**Assignment OOPs**

1. –x--Create a class called *Rectangle* that stores width

and height. Write functions to accept, display and get/set of the data. Also write method that returns its area. Test the above code by creating an implementation program. Write *get/set methods using shortcuts.*

2. ---Create a class called *Student* that stores its name, class, section, and marks in four subjects. Write

functions to accept, display and get/set of the data. Also write methods that return its total and

percentage. Test the above code by creating an implementation program.

3. ----Create a class called *Employee* that stores its name, department, designation and basic salary. Write

functions to accept, display and get/set of the data. Also write method that displays his incentives.

The incentives includes HRA (20%), DA (10%), CA(10%). Test the above code by creating an

implementation program.

4. ----Create a class called *Circle* that stores radius. Write methods to get/set, displays and accept the data.

All write methods that return its area and circumference. Create its object and implement all the

methods defined.

5. –x--Add the constructors to all the above-mentioned programs using short cuts that NetBeans provides.

6. ----Create a class called *Rectangle* derived from *Point* class. Apart from data of point class rectangle

should store its width and height. Write suitable method that accepts, displays or reads/writes the

data. Write appropriate constructors and method that returns its area. Test the class constructors and

methods by writing the implementation program.

7. ----Create a class called *Employee* that stores its code and name. Create two derive classes from

Employee named *TempEmp* and *PerEmp*. *TempEmp* should store wage grade and number of days

worked whereas *PerEmp* should store department, designation and basic salary. Write constructors

and appropriate methods.

8. ----Create a class called *Circle* derived from *Point*. Apart from Point this class should store the radius.

Write constructors and appropriate methods. Also declare a static final called PI in this class that can

be used in area and circumference methods. These methods should also be declared as final.

9. Create an abstract class called *AbstarctA,* with a method called *showData()*. Create a derive class

from this class called *DerivedA* with an integer as its member data. Write suitable constructors and

methods. In the implementation program try to create an object of class *AbstractA* and *DerivedA*.

10. Create a class that counts the number of its objects created. If the object counter is equal to 5, it

should display a message “Too Less…” on the screen at the time of object creation. Like when its is

equal to 5 then “Will Do…” and if its more than 5 then it should display “Exceeding the Limits…”

Write suitable constructors and methods. Also write methods that returns the number of objects

created.