public class Employee {

                private String name;

                private int rank;

                private int yearsOfExperince;

                Employee(String name, int rank, int yearsOfExperince)

                {

                                this.name = name;

                                this.rank = rank;

                                this.yearsOfExperince = yearsOfExperince;

                }

                Employee(Employee e){

                                name = e.name;

                                rank = e.rank;

                                yearsOfExperince = e.yearsOfExperince;

                }

                public int getRank() {

                                return rank;

                }

                public int getYearsOfExperince() {

                                return yearsOfExperince;

                }

                public void display(){

                                System.out.print("name: "+name+" Rank: "+rank+" Years of Exp: "+ yearsOfExperince +" Salary: "+ calculateSalary());

                }

                public double calculateSalary() {

                                return yearsOfExperince \* 500 + rank \* 1000;

                }

}

public class Consultant extends Employee{

                private int hoursPerWeek ;

                Consultant(String name, int rank, int yearsOfExperince, int hoursPerWeek ) {

                                super(name, rank, yearsOfExperince);

                                this.hoursPerWeek = hoursPerWeek ;

                }

                Consultant(Consultant c){

                                super(c);

                                hoursPerWeek= c.hoursPerWeek;

                }

                public int getHoursPerWeek() {

                                return hoursPerWeek;

                }

                public void display(){

                                super.display();

                                System.out.println(" Hours per Week: "+hoursPerWeek);

                }

                public double calculateSalary(){

                                return hoursPerWeek \* 1000 + getRank() \* 500 ;

                }

}

public class Manager extends Employee{

                private int nbEmps;

                Manager(String name, int rank, int yearsOfExperince, int nbEmps) {

                                super(name, rank, yearsOfExperince);

                                this.nbEmps = nbEmps;

                }

                Manager(Manager m){

                                super(m);

                                this.nbEmps = m.nbEmps;

                }

                public int getNbEmps() {

                                return nbEmps;

                }

                public void display(){

                                super.display();

                                System.out.println(" Number of employees: "+nbEmps+" Salary: ");

                }

                public double calculateSalary(){

                                return getYearsOfExperince() \* 500 + getRank() \* 1500 + nbEmps \* 10;

                }

}

public class Company {

                private String name;

                private Employee[] arrEmps;

                private int nbe;

                Company(String name, int size){

                                this.name = name;

                                arrEmps = new Employee[size];

                                nbe = 0;

                }

                public boolean addEmployee(Employee e){

                                if(nbe >= arrEmps.length)

                                                return false;

                                if(e instanceof Manager)

                                                arrEmps[nbe++] = new Manager((Manager)e);

                                else if(e instanceof Consultant)

                                                arrEmps[nbe++] = new Consultant((Consultant)e);

                                else

                                                arrEmps[nbe++] = new Employee(e);

                                return true;

                }

                public void displayAll(){

                                System.out.println("Name: "+name);

                                for(int i=0; i<nbe;i++)

                                                arrEmps[i].display(); // polymorphism here

                }

                public int countConsultant(){

                                int count=0;

                                for(int i=0;i<nbe;i++)

                                                if(arrEmps[i] instanceof Consultant)

                                                                count++;

                                return count;

                }

                public Employee[] getHighSalary(double s){

                                Employee[] res = new Employee[nbe];

                                int j=0;

                                for(int i=0; i< nbe; i++)

                                                if( arrEmps[i].calculateSalary() > s) // polymorphism here

                                                                res[j++]= arrEmps[i];

                                return res;

                }

                public Manager[] getManagers(int n){

                                Manager[] res = new Manager[nbe];

                                int j=0;

                                for(int i=0; i<nbe; i++)

                                                if(arrEmps[i] instanceof Manager)

                                                                if(((Manager)arrEmps[i]).getNbEmps()>n)

                                                                                res[j++]=(Manager) arrEmps[i];

                                return res;

                }

}