Java Week 4

Assignments: Implementing the concepts of class variable, instance variable, use of "this" keyword, use of reference variable in Java.

1. Write a Java program to implement the concept of inheritance.

```
package Week4;

class Employee{

     float salary=40000;
}

public class InheritEx  extends Employee{
      int bonus=10000;
      public static void main(String args[]){
           InheritEx p=new InheritEx();
        System.out.println("Programmer salary is:"+p.salary);
        System.out.println("Bonus of Programmer is:"+p.bonus);
}
}
Programmer salary is:40000.0
Bonus of Programmer is:10000
```

2. Write a Java program to show method overloading.

```
package Week4;

class Adder{
     static int add(int a,int b)
     {
          return a+b;
     }
     static int add(int a,int b,int c)
     {
          return a+b+c;
}
}
class OverloadingEx{

     public static void main(String[] args)
     {
          System.out.println(Adder.add(11,11));
          System.out.println(Adder.add(11,11,11));
     }
}
```

```
22
33
```

3. Write a Java program to show method overriding.

```
package Week4;

class Vehicle{
     void run()
     {
          System.out.println("Vehicle is running");
     }
}
class OverRidingEx extends Vehicle{

     void run()
     {
          System.out.println("Bike is running safely");
     }

     public static void main(String args[]){
          OverRidingEx obj = new OverRidingEx();
          obj.run();
     }
}
Bike is running safely
```

4. Write a Java program to show method hiding.

```
package Week4;

class Demo{
        public static void demoMethod() {
           System.out.println("method of super class");
        }
     }

public class HidingEx extends Demo {

     public static void demoMethod() {
           System.out.println("method of sub class");
        }

     public static void main(String args[] ) {
             HidingEx.demoMethod();
        }
     }

method of sub class
```

```
package Week4;

import java.util.Scanner;

import start.java.Box;

public class ThreeD_Class {


      static int SurfaceArea(int h, int w, int l) {

          return (2*(h*w)+2*(h*l)+2*(w*l));
     }

      static int SurfaceArea(int a) {

          return (6*a*a);
     }

      static int SurfaceArea(int h,int r) {

                return (int) (Math.PI*r*(r+Math.sqrt(Math.pow(h,
2)+Math.pow(r, 2))));
          }


      static int SurfaceArea(double h,double r) {

                return (int) (2*Math.PI*r*h+2*Math.PI*Math.pow(r, 2));
          }


      static int volume(int h, int w, int l) {

          return (h*w*l);
     }

      static int volume(int a) {

                return (a*a*a);
          }
```

```
      static int volume(int h,int r) {

                return (int) (Math.PI*Math.pow(r,2)*(h/3));
           }


      static int volume(double h,double r) {

                return (int) (Math.PI*Math.pow(r,2)*h);
           }

     public static void main(String args[])
     {
          ThreeD_Class obj=new ThreeD_Class();

          System.out.println("Box surface area
"+SurfaceArea(10,20,30));
          System.out.println("Cube surface area "+SurfaceArea(10));
          System.out.println("Cone surface area "+SurfaceArea(10,20));
          System.out.println("Sylinder surface area
"+SurfaceArea(10.0,20.0));

          System.out.println("Box volume " +volume(10,20,30));
          System.out.println("Cube volume "+volume(10));
          System.out.println("Cone volume "+volume(10,20));
          System.out.println("Sylinder volume "+volume(10.0,20.0));

     }
}
```

```
Box surface area 2200
Cube surface area 600
Cone surface area 2661
Sylinder surface area 3769
Box volume6000
Cube volume  1000
Cone volume  3769
Sylinder volume  12566
```

6. Write a program to create a class named Vehicle having protected instance variables regnNumber, speed, color, ownerName and a method showData ( ) to show “This is a vehicle class”. Inherit the Vehicle class into subclasses named Bus and Car having individual private instance variables routeNumber in Bus and manufacturerName in Car and both of them having showData ( ) method showing all details of Bus and Car respectively with content of the super class’s showData ( ) method.

```
package Week4;
```

```
class Vehicle_Main {

	int regnNumber;
	int speed;
	String color;
	String ownerName;

	public Vehicle_Main(int regnNumber, int speed, String color,
String ownerName) {

		this.regnNumber = regnNumber;
		this.speed = speed;
		this.color = color;
		this.ownerName = ownerName;
	}


}

class Bus extends Vehicle_Main{

	int routeNumber;

	public Bus(int regnNumber, int speed, String color, String
ownerName, int routeNumber) {
		super(regnNumber, speed, color, ownerName);
		this.routeNumber = routeNumber;
	}



	public String ShowData() {
		return "Bus [routeNumber=" + routeNumber + ", regnNumber=" +
regnNumber + ", speed=" + speed + ", color="
				+ color + ", ownerName=" + ownerName + "]";
	}


}

class Car extends Vehicle_Main{

	String manufacturerName ;

	public Car(int regnNumber, int speed, String color, String
ownerName, String manufacturerName ) {
		super(regnNumber, speed, color, ownerName);
```

```
            this.manufacturerName  = manufacturerName ;
      }



      public String ShowData() {
            return "Car [manufacturerName=" + manufacturerName + ",
regnNumber=" + regnNumber + ", speed=" + speed + ", color="
                        + color + ", ownerName=" + ownerName + "]";
      }


}

public class Vehicle_Drive {

      public static void main(String args[]) {

            Bus obj=new Bus(1010,70,"Black","Soumyadip",199);
            System.out.println(obj.ShowData());

            Car obj1=new Car(1010,70,"Black","Soumyadip","TATA");
            System.out.println(obj1.ShowData());

      }

}
Bus [routeNumber=199, regnNumber=1010, speed=70, color=Black,
ownerName=Soumyadip]
Car [manufacturerName=TATA, regnNumber=1010, speed=70, color=Black,
ownerName=Soumyadip]
```

7. An educational institution maintains a database of its employees. The database is divided into a number of classes whose hierarchical relationships are shown below. Write all the classes and define the methods to create the database and retrieve individual information as and when needed.

Write a driver program to test the classes.
Staff (code, name) *Teacher* (subject, publication) is a Staff
Officer (grade) is a Staff *Typist* (speed) is a Staff
RegularTypist (remuneration) is a Typist *CasualTypist* (daily wages) is a Typist.

```
package Week4;

class Staff{

      int code;
      String name;
```

```
	public Staff(int code, String name) {
		super();
		this.code = code;
		this.name = name;
	}

}
class Teacher extends Staff{

	String subject, publication;

	public Teacher(int code, String name, String subject, String
publication) {
		super(code, name);
		this.subject = subject;
		this.publication = publication;
	}


	public String ShowData() {
		return "Teacher [subject=" + subject + ", publication=" +
publication + ", code=" + code + ", name=" + name
				+ "]";
	}

}

class Officer extends Staff{

	String grade;

	public Officer(int code, String name, String grade) {
		super(code, name);
		this.grade = grade;
	}

	public String ShowData() {
		return "Officer [grade=" + grade + ", code=" + code + ",
name=" + name + "]";
	}

}


class Typist  extends Staff{

	int speed;
```

```
	public Typist(int code, String name, int speed) {
		super(code, name);
		this.speed = speed;
	}


}

class Regular_Typist  extends Typist{

	int remuneration;

	public Regular_Typist(int code, String name, int speed, int
remuneration) {
		super(code, name, speed);
		this.remuneration = remuneration;
	}

	public String ShowData() {
		return "Regular_Typist [remuneration=" + remuneration + ",
speed=" + speed + ", code=" + code + ", name=" + name
				+ "]";
	}
}


class Casual_Typist  extends Typist{

	int wages;

	public Casual_Typist(int code, String name, int speed, int wages)
{
		super(code, name, speed);
		this.wages = wages;
	}

	public String ShowData() {
		return "Casual_Typist [wages=" + wages + ", speed=" + speed
+ ", code=" + code + ", name=" + name + "]";
	}


}
public class School_Database {

	public static void main(String args[]) {

		Teacher obj=new Teacher(101,"MB","JAVA","ABC Publication");
```

```
          Officer obj1=new Officer(102,"SDC","A");
          Regular_Typist obj2=new Regular_Typist(102, "John",
40,10000);
          Casual_Typist obj3=new Casual_Typist(104,"Tom",50,5000);

          System.out.println(obj.ShowData());
          System.out.println(obj1.ShowData());
          System.out.println(obj2.ShowData());
          System.out.println(obj3.ShowData());
     }

}
Teacher [subject=JAVA, publication=ABC Publication, code=101, name=MB]
Officer [grade=A, code=102, name=SDC]
Regular_Typist [remuneration=10000, speed=40, code=102, name=John]
Casual_Typist [wages=5000, speed=50, code=104, name=Tom]
```

08. Create a base class Building that stores the number of floors of a building, number of rooms and it's total footage. Create a derived class House that inherits Building and also stores the number of bedrooms and bathrooms. Demonstrate the working of the classes.

```
package Week4;


 class Building_Main {

     int room;
     int floor;
}

class House extends Building_Main{

     int bedroom;
     int bathroom;

     public House(int bedroom, int bathroom,int room,int floor) {
          super.room=room;
          super.floor=floor;
          this.bedroom = bedroom;
          this.bathroom = bathroom;
     }

     public String ShowData() {
          return "House [bedroom=" + bedroom + ", bathroom=" +
bathroom + ", room=" + room + ", floor=" + floor + "]";
     }
}
```

```
public class Building_Drive {

	public static void main(String args[]) {

		House obj=new House(3,2, 5,2);
		System.out.println(obj.ShowData());
	}

}
```

House [bedroom=3, bathroom=2, room=5, floor=2]

9. In the earlier program, create a second derived class Office that inherits Building and stores the number of telephones and tables. Now demonstrate the working of all three classes.

```
package Week4;


 class Building_Main {

	int room;
	int floor;
}

class House extends Building_Main{

	int bedroom;
	int bathroom;

	public House(int bedroom, int bathroom,int room,int floor) {
		super.room=room;
		super.floor=floor;
		this.bedroom = bedroom;
		this.bathroom = bathroom;
	}

	public String ShowData() {
		return "House [bedroom=" + bedroom + ", bathroom=" +
bathroom + ", room=" + room + ", floor=" + floor + "]";
	}
}

class Office extends Building_Main{


	int telephone;
```

```
     int table;

     public  Office(int telephone, int table,int room,int floor) {
          super.room=room;
          super.floor=floor;
          this.telephone = telephone;
          this.table = table;
     }

     public String ShowData() {
          return "Office [telephone=" + telephone + ", table=" + table
+ ", room=" + room + ", floor=" + floor + "]";
     }
}

public class Building_Drive {

     public static void main(String args[]) {

          House obj=new House(3,2, 5,2);
          Office obj1=new Office(34,22, 10,20);
          System.out.println(obj.ShowData());
          System.out.println(obj1.ShowData());
     }

}
House [bedroom=3, bathroom=2, room=5, floor=2]
Office [telephone=34, table=22, room=10, floor=20]
```

10. Write a Java program which creates a base class Num and contains an integer number along with a method shownum() which displays the number. Now create a derived class HexNum which inherits Num and overrides shownum() which displays the hexadecimal value of the number. Demonstrate the working of the classes.

```
package Week4;

 class Num{

    int x;

     public void ShowNum()
     {
          System.out.println(x);
     }
}

 class HexNum extends Num{
```

```
      public void ShowNum()
      {
            String hex = Integer.toHexString(x);
            System.out.println(hex);
           }
 }
public class Number_Driver {

     public static void main(String args[]) {

          HexNum obj=new HexNum();
          obj.x=10;
          obj.ShowNum();

     }

}
```

Output: a

11. Write a Java program which creates a base class Num and contains an integer number along with a method shownum() which displays the number. Now create a derived class OctNum which inherits Num and overrides shownum() which displays the octal value of the number. Demonstrate the working of the classes.

```
package Week4;

 class Num{

    int x;

     public void ShowNum()
     {
          System.out.println(x);
     }
}
class OctNum extends Num{

      public void ShowNum()
      {
            String oct = Integer.toOctalString(x);
            System.out.println("Oct of " +x+" is "+oct);
           }
 }

public class Number_Driver {

     public static void main(String args[]) {
```

```
            OctNum obj=new OctNum();
            obj.x=10;
            obj.ShowNum();

      }

}
Octs of 10 is 12
```

12. Combine Question number 10 and 11 and have all the three classes together. Now m describe the working of all classes.

```
package Week4;

 class Num{

    int x;

     public void ShowNum()
     {
          System.out.println(x);
     }
}

 class HexNum extends Num{

      public void ShowNum()
      {
           String hex = Integer.toHexString(x);
           System.out.println("Hex of "+x+" is "+hex);
          }
 }
 class OctNum extends Num{

      public void ShowNum()
      {
           String hex = Integer.toOctalString(x);
           System.out.println("Oct of " +x+" is "+hex);
          }
 }

public class Number_Driver {

     public static void main(String args[]) {

          OctNum obj=new OctNum();
          obj.x=10;
```

```
            obj.ShowNum();

            HexNum obj1=new HexNum();
            obj1.x=10;
            obj1.ShowNum();
      }

}
Oct of 10 is 12
Hex of 10 is a
```

15. Write a Java program to explain “multilevel inheritance.”

```
package object_class.java;

class Student{
      private String name;
      private int year;
      void setName(String name)
      {
            this.name=name;
      }
      void setRoll(int year)
      {
            this.year=year;
      }
      protected String getName()
      {
            return name;
      }
      protected int getyear()
      {
            return year;
      }
}

class Student_bio extends Student{
      private String collage;

      protected void setCollage(String collage) {
            this.collage = collage;
      }
      protected String getCollage() {
            return collage;
      }

}
```

```
public class Inheritance extends Student_bio {

	public static void main(String args[])
	{
		Inheritance obj=new Inheritance();
		obj.setName("Soumyadip");
		obj.setRoll(3);
		obj.setCollage("UEMK");
		System.out.println("Name="+obj.getName());
		System.out.println("Year="+obj.getyear());
		System.out.println("Collage="+obj.getCollage());

	}
}
```

```
Name=Soumyadip
Year=3
Collage=UEMK
```