# ROURKELA INSTITUTE of MANAGEMENT STUDIES

(BCA 3rd Semester Online Mid-Term Exam, 2020)

| Online Internal Examination (Test-1) | |
|---|---|
| Course & Year : BCA 2nd Year | |
| Name of Student : Omishan Upadhyay | |
| Section : A | Roll Number : S3119BCA26 |
| Paper code : | Subject : JAVA Programming |
| Total Pages = 5 | Omishan Upadhyay<br>Full Signature |

# 1) Wrapper Class in JAVA

Wrapper Class is the class in JAVA which consists of all the primitive datatypes and hence, provides a way to implement them in a program. It also provides way to convert the primitive datatypes into objects and vice-versa. The Wrapper Class comes under the "java.lang" package.

As JAVA doesn't support call-by-reference method, so if we pass data variables to the parameters then the actual values will not get affected; but, if we pass the data as objects into the parameter, then the actual values will be updated as per the changes done to the parameters.

Example :-

```
void swap (int x, int y)
{
    int c = x;
    x = y;
    y = c;
}
public static void main (String args[])
{
    int a = 10, b = 20;
    swap (a, b);
    System.out.println ("A = " + a + "\tB = " + b);
}
```

Here values of 'x' and 'y' will get swapped but will not get updated to 'a' & 'b'. But if we pass them as objects then it will be updated.



Omnishom Upadhyay

2)

| Abstract Class | Interface |
|---|---|
| • Abstract class is used for data abstraction in a program | • Interface is the method used in Abstract class to achieve data abstraction. |
| • It can have both abstract as well as non-abstract methods. | • It can only have the abstract methods. |
| • Abstract class can have the methods final, non-final, static and non-static. | • Interface has only static and final variables. |
| • Keyword : "extends" | • Keyword : "implements" |

3) JAVA Package → A java package is a group of similar types of classes, interfaces and sub-packages. It is categorised into two types:-

- Built-in
- User-defined

These are used to categorize the classes, objects etc., provide secure access and avoid name collisions.

Example showing creation of user-defined package :-

```
package MyNewPack;
public class NewPack
{
    public static void main (String args[])
    {
        System.out.println ("Welcome to my new package.");
    }
}
```



Anirudhan Upadhyay

4.) Method Overloading → In a program, if a class has various methods/functions with different parameters but have same name, then it is said to be Method Over loading

Example :-

```
int change (int x, y)
void change (String a)
boolean change( int a)
```

Method Overriding → In a program, if a subclass of a class has the same methods as declared in parent class, then it is said to be Method Overriding.

Example :-

```
class One
{
    int input (int x, int y)
    {
        ---------
        ---------
        ---------
    }
}
class SubOne extends One
{
    int input (int x, int y)
    {
        ---------
        ---------
        ---------
    }
}
```



Omrishan Upadhyay

# Programming

## 1.) Decimal to Binary using array :-

```
import java.lang.*;
import java.util.*;
import java.io.*;
class DecimalToBinary
{
    public static void main(String args[])
    {
        int n;
        Scanner os = new Scanner(System.in);
        System.out.print("Input Decimal No.: "); n = os.nextInt();
        int binNum[] = new int[1000]; int i = 0;
        for(int i=0; n>0; n/=2, i++)
        {
            binNum[i] = n%2;
        }
        System.out.println("Binary Form :- ");
        for(int j = i-1; j>=0; j--)
            System.out.print(binNum(i));
    }
}
```



Omisham Upadhyay