. f2.foo(c) prints Hello world!
- I. f1.foo(c) prints Bonjour le monde!

Answer: BFG

NEW QUESTION 10

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 10

Given:

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

What is the result?

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

Answer: D

Explanation:

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

Execute Mode, Version, Inputs & Arguments

JDK 11.0.4

CommandLine Arguments

Result

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

6104 3

NEW QUESTION 13

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 15

Given the code fragment:

```
var pool = Executors.newFixedThreadPool(5);
```

```
Future outcome = pool.submit(() > 1);
```

Which type of lambda expression is passed into submit()?

- A. `java.lang.Runnable`
- B. `java.util.function.Predicate`
- C. `java.util.function.Function`
- D. `java.util.concurrent.Callable`

Answer: D

NEW QUESTION 16

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:

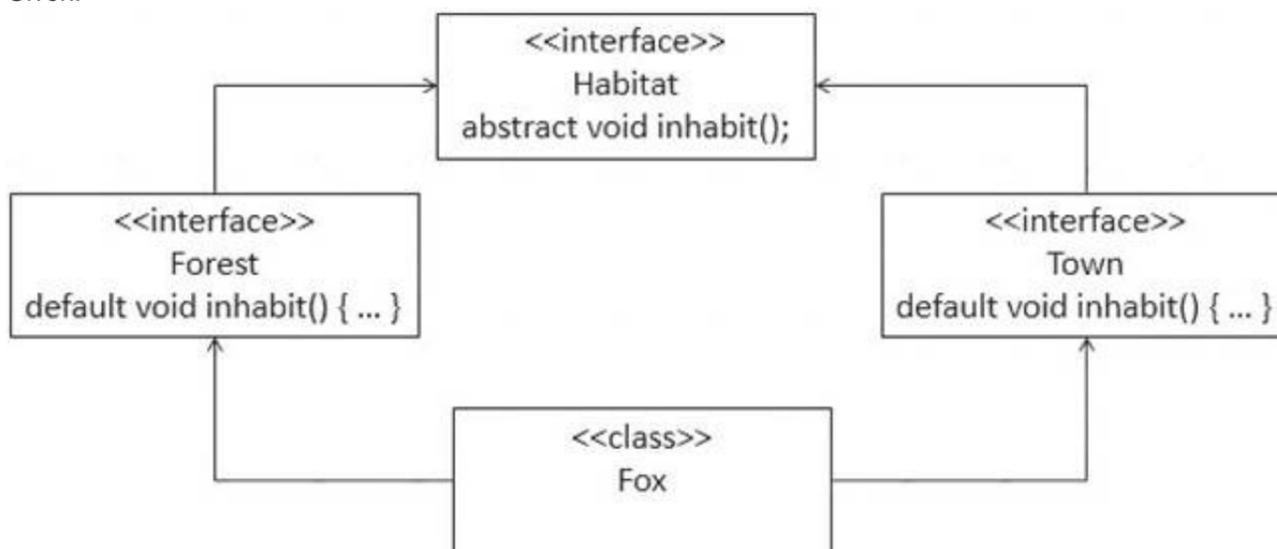
```

cannot find symbol
symbol:   class File
location: class Main
cannot find symbol
symbol:   class File
location: class Main
checkConfiguration(String filename) {
4   File file = new File(filename);
5   if(!file.exists()) {
6       throw new Error("Fatal ErrorL Configuration File, "
7           + filename + ", is missing.");
8   }
9
10  }
11  public static void main(String[] args) {
12      checkConfiguration("App.config");
13      System.out.println("Configuration is OK");
14  }
15  }
16  |

```

NEW QUESTION 18

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 23

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 27

Given:

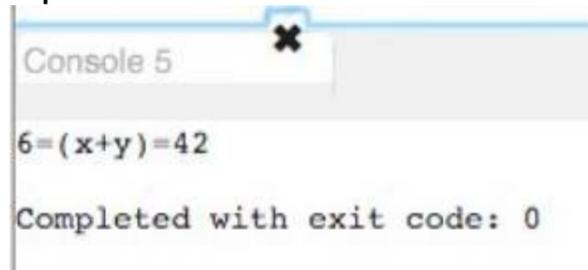
```
public class Tester {
    public static void main(String[] args) {
        int x = 4;
        int y = 2;
        System.out.println(x+y+"=(x+y)="+x+y);
    }
}
```

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 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 34

Given:

```
LocalDate d1 = LocalDate.of(1997,2,7); DateTimeFormatter dtf = DateTimeFormatter.ofPattern( /*insert code here*/ ); System.out.println(dtf.format (d1));
```

Which pattern formats the date as Friday 7th of February 1997?

- A. "eeee dd+'th of'+ MMM yyyy"
- B. "eeee dd'th of' MMM yyyy"
- C. "eeee d+'th of'+ MMMM yyyy"
- D. "eeee d'th of' MMMM yyyy"

Answer: B

NEW QUESTION 39

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 40

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:

```
@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 45

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 48

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 49

Given:

```

Integer[] intArray = {2, 1, 3, 4, 5};
List<Integer> list =
new ArrayList<>(Arrays.asList (intArray));
list.parallelStream()
    .forEach(e -> System.out.print(e + " "));

```

Which two are correct? (Choose two.)

- A. The output will be exactly 2 1 3 4 5.
- B. The program prints 1 4 2 3, but the order is unpredictable.
- C. Replacing forEach() with forEachOrdered(), the program prints 2 1 3 4 5, but the order is unpredictable.
- D. Replacing forEach() with forEachOrdered(), the program prints 1 2 3 4 5.
- E. Replacing forEach() with forEachOrdered(), the program prints 2 1 3 4 5.

Answer: BD

Explanation:

```

8 public class Secret {
9     public static void main(String[] args) {
10        Integer[] intArray = {1, 2, 3, 4, 5};
11        List<Integer> list =
12        new ArrayList<> (Arrays.asList (intArray));
13        list.parallelStream()
14        .forEachOrdered(e -> System.out.print(e + " "));
15    }
16 }

```

Execute Mode, Version, Inputs & Arguments

JDK 11.0.4

CommandLine Arguments

Result

CPU Time: 0.32 sec(s), Memory: 37040 kilobyte(s)

1 2 3 4 5

NEW QUESTION 52

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 55

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).forEach(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 59

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 63

Given:

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

and

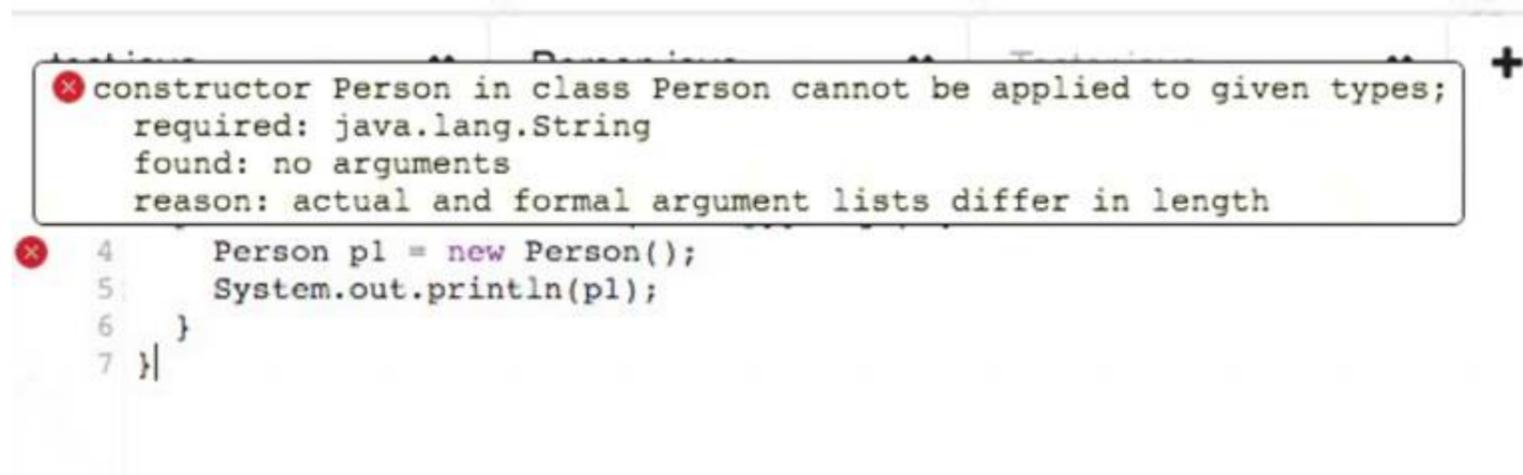
```
public class Tester {
    public static void main(String[] args) {
        Person p1 = new Person(); // line 1
        System.out.println(p1);
    }
}
```

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:



```
✖ constructor Person in class Person cannot be applied to given types;
  required: java.lang.String
  found: no arguments
  reason: actual and formal argument lists differ in length

✖ 4   Person p1 = new Person();
   5   System.out.println(p1);
   6   }
   7 }
```

NEW QUESTION 65

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 67

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 69

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 72

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 75

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 80

Given:

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

Which is true?

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

Answer: D

NEW QUESTION 85

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 import java.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 86

Given:

```

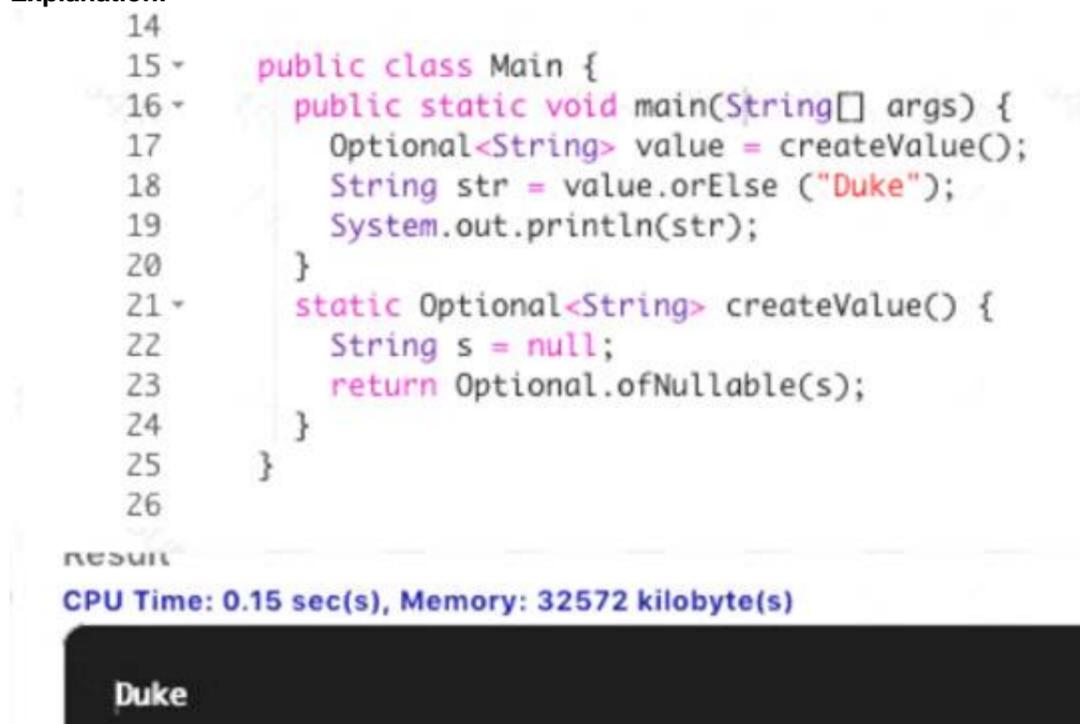
public class Main {
    public static void main(String[] args) {
        Optional<String> value = createValue();
        String str = value.orElse ("Duke");
        System.out.println(str);
    }
    static Optional<String> createValue() {
        String s = null;
        return Optional.ofNullable(s);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

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:



```

14
15 - public class Main {
16 -     public static void main(String[] args) {
17         Optional<String> value = createValue();
18         String str = value.orElse ("Duke");
19         System.out.println(str);
20     }
21 -     static Optional<String> createValue() {
22         String s = null;
23         return Optional.ofNullable(s);
24     }
25 }
26

result
CPU Time: 0.15 sec(s), Memory: 32572 kilobyte(s)

Duke
    
```

NEW QUESTION 91

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 93

Given:

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

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

What is the result?

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

Answer: C

NEW QUESTION 97

Given:

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

Which annotation should be used to remove warnings from compilation?

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

Answer: B

Explanation:

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

NEW QUESTION 101

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 102

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 104

Given:

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

    void setACount(int cCount){
        cCount = cCount;
    }
    void setTCount(){
        this.tCount = tCount;
    }
    int setCCount(){
        return cCount;
    }
    int setGCount(int g){
        gCount = g;
        return gCount;
    }
    void setAllCounts(int x){
        aCount = tCount = this.cCount = setGCount(x);
    }
}
```

Which two methods modify field values? (Choose two.)

- A. setAllCounts
- B. setACount
- C. setGCount
- D. setCCount
- E. setTCount

Answer: AC

NEW QUESTION 109

Given:

```
1. void insertionSort(int values[]) {
2.     int n = values.length;
3.     for (int j = 1; j < n; j++) {
4.         int tmp = values[j];
5.         int i = j - 1;
6.         while ( (i > -1) && (values[i] > tmp) ) {
7.             values[i + 1] = values[i];
8.             i--;
9.         }
10.        values[i + 1] = tmp;
11.    }
12. }
```

After which line can we insert `assert i < 0 || values[i] <= values[i + 1];` to verify that the values array is partially sorted?

- A. after line 8
- B. after line 6
- C. after line 5
- D. after line 10

Answer: B

Explanation:

```
1 import java.util.*;
2 import java.io.*;
3 import java.lang.Thread;
4 import java.util.ArrayList;
5 import java.util.LinkedList;
6 import java.util.List;
7 import java.util.function.Consumer;
8 import java.util.stream.Stream;
9 import java.util.stream.IntStream;
10
11
12 public class Main {
13
14
15 void insertionSort (int values[]) {
16     int n = values.length;
17     for (int j = 1; j < n; j++) {
18         int tmp = values[j];
19
20         int i = j - 1;
21         assert i < 0 || values[i] <= values[i + 1];
22         while ((i > 1) && (values[i] > tmp) ) {
23             values[i + 1] = values[i];
24             i--;
25
26         }
27         values[i + 1] = tmp;
28
29     }
30 }
31 }
```

NEW QUESTION 111

Which code is correct?

- A. Runnable r = "Message" > System.out.println();
- B. Runnable r = () > System.out::print;
- C. Runnable r = () -> {System.out.println("Message");};
- D. Runnable r = > System.out.println("Message");
- E. Runnable r = {System.out.println("Message");};

Answer: C

NEW QUESTION 112

.....

Relate Links

100% Pass Your 1Z0-819 Exam with ExamBible Prep Materials

<https://www.exambible.com/1Z0-819-exam/>

Contact us

We are proud of our high-quality customer service, which serves you around the clock 24/7.

Viste - <https://www.exambible.com/>