# Oracle

## Exam 1z0-803

### Java SE 7 Programmer I

**Verson: Demo**

**[ Total Questions:  10 ]**

## Question No : 1

Given the code fragment:

```
int b = 3;
if ( !(b > 3)) {
System.out.println("square ");
}{
System.out.println("circle ");
}
System.out.println("...");
```

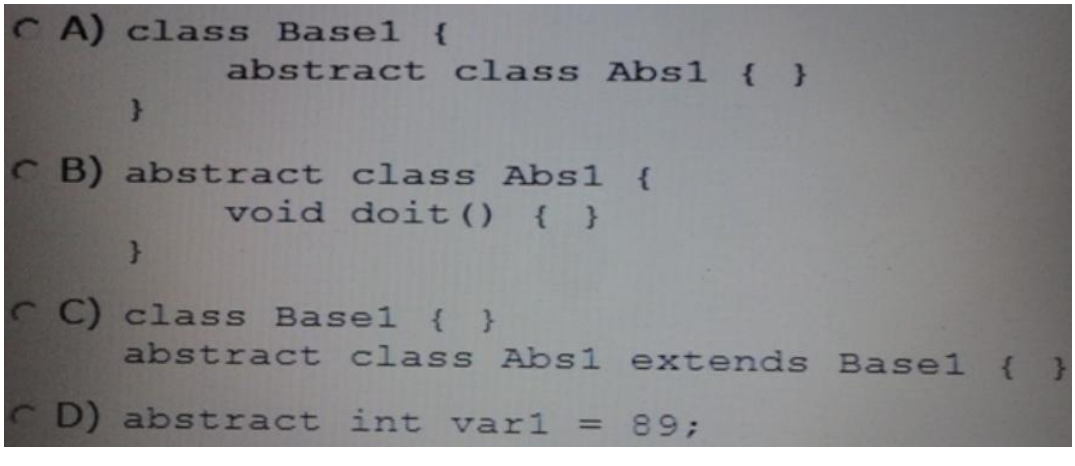
What is the result?

**A.** square...
**B.** circle...
**C.** squarecircle...
**D.** Compilation fails.

**Answer: C**

## Question No : 2

Which code fragment is illegal?



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

**Answer: D**

**Explanation:**

The abstract keyword cannot be used to declare an int variable.

The abstract keyword is used to declare a class or method to beabstract[3]. An abstract method has no implementation; all classes containing abstract methods must themselves be abstract, although not all abstract classes have abstract methods.

**Question No : 3**

Given the code fragment:

```
public static void main(String[] args) {
    ArrayList<String> list = new ArrayList<>();

    list.add("SE");
    list.add("EE");
    list.add("ME");
    list.add("SE");
    list.add("EE");

    list.remove("SE");

    System.out.print("Values are : " + list);
}
```

What is the result?

**A.** Values are : [EE, ME]
**B.** Values are : [EE, EE, ME]
**C.** Values are : [EE, ME, EE]
**D.** Values are : [SE, EE, ME, EE]
**E.** Values are : [EE, ME, SE, EE]

**Answer: E**

**Question No : 4**

Which two are valid declarations of a two-dimensional array?

**A.** int [] [] array2D;
**B.** int [2] [2] array2D;
**C.** int array2D [];
**D.** int [] array2D [];
**E.** int [] [] array2D [];

**Answer: A,D**

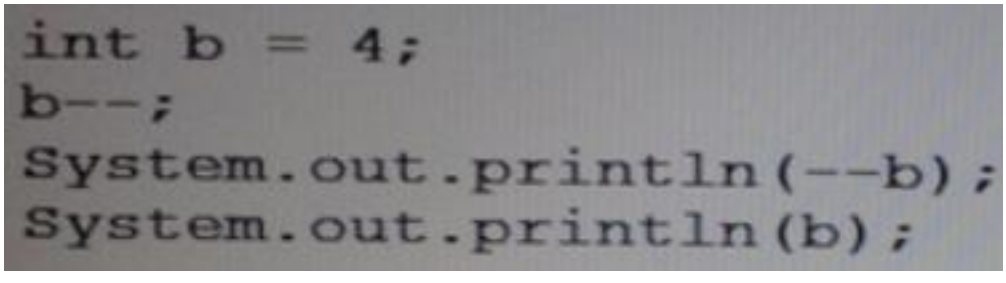
**Explanation:**

int[][] array2D; is the standard convention to declare a 2-dimensional integer array.

int[] array2D[]; works as well, but it is not recommended.

**Question No : 5**

Given the code fragment:




What is the result?

**A.** 2
2
**B.** 1
2
**C.** 3
2
**D.** 3
3

**Answer: A**

**Explanation:**
Variable b is set to 4.
Variableb is decreased to 3.
Variable b is decreased to 2 and then printed. Output: 2
Variable b is printed. Output: 2

## Question No : 6

Given:

```
interface Pet { }

class Dog implements Pet { }

public class Beagle extends Dog{ }
```

Which three are valid?

**A.** Pet a = new Dog();
**B.** Pet b = new Pet();
**C.** Dog f = new Pet();
**D.** Dog d = new Beagle();
**E.** Pet e = new Beagle();
**F.** Beagle c = new Dog();

**Answer: A,D,E**

**Explanation:**
Incorrect:
Not B, not C: Pet is abstact, cannot be instantiated.
Not F: incompatible type. Required Beagle, found Dog.

## Question No : 7

Given:

```
package p1;

public class Test {

static double dvalue;

static Test ref;

public static void main(String[] args) {

System.out.println(ref);

System.out.println(dvalue);

}

}
```

What is the result?

**A.** p1.Test.class
0.0
**B.** <the summary address refrenced by ref>
0.000000
**C.** Null
0.0
**D.** Compilation fails
**E.** A NullPointerException is thrown at runtime

**Answer: C**

**Question No : 8**

Given:

```
class Cake {

int model;

String flavor;

Cake() {

model = 0;

flavor = "Unknown";
```

```
}

}

public class Test {

public static void main(String[] args) {

Cake c = new Cake();

bake1(c);

System.out.println(c.model + " " + c.flavor);

bake2(c);

System.out.println(c.model + " " + c.flavor);

}

public static Cake bake1(Cake c) {

c.flavor = "Strawberry";

c.model = 1200;

return c;

}

public static void bake2(Cake c) {

c.flavor = "Chocolate";

c.model = 1230;

return;

}

}
```

What is the result?

**A.** 0 unknown
0 unknown
**B.** 1200 Strawberry
1200 Strawberry
**C.** 1200 Strawberry
1230 Chocolate

**D.** Compilation fails

**Answer: C**

**Explanation:** 1200 Strawberry
1230 Chocolate

## Question No : 9

Which two actions will improve the encapsulation of a class?

**A.** Changing the access modifier of a field from public to private
**B.** Removing the public modifier from a class declaration
**C.** Changing the return type of a method to void
**D.** Returning a copy of the contents of an array or ArrayList instead of a direct reference

**Answer: A,D**
Reference:

http://www.tutorialspoint.com/java/java_access_modifiers.htm

## Question No : 10

View the Exhibit.

```
public class Hat {

public int ID =0;

public String name = "hat";

public String size = "One Size Fit All";

public String color="";

public String getName() { return name; }

public void setName(String name) {
```

```
this.name = name;

}

}
```

**Given**

```
public class TestHat {

public static void main(String[] args) {

Hat blackCowboyHat = new Hat();

}

}
```

Which statement sets the name of the Hat instance?

**A.** blackCowboyHat.setName = "Cowboy Hat";
**B.** setName("Cowboy Hat");
**C.** Hat.setName("Cowboy Hat");
**D.** blackCowboyHat.setName("Cowboy Hat");

**Answer: D**