1Z0-819 Dumps

Java SE 11 Developer

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

NEW QUESTION 1

Given:

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

What is the result?

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

Answer: D

Explanation:

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

```
(command line arguments)

        COMPILE & EXECUTE                    PASTE SOURCE

Successfully compiled /tmp/java_82Tlan/Tester.java <-- main method          5
```

NEW QUESTION 2

Given:

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

and

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

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

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

Answer: BC

NEW QUESTION 3

A bookstore's sales are represented by a list of Sale objects populated with the name of the customer and the books they purchased.
public class Sale { private String customer;
private List<Book> items;
// constructor, setters and getters not shown
}
public class Book { private String name; private double price;
// constructor, setters and getters not shown
}
Given a list of Sale objects, tList, which code fragment creates a list of total sales for each customer in ascending order?

```
A. List<String> totalByUser = tList.stream()
        .collect(flatMapping(t -> t.getItems().stream(),
                groupingBy(Sale::getCustomer,
                summingDouble(Book::getPrice))))
        .entrySet().stream()
        .sorted(Comparator.comparing(Entry::getValue))
        .collect(mapping(e -> e.getKey() + ":" + e.getValue(),toList()));

B. List<String> totalByUser = tList.stream()
        .collect(groupingBy(Sale::getCustomer,
                flatMapping(t -> t.getItems().stream(),
                summingDouble(Book::getPrice))))
        .sorted(Comparator. comparing (Entry::getValue))
        .collect(mapping(e -> e.getKey() + ":" + e.getValue(),toList()));

C. List<String> totalByUser = tList.stream()
        .collect(groupingBy(Sale::getCustomer,
                flatMapping(t -> t.getItems().stream(),
                summingDouble(Book::getPrice))))
        .entrySet().stream()
        .sorted(Comparator.comparing(Entry::getValue))
        .collect(mapping(e -> e.getKey() + ":" + e.getValue(),toList()));

D. List<String> totalByUser = tList.stream()
        .collect(flatMapping(t -> t.getItems().stream(),
                groupingBy(Sale::getCustomer,
                summingDouble(Book::getPrice))))
        .sorted(Comparator.comparing (Entry::getValue))
        .collect(mapping(e -> e.getKey() + ":" + e.getValue(),toList()));
```

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

Answer: C

NEW QUESTION 4
Given:

```
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 5
Which two commands are used to identify class and module dependencies? (Choose two.)

A. jmod describe
B. java Hello.java

C. jdeps --list-deps
D. jar --show-module-resolution
E. java --show-module-resolution

Answer: CE

NEW QUESTION 6
Given the code fragment:

```
int x = 0;
while(x < 10){
    System.out.print(x++);
}
```

Which "for" loop produces the same output?

A.

```
int b = 0;
for( ; b < 10; ){
    System.out.print(++b);
}
```

B.

```
for(a; a < 10; a++){
    System.out.print(a);
}
```

C.

```
for(int d = 0; d < 10; ){
    System.out.print(d);
    ++d;
}
```

D.

```
for(int c = 0; ; c++){
    System.out.print(c);
    if(c == 10){
        break;
    }
}
```

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

Answer: C

NEW QUESTION 7
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 8
Given:

```
public class A {
  private boolean checkValue(int val) {
    return true;
  }
}
```

and

```
public class B extends A {
  public int modifyVal(int val) {
    if(checkValue(val)) {
      return val;
    } else {
      return 0;
    }
  }
  public static void Main(String[] args) {
    B b = new B();
    System.out.println(b.modifyVal(10));
  }
}
```

What is the result?

A. nothing
B. It fails to compile.
C. A java.lang.IllegalArgumentException is thrown.
D. 10

Answer: B

Explanation:

```
1 ▾ public class A {
2 ▾     private boolean checkValue(int val) {
3           return true;
4       }
5   }
6   and
7 ▾ public class B extends A {
8 ▾     public int modifyVal(int val) {
9 ▾         if(checkValue(val)) {
10              return val;
11 ▾        } else {
12              return 0;
13          }
14      }
15 ▾     public static void Main(String[] args) {
16          B b = new B();
17          system.out.println(b.modfiyVal (10));
18      }
19  }
```

❯ Execute Mode, Version, Inputs & Arguments

JDK 11.0.4 ❯

CommandLine Arguments

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

```
/A.java:6: error: class, interface, or enum expected
and
 ^
1 error
```

NEW QUESTION 9
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 10
Examine this excerpt from the declaration of the java.se module:

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

What does the transitive modifier mean?

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

Answer: A

NEW QUESTION 10
Given the code fragment:
Path currentFile = Paths.get("/scratch/exam/temp.txt"); Path outputFile = Paths get("/scratch/exam/new.txt"); Path directory = Paths.get("/scratch/");
Files.copy(currentFile, outputFile); Files.copy(outputFile, directory);
Files.delete (outputFile);
The /scratch/exam/temp.txt file exists. The /scratch/exam/new.txt and /scratch/new.txt files do not exist. What is the result?

A. /scratch/exam/new.txt and /scratch/new.txt are deleted.
B. The program throws a FileaAlreadyExistsException.
C. The program throws a NoSuchFileException.
D. A copy of /scratch/exam/new.txt exists in the /scratch directory and /scratch/exam/new.txt is deleted.

Answer: C

Explanation:

```
27     public class Main {
28       public static void main(String[] args) {
29       Path currentFile = Paths.get("/scratch/exam/temp.txt");
30       Path outputFile = Paths.get("/scratch/exam/new.txt");
31       Path directory = Paths.get("/scratch/");
32
33       Files.copy(currentFile, outputFile);
34       Files.copy(outputFile, directory);
35       Files.delete (outputFile);
36     }
37     }
38
```

NEW QUESTION 12
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 15
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 20
Given:
jdeps -jdkinternals C:\workspace4\SimpleSecurity\jar\classes.jar
Which describes the expected output?

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

Answer: A

Explanation:
-jdkinternals option analyzes all classes in the .jar for class-level dependencies on JDK internal APIs. If any are found, the results with suggested replacements are output in the console.
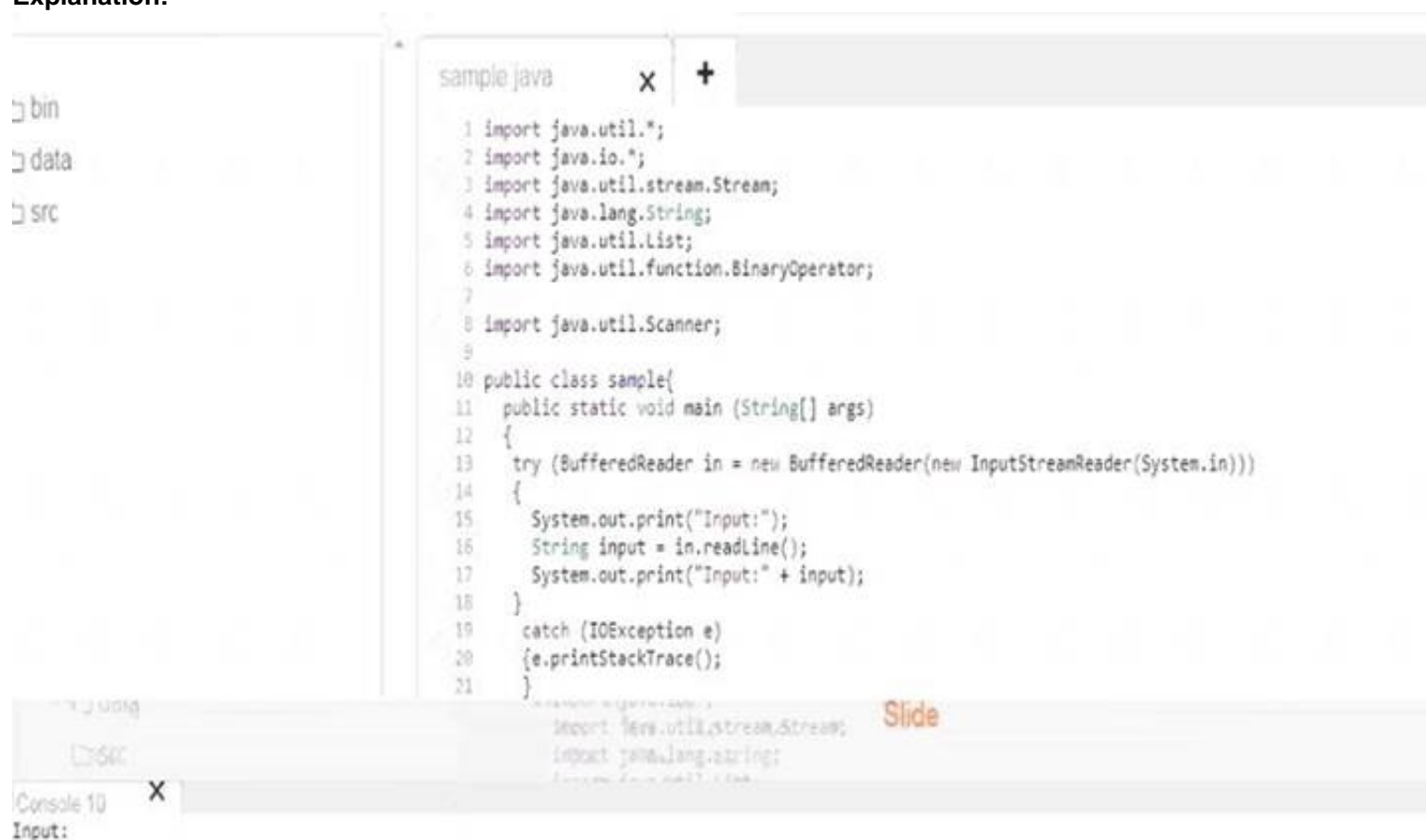
NEW QUESTION 23
Given:
```
public class Main {
    public static void main(String[] args) {
        try(BufferedReader in = new BufferedReader(new InputStreamReader(System.in))) {
            System.out.print("Input: ");
            String input = in.readLine();
            System.out.println("Echo: " + input);
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```
And the command: java Main Helloworld What is the result ?

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

Answer: C

Explanation:

NEW QUESTION 26

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

Answer: B

NEW QUESTION 29
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 33
Which command line runs the main class com.acme.Main from the module com.example?

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

Answer: D

NEW QUESTION 38
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 39
Given the code fragment:
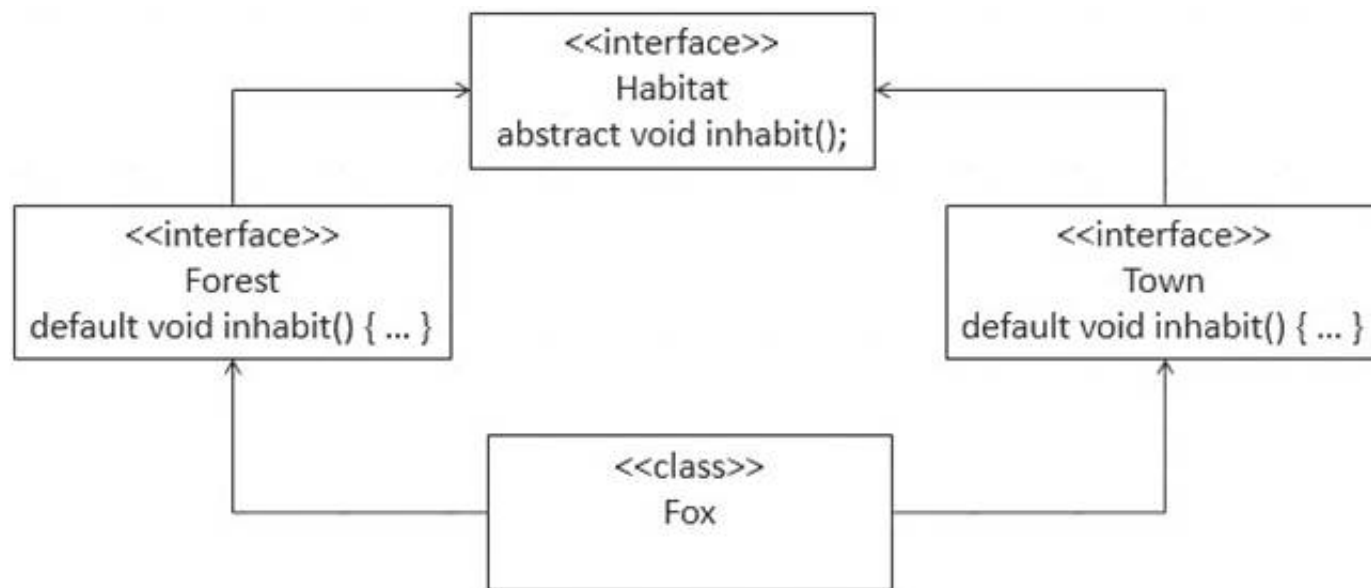var pool = Executors.newFixedThreadPool(5);
Future outcome = pool.submit(() > 1);
Which type of lambda expression is passed into submit()?

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

Answer: D

NEW QUESTION 40
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 45
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 47
What makes Java dynamic?

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

Answer: A

NEW QUESTION 49
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 52

Given:

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

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

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

Answer: D

Explanation:

```
1   import java.util.*;
2   import java.io.*;
3   import java.lang.Thread;
4   import java.util.ArrayList;
5   import java.util.LinkedList;
6   import java.util.List;
7
8   public class Test {
9
10      public static void doThings() throws GeneralException {
11          try{
12              throw new RuntimeException("Something happened");
13          } catch (Exception e) {
14              throw new SpecificException (e.getMessage());
15
16          }
17      }
18
19      public static void main(String args[]) {
20          try{
21              Test.doThings();
22          }catch (Exception e) {
23              System.out.println(e.getMessage());
24          }
25      }
26      class GeneralException extends Exception {
27          public GeneralException(String s) { super(s); }
28      }
29      class SpecificException extends GeneralException {
30          public SpecificException(String s) { super(s);}
31      }
32  }
```

NEW QUESTION 56

Consider this method declaration:

```
void setSessionUser(Connection conn, String user) throws SQLException {
    Statement stmt = conn.createStatement();
    String sql = <EXPRESSION>;
    stmt .execute();
}
```

A) "SET SESSION AUTHORIZATION " + user
B) "SET SESSION AUTHORIZATION " + stmt.enquoteIdentifier(user) Is A or B the correct replacement for <EXPRESSION> and why?

A. A, because it sends exactly the value of user provided by the calling code.
B. B, because enquoting values provided by the calling code prevents SQL injection.
C. A and B are functionally equivalent.
D. A, because it is unnecessary to enclose identifiers in quotes.
E. B, because all values provided by the calling code should be enquoted.

Answer: A

NEW QUESTION 57
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 59
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 61
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 64
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 68
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 70
Given:

```
class Employee {
    String office;
}
```

and the code fragment:

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

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

A. line 12
B. line 6
C. line 9
D. line 8
E. line 7

Answer: BE

NEW QUESTION 75
A company has an existing sales application using a Java 8 jar file containing packages: com.company.customer;
com.company.customer.orders; com.company.customer.info; com.company.sales;
com.company.sales.leads; com.company.sales.closed; com.company.orders; com.company.orders.pending; com.company.orders.shipped.

To modularize this jar file into three modules, customer, sales, and orders, which module-info.java would be correct?
A)
```
module com.company.customer {
    opens com.company.customer;
}
module com.company.sales{
    opens com.company.sales;
}
module com.company.orders {
    opens com.company.orders;
}
```
B)
```
module com.company.customer {
    exports com.company.customer;
}
module com.company.sales{
    exports com.company.sales;
}
module com.company.orders{
    exports com.company.orders;
}
```
C)
```
module com.company.customer {
    requires com.company.customer;
}
module com.company.sales{
    requires com.company.sales;
}
module com.company.orders {
    requires com.company.orders;
}
```
D)
```
module com.company.customer {
    provides com.company.customer;
}
module com.company.sales{
    provides com.company.sales;
}
module com.company.orders {
    provides com.company.orders;
}
```

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

Answer: C

NEW QUESTION 80
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 83
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 88
Given the code fragment:

```
int x = 0;
do {
    x++;
    if (x == 1) {
        continue;
    }
    System.out.println(x);
} while(x < 1);
```

What is the result?

A. 01
B. 1
C. The program prints nothing.
D. It prints 1 in the infinite loop.

Answer: D

NEW QUESTION 89
Given:

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

Examine these requirements:

› Eliminate code duplication.

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

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

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

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

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

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

Answer: B

NEW QUESTION 94
Given:

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

Which is true?

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

Answer: C

NEW QUESTION 96
Given:

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

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

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

Answer: B

NEW QUESTION 98
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 102
Which three guidelines are used to protect confidential information? (Choose three.)

A. Limit access to objects holding confidential information.
B. Clearly identify and label confidential information.
C. Manage confidential and other information uniformly.
D. Transparently handle information to improve diagnostics.
E. Treat user input as normal information.
F. Validate input before storing confidential information.
G. Encapsulate confidential information.

Answer: ADF

NEW QUESTION 104
Given these two classes:

```
public class Resource {
    public Worker owner;
    public synchronized boolean claim(Worker worker) {
        if (owner == null) {
            owner = worker;
            return true;
        }
        else return false;
    }
    public synchronized void release() {
        owner = null;
    }
}
public class Worker {
    public synchronized void work(Resource... resources) {
        for (int i = 0; i < 10; i++) {
            while (!resources[0].claim(this)) { }
            while (!resources[1].claim(this)) { }
            // do work with resource
            resources[1].release();
            resources[0].release();
        }
    }
}
```

And given this fragment:

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

Which describes the fragment?

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

Answer: D

NEW QUESTION 105
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 108
Which three initialization statements are correct? (Choose three.)

A. int x = 12_34;
B. short sh = (short)'A';
C. String contact# = "(+2) (999) (232)";
D. boolean true = (4 == 4);
E. float x = 1.99;
F. int[][] e = {{1,1},{2,2}};
G. byte b = 10;char c = b;

Answer: ABF

NEW QUESTION 113
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:

```
Console 4   ✖    Console 5   ✖
hello
Completed with exit code: 0
```

NEW QUESTION 114
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 118
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.tim
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 119
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 123
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 strea
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 strea
F. The stream is already open.

Answer: D

NEW QUESTION 125
Given:

```
public class Main {
    public static void main(String[] args) {
        Thread t1 = new Thread(new MyThread());
        Thread t2 = new Thread(new MyThread());
        Thread t3 = new Thread(new MyThread());

        t1.start();
        t2.run();
        t3.start();

        t1.start();
    }
}
class MyThread implements Runnable {
    public void run() {
        System.out.println("Running.");
    }
}
```

Which one is correct?

A. An IllegalThreadStateException is thrown at run time.
B. Three threads are created.
C. The compilation fails.
D. Four threads are created.

Answer: A

Explanation:

```
CPU Time: 0.15 sec(s), Memory: 32150 kilobyte(s)

Running.
Running.
Running.

Exception in thread "main" java.lang.IllegalThreadStateException
    at java.base/java.lang.Thread.start(Thread.java:794)
    at Main.main(Main.java:12)
```
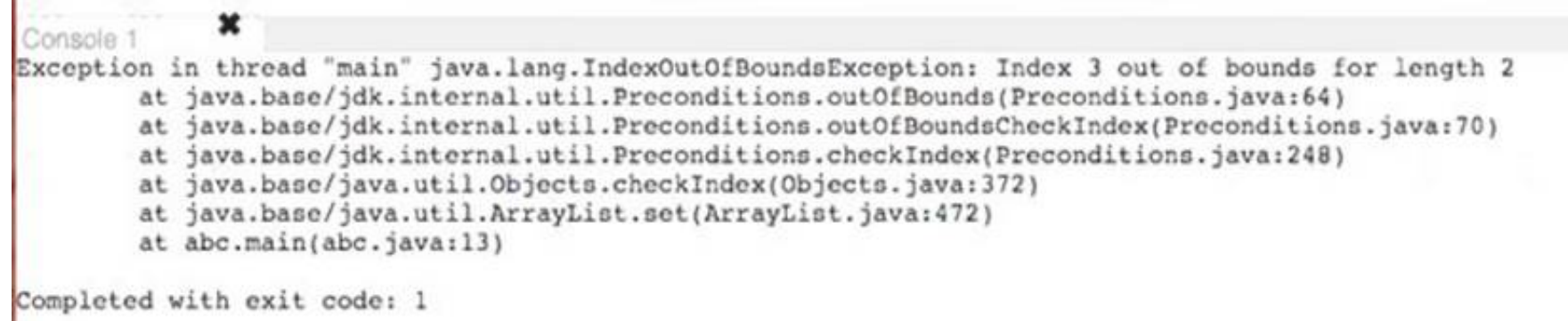
NEW QUESTION 126
Given:
var data = new ArrayList<>(); data.add("Peter");

data.add(30); data.add("Market Road"); data.set(1, 25); data.remove(2); data.set(3, 1000L); System.out.print(data); What is the output?

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

Answer: D

Explanation:

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

Completed with exit code: 1
```

NEW QUESTION 130
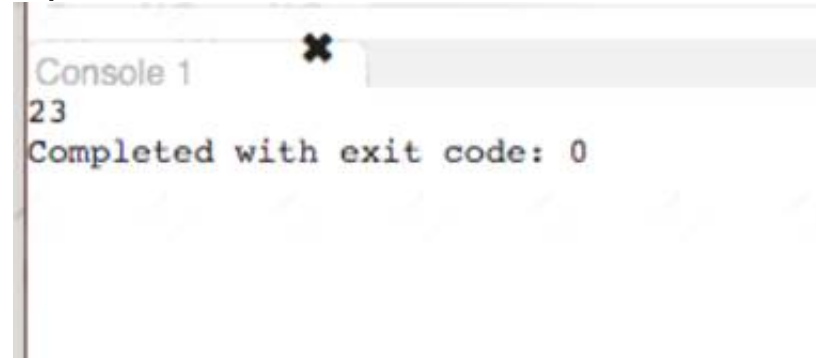Given the code fragment:

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

What is the result?

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

Answer: A

Explanation:

```
Console 1           ✖
23
Completed with exit code: 0
```

NEW QUESTION 131
Given:

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

and

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

What is the result?

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

Answer: D

Explanation:

```
Console 1        ✖   Console 2    ✖   Console 3    ✖
null
Mary

Completed with exit code: 0
```

NEW QUESTION 134
Given:
```java
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 136
Which statement about access modifiers is correct?

A. An instance variable can be declared with the static modifier.
B. A local variable can be declared with the final modifier.
C. An abstract method can be declared with the private modifier.
D. An inner class cannot be declared with the public modifier.
E. An interface can be declared with the protected modifier.

Answer: B

NEW QUESTION 137
Given:
```
public class Test {
   private String[] strings;
}
```
Which two constructors will compile and set the class field strings? (Choose two.)
A.
```
public Test(List<String> strings) {
   this.strings = strings;
}
```

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

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

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

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

A. Option A

B. Option B
C. Option C
D. Option D
E. Option E

Answer: CE

NEW QUESTION 138
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)
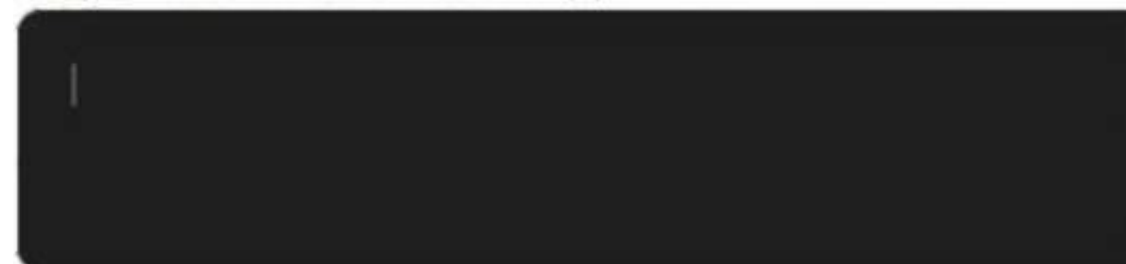
NEW QUESTION 143
Given the Person class with age and name along with getter and setter methods, and this code fragment:

```
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 148
Given:
/code/a/Test.java containing:

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

and
/code/b/Best.java containing: package b;
public class Best { }
Which is the valid way to generate bytecode for all classes?

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

Answer: E

NEW QUESTION 151
Given:
public class X {
}
and
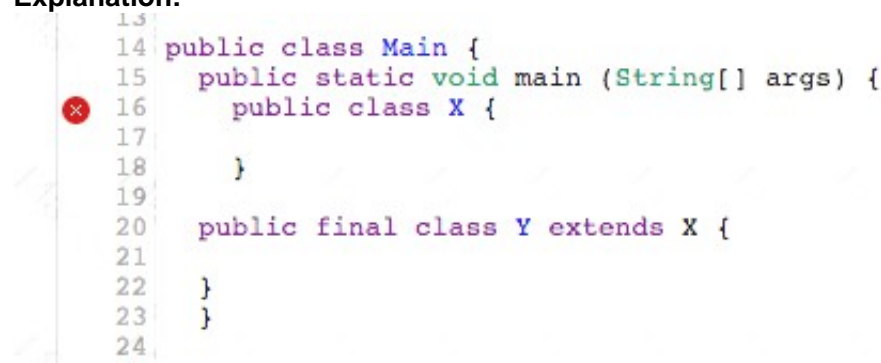public final class Y extends X {
}
What is the result of compiling these two classes?

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

Answer: B

Explanation:

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

NEW QUESTION 155
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 158
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 163
Given:
```
public class Employee {
    private String name;
    private LocalDate birthday;
    // the constructors, getters, and setters methods go here
}
```
and
```
List<Employee> roster = new ArrayList<>();
// ...
Predicate<Employee> y = (Employee e) -> e.getBirthday()
    .isBefore(IsoChronology.INSTANCE.date(1989, 1, 1));
Set<String> s1 = roster.stream()
// Line 1
```

Which code fragment on line 1 makes the s1 set contain the names of all employees born before January 1, 1989?

```
A. .collect(Collectors.partitioningBy(y))
    .get(true)
    .stream()
    .map(Employee::getName)
    .collect(Collectors.toCollection(TreeSet::new));

B. .collect(Collectors.partitioningBy(y))
    .get(true)
    .map(Employee::getName)
    .collect(Collectors.toSet());

C. .collect(Collectors.partitioningBy(y, Collectors.mapping(
        Employee::getName, Collectors.toSet())));

D. .collect(Collectors.partitioningBy(y, Collectors.groupingBy(
        Employee::getName, Collectors.toCollection(TreeSet::new))));
```

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

Answer: B

NEW QUESTION 165
Given:
```
public class Main {
    public static void main(String[] args) {
        var numbers = List.of(1,2,3,4,5,6,7,8,9,10);
        Optional<Integer> result = numbers.stream().filter(x -> x % 3 != 0).reduce((i, j)
-> i + j);
        result.ifPresent(System.out::print); // line 1
```

Which is true about line 1?

A. If the value is not present, a NoSuchElementException is thrown at run time.
B. It always executes the System.out::print statement.
C. If the value is not present, a NullPointerException is thrown at run time.
D. If the value is not present, nothing is done.

Answer: D

Explanation:

```
1   import java.util.*;
2   import java.io.*;
3   import java.lang.Thread;
4   import java.util.ArrayList;
5   import java.util.LinkedList;
6   import java.util.List;
7   import java.util.function.Consumer;
8   import java.util.stream.Stream;
9   import java.util.stream.IntStream;
10  import java.util.Optional.
11
12
13 · public class Main {
14 ·     public static void main(String[] args) {
15          var numbers = List.of(1,2,3,4,5,6,7,8,9,10);
16          Optional<Integer> result = numbers.stream().filter (x -> x % 3 != 0).reduce( (i, j) -> i + j);
17
18      }
19  }
```
Result

CPU Time: 0.18 sec(s), Memory: 33380 kilobyte(s)

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

NEW QUESTION 166
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 169
Given:

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

and

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

and

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

What is the result?

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

Answer: A

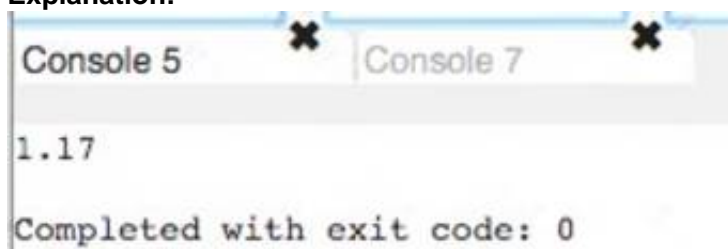**NEW QUESTION 170**
Given:

```
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:

Console 5   ✖   Console 7

1.17

Completed with exit code: 0

NEW QUESTION 172
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 175
Given:

```
List<Reader> dataFiles = new ArrayList<>();
File indexFile = new File("MyIndex.idx");
try (BufferedReader indexReader =
        new BufferedReader(new FileReader(indexFile))) {
    for(String file = indexReader.readbine(); file != null;
        file = indexReader.readLine()) {
     BufferedReader dataReader = new BufferedReader (
        new FileReader(new File(file))); // Line 1
     dataFiles.add(dataReader); // Line 2
     processData(dataReader); // Line 3
    }
} catch (IOException ex) {
    ...
} finally {
    for(Reader r : dataFiles) {
        try {
            r.close();
        } catch (IOException ex) (
            ...
        } // Line 4
    }
}
```

What will secure this code from a potential Denial of Service condition?

A. After Line 4, add indexReader.close().
B. On Line 3, enclose processData(dataReader) with try with resources.
C. After Line 3, add dataReader.close().
D. On Line 1, use try with resources when opening each dataReader.
E. Before Line 1, check the size of dataFiles to make sure it does not exceed a threshold.

Answer: B

NEW QUESTION 178
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 183
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 184
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 187
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);
  }
}
```
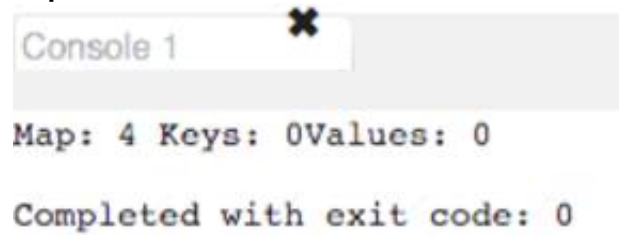
What is the result?

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: 0Values: 0

Completed with exit code: 0

NEW QUESTION 192
Given this enum declaration:

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

Examine this code: System.out.println(Alphabet.getFirstLetter());
What code should be written at line 3 to make this code print A?

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

Answer: C

NEW QUESTION 197
......

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