Vendor: Oracle

Exam Code: 1Z0-808

Exam Name: Java SE 8 Programmer I

Question 31—Question 40

QUESTION 31
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

Answer: BE

QUESTION 32
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.

Answer: A
Explanation:
Class B doesn't compile because we can't use access modifiers (private) inside methods.
Class C doesn't compile because if the class is part of a package (p1), the package statement must be the first line in the source code file, before any import statements (java.io.*) that may be present.

QUESTION 33
Given the code fragment:

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

And given the requirements:

```
1. Process all the elements of the array in the order of entry.
2. Process all the elements of the array in the reverse order of
entry.
3. 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.

Answer: BD

QUESTION 34
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.

Answer: A

QUESTION 35
Given the following class declarations:

- public abstract class Animal
- public interface Hunter
- public class Cat extends Animal implements Hunter
- public class Tiger extends Cat

Which answer fails to compile?

A)
```
ArrayList<Animal> myList = new ArrayList<>();
myList.add(new Tiger());
```

B)
```
ArrayList<Hunter> myList = new ArrayList<>();
myList.add(new Cat());
```

C)
```
ArrayList<Hunter> myList = new ArrayList<>();
myList.add(new Tiger());
```

D)
```
ArrayList<Tiger> myList = new ArrayList<>();
myList.add(new Cat());
```

E)
```
ArrayList<Animal> myList = new ArrayLIst<>();
myList.add(new Cat());
```

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

Answer: D

Explanation:

Cat cannot be converted to Tiger.

One Tiger is a Cat but one Cat isn't a Tiger.

QUESTION 36

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.

Answer: C

Explanation:

We are talking about byte code so the Java program has been compiled.

The question ask for what we need to run the byte code.

https://www.java.com/en/download/faq/whatis_java.xml

http://www.researchgate.net/post/Run_Java_Application_Without_Installing_Java_Runtime

QUESTION 37

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

Answer: A
Explanation:
Only the statement "MarList obj1 = new MarkList();" creates an instance of MarkList.

QUESTION 38
Given:

```
public class Triangle {
    static double area;
    int b = 2, h = 3;
    public static void main(String[] args) {
        double p, b, h;          //line n1
        if (area == 0) {
            b = 3;
            h = 4;
            p = 0.5;
        }
        area = p * b * h;        //line n2
        System.out.println("Area is " + area);
    }
}
```

What is the result?

A. Area is 6.0
B. Area is 3.0
C. Compilation fails at line n1
D. Compilation fails at line n2.

Answer: D

QUESTION 39
Given the code fragment:

```
public class Test {
    public static void main(String[] args) {
        //line n1
        switch (x) {
            case 1:
                System.out.println("One");
                break;
            case 2:
                System.out.println("Two");
                break;
        }
    }
}
```

Which three code fragments can be independently inserted at line nl to enable the code to print one?

A. Byte x = 1;
B. short x = 1;
C. String x = "1";
D. Long x = 1;
E. Double x = 1;
F. Integer x = new Integer ("1");

Answer: ABF

QUESTION 40
Given:

```
public class App {

    public static void main(String[] args) {
        Boolean[] bool = new Boolean[2];

        bool[0] = new Boolean(Boolean.parseBoolean("true"));
        bool[1] = new Boolean(null);

        System.out.println(bool[0] + " " + bool[1]);
    }
}
```

What is the result?

A. True false
B. True null
C. Compilation fails
D. A NullPointerException is thrown at runtime

Answer: A
Explanation:
With the statement “bool[1] = new Boolean(null);” we are creating a wrapped Boolean object with value null.
Java evaluates it to false since it cannot evaluate to true;