NAME:ROHAN NYATI

SAP ID:500075940

ROLL NO. : R177219148

BATCH-5(AI&ML)

Experiment-4

TITLE: Inheritance

1. Write a program in Java to create a Player class. Inherit the classes Cricket _Player, Football _Player and Hockey_ Player from Player class.

CODE

```
package rohan;
public class c {


public static void main(String args[])
{
criket_player c=new criket_player("Ameer","criket",25);
football_player f=new football_player("arun","foot ball",25);
hockey_player h=new hockey_player("Ram","hockey",25);
c.show();
f.show();
h.show();
}
}
class Player
{
String name;
int age;
Player(String n,int a)
{ name=n; age=a; }
void show()
{
System.out.println("\n");
System.out.println("Player name : "+name);
System.out.println("Age : "+age);
}
```

```
}
class criket_player extends Player
{
String type;
criket_player(String n,String t,int a)
{
super(n,a);
type=t;
}
public void show()
{
super.show();
System.out.println("Player type : "+type);
}
}
class football_player extends Player
{
String type;
football_player(String n,String t,int a)
{
super(n,a);
type=t;
}
public void show()
{
super.show();
System.out.println("Player type : "+type);
}
}
class hockey_player extends Player
{
String type;
hockey_player(String n,String t,int a)
{
super(n,a);
type=t;
}
public void show()
{
super.show();
System.out.println("Player type : "+type);
}
}
```

OUTPUT

```
Player name : Ameer
Age : 25
Player type : criket


Player name : arun
Age : 25
Player type : foot ball


Player name : Ram
Age : 25
Player type : hockey
```

2. Write a Java program to show that private member of a super class cannot be accessed from derived classes.

CODE

```
package rohan;
public class R
{
public static void main(String args[])
{
class_room cr=new class_room(10,20,15);
int a1=cr.area();
int v1=cr.volume();
System.out.println("Area of Room : "+a1);
System.out.println("Volume of Room : "+v1);
}
}
class room
{
        private int l,b;
        room(int x,int y)
        { l=x; b=y;}
        int area()
        {return(l*b);
        }
}
class class_room extends room
```

```
{
        int h;
        class_room(int x,int y,int z)
        {
        super(x,y);
        h=z;
        }
        int volume()
        {
        return(area()*h);
        }
}
```

OUTPUT

```
Area of Room : 200
Volume of Room : 3000
```

3. Write a class Worker and derive classes DailyWorker and SalariedWorker from it. Every worker has a name and a salary rate. Write method ComPay (int hours) to compute the week pay of every worker. A Daily Worker is paid on the basis of the number of days he/she works. The Salaried Worker gets paid the wage for 40 hours a week no matter what the actual hours are. Test this program to calculate the pay of workers. You are expected to use the concept of polymorphism to write this program.

CODE

```
package rohan;
class W
{
public static void main(String args[])
{
dailyworker d=new dailyworker(254,"Arjun",75);
salariedworker s=new salariedworker(666,"Unni",100);
d.compay(45);
s.compay();
}
}
class worker
{
```

```
String name;
int empno;
worker(int no,String n)
{ empno=no; name=n; }
void show()
{
System.out.println("\n-------------------------");
System.out.println("Employee number : "+empno);
System.out.println("Employee name : "+name);
}
}
class dailyworker extends worker
{
int rate;
dailyworker(int no,String n,int r)
{
super(no,n);
rate=r;
}
void compay(int h)
{
show();
System.out.println("Salary : "+rate*h);
}
}
class salariedworker extends worker
{
int rate;
salariedworker(int no,String n,int r)
{
super(no,n);
rate=r;
}
int hour=40;
void compay()
{
show();
System.out.println("Salary : "+rate*hour);
}
}
```

OUTPUT

```
-------------------------
Employee number : 254
Employee name : Arjun
Salary : 3375

-------------------------
Employee number : 666
Employee name : Unni
Salary : 4000
```

4. Consider the trunk calls of a telephone exchange. A trunk call can be ordinary, urgent or lightning. The charges depend on the duration and the type of the call. Write a program using the concept of polymorphism in Java to calculate the charges.

CODE

```
import java.util.Scanner;
class Telephone{
	int callnumber;
	String calltype;
	Telephone(int c,String s){
		callnumber = c;
		calltype = s;
	}
	void show() {
		System.out.println("call number"+" "+callnumber);
		System.out.println("call type"+" "+calltype);
		}
}
class Ordinary extends Telephone{
	float cost;
	Ordinary(int c,String s,float co){
		super(c,s);
		this.cost = co;
	}
	void charge(double time) {
		super.show();//calling show() of parent class
		System.out.println("call charges"+" "+cost*time);
	}
}
class Urgent extends Telephone{
	float cost;
	Urgent(int c,String s,float co){
		super(c,s);
```

```
                this.cost = co;
        }
        void charge(double time) {
                super.show();
                System.out.println("call charges"+" "+cost*time);
                }
}

class Lightening extends Telephone{
        float cost;
        Lightening(int c,String s,float co){
                super(c,s);
                this.cost = co;
        }
        void charge(double time) {
                super.show();
                System.out.println("call charges"+" "+cost*time);
                }
}
public class Exp_4_4 {

        public static void main(String[] args) {
                // TODO Auto-generated method stub
                Scanner sc=new Scanner(System.in);
                int a,no;
                double t;
                System.out.println("Enter 1 for ordinary call,2 for urgent call or 3
for lightening call");
                a=sc.nextInt();
                switch(a) {
                case 1:
                        System.out.println("Enter the number to which call is to be
made");
                         no=sc.nextInt();
                        Ordinary obj1 = new Ordinary(no,"ordinary call",3);
                        System.out.println("Enter the duration of the call");
                        t=sc.nextDouble();
                        obj1.charge(t);
                        break;
                case 2:
                        System.out.println("Enter the number");

                        no=sc.nextInt();
```

```
                Urgent obj2=new Urgent(no,"Urgent call",4);
                System.out.println("Enter the duration of the call");
                t=sc.nextDouble();
                obj2.charge(t);
                break;
            case 3:
                System.out.println("Enter the number");
                 no=sc.nextInt();
                Lightening obj3 = new Lightening(no,"Lightening call",5);
                System.out.println("Enter the duration of the call");
                t=sc.nextDouble();
                obj3.charge(t);
            }


    }
}
```

OUTPUT

```
Enter 1 for ordinary call,2 for urgent call or 3 for lightening call
2
Enter the number
345656
Enter the duration of the call
35.8
call number 345656
call type Urgent call
call charges 143.2
```

5. Design a class employee of an organization. An employee has a name, empid, and salary. Write the default constructor, a constructor with parameters (name, empid, and salary) and methods to return name and salary. Also write a method increaseSalary that raises the employee's salary by a certain user specified percentage. Derive a subclass Manager from employee. Add an instance variable named department to the manager class. Supply a test program that uses theses classes and methods.

CODE

```
import java.util.Scanner;
class Employee{
        int empid;
    String name;
    double salary;
    Employee(){
        System.out.println("This is employee default constructor");
    }

   Employee(int emid,String name,double salary){
            this.empid = emid;
            this.name = name;
            this.salary = salary;
   }
    String name()
   {
          return name;
   }
    double salary()
    {
        return salary;
    }
    void increase_salary(int i) {
        double x;
        x=((salary*i)/100)+salary;
      System.out.print("Increased salary ="+" "+x);
   }

}
class Manager extends Employee{
        String department;
        Manager(int e,String n,double sal,String d){
                super(e,n,sal);
                department = d;
        }
        void display() {
                Scanner sc=new Scanner(System.in);
                 System.out.println("Name of the employee"+" "+super.name());
                   System.out.println("salary of a employee"+" "+super.salary());

                System.out.print("department is"+ " "+ department+"\n");
                System.out.println("Enter the percentage by which a salary is to be
increased");
```

```
            int ps=sc.nextInt();
                super.increase_salary(ps);

        }

}
public class Exp_5_5 {

        public static void main(String[] args) {
                Scanner sc=new Scanner(System.in);
                System.out.println("Enter the name of the employee");
                String name=sc.nextLine();
                System.out.println("Enter the id of the employee");
                int id=sc.nextInt();
                System.out.println("Enter the salary of the employee");
                double sal=sc.nextDouble();
                Manager M = new Manager(id,name,sal,"CS");
                M.display();

    }
}
```

OUTPUT

```
Enter the name of the employee
Aviral Mehra
Enter the id of the employee
500076136
Enter the salary of the employee
500000
Name of the employee Aviral Mehra
salary of a employee 500000.0
department is CS
Enter the percentage by which a salary is to be increased
10
Increased salary = 550000.0
```