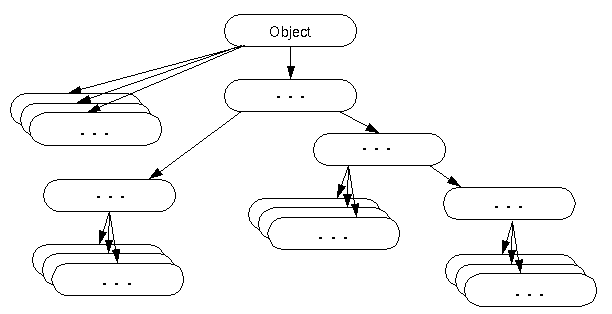
**Lab 4**

**Object class in Java**

The **Object class** is the **parent class of all the classes** in java by default. In other words, it is the topmost class of java. It contains a number of methods, one of which is given below:

**public String toString()** → returns the string representation of this object.



**Task 1**

Design a “Vehicle” class. A vehicle assumes that the whole world is a 2D graph paper. It maintains its **x and y coordinates** (both are integers). The vehicle gets manufactured (constructed?) at (0,0) coordinate.

Write a user class called “**Vehicle**”. It must have methods to **move up, down, left, right** and a **printCoords** method for printing current coordinates.

Note: All moves are 1 step. That means a single call to any move method changes the value of either x or y or both by 1.

|  |  |
| --- | --- |
| public class VehicleUser{  public static void main(String[] args){  Vehicle car = new Vehicle();  System.out.println(car.printCoords());  car.moveUp();  System.out.println(car.printCoords());  car.moveLeft();  System.out.println(car.printCoords());  car.moveDown();  System.out.println(car.printCoords());  car.moveRight();  System.out.println(car.printCoords());  }  } | **Expected Output**  (0, 0)  (0, 1)  (-1, 1)  (-1, 0)  (0,0) |

**Task 2**

Write a java program, which calculates “the area of a circle” and “the area of a sphere” by overriding the space() method in the subclass. [Extend the following super class ‘Point’ with necessary overriding of specific method. DO NOT CHANGE THE POINT CLASS].

class Point{

private double radius;

Point (double r) {

radius = r;

}

double space( ){

System.out.println("Space for a Point can’t be defined");

return 0;

}

protected double getRadius(){

return radius;

}

}

Implement the above program to include constructors for all the subclasses & use ‘super( )’ to call super-class constructors. You must reuse the radius the variable of the Point class. The main method is given below:

import java.util.Scanner;

public class CircleTest{

public static void main(String[]args){

Scanner kb = new Scanner(System.in);

System.out.println("Enter radius of circle");

double circleRadius = kb.nextDouble();

Circle circleObj = new Circle(circleRadius);

System.out.println("The area of the circle is "+circleObj.space());

System.out.println("Enter radius of sphere");

double spehereRadius = kb.nextDouble();

Sphere sphereObj = new Sphere(spehereRadius);

System.out.println("The area of the sphere is "+sphereObj.space());

}

}

**Sample Input/Output**

Enter radius of Circle: **5**  
Creating a circle...done!  
The area of the circle is 78.53981633974483

Enter radius of Sphere: **7**  
Creating a Sphere...done!  
The area of the sphere is 615.7521601035994

**Task 3**

Given the following classes, write the code for the **BBAStudent** class so that the following output is printed when we run the **TestStudent** class.

Name : Default BBA Student Department: BBA

Name : Humpty Dumpty Department: BBA

Name : Little Bo Peep Department: BBA

|  |
| --- |
| class Student{  private String name = "Just a Student";  private String department = "nothing";  public void setDepartment(String dpt){  this.department = dpt;  }  protected String getName(){  return name;  }  protected void setName(String name){  this.name = name;  }  public String toString(){  return "Name : " + name + " Department: " + department;  }  }  public class TestStudent{  public static void printName(Student s){  System.out.println(s.toString());  }  public static void main(String [] args){  printName(new BBAStudent());  printName(new BBAStudent("Humpty Dumpty"));  printName(new BBAStudent("Little Bo Peep"));  }  } |

**Task 4**

Write the **CheckingAccount** class so that the following code generates the output below

Number of Checking Accounts: 0

Account Balance: 0.0

Account Balance: 100.0

Account Balance: 200.0

Number of Checking Accounts: 3

|  |
| --- |
| public class Account{ |
| protected double balance = 0.0; |
| public Account(double balance){ |
| this.balance = balance; |
| } |
| public double getBalance(){ |
| return balance; |
| } |
| } |
| public class TestAccount{ |
| public static void printBalance(Account a){ |
| System.out.println("Account Balance: " + a.getBalance()); |
| } |
| public static void main(String [] args) |
| { |
| System.out.println("Number of Checking Accounts: " + CheckingAccount.numberOfAccount); |
| printBalance(new CheckingAccount()); |
| printBalance(new CheckingAccount(100.00)); |
| printBalance(new CheckingAccount(200.00)); |
| System.out.println("Number of Checking Accounts: " + CheckingAccount.numberOfAccount); |
| } |
| } |