1Z0-808 Mock Exam

ExamId: 100
Items: 56
Dificulty: HARD

August 5, 2025

(1) (questionId: 101125) What is the output of the following code fragment?

```
int val = 0;
loop1:
for (int i = 0; i < 2; i++) {
    for (int j = 0; j < 4; j++) {
        val++;
        if (j >= 1) {
            continue loop1;
        }
    }
}
System.out.println(val);
```

Choose the most correct answer.

- 0) 2
- 1) 3
- 2) 4
- 3) 8

(2) (questionId: 100028) You are in the directory `/root`. You have the following files: `/root/com/example/App.java` `/root/lib/helper.jar` The class `App` depends on a class inside `helper.jar`. Which command(s) will successfully compile `App.java`? (Choose all that apply) Choose all the correct answer.

- 0) `javac -cp lib/helper.jar com/example/App.java`
- 1) `javac -classpath lib/helper.jar com/example/App.java`
- 2) `javac com/example/App.java -cp lib/helper.jar`
- 3) `javac -cp lib/helper.jar;com/example/App.java`
- 4) `javac -d . -cp lib/helper.jar com/example/App.java`

(3) (questionId: 100226) What happens if a ‘package-info.java‘ file contains a class declaration? Choose the most correct answer.

- 0) The class is associated with the package as metadata.
- 1) It is treated like any other class in the package.
- 2) A compilation error occurs.
- 3) A runtime error occurs when the package is loaded.

(4) (questionId: 100121) What is the result of attempting to compile and run the following code?

```
public class TrickyMain {
    public static void main(String args) {
        System.out.println("Hello");
    }
}
```

Choose the most correct answer.

- 0) It compiles and runs, printing ”Hello”.
- 1) It fails to compile because the ‘main‘ parameter is not an array.
- 2) It compiles, but at runtime the JVM reports that ‘main‘ is not found.
- 3) It compiles and runs, but ‘args‘ is null.

(5) (questionId: 102728) What is the output of this code?

```
List<String> data = new ArrayList<>();
data.add("C");
data.add(null);
data.add("A");
data.sort(Comparator.nullsFirst(Comparator.naturalOrder()));
System.out.println(data);
```

Choose the most correct answer.

- 0) ‘[null, A, C]‘
- 1) ‘[A, C, null]‘
- 2) A ‘NullPointerException‘ is thrown.
- 3) The code does not compile.

(6) (questionId: 100222) You are in the directory ‘/app/bin/com/corp/‘, which contains ‘Main.class‘. The class is declared in package ‘com.corp‘. You run ‘java Main‘. What is the result? Choose the most correct answer.

- 0) The program runs successfully.
- 1) A ‘ClassNotFoundException‘ is thrown.
- 2) A ‘NoClassDefFoundError‘ is thrown with a message about ‘com/corp/Main‘ being found in the wrong place.
- 3) A ‘SecurityException‘ is thrown.

(7) (questionId: 103651) What is the output of the following program?

```
public class ArrayOfObjects {
    static class Bulb { boolean on = false; }

    public static void main(String[] args) {
        Bulb[] bulbs = {new Bulb(), new Bulb()};
        turnOn(bulbs);
        System.out.println(bulbs[0].on + "," + bulbs[1].on);
    }

    public static void turnOn(Bulb[] lights) {
        lights[0].on = true;
        lights[1] = new Bulb();
        lights[1].on = true;
    }
}
```

Choose the most correct answer.

- 0) ‘true,true‘
- 1) ‘true,false‘
- 2) ‘false,true‘
- 3) ‘false,false‘

(8) (questionId: 100427) Which statements are true about division in Java? (Choose all that apply) Choose all the correct answer.

- 0) Dividing a non-zero floating-point number by ‘0.0‘ results in ‘Infinity‘ or ‘-Infinity‘ and does not throw an exception.
- 1) Dividing any integer by ‘0‘ will always result in a compile-time error.
- 2) The expression ‘0.0 / 0.0‘ evaluates to ‘NaN‘ (Not a Number).
- 3) The expression ‘10 / 4‘ evaluates to ‘2.5‘.

(9) (questionId: 103351) What is the result of attempting to compile and run the following code? This question checks your knowledge of object instantiation rules for the Date-Time API.

```
import java.time.LocalDate;

public class ConstructorTest {
    public static void main(String[] args) {
        LocalDate date = new LocalDate(2025, 8, 2);
        System.out.println(date);
    }
}
```

Choose the most correct answer.

- 0) It prints ‘2025-08-02‘.
- 1) It throws a ‘DateTimeException‘ at runtime.
- 2) It fails to compile.
- 3) It prints a reference to the object.

(10) (questionId: 103123) What is the output of this code?

```
class R implements AutoCloseable {
    public R() throws Exception { throw new Exception("R_INIT"); }
    public void close() { System.out.print("R_CLOSE"); }
}
public class TestFinal {
    public static void main(String[] args) {
        try (R r = new R()) {
            System.out.print("TRY");
        } catch (Exception e) {
            System.out.print(e.getMessage());
        } finally {
            System.out.print("_FINAL");
        }
    }
}
```

Choose the most correct answer.

- 0)

  ‘R_INIT_FINAL‘

- 1)

  ‘TRY_R_CLOSE_FINAL‘

- 2)

  ‘R_INIT_R_CLOSE_FINAL‘

- 3)

  ‘R_INIT‘

(11) (questionId: 101222) Examine the following code. What is the result?

```
public enum Operation {
    PLUS {
        public double apply(double x, double y) { return x + y; }
    },
    MINUS {
        public double apply(double x, double y) { return x - y; }
    };
    public abstract double apply(double x, double y);
}

class Test {
    public static void main(String[] args) {
        System.out.println(Operation.PLUS.apply(5, 3));
    }
}
```

Choose the most correct answer.

- 0) ‘8.0‘
- 1) The code fails to compile because an enum cannot be ‘abstract‘.
- 2) The code fails to compile because ‘apply‘ is not defined for the ‘Operation‘ enum itself.

- 3) The code fails to compile because an enum constant cannot provide a method implementation.

(12) (questionId: 101623) What is the output of this program?

```
public class StaticForward {
    static {
        System.out.print(x + " ");
    }
    private static int x = initX();
    static {
        System.out.print(x + " ");
    }

    private static int initX() {
        System.out.print("initX ");
        return 10;
    }

    public static void main(String[] args) {
        // Class loading is triggered by main method lookup
    }
}
```

Choose the most correct answer.

- 0) 0 initX 10
- 1) initX 10 10
- 2) The code fails to compile due to illegal forward reference.
- 3) initX 0 10

(13) (questionId: 102424) Which of the following statements are true? (Choose all that apply) Choose all the correct answer.

- 0) `int[] x, y[];` declares `x` as a 1D array and `y` as a 2D array.
- 1) An array's size can be changed after it has been created.
- 2) `new int[0]` creates an array of size 0.
- 3) An `ArrayStoreException` is a checked exception.

(14) (questionId: 101324) How many 'String' objects are created in the following code, not including any pre-existing literals in the string pool?

```
String s1 = new String("Hello");
String s2 = " World";
String s3 = s1 + s2;
```

Choose the most correct answer.

- 0) 1

- 1) 2
- 2) 3
- 3) 4

(15) (questionId: 101721) What is the output of the following code? This question tests method hiding.

```
class Animal {
    static void eat() { System.out.println("Animal eats"); }
}
class Dog extends Animal {
    static void eat() { System.out.println("Dog eats"); }
}
public class Test {
    public static void main(String[] args) {
        Animal myAnimal = new Dog();
        myAnimal.eat();
    }
}
```

Choose the most correct answer.

- 0) Animal eats
- 1) Dog eats
- 2) The code fails to compile.
- 3) A runtime exception is thrown.

(16) (questionId: 101826) Select all lines of code after which at least one ‘Gadget‘ object becomes eligible for garbage collection.

```
class Gadget {}
public class GadgetFactory {
    static Gadget staticGadget = new Gadget(); // Line 1
    Gadget instanceGadget = new Gadget();      // Line 2

    public static void main(String[] args) {
        GadgetFactory gf = new GadgetFactory(); // Line 3
        Gadget g1 = new Gadget();               // Line 4
        gf.build(g1);
        g1 = null;                              // Line 5
        gf = null;                              // Line 6
    }

    void build(Gadget g) {
        Gadget g2 = new Gadget();               // Line 7
    } // End of build method is effectively Line 8
}
```

Choose all the correct answer.

- 0) Line 5
- 1) Line 6
- 2) Line 8
- 3) The line after the ‘main‘ method completes.
- 4) Line 3

(17) (questionId: 102822) What is the outcome of compiling and running the following code?

```
public class Test {
    static {
        if (true) {
            throw new NullPointerException("Error in static block");
        }
    }
    public static void main(String[] args) {
        System.out.println("Hello");
    }
}
```

Choose the most correct answer.

- 0) A ‘NullPointerException‘ is caught by the JVM and ‘Hello‘ is printed.
- 1) The program prints ‘Hello‘ and exits normally.
- 2) An ‘ExceptionInInitializerError‘ is thrown, and the program terminates.
- 3) A ‘NullPointerException‘ is thrown, and the program terminates.

(18) (questionId: 100728) Which statements about the following code are correct? (Choose all that apply)

```
public class Outer {
    private String name = "Outer";

    class Inner {
        private String name = "Inner";

        void printNames() {
            String name = "Local";
            System.out.println(name);
            System.out.println(this.name);
            System.out.println(Outer.this.name);
        }
    }

    public static void main(String... args) {
        new Outer().new Inner().printNames();
    }
}
```

Choose all the correct answer.

- 0) The code will fail to compile.
- 1) The output will be: Local
- 2) The output will be: Local Inner Outer
- 3) `this.name` refers to the instance variable of the `Inner` class.
- 4) `Outer.this.name` is used to access the instance variable of the enclosing `Outer` class.

(19) (questionId: 103227) Which of the following assignments will cause a compilation error?

```
import java.util.function.*;
import java.io.IOException;
```

Choose the most correct answer.

- 0) ‘Function¡String, Integer¿ f = s -¿ if(s==null) throw new IOException(); return s.length(); ;‘
- 1) ‘Runnable r = () -¿ try Thread.sleep(100); catch (Exception e) ;‘
- 2) ‘Predicate¡String¿ p = (final String s) -¿ s.isEmpty();‘
- 3) ‘Object o = (Runnable) () -¿ System.out.println(”Hi”);‘

(20) (questionId: 100321) What is the result of attempting to compile the following code?

```
public class NestedComment {
    /*
     * This is an outer comment.
     * /* This is a nested comment. */
     * The outer comment ends here.
     */
    public static void main(String[] args) {
        System.out.println("Hello");
    }
}
```

Choose the most correct answer.

- 0) Compilation is successful, and the program prints ”Hello”.
- 1) Compilation fails due to an unclosed comment.
- 2) Compilation is successful, but a warning is issued about nested comments.
- 3) Compilation fails due to illegal syntax inside a comment.

(21) (questionId: 100828) What is the output of the code below?

```
int i = -1;
i = i >>> 30;
System.out.println(i);
```

Choose the most correct answer.

- 0) -1
- 1) 0
- 2) 1
- 3) 3

(22) (questionId: 101527) Which statement about the ‘final‘ instance variable ‘ID‘ is correct?

```
public class Record {
    private final int ID;

    public Record(int id) {
        this.ID = id;
    }

    public void setId(int id) {
        // Line X
        ID = id;
    }
}
```

Choose the most correct answer.

- 0) The code is correct as is.
- 1) The code will fail to compile because a ‘final‘ variable cannot be assigned in a constructor.
- 2) The code will fail to compile at Line X because a ‘final‘ variable cannot be reassigned.
- 3) The code will compile but throw a runtime exception at Line X.

(23) (questionId: 101023) What is the result of attempting to compile and run this code?

```
import java.util.List;

public class NullEnhancedFor {
    public static void main(String[] args) {
        List<String> list = null;
        for (String s : list) {
            System.out.println("This will not be printed");
        }
    }
}
```

Choose the most correct answer.

- 0) It compiles and runs, producing no output.
- 1) It fails to compile.
- 2) It compiles, but throws a ‘NullPointerException‘ at runtime.
- 3) It compiles, but throws an ‘IllegalStateException‘ at runtime.

(24) (questionId: 103453) Consider an interface with a static method (a Java 8 feature). What is the result of this code?

```
// File: I.java
public interface I {
    static void run() { System.out.println("I"); }
}

// File: C.java
public class C {
    public static void run() { System.out.println("C"); }
}

// File: Main.java
import static I.*;
import static C.*;

public class Main {
    public static void main(String[] args) {
        run();
    }
}
```

Choose the most correct answer.

- 0) It prints ‘I‘.
- 1) It prints ‘C‘.
- 2) The code fails to compile due to ambiguity.
- 3) The code fails to compile because you cannot statically import methods from an interface.

(25) (questionId: 102524) Consider the following code:

```
import java.util.List;
import java.util.ArrayList;
import java.util.Arrays;

public class Test {
    public static void main(String[] args) {
        List<String> list1 = new ArrayList<>(Arrays.asList("A","B"));
        List<String> list2 = new ArrayList<>(Arrays.asList("B","A"));
        List<String> list3 = new ArrayList<>(Arrays.asList("A","B"));
        System.out.print(list1.equals(list2));
        System.out.print(", ");
        System.out.print(list1.equals(list3));
    }
}
```

What is the output? Choose the most correct answer.

- 0) true, true
- 1) true, false
- 2) false, true
- 3) false, false

(26) (questionId: 100723) Consider the following class. What is the outcome?

```
public class Test {
    static {
        i = 20; // Forward reference is ok in assignment
    }
    static int i = 10;

    public static void main(String[] args) {
        System.out.println(i);
    }
}
```

Choose the most correct answer.

- 0) 20
- 1) 10
- 2) Compilation fails due to illegal forward reference.
- 3) 0

(27) (questionId: 102322) What is the result of compiling the following code?

```
public class Test {
    public void process() {
        final int x;
        try {
            x = 10;
        } catch (Exception e) {
            // x is not initialized here
        }
        // System.out.println(x); // Uncomment this line
    }
}
```

If the final line is uncommented, what happens? Choose the most correct answer.

- 0) The code compiles fine.
- 1) A compile-time error occurs because ‘x‘ may not have been initialized.
- 2) A compile-time error occurs because a final variable cannot be initialized inside a ‘try‘ block.
- 3) The code compiles but throws an ‘IllegalStateException‘ at runtime.

(28) (questionId: 101427) Given ‘StringBuilder sb = new StringBuilder(”abcde”);‘. Which statements about its capacity are true? (Choose all that apply) Choose all the correct answer.

- 0) The initial capacity is 21 (5 for ”abcde” + 16 default).
- 1) ‘sb.trimToSize();‘ will likely change its capacity to 5.
- 2) After ‘sb.append(”fghijklmnopqrstuvwxyz”);‘, the capacity will be larger than its length.
- 3) ‘sb.ensureCapacity(10);‘ will not change the capacity.

(29) (questionId: 101821) Examine this code carefully:

```
public class Zombie {
    static Zombie zombie;
    @Override
    protected void finalize() {
        System.out.print("X");
        zombie = this; // Resurrection
    }

    public static void main(String[] args) throws InterruptedException {
        Zombie z = new Zombie();
        z = null;
        System.gc();
        Thread.sleep(100); // Allow time for finalization

        if (zombie != null) {
            zombie = null;
            System.gc();
            Thread.sleep(100); // Allow time for GC again
        }
        System.out.print("Y");
    }
}
```

What is the most likely output? Choose the most correct answer.

- 0) XY
- 1) XXY
- 2) Y

- 3) YX
- 4) The output is unpredictable.

(30) (questionId: 103551) What is the result of attempting to compile and run this code? This tests overloading resolution with autoboxing and varargs.

```
public class BoxingTest {
    static void run(Integer i) { System.out.println("Integer"); }
    static void run(long... l) { System.out.println("long..."); }

    public static void main(String[] args) {
        int myInt = 10;
        run(myInt);
    }
}
```

Choose the most correct answer.

- 0) ‘Integer‘
- 1) ‘long...‘
- 2) The code fails to compile due to ambiguity.
- 3) The code fails to compile for another reason.

(31) (questionId: 100626) Which two wrapper classes have caches that are mandated by the Java Language Specification to be at least for the range -128 to 127? Choose the most correct answer.

- 0) `Integer` and `Long`
- 1) `Integer` and `Short`
- 2) `Byte` and `Short`
- 3) The caching behavior is implementation-specific for all wrapper types.

(32) (questionId: 100829) What is the result of the following code? (Choose all that apply)

```
public class Test {
    public static void main(String[] args) {
        int i = 0;
        boolean t = true;
        boolean f = false, b;
        b = (t || ((i++) == 0));
        System.out.println(i);
        b = (f || ((i++) == 0));
        System.out.println(i);
    }
}
```

Choose all the correct answer.

- 0) The first output is 0.
- 1) The first output is 1.
- 2) The second output is 0.
- 3) The second output is 1.

(33) (questionId: 100425) Which of the following lines of code will result in a compilation error? (Choose all that apply) Choose all the correct answer.

- 0) ‘byte b = 127; b++;‘
- 1) ‘char c = -1;‘
- 2) ‘float f = 1.0;‘
- 3)

  ```
  ‘int i = 1_00L;‘
  ```

(34) (questionId: 100925) What is the output of the following code?

```
public enum Color { RED, GREEN, BLUE }

public class EnumSwitch {
    public static void main(String[] args) {
        Color color = Color.BLUE;
        switch (color) {
            case RED:
                System.out.print("R");
                break;
            case GREEN:
                System.out.print("G");
                break;
            default:
                System.out.print("X");
            case BLUE:
                System.out.print("B");
        }
    }
}
```

Choose the most correct answer.

- 0) B
- 1) XB
- 2) X
- 3) A compilation error occurs.

(35) (questionId: 102228) What is the result of attempting to compile and run the Test class?

```
interface I1 {
```

```
    default void go() { System.out.println("I1"); }
}
interface I2 {
    default void go() { System.out.println("I2"); }
}
class C1 implements I1, I2 {
    public void go() {
        I1.super.go();
    }
}
public class Test {
    public static void main(String[] args) {
        new C1().go();
    }
}
```

Choose the most correct answer.

- 0) A compile-time error at ‘class C1‘.
- 1) The code compiles and prints ”I1”.
- 2) The code compiles and prints ”I2”.
- 3) A compile-time error at ‘I1.super.go();‘ because ‘super‘ can only be used with classes.

(36) (questionId: 102021) What is the output of this code?

```
class Mammal {
    public Mammal(int age) {
        System.out.print("Mammal");
    }
}
class Platypus extends Mammal {
    public Platypus() {
        super(5);
        System.out.print("Platypus");
    }
}
public class TestOrder extends Platypus {
    public TestOrder() {
        System.out.print("TestOrder");
    }
    public static void main(String[] args) {
        new TestOrder();
    }
}
```

Choose the most correct answer.

- 0) TestOrderPlatypusMammal

- 1) MammalPlatypusTestOrder
- 2) TestOrder
- 3) Compilation fails because of constructor issues in ‘TestOrder‘.
- 4) Compilation fails because of constructor issues in ‘Platypus‘.

(37) (questionId: 101121) What is the result of attempting to compile and run this code?

```
public class LabeledBlock {
    public static void main(String[] args) {
        int x = 5;
        myBlock: {
            if (x == 5) {
                break myBlock;
            }
            System.out.print("Inside");
        }
        System.out.print("Outside");
    }
}
```

Choose the most correct answer.

- 0) It prints ‘InsideOutside‘.
- 1) It prints ‘Outside‘.
- 2) It prints ‘Inside‘.
- 3) It fails to compile.

(38) (questionId: 102127) Which of the following statements about polymorphism and casting in Java are true? (Choose all that apply) Choose all the correct answer.

- 0) A compile-time error will occur if an ‘instanceof‘ check is performed on an object against a final class that is not in its direct inheritance hierarchy (e.g. ‘”hello” instanceof Integer‘).
- 1) Casting a ‘null‘ reference to any object type will result in a ‘NullPointerException‘.
- 2) When accessing instance variables, the reference type at compile-time determines which variable is used, regardless of the actual object type at runtime.
- 3) Static methods cannot be overridden, but they can be hidden. The version that gets called is determined by the object’s type at runtime.

(39) (questionId: 100529) Which of these code fragments will fail to compile? (Choose all that apply) Choose all the correct answer.

- 0)

  ```
  byte b1 = 1;
  final byte b2 = 2;
  byte b3 = b1 + b2;
  ```

- 1)

  ```
  short s = Short.MAX_VALUE;
  s += 1;
  ```

- 2)

  ```
  char c = 0;
  short s2 = c;
  ```

- 3)

  ```
  float f = 1.0F;
  long l = f;
  ```

(40) (questionId: 102927) What happens when this method is called?

```
public class Test {
    public void go() {
        try {
            System.out.println("Trying");
            return;
        } finally {
            System.out.println("Finalizing");
            throw new RuntimeException("Error in finally");
        }
    }
}
```

Choose the most correct answer.

- 0) The method prints ‘Trying‘ and ‘Finalizing‘ and then returns normally.
- 1) The method prints ‘Trying‘ and then returns normally.
- 2) The method prints ‘Trying‘ and ‘Finalizing‘ and then completes abruptly with a ‘RuntimeException‘.
- 3) The code does not compile.

(41) (questionId: 101325) What is the output of the following code?

```
String text = "a.b.c";
String[] parts = text.split(".");
System.out.println(parts.length);
```

Choose the most correct answer.

- 0) 0
- 1) 1
- 2) 3
- 3) An exception is thrown at runtime.

(42) (questionId: 101026) What will be printed after this code executes?

```
String[] data = {"a", "b", "c"};
int x = 0;
for(;;){
    try {
        System.out.print(data[x++]);
    } catch (ArrayIndexOutOfBoundsException e) {
        break;
    }
}
```

Choose the most correct answer.

- 0) abc
- 1) ab
- 2) a
- 3) An infinite loop occurs.

(43) (questionId: 100623) Which of the following lines will compile without errors? (Choose all that apply) Choose all the correct answer.

- 0) `Integer i = new Integer(null);`
- 1) `Double d = null; double d2 = d;`
- 2) `Byte b = 25;`
- 3) `Short s = new Short((short)10);`
- 4) `long l = new Integer(100);`

(44) (questionId: 101223) What happens when you attempt to compile and run the following code?

```
public enum MyEnum {
    FIRST, SECOND;

    MyEnum() {
        System.out.print(this.ordinal());
    }

    static {
        System.out.print("S");
    }
}

class Test {
    public static void main(String[] args) {
        System.out.print("M");
        MyEnum e = MyEnum.FIRST;
    }
}
```

Choose the most correct answer.

- 0) ‘SM01‘
- 1) ‘01SM‘
- 2) ‘S01M‘
- 3) ‘MS01‘

(45) (questionId: 101723) What is the output of the following code? This question tests static initialization order.

```
public class Init {
    static { a = b * 2; }
    static int a = 10;
    static int b = 5;
    static { a = b * 3; }

    public static void main(String[] args) {
        System.out.println(a);
    }
}
```

Choose the most correct answer.

- 0) 10
- 1) 15
- 2) 30
- 3) The code fails to compile.

(46) (questionId: 102622) Which of these lines causes a compilation error?

```
import java.util.*;

class Mammal {}
class Primate extends Mammal {}
class Human extends Primate {}

public class Test {
    public static void main(String[] args) {
        List<? super Primate> primates = new ArrayList<Mammal>(); // Line 1
        primates.add(new Human());                                // Line 2
        primates.add(new Primate());                              // Line 3
        primates.add(new Mammal());                               // Line 4
    }
}
```

Choose the most correct answer.

- 0) Line 1
- 1) Line 2

- 2) Line 3
- 3) Line 4

(47) (questionId: 101626) What is the output of the following code?

```
public class TrickyInit {
    TrickyInit(int i) {
        System.out.print("C(" + i + ")");
    }

    { System.out.print("I1 "); }

    int x = 1;

    TrickyInit() {
        this(2);
        System.out.print("C() ");
    }

    { System.out.print("I2 "); }

    public static void main(String... args) {
        new TrickyInit();
    }
}
```

Choose the most correct answer.

- 0) I1 I2 C(2)C()
- 1) I1 C(2)I2 C()
- 2) C(2)C() I1 I2
- 3) I1 I2 C()C(2)

(48) (questionId: 101421) What is the output of the following program?

```
public class Test {
    public static void main(String[] args) {
        StringBuilder sb = new StringBuilder("Initial");
        reassign(sb);
        System.out.print(sb + ":");
        modify(sb);
        System.out.print(sb);
    }
    static void reassign(StringBuilder sb) {
        sb = new StringBuilder("New");
    }
    static void modify(StringBuilder sb) {
        sb.append("-Mod");
    }
}
```

Choose the most correct answer.

- 0) ‘Initial:Initial-Mod‘
- 1) ‘New:New-Mod‘
- 2) ‘Initial:Initial‘
- 3) ‘New:Initial-Mod‘

(49) (questionId: 100125) Consider the following code:

```
package com.test;
public class Runner {
    public static void main(String[] args) {
        System.out.println("OK");
    }
}
```

After compiling with ‘javac -d . com/test/Runner.java‘, you are in the ‘com/test‘ directory. You execute ‘java Runner‘. What is the result? Choose the most correct answer.

- 0) It prints ”OK”.
- 1) A ‘ClassNotFoundException‘ is thrown.
- 2) A ‘NoClassDefFoundError‘ is thrown.
- 3) A ‘SecurityException‘ is thrown.

(50) (questionId: 101524) What is the output of the following code?

```
class Wallet {
    public int cash;
}

public class Thief {
    public static void main(String[] args) {
        Wallet w = new Wallet();
        w.cash = 100;
        steal(w);
        System.out.println(w.cash);
    }

    public static void steal(Wallet victimWallet) {
        victimWallet.cash -= 50;
        victimWallet = new Wallet(); // Thief gets a new wallet
        victimWallet.cash = 10;
    }
}
```

Choose the most correct answer.

- 0) 100
- 1) 50
- 2) 10
- 3) 0

(51) (questionId: 101925) Examine the following code:

```
// In package p1
package p1;
public interface CanFly {
    void fly(); // public abstract by default
}

// In package p1
package p1;
public abstract class Bird {
    protected abstract void sing();
}

// In package p2
package p2;
import p1.*;
class Robin extends Bird implements CanFly {
    // Which implementation of fly() is valid?
    // Which implementation of sing() is valid?
}
```

Which pair of method implementations, when inserted into the ‘Robin‘ class, will allow the code to compile? Choose the most correct answer.

- 0) ‘void fly() ‘ and ‘protected void sing() ‘
- 1) ‘public void fly() ‘ and ‘private void sing() ‘
- 2) ‘protected void fly() ‘ and ‘void sing() ‘
- 3) ‘public void fly() ‘ and ‘public void sing() ‘

(52) (questionId: 103022) What is the result of attempting to compile and run the following code?

```
public class StaticFail {
    static {
        if (true) {
            throw new RuntimeException("Initialization failed");
        }
    }

    public static void main(String[] args) {
        System.out.println("Hello");
    }
}
```

Choose the most correct answer.

- 0) The code compiles and prints ‘Hello‘.
- 1) The code does not compile.
- 2) The code compiles, but throws a ‘RuntimeException‘ when run.
- 3) The code compiles, but throws an ‘ExceptionInInitializerError‘ when run.
- 4) The code compiles, but throws a ‘NoClassDefFoundError‘ when run.

(53) (questionId: 100329) Consider the following line of code. How does the Java compiler interpret it?

```
// http://www.example.com?value=1\u0026value=2
```

Choose the most correct answer.

- 0) As a single-line comment, with no special behavior.
- 1) It causes a compilation error because ‘0̆026‘ is not a valid Unicode escape for a character.
- 2) It is interpreted as a comment, but the compiler issues a warning about the unknown Unicode escape.
- 3) It causes a compilation error because ‘‘ is not a valid character to be escaped with ‘‘̆ in this context.

(54) (questionId: 100025) You have a class `com.app.Main` in a compiled JAR file `app.jar`. Which command correctly runs this class? Choose the most correct answer.

- 0) `java app.jar com.app.Main`
- 1) `java -jar com.app.Main app.jar`
- 2) `java -cp app.jar com.app.Main`
- 3) `java com.app.Main -cp app.jar`

(55) (questionId: 100926) Examine this code carefully. What is the result?

```
public class Test {
    public static void main(String[] args) {
        Integer i = 128;
        Integer j = 128;
        int k = 128;

        if (i == j) {
            System.out.print("A");
        }
        if (i == k) {
            System.out.print("B");
        }
    }
}
```

Choose the most correct answer.

- 0) A
- 1) B
- 2) AB
- 3) No output is produced.

(56) (questionId: 100524) What is the final value of ‘s‘?

```
short s = 32767;
s++;
```

Choose the most correct answer.

- 0) ‘32768‘
- 1) ‘-32768‘
- 2) ‘0‘
- 3) The code does not compile.