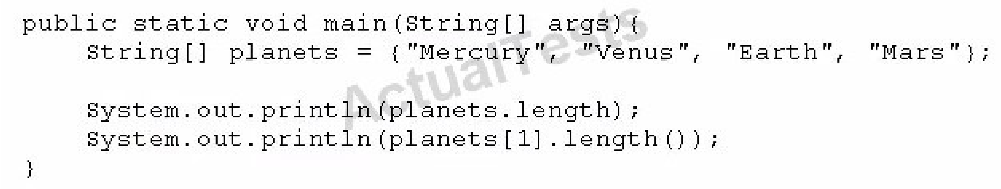
**QUESTION NO: 41**

Given the following code:



```
public static void main(String[] args){
    String[] planets = {"Mercury", "Venus", "Earth", "Mars"};

    System.out.println(planets.length);
    System.out.println(planets[1].length());
}
```

What is the output?

**A.**
4
4

**B.**
3
5

**C.**
4
7

**D.**
5
4

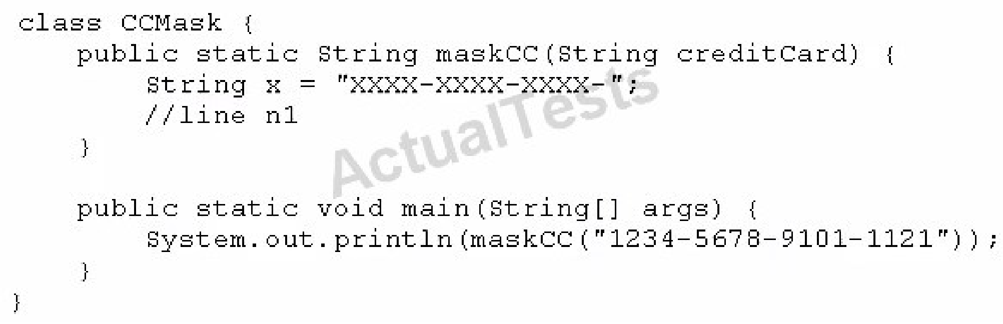
**E.**
4
5

**F.**
4
21

**QUESTION NO: 42**

You are developing a banking module. You have developed a class named ccMask that has a maskcc method.

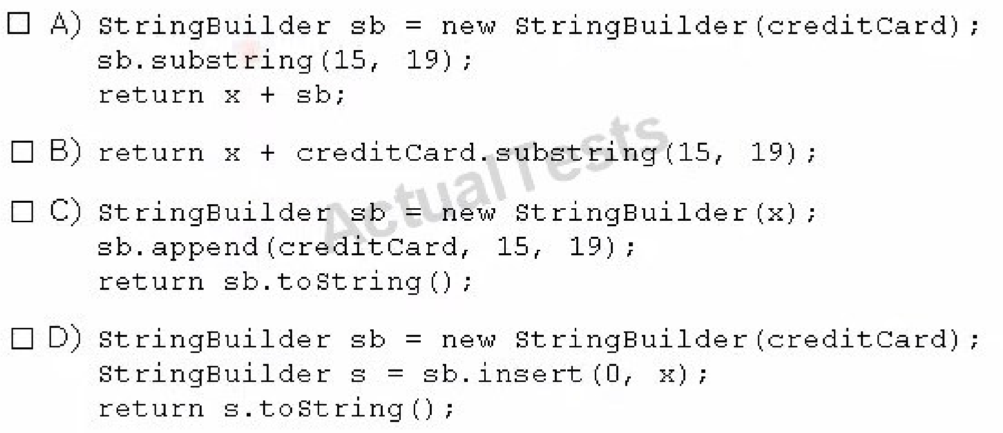
Given the code fragment:



```
class CCMask {
    public static String maskCC(String creditCard) {
        String x = "XXXX-XXXX-XXXX-";
        //line n1
    }

    public static void main(String[] args) {
        System.out.println(maskCC("1234-5678-9101-1121"));
    }
}
```

You must ensure that the maskcc method returns a string that hides all digits of the credit card number except the four last digits (and the hyphens that separate each group of four digits).

Which two code fragments should you use at line n1, independently, to achieve this requirement?



```
☐ A) StringBuilder sb = new StringBuilder(creditCard);
     sb.substring(15, 19);
     return x + sb;

☐ B) return x + creditCard.substring(15, 19);

☐ C) StringBuilder sb = new StringBuilder(x);
     sb.append(creditCard, 15, 19);
     return sb.toString();

☐ D) StringBuilder sb = new StringBuilder(creditCard);
     StringBuilder s = sb.insert(0, x);
     return s.toString();
```

**A.**
Option A

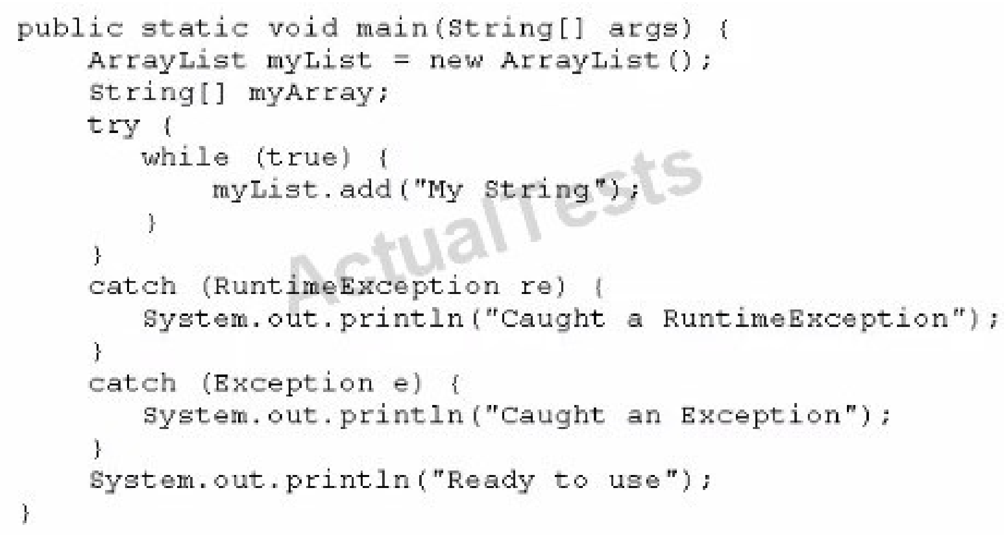
**B.**
Option B

**C.**
Option C

**D.**
Option D

**QUESTION NO: 45**

Given the code fragment:



```
public static void main(String[] args) {
     ArrayList myList = new ArrayList();
     String[] myArray;
     try {
        while (true) {
            myList.add("My String");
        }
     }
     catch (RuntimeException re) {
        System.out.println("Caught a RuntimeException");
     }
     catch (Exception e) {
        System.out.println("Caught an Exception");
     }
     System.out.println("Ready to use");
}
```

What is the result?

**A.** Execution terminates in the first catch statement, and caught a RuntimeException is printed to the console.

**B.** Execution terminates in the second catch statement, and caught an Exception is printed to the console.

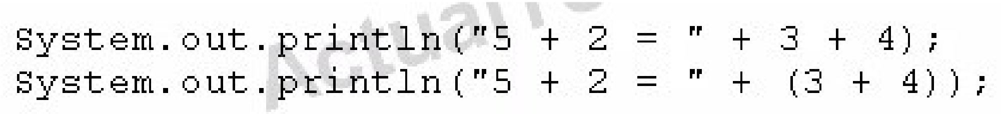
**C.** A runtime error is thrown in the thread "main".

**D.** Execution completes normally, and Ready to use is printed to the console.

**E.** The code fails to compile because a throws keyword is required.

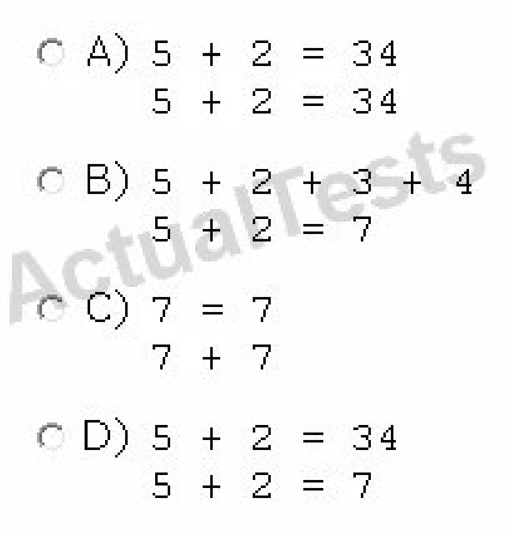
**QUESTION NO: 46**

Given:



```
System.out.println("5 + 2 = " + 3 + 4);
System.out.println("5 + 2 = " + (3 + 4));
```

What is the result?



```
○ A) 5 + 2 = 34
     5 + 2 = 34

○ B) 5 + 2 + 3 + 4
     5 + 2 = 7

○ C) 7 = 7
     7 + 7

○ D) 5 + 2 = 34
     5 + 2 = 7
```

**A.** Option A

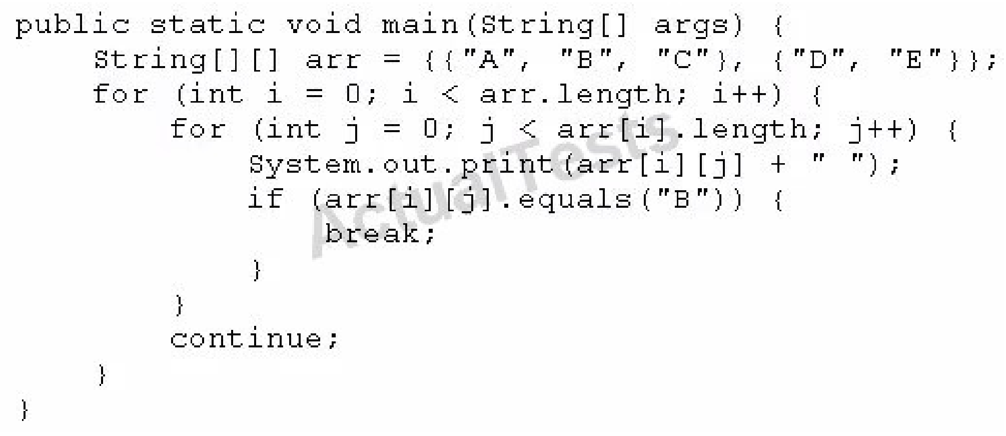
**B.** Option B

**C.** Option C

**D.** Option D

**QUESTION NO: 48**

Given the code fragment:



```
public static void main(String[] args) {
    String[][] arr = {{"A", "B", "C"}, {"D", "E"}};
    for (int i = 0; i < arr.length; i++) {
        for (int j = 0; j < arr[i].length; j++) {
            System.out.print(arr[i][j] + " ");
            if (arr[i][j].equals("B")) {
                break;
            }
        }
        continue;
    }
}
```

What is the result?

**A.** A B C

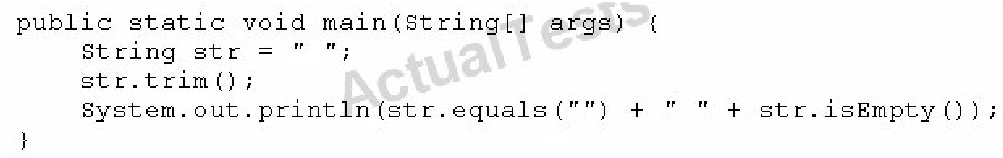
**B.** A B C D E

**C.** A B D E

**D.** Compilation fails.

**QUESTION NO: 49**

Given the code fragment:



```
public static void main(String[] args) {
     String str = " ";
     str.trim();
     System.out.println(str.equals("") + " " + str.isEmpty());
}
```

What is the result?

**A.** true true

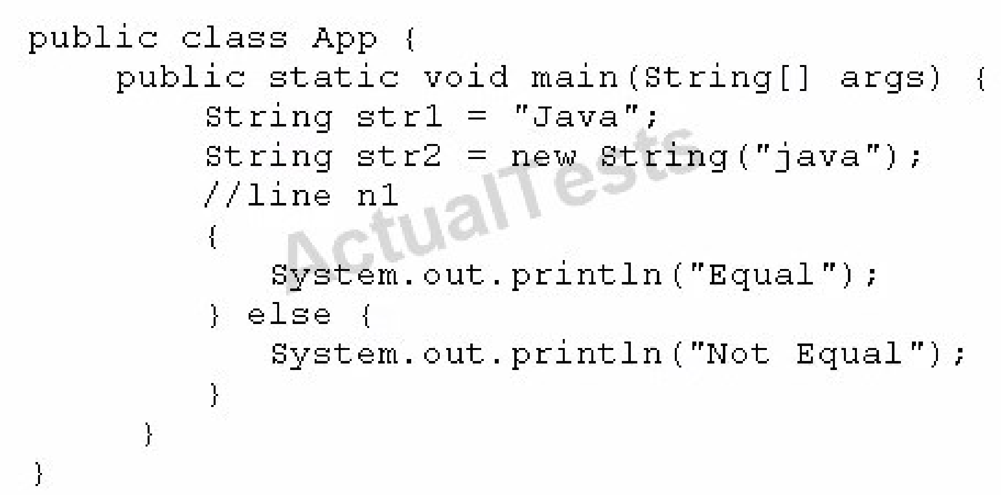
**B.** true false

**C.** false false

**D.** false true

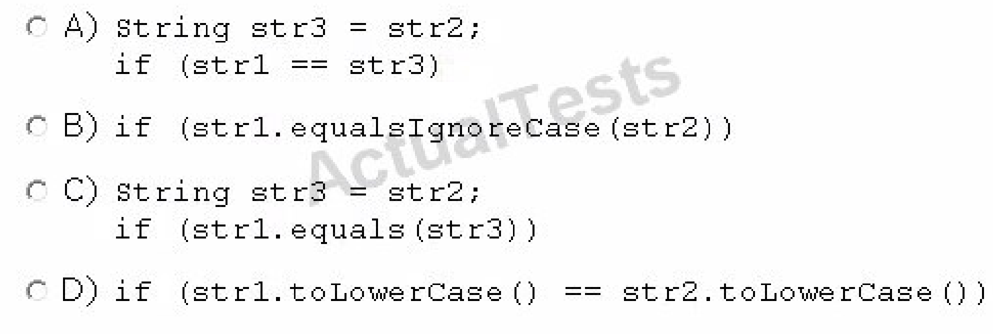
**QUESTION NO: 50**

Given the code fragment:



```
public class App {
    public static void main(String[] args) {
        String str1 = "Java";
        String str2 = new String("java");
        //line n1
        {
           System.out.println("Equal");
        } else {
           System.out.println("Not Equal");
        }
     }
}
```

Which code fragment, when inserted at line n1, enables the App class to print Equal?



```
○ A) String str3 = str2;
     if (str1 == str3)

○ B) if (str1.equalsIgnoreCase(str2))

○ C) String str3 = str2;
     if (str1.equals(str3))

○ D) if (str1.toLowerCase() == str2.toLowerCase())
```

**A.** Option A

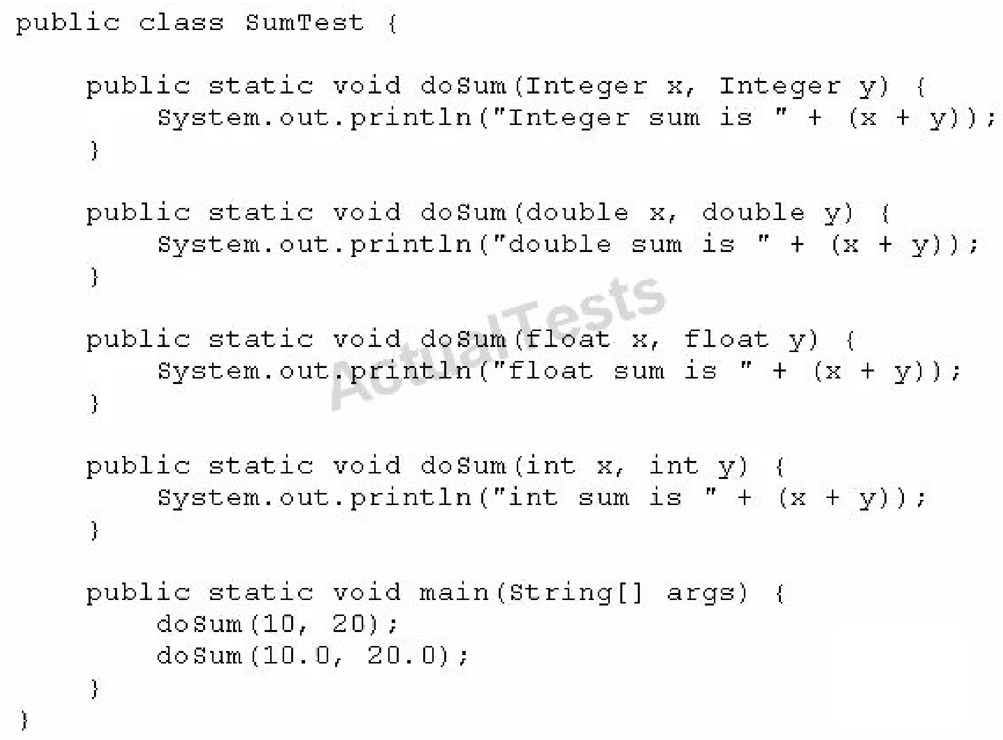
**B.** Option B

**C.** Option C

**D.** Option D

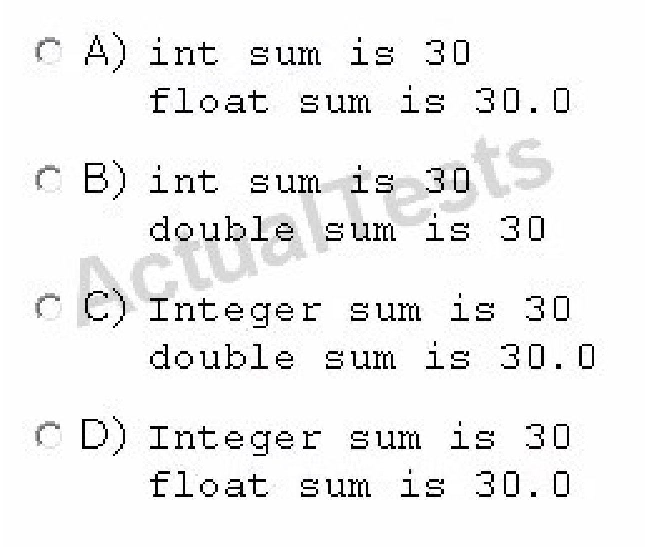
## QUESTION NO: 51

Given:



```
public class SumTest {

    public static void doSum(Integer x, Integer y) {
        System.out.println("Integer sum is " + (x + y));
    }

    public static void doSum(double x, double y) {
        System.out.println("double sum is " + (x + y));
    }

    public static void doSum(float x, float y) {
        System.out.println("float sum is " + (x + y));
    }

    public static void doSum(int x, int y) {
        System.out.println("int sum is " + (x + y));
    }

    public static void main(String[] args) {
        doSum(10, 20);
        doSum(10.0, 20.0);
    }
}
```

What is the result?



```
○ A) int sum is 30
     float sum is 30.0

○ B) int sum is 30
     double sum is 30

○ C) Integer sum is 30
     double sum is 30.0

○ D) Integer sum is 30
     float sum is 30.0
```

**A.** Option A

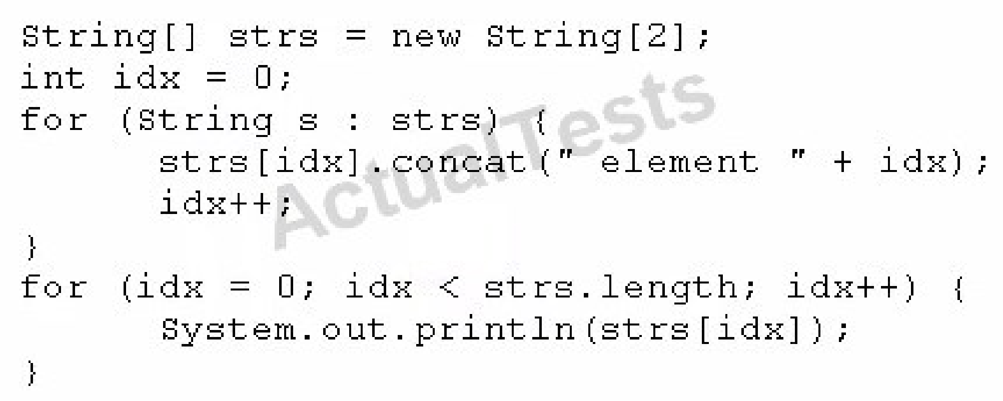
**B.** Option B

**C.** Option C

**D.** Option D

**QUESTION NO: 52**

Given the code fragment:



```
String[] strs = new String[2];
int idx = 0;
for (String s : strs) {
       strs[idx].concat(" element " + idx);
       idx++;
}
for (idx = 0; idx < strs.length; idx++) {
       System.out.println(strs[idx]);
}
```

What is the result?

**A.**
Element 0
Element 1

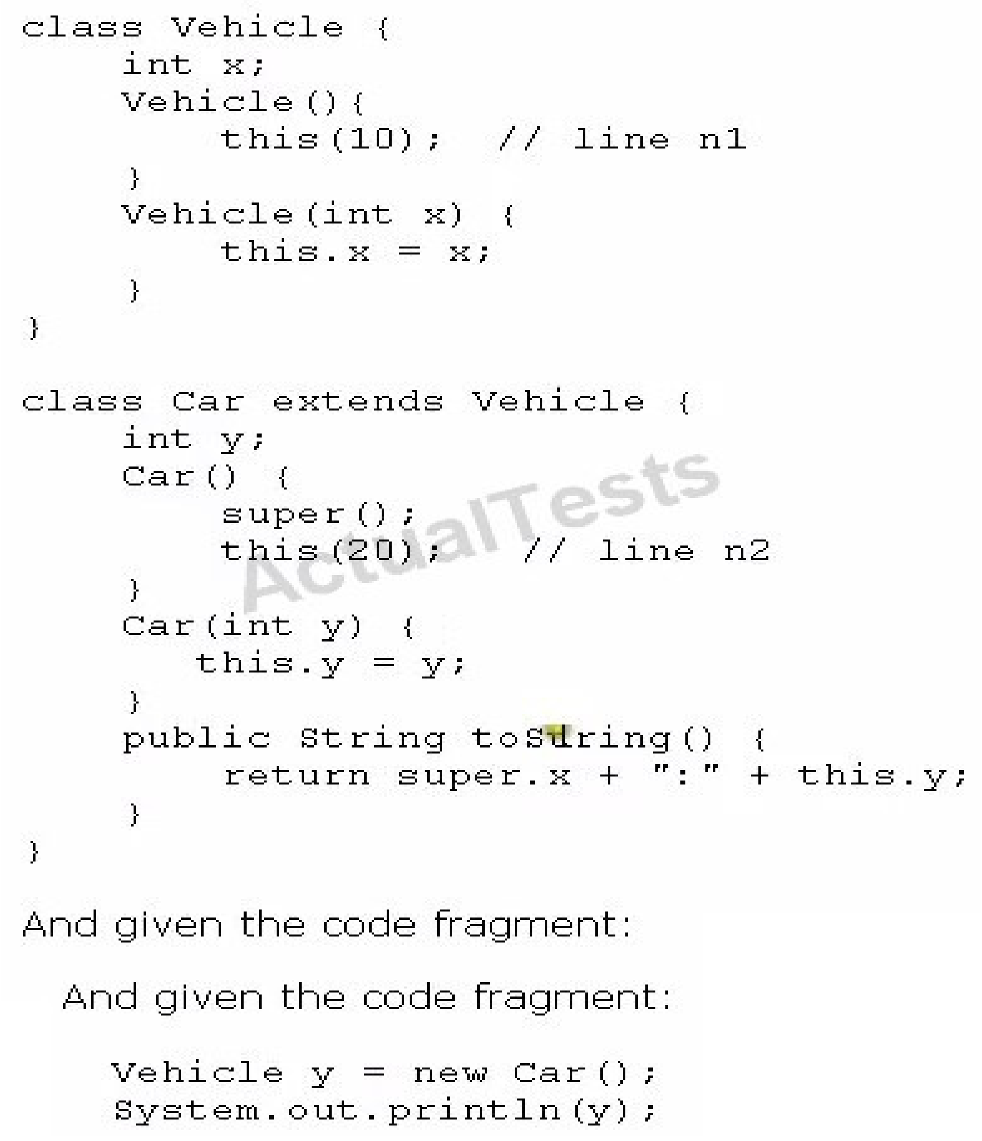
**B.**
Null element 0
Null element 1

**C.**
Null
Null

**D.**
A NullPointerException is thrown at runtime.

**QUESTION NO: 53**

Given:



```
class Vehicle {
     int x;
     Vehicle(){
          this(10);  // line n1
     }
     Vehicle(int x) {
          this.x = x;
     }
}

class Car extends Vehicle {
     int y;
     Car() {
          super();
          this(20);   // line n2
     }
     Car(int y) {
        this.y = y;
     }
     public String toString() {
          return super.x + ":" + this.y;
     }
}
```

And given the code fragment:

And given the code fragment:

```
Vehicle y = new Car();
System.out.println(y);
```

What is the result?

**A.** 10:20

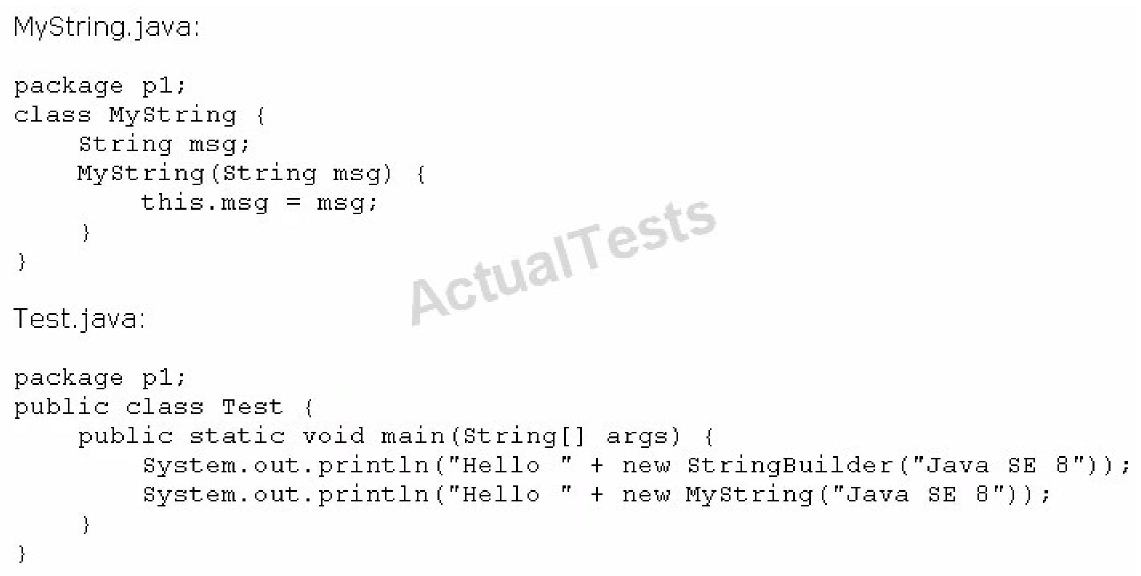
**B.** 0:20

**C.** Compilation fails at line n1

**D.** Compilation fails at line n2

**QUESTION NO: 54**

Given the definitions of the MyString class and the Test class:



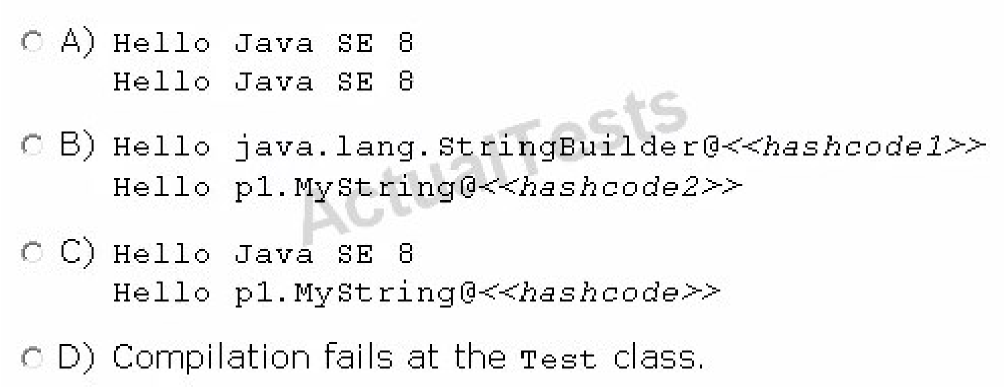
MyString.java:

```
package p1;
class MyString {
    String msg;
    MyString(String msg) {
        this.msg = msg;
    }
}
```

Test.java:

```
package p1;
public class Test {
    public static void main(String[] args) {
        System.out.println("Hello " + new StringBuilder("Java SE 8"));
        System.out.println("Hello " + new MyString("Java SE 8"));
    }
}
```

What is the result?



○ A) `Hello Java SE 8`
`Hello Java SE 8`

○ B) `Hello java.lang.StringBuilder@`*`<<hashcode1>>`*
`Hello p1.MyString@`*`<<hashcode2>>`*

○ C) `Hello Java SE 8`
`Hello p1.MyString@`*`<<hashcode>>`*

○ D) Compilation fails at the `Test` class.

**A.** Option A

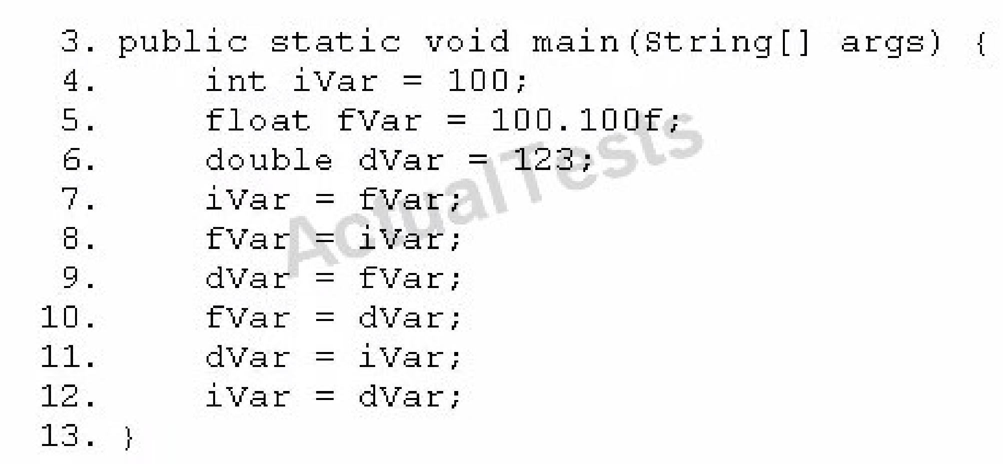
**B.** Option B

**C.** Option C

**D.** Option D

**QUESTION NO: 55**

Given the code fragment:



```
 3. public static void main(String[] args) {
 4.     int iVar = 100;
 5.     float fVar = 100.100f;
 6.     double dVar = 123;
 7.     iVar = fVar;
 8.     fVar = iVar;
 9.     dVar = fVar;
10.     fVar = dVar;
11.     dVar = iVar;
12.     iVar = dVar;
13. }
```

Which three lines fail to compile?

**A.** Line 7

**B.** Line 8

**C.** Line 9

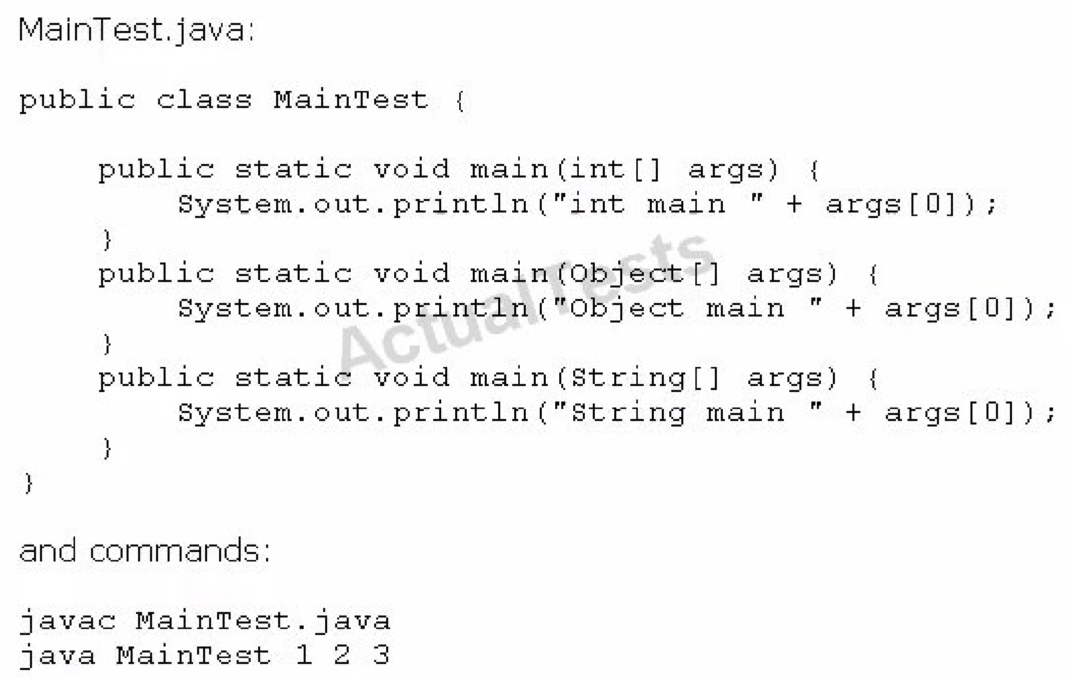
**D.** Line 10

**E.** Line 11

**F.** Line 12

**QUESTION NO: 56**

Given:



MainTest.java:

```
public class MainTest {

    public static void main(int[] args) {
        System.out.println("int main " + args[0]);
    }
    public static void main(Object[] args) {
        System.out.println("Object main " + args[0]);
    }
    public static void main(String[] args) {
        System.out.println("String main " + args[0]);
    }
}
```

and commands:

```
javac MainTest.java
java MainTest 1 2 3
```

What is the result?

**A.** int main 1

**B.** Object main 1

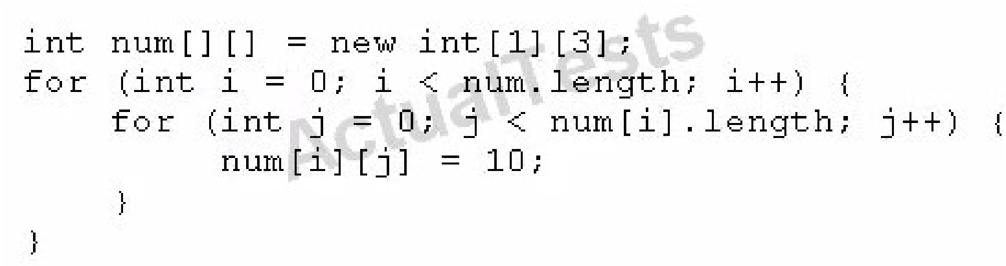
**C.** String main 1

**D.** Compilation fails

**E.** An exception is thrown at runtime

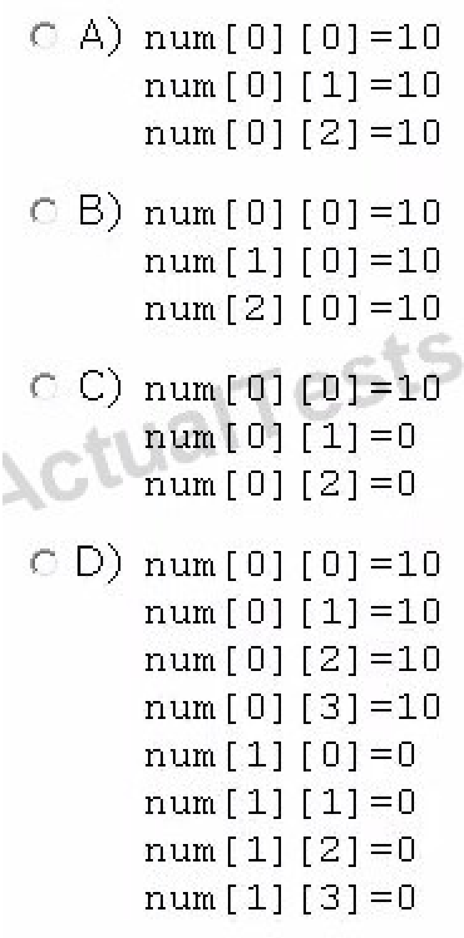
**QUESTION NO: 57**

Given the code fragment:



```
int num[][] = new int[1][3];
for (int i = 0; i < num.length; i++) {
    for (int j = 0; j < num[i].length; j++) {
         num[i][j] = 10;
    }
}
```

Which option represents the state of the num array after successful completion of the outer loop?



```
○ A) num[0][0]=10
     num[0][1]=10
     num[0][2]=10

○ B) num[0][0]=10
     num[1][0]=10
     num[2][0]=10

○ C) num[0][0]=10
     num[0][1]=0
     num[0][2]=0

○ D) num[0][0]=10
     num[0][1]=10
     num[0][2]=10
     num[0][3]=10
     num[1][0]=0
     num[1][1]=0
     num[1][2]=0
     num[1][3]=0
```

**A.** Option A

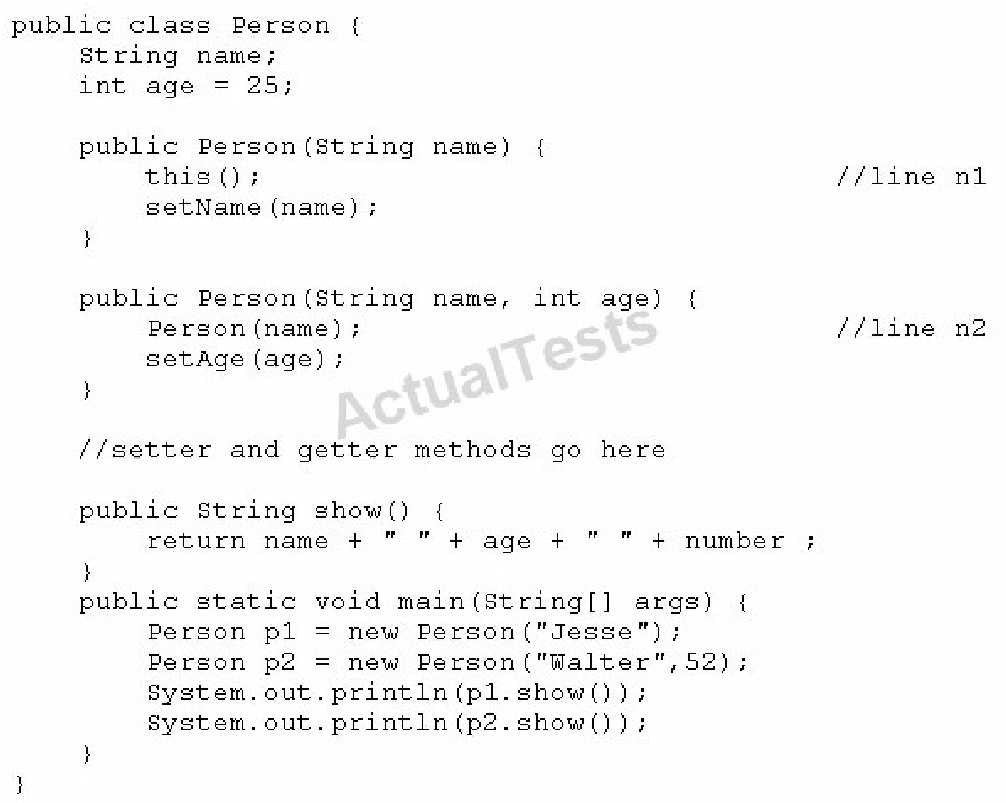
**B.** Option B

**C.** Option C

**D.** Option D

**QUESTION NO: 58**

Given the code fragment:



```
public class Person {
    String name;
    int age = 25;

    public Person(String name) {
        this();                                  //line n1
        setName(name);
    }

    public Person(String name, int age) {
        Person(name);                            //line n2
        setAge(age);
    }

    //setter and getter methods go here

    public String show() {
        return name + " " + age + " " + number ;
    }
    public static void main(String[] args) {
        Person p1 = new Person("Jesse");
        Person p2 = new Person("Walter",52);
        System.out.println(p1.show());
        System.out.println(p2.show());
    }
}
```

What is the result?

**A.**
Jesse 25
Walter 52

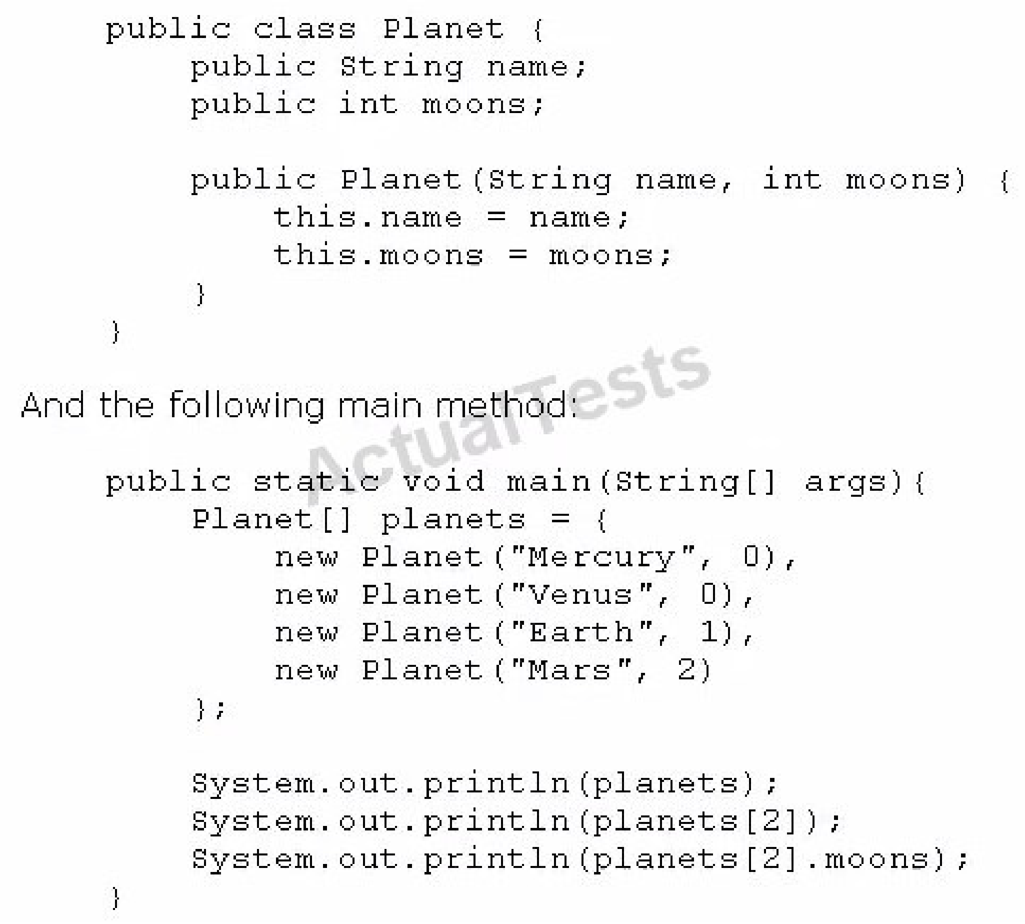
**B.** Compilation fails only at line n1

**C.** Compilation fails only at line n2

**D.** Compilation fails at both line n1 and line n2

**QUESTION NO: 59**

Given the following code for a Planet object:

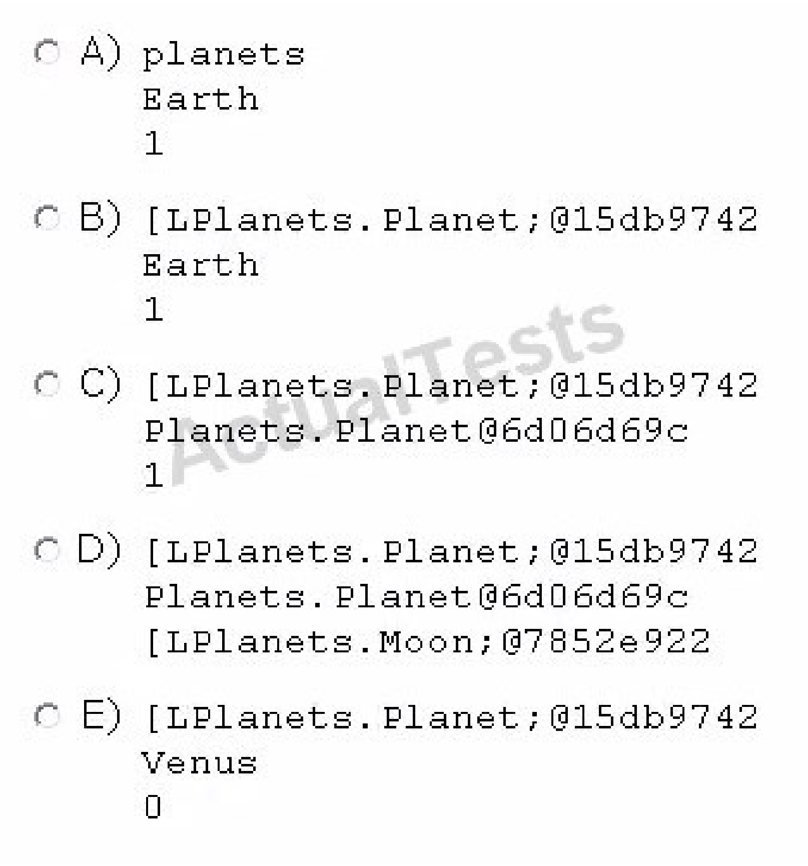


```
public class Planet {
    public String name;
    public int moons;

    public Planet(String name, int moons) {
        this.name = name;
        this.moons = moons;
    }
}
```

And the following main method:

```
public static void main(String[] args){
    Planet[] planets = {
        new Planet("Mercury", 0),
        new Planet("Venus", 0),
        new Planet("Earth", 1),
        new Planet("Mars", 2)
    };

    System.out.println(planets);
    System.out.println(planets[2]);
    System.out.println(planets[2].moons);
}
```

What is the output?



```
○ A) planets
     Earth
     1

○ B) [LPlanets.Planet;@15db9742
     Earth
     1

○ C) [LPlanets.Planet;@15db9742
     Planets.Planet@6d06d69c
     1

○ D) [LPlanets.Planet;@15db9742
     Planets.Planet@6d06d69c
     [LPlanets.Moon;@7852e922

○ E) [LPlanets.Planet;@15db9742
     Venus
     0
```

**A.** Option A

**B.** Option B

**C.** Option C

**D.** Option D

**E.** Option E

**QUESTION NO: 60**

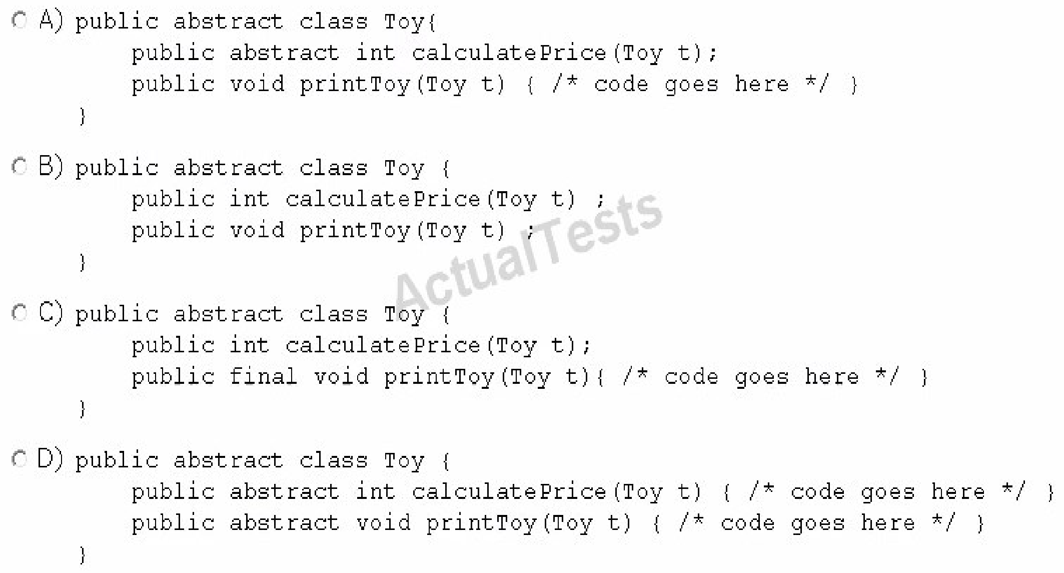
You are asked to develop a program for a shopping application, and you are given the following information:

The application must contain the classes Toy, EduToy, and ConsToy. The Toy class is the superclass of the other two classes.

The int caicuiatePrice (Toy t) method calculates the price of a toy.

The void printToy (Toy t) method prints the details of a toy.

Which definition of the Toy class adds a valid layer of abstraction to the class hierarchy?



```
○ A) public abstract class Toy{
        public abstract int calculatePrice(Toy t);
        public void printToy(Toy t) { /* code goes here */ }
     }

○ B) public abstract class Toy {
        public int calculatePrice(Toy t) ;
        public void printToy(Toy t) ;
     }

○ C) public abstract class Toy {
        public int calculatePrice(Toy t);
        public final void printToy(Toy t){ /* code goes here */ }
     }

○ D) public abstract class Toy {
        public abstract int calculatePrice(Toy t) { /* code goes here */ }
        public abstract void printToy(Toy t) { /* code goes here */ }
     }
```

**A.** Option A

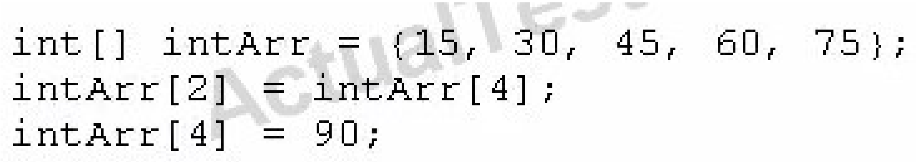
**B.** Option B

**C.** Option C

**D.** Option D

**QUESTION NO: 61**

Given the following code:



```
int[] intArr = {15, 30, 45, 60, 75};
intArr[2] = intArr[4];
intArr[4] = 90;
```

What are the values of each element in intArr after this code has executed?

**A.** 15, 60, 45, 90, 75

**B.** 15, 90, 45, 90, 75

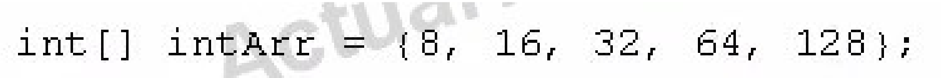
**C.** 15, 30, 75, 60, 90

**D.** 15, 30, 90, 60, 90

**E.** 15, 4, 45, 60, 90

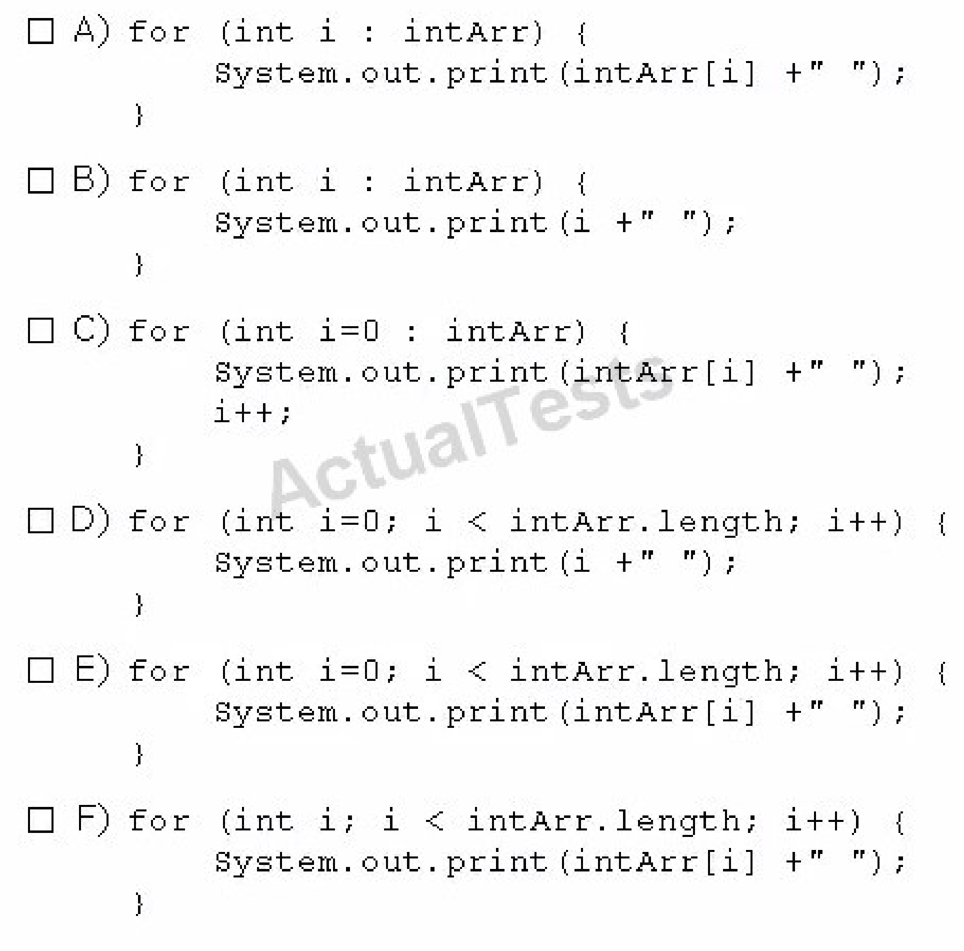
**QUESTION NO: 62**

Given the following array:



```
int[] intArr = {8, 16, 32, 64, 128};
```

Which two code fragments, independently, print each element in this array?



```
□ A) for (int i : intArr) {
          System.out.print(intArr[i] +" ");
     }

□ B) for (int i : intArr) {
          System.out.print(i +" ");
     }

□ C) for (int i=0 : intArr) {
          System.out.print(intArr[i] +" ");
          i++;
     }

□ D) for (int i=0; i < intArr.length; i++) {
          System.out.print(i +" ");
     }

□ E) for (int i=0; i < intArr.length; i++) {
          System.out.print(intArr[i] +" ");
     }

□ F) for (int i; i < intArr.length; i++) {
          System.out.print(intArr[i] +" ");
     }
```

**A.** Option A

**B.** Option B

**C.** Option C

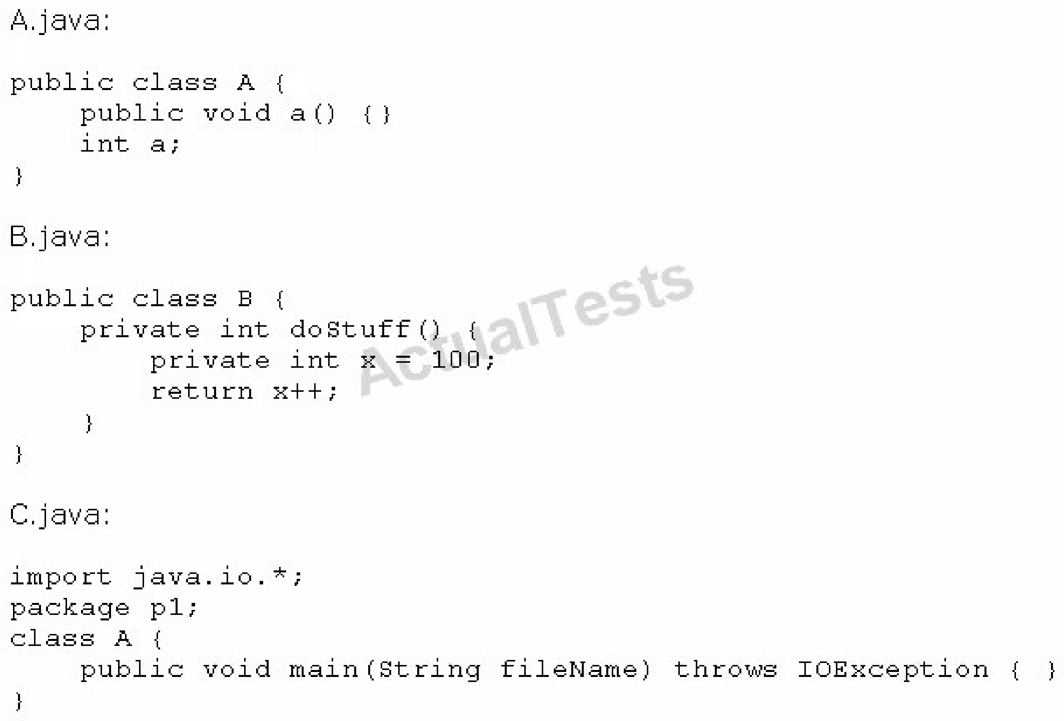
**D.** Option D

**E.** Option E

**F.** Option F

**QUESTION NO: 63**

Given the content of three files:



```
A.java:

public class A {
    public void a() {}
    int a;
}

B.java:

public class B {
    private int doStuff() {
        private int x = 100;
        return x++;
    }
}

C.java:

import java.io.*;
package p1;
class A {
    public void main(String fileName) throws IOException { }
}
```

Which statement is true?

**A.** Only the A.Java file compiles successfully.

**B.** Only the B.java file compiles successfully.

**C.** Only the C.java file compiles successfully.

**D.** The A.Java and B.java files compile successfully.

**E.** The B.java and C.java files compile successfully.

**F.** The A.Java and C.java files compile successfully.

**QUESTION NO: 64**

Given the code fragment:

```
int[] array = {1, 2, 3, 4, 5};
```

And given the requirements:

Process all the elements of the array in the order of entry.

Process all the elements of the array in the reverse order of entry.

Process alternating elements of the array in the order of entry.

Which two statements are true?

**A.** Requirements 1, 2, and 3 can be implemented by using the enhanced for loop.

**B.** Requirements 1, 2, and 3 can be implemented by using the standard for loop.

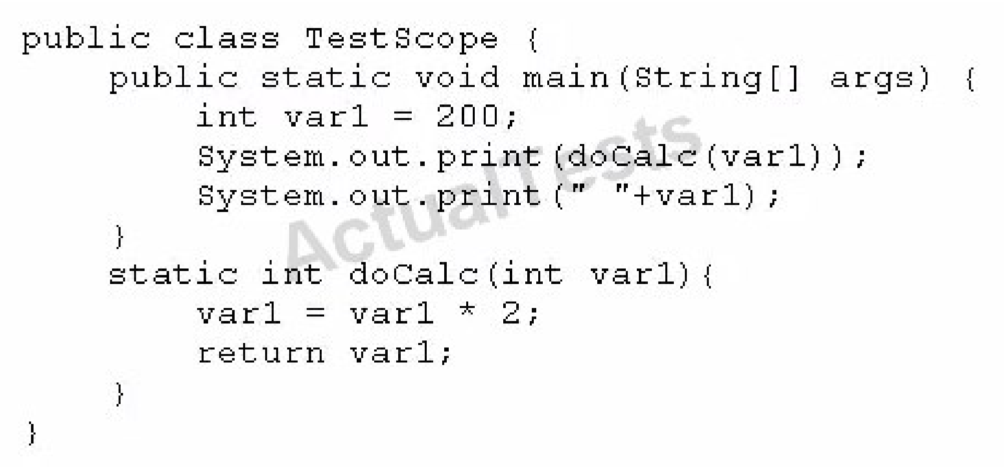
**C.** Requirements 2 and 3 CANNOT be implemented by using the standard for loop.

**D.** Requirement 1 can be implemented by using the enhanced for loop.

**E.** Requirement 3 CANNOT be implemented by using either the enhanced for loop or the standard for loop.

**QUESTION NO: 65**

Given:



```
public class TestScope {
    public static void main(String[] args) {
        int var1 = 200;
        System.out.print(doCalc(var1));
        System.out.print(" "+var1);
    }
    static int doCalc(int var1){
        var1 = var1 * 2;
        return var1;
    }
}
```

What is the result?

**A.** 400 200

**B.** 200 200

**C.** 400 400

**D.** Compilation fails.

**QUESTION NO: 67**

Which statement is true about Java byte code?

**A.** It can run on any platform.

**B.** It can run on any platform only if it was compiled for that platform.

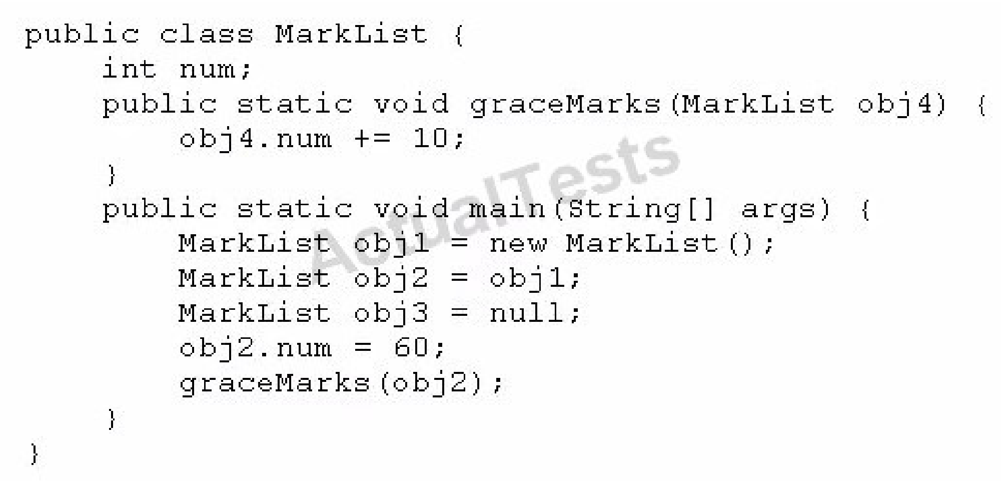
**C.** It can run on any platform that has the Java Runtime Environment.

**D.** It can run on any platform that has a Java compiler.

**E.** It can run on any platform only if that platform has both the Java Runtime Environment and a Java compiler.

**QUESTION NO: 68**

Given:



```
public class MarkList {
    int num;
    public static void graceMarks(MarkList obj4) {
        obj4.num += 10;
    }
    public static void main(String[] args) {
        MarkList obj1 = new MarkList();
        MarkList obj2 = obj1;
        MarkList obj3 = null;
        obj2.num = 60;
        graceMarks(obj2);
    }
}
```

How many MarkList instances are created in memory at runtime?

**A.** 1

**B.** 2

**C.** 3

**D.** 4