

1Z0-819 Dumps

Java SE 11 Developer

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



NEW QUESTION 1

Given:

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

What is the result?

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

Answer: D

Explanation:

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

(command line arguments)

COMPILE & EXECUTE

PASTE SOURCE

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

5

NEW QUESTION 2

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

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 4

Given:

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

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

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

Answer: CE

NEW QUESTION 5

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 6

Given:

```
public static void main(String[] args) {
    final List<String> fruits =
        List.of("Orange", "Apple", "Lemmon", "Raspberry");
    final List<String> types =
        List.of("Juice", "Pie", "Ice", "Tart");
    final var stream =
        IntStream.range(0, Math.min(fruits.size(), types.size()))
            .mapToObj((i) -> fruits.get(i) + " " + types.get(i) );
    stream. forEach(System.out::println);
}
```

What is the result?

- A. Orange Juice
- B. The compilation fails.
- C. Orange Juice Apple Pie Lemmon Ice Raspberry Tart
- D. The program prints nothing.

Answer: C

Explanation:

```
12 public class Person {
13     public static void main (String[] args) {
14         final List<String> fruits =
15             List.of("Orange", "Apple", "Lemmon", "raspberry");
16         final List<String> types =
17             List.of("Juice", "Pie", "Ice", "Tart");
18         final var stream =
19             IntStream.range(0, Math.min(fruits.size(), types.size()))
20                 .mapToObj ((i) -> fruits.get(i) + " " + types.get(i) );
21         stream. forEach(System.out::println);
22     }
23 }
24 }
```

Result

compiled and executed in 1.227 sec(s)

```
Orange Juice
Apple Pie
Lemmon Ice
raspberry Tart
```

NEW QUESTION 7

Given this enum declaration:

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

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

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

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

Answer: C

Explanation:

```

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

```

Result

compiled and executed in 1.099 sec(s)

200

NEW QUESTION 8

Given:

```

public class MethodTest {
    // line 1
}

```

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

- A.


```

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

```
- B.


```

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

```
- C.


```

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

```
- D.


```

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

```
- E.


```

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

```

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

E. Option E

Answer: AE

NEW QUESTION 9

Given:

```
public class Tester {
    static class Person implements /* line 1 */ {
        private String name;
        Person(String name) { this.name = name; }
        /* line 2 */
    }
    public static void main(String[] args) {
        Person[] people = {new Person("Joe"),
                           new Person("Jane"),
                           new Person("John")};
        Arrays.sort(people);
        for(Person person: people) {
            System.out.println(person.name);
        }
    }
}
```

You want the code to produce this output:

John
Joe Jane

Which code fragment should be inserted on line 1 and line 2 to produce the output?

- A. Insert Comparator<Person> on line 1. Insert public int compare(Person p1, Person p2) { return p1.name.compare(p2.name);} on line 2.
- B. Insert Comparator<Person> on line 1. Insert public int compareTo(Person person) { return person.name.compareTo(this.name);} on line 2.
- C. Insert Comparable<Person> on line 1. Insert public int compare(Person p1, Person p2) { return p1.name.compare(p2.name);} on line 2.
- D. Insert Comparator<Person> on line 1. Insert public int compare(Person person) { return person.name.compare(this.name);} on line 2.

Answer: B

NEW QUESTION 10

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 10

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 13

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.");
        }
    }

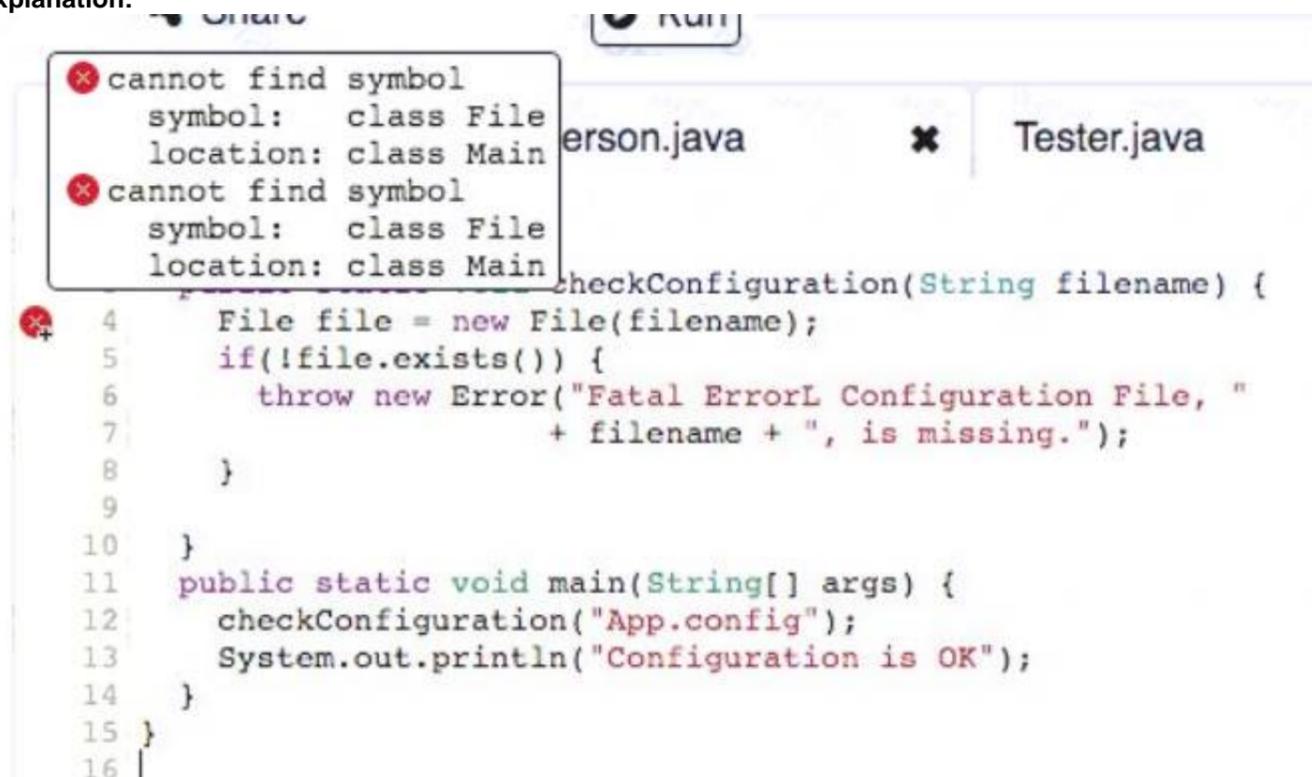
    public static void main(String[] args) {
        checkConfiguration("App.config");
        System.out.println("Configuration is OK");
    }
}
```

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 17

Given:

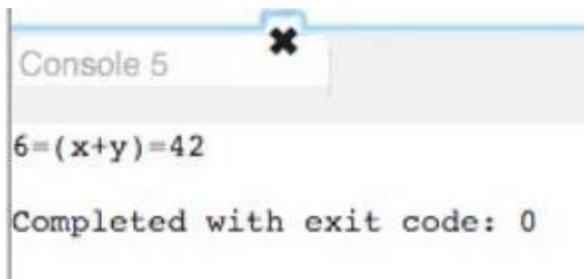
```
public class Tester {
    public static void main(String[] args) {
        int x = 4;
        int y = 2;
        System.out.println(x+y+"=(x+y)="+x+y);
    }
}
```

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:



```
Console 5
6=(x+y)=42
Completed with exit code: 0
```

NEW QUESTION 19

Given the code fragment:

```
String s1 = new String("ORACLE");
String s2 = "ORACLE";
String s3 = s1.intern();

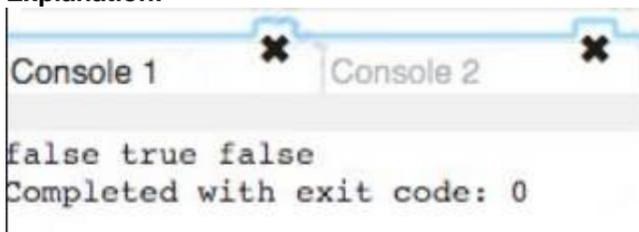
System.out.print((s1==s2) + " ");
System.out.print((s2==s3) + " ");
System.out.println(s1==s3);
```

What is the result?

- A. false true true
- B. true false false
- C. false false true
- D. false true false

Answer: D

Explanation:



```
Console 1 Console 2
false true false
Completed with exit code: 0
```

NEW QUESTION 21

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 23

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 27

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 30

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 31

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 36

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 38

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 41

Given:

```

1. public class Main {
2.     public static void greet(String... args) {
3.         System.out.print("Hello ");
4.         for (String arg : args) {
5.             System.out.println(arg);
6.         }
7.     }
8.     public static void main(String[] args) {
9.         Main c = null;
10.        c.greet();
11.    }
12. }

```

What is the result?

- A. NullPointerException is thrown at line 4.
- B. NullPointerException is thrown at line 10.
- C. A compilation error occurs.
- D. Hello

Answer: D

Explanation:



NEW QUESTION 44

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 46

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 49

Given:

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

What is the result?

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

Answer: C

Explanation:

```

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

```

✖ incompatible types: SomeClass cannot be converted to AnotherClass

NEW QUESTION 53

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 54

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 56

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:

```

2 public class Car extends Automobile {
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 57

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 }

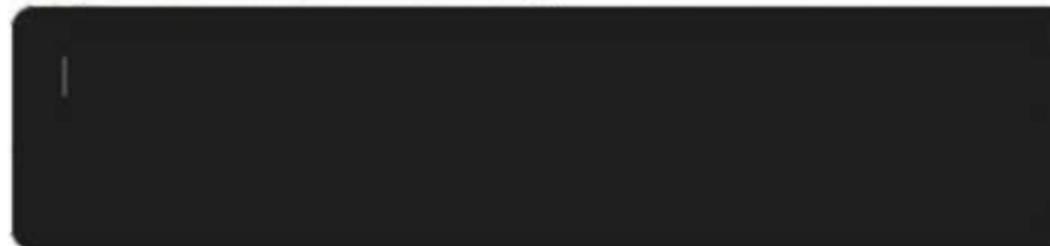
```

Execute Mode, Version, inputs & Arguments

JDK 11.0.4

Result

compiled and executed in 1.173 sec(s)



NEW QUESTION 59

Given the Person class with age and name along with getter and setter methods, and this code fragment:

```

List<Person> persons = new ArrayList(List.of(new Person(44, "Tom"),
                                             new Person(40, "Aman"),
                                             new Person(40, "Peter")));
persons.sort(Comparator.comparing((Person::getAge))
                  .thenComparing(Person::getName)
                  .reversed());
persons.forEach(p1->System.out.print(" "+p1.getName()));

```

What will be the result?

- A. Aman Tom Peter
- B. Tom Aman Peter
- C. Aman Peter Tom
- D. Tom Peter Aman

Answer: C

NEW QUESTION 64

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 69

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 74

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 75

Given:

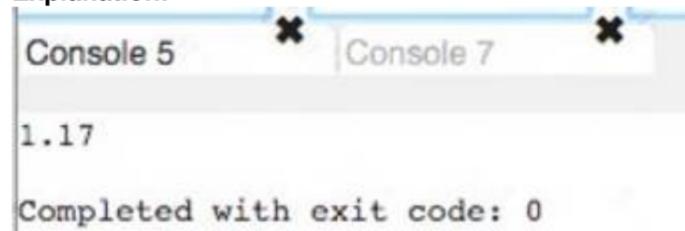
```
public class Tester {
    public static void main(String[] args) {
        byte x = 7, y = 6;
        // line 1
        System.out.println(z);
    }
}
```

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 78

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 80

Given:

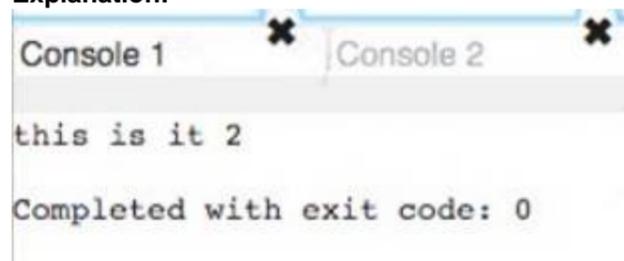
```
public class Tester {
    public static void main(String[] args) {
        String s = "this is it";
        int x = s.indexOf("is");
        s.substring(x+3);
        x = s.indexOf("is");
        System.out.println(s+" "+x);
    }
}
```

What is the result?

- A. is it 1
- B. An IndexOutOfBoundsException is thrown at runtime.
- C. is it 0
- D. this is it 2
- E. this is it 3

Answer: D

Explanation:



NEW QUESTION 81

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 83

Which three annotation uses are valid? (Choose three.)

- A. Function<String, String> func = (@NonNull x) > x.toUpperCase();
- B. var v = "Hello" + (@Interned) "World"
- C. Function<String, String> func = (var @NonNull x) > x.toUpperCase();
- D. Function<String, String> func = (@NonNull var x) > x.toUpperCase();
- E. var myString = (@NonNull String) str;
- F. var obj = new @Interned MyObject();

Answer: ACF

NEW QUESTION 87

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 88

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 }
17
```

NEW QUESTION 93

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 97

Given:

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

What is the result?

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

Answer: A

Explanation:

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

✖ variable value might not have been initialized

NEW QUESTION 99

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]);
        }
    }

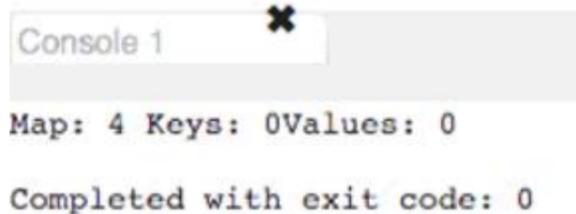
    public static void main(String[] args) {
        keys.clear();
        values = new String[0];
        System.out.println("Map: " + map.size() +
            " Keys: " + keys.size() +
            " Values: " + values.length);
    }
}
```

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 101

Given:

```
List<String> list1 = new LinkedList<String>();
Set<String> hs1 = new HashSet<String>();
String[] v = {"a", "b", "c", "b", "a"};
for (String s: v) {
    list1.add(s);
    hs1.add(s);
}
System.out.print(hs1.size() + " " + list1.size() + " ");
HashSet hs2 = new HashSet(list1);
LinkedList list2 = new LinkedList(hs1);
System.out.print(hs2.size() + " " + list2.size());
```

What is the result?

- A. 3 5 3 3
- B. 3 3 3 3
- C. 3 5 3 5
- D. 5 5 3 3

Answer: A

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         List<String> list1 = new LinkedList<String>();
16         Set<String> hs1 = new HashSet<String>();
17         String[] v = {"a", "b", "c", "b", "a"};
18         for (String s: v) {
19             list1.add(s);
20             hs1.add(s);
21         }
22         System.out.println(hs1.size() + "" + list1.size() + "");
23         HashSet hs2 = new HashSet(list1);
24         LinkedList list2 = new LinkedList(hs1);
25         System.out.print(hs2.size() + "" + list2.size());
26
27     }
28 }

```

Result

CPU Time: 0.28 sec(s). Memory: 36204 kilobyte(s)

```

35
33

```

NEW QUESTION 105

Given:

```

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

```

What is the type of the local variable x?

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

Answer: D

NEW QUESTION 107

.....

Thank You for Trying Our Product

* 100% Pass or Money Back

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

* One year free update

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

* Trusted by Millions

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

* Shop Securely

All transactions are protected by VeriSign!

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

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