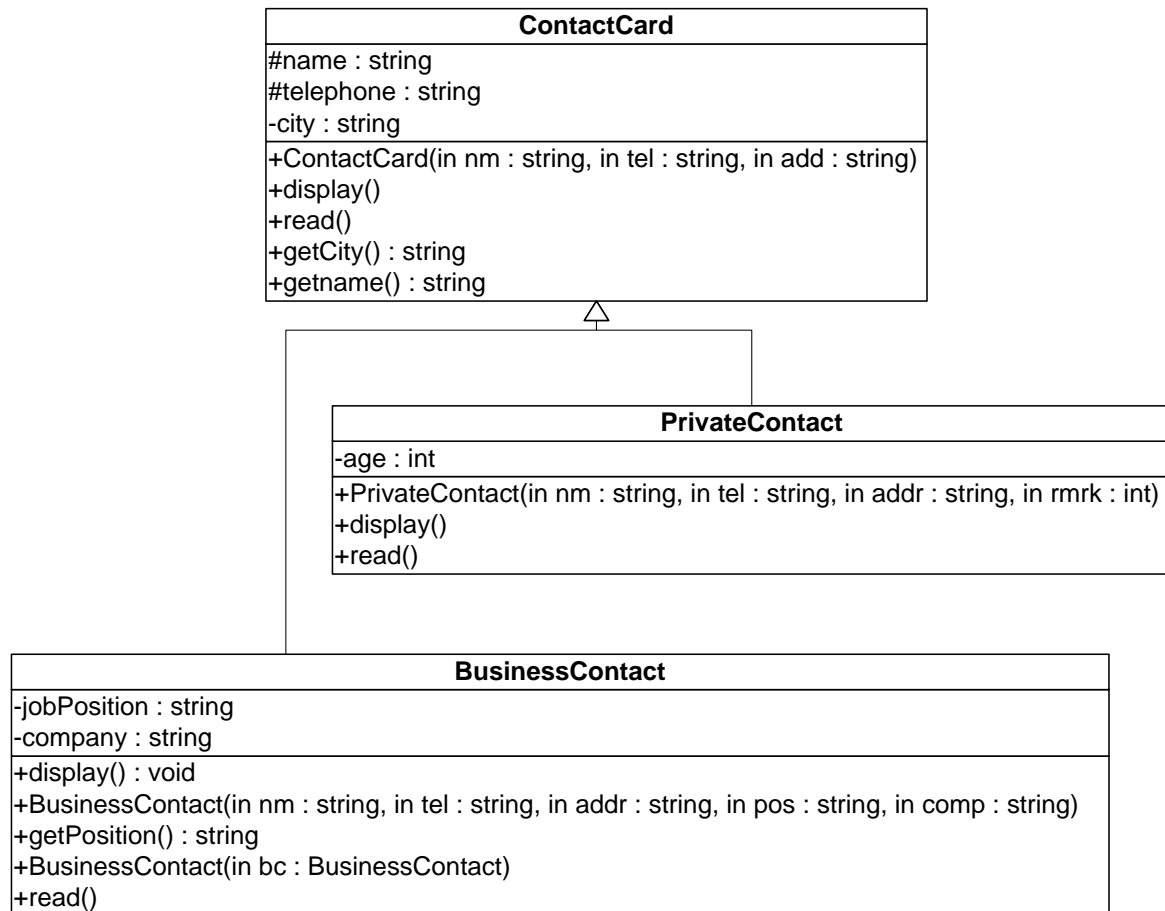


KSU/CCIS/CS	CSC 113 - Mid 1	Fall 30/31
Name:	Student ID:	Dr:

Question 1

Given the following UML diagram:



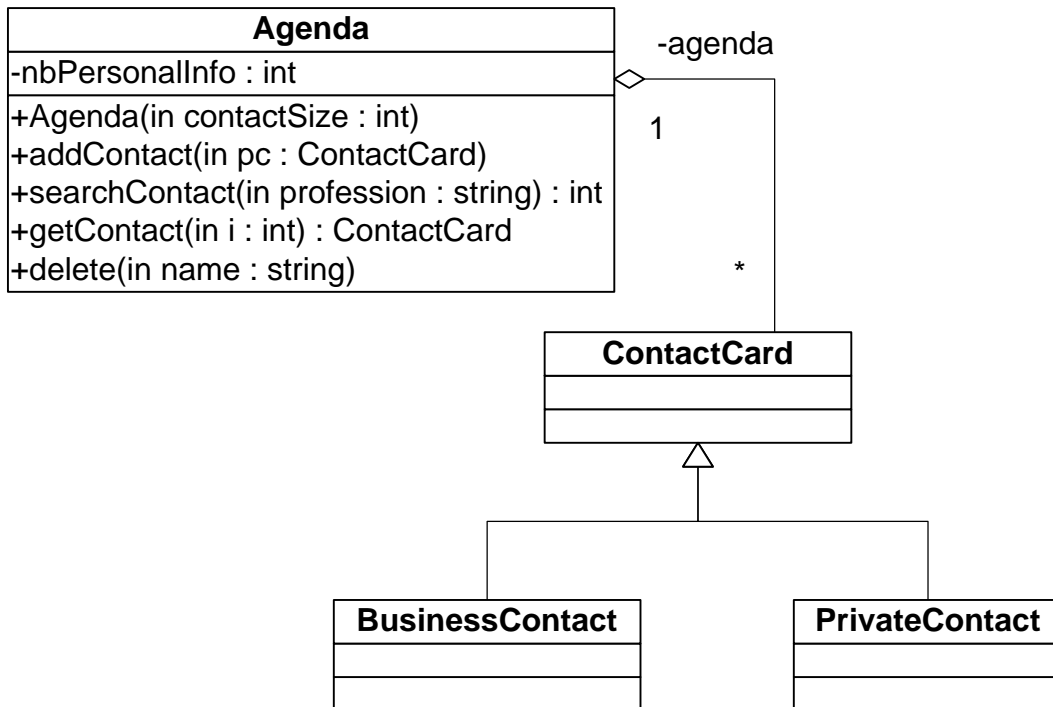
Where

- Display() method display all features of the class.
- BusinessContact(String, String, String, String, String) and BusinessContact(BusinessContact) are two constructors.
- Write in Java the entire BusinessContact class.

KSU/CCIS/CS	CSC 113 - Mid 1	Fall 30/31
Name:	Student ID:	Dr:

Question 2:

Given the following UML diagram (notice that ContactCard is the superclass defined in Question 1):



Where :

- **agenda(int)** allows creating an instance of the class agenda
- **addContact(ContactCard)** adds a new contact to the agenda
- **searchContact(String):int** returns the location of the `ContactCard` instance in the agenda according to the given job position.
- **getContacts(index:int):ContactCard** return the contact card located in the position index
- **Delete(String)** deletes a Contact card from the Agenda according to its attribute name.

Q1. Write in Java the class Agenda (except the methods `addContact` and `delete`)

Q2. Write a main program that :

- Reads and inserts two business and one private contact cards in the agenda.
- Display the contact card information of "Fahd AlOmary"

KSU/CCIS/CS	CSC 113 - Mid 1	Fall 30/31
Name:	Student ID:	Dr:

Question 3:

What is output of the following code:

```
public class Tank {
    protected final static double NUM= 0.5;
    protected int size = 6;
    protected int x,y;
    public Tank ( ) { System.out.println (" OK you can start now ..."); }

    public Tank (int x,int y) { this.y = y; this.x=x; }
    protected void setX(int x) { this.x=x; }
    protected int getX() { return x; }
    public int getSize() { return size; }
    public String toString() { return " Tank"; }
    public double method() { return size; }
}
//-----
public class OilTank extends Tank{
    public OilTank (int x,int y) { super(x,y); }
    public void setR(int r) { super.setX(r); }
    public int getR() { return super.getX(); }
    public String toString () { return super.toString(); }
}
//-----
public class WaterTank extends Tank{
    protected final static double NUMC= 2.0;
    private double var1,var2;
    public WaterTank (int x,int y,double z,double w) {
        super(x,y); var1=z; var2=w;
    }
    public double getVar1() { return var1; }
    public void setVar1(double d) { var1=d; }
    public double method() { return (var1*var2) % NUMC; }
    public String toString() { return super.toString(); }
}
//-----
public class MilkTank extends WaterTank{
    public MilkTank(int x,int y, double r) { super(x,y,r,r); }
    protected String getName() { return " MilkTank "; }
    public double method() { return NUM *
super.getVar1()*super.getVar1(); }
    public String toString () { return "WaterTank "; }
}
```

KSU/CCIS/CS	CSC 113 - Mid 1	Fall 30/31
Name:	Student ID:	Dr:

```
//----- Main -----
public class test {
    public static void main(String[] args) {
        int i;
        int value=3;
        String names[] = {" WaterTank ", "MilkTank ", "OilTank "};
        Tank x = new Tank ();
        Tank [] tankList = new Tank [x.getSize()];
        tankList [0]= new WaterTank (4,3,2,1);
        tankList [1]= new MilkTank (5,6,2);
        tankList [2]= new OilTank (2,4);
        // The following loop will produce 3 output lines where each line
        has mainly 3 parts
        for (i=0; i< 3; i++) {
            System.out.print("{ " + "<tankList[" + i + "]" + " is a "names[i % value]+ " > , ");
            System.out.print("< tankList[" + i + "]" + ".method returns " + tankList[i].method()
                + " > , ");
            System.out.println("< " + names[i % value] + " has <is-a> relation to " +
                tankList[i].toString()+ " > }");
        }
    }
}
```