➢ **Vendor: Oracle**

➢ **Exam Code: 1Z0-808**

➢ **Exam Name: Java SE 8 Programmer I**

➢ **Question 21 -- Question 40**

Visit PassLeader and Download Full Version 1Z0-808 Exam Dumps

QUESTION 21
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.

Answer: D

QUESTION 22
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

Answer: D
Explanation:
this() and super() can't be used in the same constructor
Here is a good reference for the question
http://stackoverflow.com/questions/10381244/why-cant-this-and-super-both-be-used-together-in-a-constructor

QUESTION 23
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?

```
C A) Hello Java SE 8
      Hello Java SE 8

C B) Hello java.lang.StringBuilder@<<hashcode1>>
      Hello p1.MyString@<<hashcode2>>

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

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

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

Answer: C

QUESTION 24
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

Answer: ADF
Explanation:
See "Assignment Compatibility" at
http://docstore.mik.ua/orelly/java/langref/ch04_13.htm

QUESTION 25
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

Answer: C
Explanation:
All methods have the same name but different signature since the parameters are different. There is no problem with that.
JVM will call the method with signature "public static void main(String[] args)"
https://docs.oracle.com/javase/tutorial/java/javaOO/methods.html

QUESTION 26
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

Answer: A
Explanation:
At first look we can exclude option D because the number of elements in the array is 3, the result of multiplying the two array dimensions 1 x 3.
We can run the code

```java
public class Main {
public static void main(String[] args) {
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;
System.out.println("num[" + i + "][" + j + "]= " + num[i][j]);
}
}
}
}
```

the output is
num[0][0]= 10
num[0][1]= 10
num[0][2]= 10

QUESTION 27
Given the code fragment:

```java
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

Answer: D
Explanation:
At line n1, Person class hasn't any constructor without arguments.
At line n2, there isn't any method Person. If we want to call the constructor that should be "this(name)".

QUESTION 28
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

Answer: C

QUESTION 29
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

Answer: B
Explanation:
https://docs.oracle.com/javase/tutorial/java/IandI/abstract.html

QUESTION 30
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

Answer: C

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: DE

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.

Visit PassLeader and Download Full Version 1Z0-808 Exam Dumps