

Exam Questions 1Z0-819

Java SE 11 Developer

<https://www.2passeasy.com/dumps/1Z0-819/>



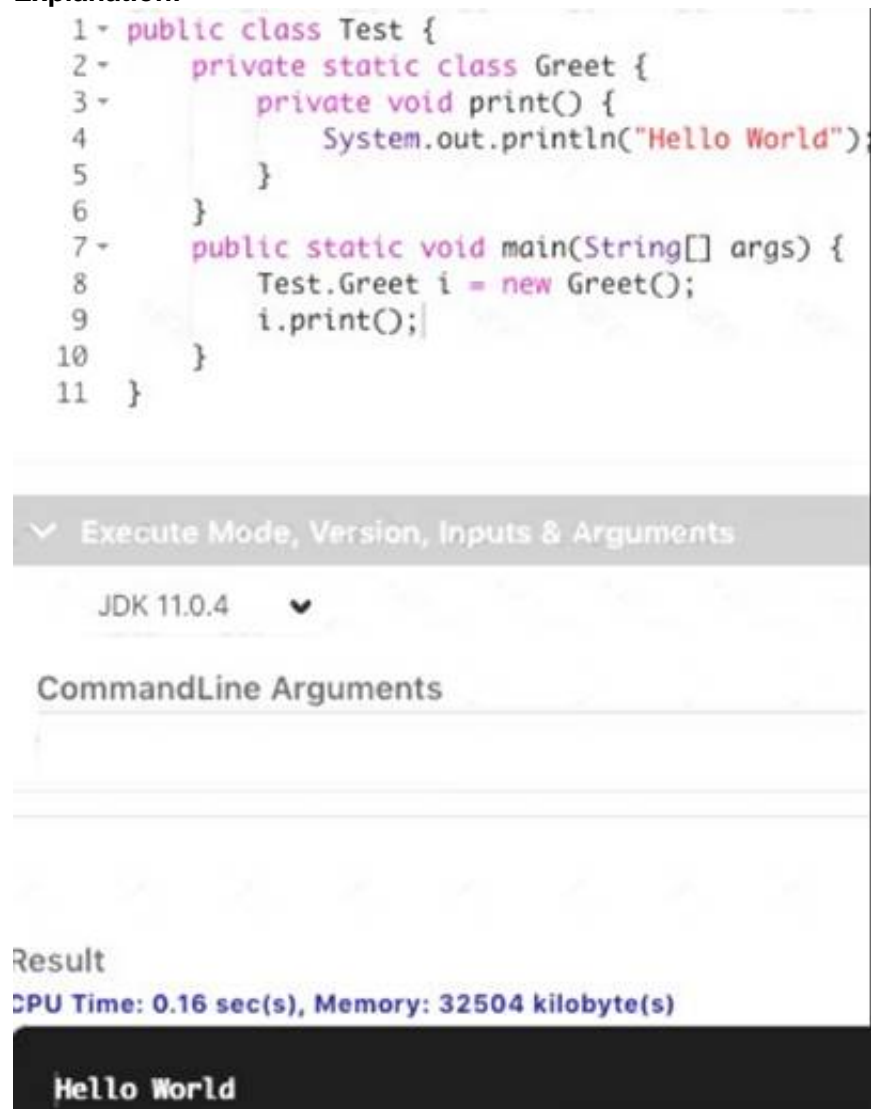
NEW QUESTION 1

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:

```
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 3

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 4

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 5

Given:

```
public class Foo {
    public <T> Collection<T> foo(Collection<T> arg) { ... }
}
```

and

```
public class Bar extends Foo { ... }
```

Which two statements are true if the method is added to Bar? (Choose two.)

- A. public Collection<String> foo(Collection<String> arg) { ... } overrides Foo.foo.
- B. public <T> Collection<T> foo(Stream<T> arg) { ... } overloads Foo.foo.
- C. public <T> List<T> foo(Collection<T> arg) { ... } overrides Foo.foo.
- D. public <T> Collection<T> foo(Collection<T> arg) { ... } overloads Foo.foo.
- E. public <T> Collection<T> bar(Collection<T> arg) { ... } overloads Foo.foo.
- F. public <T> Iterable<T> foo(Collection<T> arg) { ... } overrides Foo.foo.

Answer: CF

NEW QUESTION 6

Given:

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

Which is true?

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

Answer: A

NEW QUESTION 7

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 8

Given:

```

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

```

When is the readObject method called?

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

Answer: B

NEW QUESTION 9

Given:

```

class Mycar {
}

```

and

```

javac C:\workspace4\Mycar.java

```

What is the expected result of javac?

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

Answer: B

NEW QUESTION 10

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 10

Given:

```
var numbers = List.of(1,2,3,4,5,6,7,8,9,10);
// line 1
StringBuilder sb = new StringBuilder();
for(int a: numbers) {
    sb.append(f.apply(a));
    sb.append(" ");
}
System.out.println(sb.toString());
```

Which statement on line 1 enables this code to compile?

- A. `Function<Integer, Integer> f = n -> n * 2;`
- B. `Function<Integer> f = n -> n * 2;`
- C. `Function<int> f = n -> n * 2;`
- D. `Function<int, int> f = n -> n * 2;`
- E. `Function f = n -> n * 2;`

Answer: A

Explanation:

```
15
16 - public class Main {
17 -     public static void main(String[] args) {
18         var numbers = List.of(1,2,3,4,5,6,7,8,9,10);
19         Function<Integer, Integer> f = n -> n * 2;
20         StringBuilder sb = new StringBuilder();
21 -         for(int a: numbers) {
22             sb.append(f.apply(a));
23             sb.append(" ");
24         }
25         System.out.println(sb.toString());
26     }
27 }
28
```

Result

CPU Time: 0.22 sec(s), Memory: 33056 kilobyte(s)

2 4 6 8 10 12 14 16 18 20

NEW QUESTION 11

Given the code fragment:

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

What is the output?

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

Answer: D

NEW QUESTION 12

Given:

```
public class Test{  
    private int num = 1;  
    private int div = 0;  
  
    public void divide() {  
        try {  
            num = num / div;  
            System.out.print("Exception");  
        }  
        catch(ArithmeticException ae) { num = 100; }  
        catch(Exception e) { num = 200; }  
        finally { num = 300; }  
        System.out.print(num);  
    }  
    public static void main(String args[])  
    {  
        Test test = new Test();  
        test.divide();  
    }  
}
```

What is the output?

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

Answer: A

Explanation:


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

Execute Mode, Version, Inputs & Arguments

JDK 11.0.4

☐ In

CommandLine Arguments

Result

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

300

NEW QUESTION 14

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 16

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 17

Given:

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

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

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

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

Answer: BE

NEW QUESTION 18

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 23

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 26

Given the code fragment:

```

int[] secA = { 2, 4, 6, 8, 10 };
int[] secB = { 2, 4, 8, 6, 10 };
int res1 = Arrays.mismatch(secA, secB);
int res2 = Arrays.compare(secA, secB);
System.out.print(res1 + " : " + res2);

```

What is the result?

- A. -1 : 2
- B. 2 : -1
- C. 2 : 3
- D. 3 : 0

Answer: B

NEW QUESTION 30

Given:

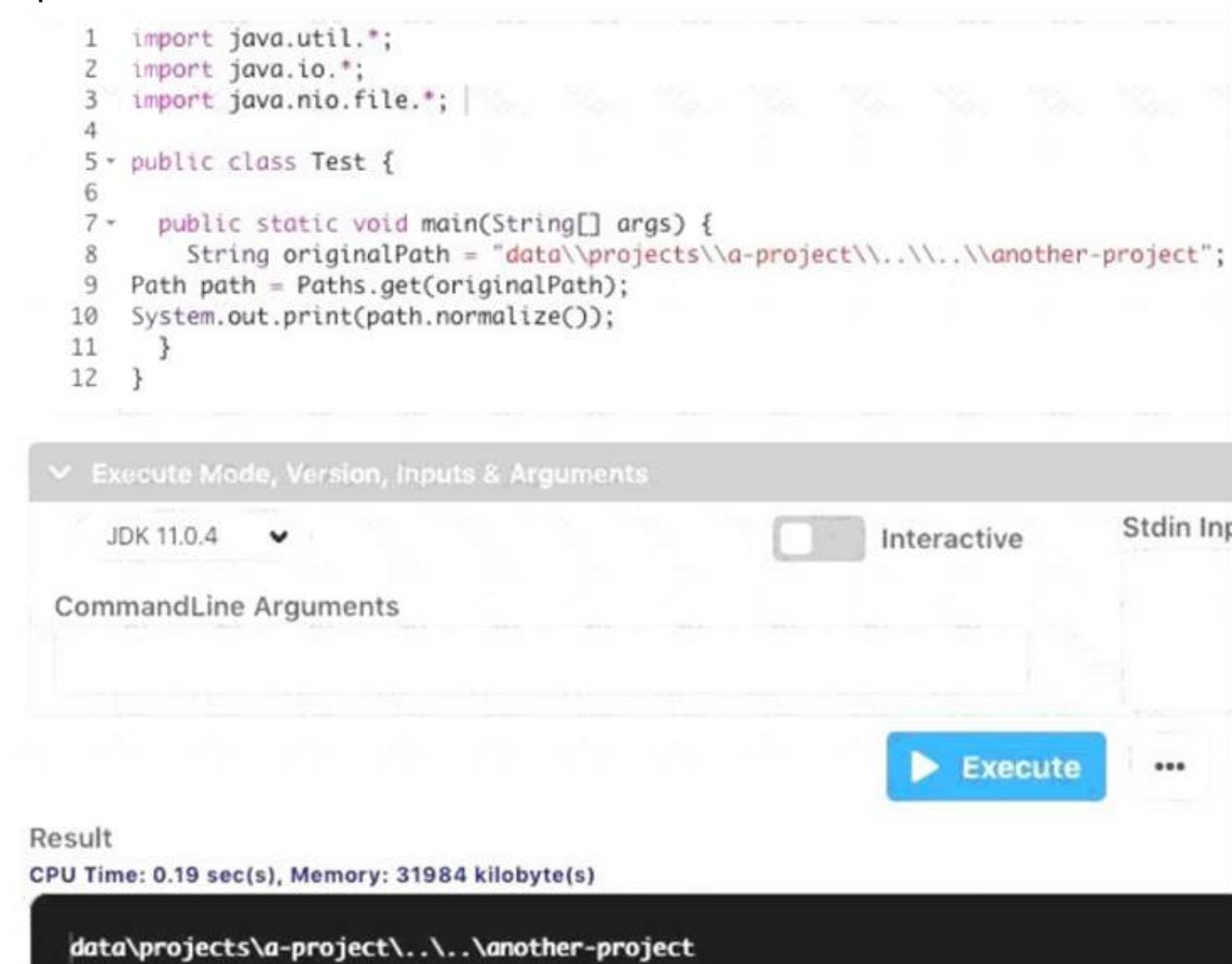
```
String originalPath = "data\\projects\\a-project\\..\\..\\another-project"; Path path = Paths.get(originalPath); System.out.print(path.normalize());
```

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 34

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 37

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 39

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 41

Given:

```

public class Confidential implements Serializable{
    private String data;

    public Confidential(String data) {
        this.data = data;
    }
}

```

Which two are secure serialization of these objects? (Choose two.)

- A. Define the serialPersistentFields array field.
- B. Declare fields transient.

- C. Implement only readResolve to replace the instance with a serial proxy and not writeReplace.
- D. Make the class abstract.
- E. Implement only writeReplace to replace the instance with a serial proxy and not readResolve.

Answer: AC

NEW QUESTION 42

Given:

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

It is required that if p instanceof Pair then p.isValid() returns true.

Which is the smallest set of visibility changes to insure this requirement is met?

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

Answer: B

NEW QUESTION 43

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 48

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 53

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 54

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 58

Given:

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

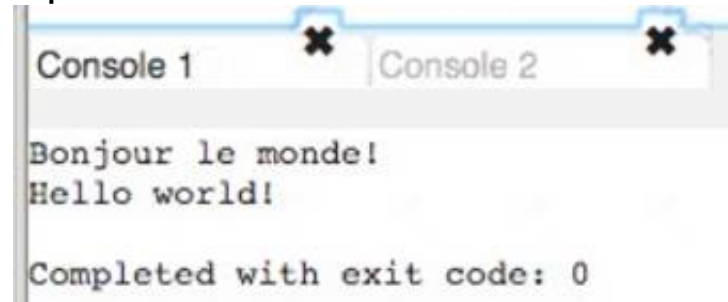
public class Bar extends Foo {
    private void print() {
        System.out.println("Hello world!");
    }
    public void bar() {
        print();
    }
    public static void main(String... args) {
        Bar b = new Bar();
        b.foo();
        b.bar();
    }
}
```

What is the output?

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

Answer: C

Explanation:



NEW QUESTION 61

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:


```
× Car is not abstract and does not override abstract method wheels() in
  Automobile
×
1 public class Car extends Automobile {
2
3
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 62

Which two statements independently compile? (Choose two.)

- A. List<? super Short> list = new ArrayList<Number>();
- B. List<? super Number> list = new ArrayList<Integer>();
- C. List<? extends Number> list = new ArrayList<Byte>();
- D. List<? extends Number> list = new ArrayList<Object>();
- E. List<? super Float> list = new ArrayList<Double>();

Answer: AC

Explanation:

```
1 import java.util.*;
2 import java.text.*;
3 import java.io.*;
4 import java.lang.Thread;
5 import java.util.ArrayList;
6 import java.util.LinkedList;
7 import java.util.List;
8 import java.util.function.Consumer;
9 import java.util.stream.Stream;
10 import java.util.stream.IntStream;
11 import java.util.Optional;
12
13 public class Intel {
14     public static void main (String[] args) {
15         List<? extends Number> list = new ArrayList<Byte>()
16     }
17 }
```

Execute Mode, Version, Inputs & Arguments

JDK 11.0.4

Result

compiled and executed in 1.173 sec(s)

**NEW QUESTION 65**

Analyze the code:


```
public class Test {
    static String prefix = "Global:";
    private String name = "namespace";
    public static String getName() {
        return new Test().name;
    }
    public static void main(String[] args) {
        Test t = new Test();
        System.out.println(/* Insert code here */);
    }
}
```

Which two options can you insert inside println method to produce Global:namespace? (Choose two.)

- A. Test.prefix+Test.name
- B. new Test().prefix+new Test().name
- C. Test.prefix+Test.getName()
- D. Test.getName+prefix
- E. prefix+Test.name
- F. prefix+name

Answer: BC

NEW QUESTION 70

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 72

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 76

Given the code fragment:

Path source = Paths.get("/repo/a/a.txt"); Path destination = Paths.get("/repo"); Files.move(source, destination); // line 1
 Files.delete (source); // line 2
 Assuming the source file and destination folder exist, what is the result?

- A. A java.nio.file.FileAlreadyExistsException is thrown on line 1.
- B. A java.nio.file.NoSuchFileException is thrown on line 2.
- C. A copy of /repo/a/a.txt is moved to the /repo directory and /repo/a/a.txt is deleted.
- D. a.txt is renamed repo.

Answer: C

NEW QUESTION 80

Given:

```

public class Test {
    private int sum;
    public int compute() {
        int x = 0;
        while(x < 3) {
            sum += x++;
        }
        return sum;
    }
    public static void main(String[] args) {
        Test t = new Test();
        int sum = t.compute();
        sum = t.compute();
        t.compute();
        System.out.println(sum);
    }
}

```

What is the result?

- A. 9
- B. An exception is thrown at runtime.
- C. 3
- D. 6

Answer: D

Explanation:

Console 1 ✕ Console 2 ✕ Console 3 ✕

6

Completed with exit code: 0

NEW QUESTION 82

Given:

```
import java.io.*;
public class Tester {
    public static void main(String[] args) {
        try {
            doA();
            doB();
        } catch (IOException e) {
            System.out.print("c");
            return;
        } finally{
            System.out.print("d");
        }
        System.out.print("f");
    }
    private static void doA() {
        System.out.print("a");
        if (false) {
            throw new IndexOutOfBoundsException();
        }
    }
    private static void doB() throws FileNotFoundException {
        System.out.print("b");
        if (true) {
            throw new FileNotFoundException();
        }
    }
}
```

What is the result?

- A. The compilation fails.
- B. abdf
- C. abd
- D. adf
- E. abcd

Answer: E

NEW QUESTION 84

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 89

Given:

```

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

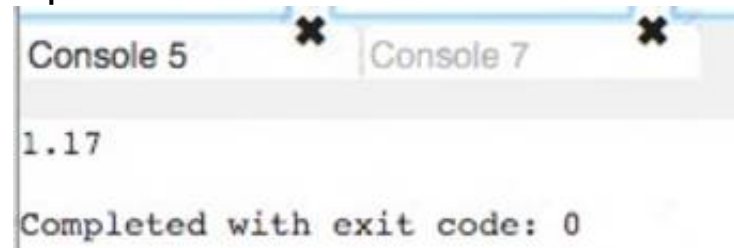
```

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 94

Given:

```

public interface InterfaceOne {
    void printOne();
}

```

Which three classes successfully override printOne()? (Choose three.)

- A.
- ```
public abstract class TestClass implements InterfaceOne {
 public abstract void printOne();
}
```
- B.
- ```
public class TestClass implements InterfaceOne {  
    private void printOne(){  
        System.out.println("one");  
    }  
}
```
- C.
- ```
public class TestClass implements InterfaceOne {
 public void printOne(){
 System.out.println("one");
 }
}
```
- D.
- ```
public abstract class TestClass implements InterfaceOne {  
    public void printOne(){  
        System.out.println("one");  
    }  
}
```
- E.
- ```
public abstract class TestClass implements InterfaceOne {
 public String printOne(){
 return "one";
 }
}
```
- F.
- ```
public class TestClass{  
    public void printOne(){  
        System.out.println("one");  
    }  
}
```

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

Answer: ACD

NEW QUESTION 95

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:



Map: 4 Keys: 0Values: 0

Completed with exit code: 0

NEW QUESTION 99

.....

THANKS FOR TRYING THE DEMO OF OUR PRODUCT

Visit Our Site to Purchase the Full Set of Actual 1Z0-819 Exam Questions With Answers.

We Also Provide Practice Exam Software That Simulates Real Exam Environment And Has Many Self-Assessment Features. Order the 1Z0-819 Product From:

<https://www.2passeasy.com/dumps/1Z0-819/>

Money Back Guarantee

1Z0-819 Practice Exam Features:

- * 1Z0-819 Questions and Answers Updated Frequently
- * 1Z0-819 Practice Questions Verified by Expert Senior Certified Staff
- * 1Z0-819 Most Realistic Questions that Guarantee you a Pass on Your FirstTry
- * 1Z0-819 Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year