HackerRank For Work

AED-Fall2018-Quiz4 (09:00-... 20 minutes

Question - 1

SCORE: **10 points**

Java: Find the Output

Java | General Programming | Exception Handling | Language-Specific | Core CS | Basic Programming

Consider the following Java code snippet:

```
public int divide(int a, int b) {
    int c = -1;

    try {
        c = a / b;
    }
    catch (Exception e) {
        System.err.print("Exception ");
    }
    finally {
        System.err.println("Finally ");
    }

    return c;
}
```

What will our code *print* when we call *divide(4, 0)*?

- (●) Exception Finally
- ( ) Finally Exception
- ( ) Exception
- ( ) Finally
- ( ) No output
- ( ) -1

Question - 2

SCORE: **10 points**

Application Development | Java

What is the correct declaration of an abstract method that is intended to be public?

- (●) public abstract void add();
- ( ) public abstract void add() {}
- ( ) public abstract add();
- ( ) public virtual add();

Question - 3

SCORE: **10 points**

In order to get all even number between the range 0 to 10 within given list, which sentence should be used for if statement?

```
private static List<Integer>
select(ArrayList<Integer> list) {
    ArrayList<Integer> result = new ArrayList<>
();
    // select even number in range [0 - 10]
    for (Integer i: list) {
        if (_______?__________) {
            result.add(i);
        }
    }
    return result;
}
```

- i % 2 == 0 || i >= 0 || i <= 10
- i % 2 == 0 || (i >= 0 && i <= 10)
- i % 2 == 0 && (i >= 0 || i <= 10)
- **i % 2 == 0 && i >= 0 && i <= 10**

Question - 4

SCORE: **10 points**

Choose the correct output according to following code:

```
import java.util.*;
public class Demo {
    public static void main(String []args) {
        ArrayList<String> list = new ArrayList<>
();
        list.add("2");
        list.add("0");
        list.add("1");
        list.add("8");
        List<String> newList = next(list);
        list.add("9");
        for (String i: newList) {
            System.out.print(i + " ");
        }
    }

    private static List<String> next(List<String>
list) {
        list.remove(list.size()-1);
        return list;
    }
}
```

- "2" "0" "1"
- "2" "0" "1" "9"
- 2 0 1
- **2 0 1 9**

Question - 5

SCORE: **10 points**

static

Choose the correct output according to following code:

```
public class Q {
        static int c = 0;
        public static void main(String[] args) {
            Q q1 = c();
            Q q2 = c(q1);
            Q q3 = c(q2);
            Q q4 = c(q3);
        }
        private Q() {
            System.out.print("c = " + c + " ");
        }
        static Q c() {
            return c++ <= 0 ? new Q() : null;
        }
        static Q c(Q w) {
            return w.c++ == 1 ? new Q() : null;
        }
}
```

- (●) c = 1 c = 2
- ( ) c = 1 c = 2 c = 3
- ( ) c = 1 c = 2 c = 3 c = 4
- ( ) Compilation Error
- ( ) NullPointerException

Question - 6

SCORE: **10 points**

Which of the following is incorrect?

Which of the following is incorrect?

X: `super` keyword and `this` keyword can not be used in a `static` method.
Y: `static` variables are shared by all objects and initialized when the `class` is first loaded.

- ( ) Only X
- ( ) Only Y
- (●) Both are correct
- ( ) Both are incorrect

Question - 7

SCORE: **10 points**

Say that there are three classes: *Computer*, *AppleComputer*, and *IBMComputer*. What are the likely relationships between these classes?

- (●) Computer is the superclass, AppleComputer and IBMComputer are subclasses of Computer.
- ( ) Computer, AppleComputer and IBMComputer are sibling classes.
- ( ) IBMComputer is the superclass, AppleComputer and Computer are subclasses of IBMComputer.
- ( ) Computer is a superclass, AppleComputer is a subclasses of Computer, and IBMComputer is a sublclas of AppleComputer.

Question - 8

SCORE: **10 points**

What is the output of this program?

```
public class Demo {
    public static void main(String args[]) {
        B obj = new B();
        obj.i=1;
        obj.j=2;
        obj.display();
    }
}

class A {
    int i;
}
class B extends A {
    int j;
    void display() {
        super.i = j + 1;
        System.out.println(j + " " + i);
    }
}
```

- ( ) 3 3
- ( ) 2 2
- ( ) 3 2
- (●) 2 3
- ( ) Compilation Error