**Question Bank**

Subject: Object Oriented Programming using Java

Session: July-December 2019

Subject Code: 18CSI302

Unit 3: String Handling and Exception Handling

1. Suppose that s1, s2, s3, and s4 are four strings, given as follows:
   String s1 = "Welcome to Java";
   String s2 = s1;
   String s3 = new String("Welcome to Java");
   String s4 = "Welcome to Java";
   **What are the results of the following expressions?**
   a. s1 == s2
   b. s2 == s3
   c. s1.equals(s2)
   d. s2.equals(s3)
   e. s1.compareTo(s2)
   f. s2.compareTo(s3)
   g. s1 == s4
   h. s1.charAt(0)
   i. s1.indexOf('j')
   j. s1.indexOf("to")
   k. s1.lastIndexOf('a')
   l. s1.lastIndexOf("o", 15)
   m. s1.length()
   n. s1.substring(**5**)
   o. s1.substring(**5**, **11**)
   p. s1.startsWith(**"Wel"**)
   q. s1.endsWith(**"Java"**)
   r. s1.toLowerCase()
   s. s1.toUpperCase()
   t. **"Welcome "**.trim()
   u. s1.replace(**'o'**, **'T'**)
   v. s1.replaceAll(**"o"**, **"T"**)
   w. s1.replaceFirst(**"o"**, **"T"**)
   x. s1.toCharArray()

2. To create the string Welcome to Java, you may use a statement like this:
   String s = "Welcome to Java";
   or:
   String s = new String("Welcome to Java");
   Which one is better? Why?
3. Suppose that s1 and s2 are two strings. Which of the following statements or expressions are incorrect?

String s = new String("new string");
   String s3 = s1 + s2;
   String s3 = s1 - s2;
   s1 == s2;

s1 >= s2;
s1.compareTo(s2);
inti = s1.length();
charc = s1(0);
charc = s1.charAt(s1.length());

4. What is the printout of the following code?
   String s1 = "Welcome to Java";
   String s2 = s1.replace("o", "abc");
   System.out.println(s1);
   System.out.println(s2);
   Let s1 be " Welcome " and s2 be " welcome ". Write the code for the following statements:

(a). Check whether **s1** is equal to **s2** and assign the result to a Boolean variable **isEqual**.
(b). Check whether **s1** is equal to **s2**, ignoring case, and assign the result to a Boolean variable **isEqual**.
(c). Compare **s1** with **s2** and assign the result to an **int**variable **x**.
(d). Compare **s1** with **s2**, ignoring case, and assign the result to an **int**variable **x**.
(e). Check whether **s1** has the prefix **AAA** and assign the result to a Boolean variable **b**.
(f). Check whether **s1** has the suffix **AAA** and assign the result to a Boolean variable **b**.

5. Write a program that prompts the user to enter a string and reports whether the string is a palindrome.
6. Define Mutable and Immutable Strings in Java.
7. Write a program to reverse a string without using StringBuffer class.
8. Explain Exception handling mechanism used in java by giving suitable examples.
9. What is the advantage of using exception handling?
10. What does the JVM do when an exception occurs? How do you catch an exception?
11. What is the output of the following code?

```
public class Test {
public static void main(String[] args) {
 try{
        intvalue = 30;
 if(value <40)
        throw new Exception("value is too small");
}
catch(Exception ex) {
        System.out.println(ex.getMessage());
}
        System.out.println("Continue after the catch block");
}
}
```

What would be the printout if the line
intvalue = 30;
were changed to
intvalue = 50;

12. What is the purpose of declaring exceptions? How do you declare an exception, and where? Can you declare multiple exceptions in a method header?
13. What is a checked exception, and what is an unchecked exception?
14. Suppose that statement2 causes an exception in the following try-catch block:

```
 try{
```

```
        statement1;
        statement2;
        statement3;
    }
    catch(Exception1 ex1) {
    }
    catch(Exception2 ex2) {
    }
        statement4;
```

**Answer the following questions:**

- Will statement3 be executed?
- If the exception is not caught, will statement4 be executed?
- If the exception is caught in the catch block, will statement4 be executed?

15. What is displayed when the following program is run?

```
public class Test {
public static void main(String[] args) {
    try{
        int[] list = new int[10];
        System.out.println("list[10] is " + list[10]);
    }
    catch(ArithmeticException ex) {
        System.out.println("ArithmeticException");
    }
    catch(RuntimeException ex) {
        System.out.println("RuntimeException");
    }
    catch(Exception ex) {
        System.out.println("Exception");
    }
}}
```

16. What does the method **getMessage()** do?
17. What does the method **printStackTrace**do?
18. Does the presence of a **try-catch** block impose overhead when no exception occurs?
19. Suppose that statement2 causes an exception in the following statement:

```
    try {
        statement1;
        statement2;
        statement3;
    }
    catch (Exception1 ex1) {
    }
    finally {
        statement4;
    }
    statement5;
```

**Answer the following questions:**

- If no exception occurs, will statement4 be executed, and will statement5 be executed?

- If the exception is of type Exception1, will statement4 be executed, and will statement5 be executed?
- If the exception is not of type Exception1, will statement4 be executed, and will statement5 be executed?

Unit 4: Interface and Packages

1. Define an Interface. With the help of a program explain how to implement an interface.
2. Compare and contrast an interface and an abstract class with an example.
3. Write a Java program to find area and perimeter of a rectangle, triangle and circle using an interface.
4. With the help of a suitable program explain how multiple inheritance can be achieved in Java using interface.
5. Develop an Interest interface which contains simpleInterest and compInterest methods and static final field of Rate 25%. Write a class to implement those methods.
6. With a suitable example program explain the need of interface in Java.
7. "Interface variables are static and final by default in Java"- Support this statement with proper explanation.
8. Explain the similarities between interfaces and classes.
9. With an example program explain how to define a group of constants that will be used by many classes.
10. While implementing multiple interface, what happens if both the interface declare the same method? Explain it with suitable example program.
11. Find out the error in the following interface program and fix the error and use it in other class.

```
public interface SomethingIsWrong
{
voidaMethod(intaValue){
System.out.println("Hi Mom");
}
```