

Name:	Section:	Fall 2008
ID:	CSC111 Final	Dr. Mohamed

**Question 1 (10 marks):** Electricity Company wants to manage bills; we create the shown UML class **Electricity\_Bill** containing the following *attributes*:

- id: customer account number (*example*: id = "546234").
- kwatts: kilowatts used (*example*: kwatts = 3000).

Electricity_Bill
- id : String - kwatts: double
+ Electricity_Bill(id: String) + compute_Electricity_Charge(): double + differenc(Electricity_Bill eBill) : double + isSaving(Electricity_Bill eBill) : void + display_Bill(): void

*Methods:*

- **compute\_Electricity\_Charge ()** : returns the electricity charge in Riyal for a consuming kilowatt-hours by using the following table:

Charge kilowatt/hour in Riyal	Kilowatts
0.05	0 - 4000 (The first 4000 kilowatts)
0.10	Between 4000 and 6000
0.30	More than 6000

- **differenc(Electricity\_Bill eBill)**: this method returns the difference in Riyals between the electricity charge of current Electricity Bill and the electricity charge eBill.
- **isSaving(Electricity\_Bill eBill)** : this method displays a message telling us whether the current bill is less charged than eBill or not.
- **display\_Bill()** : this method displays an electricity bill (see the following two examples)

Electricity Bill	
customer number: <b>546234</b>	
Consumed Kilowatts : <b>3000</b> kilowatts	Electricity charge: <b>150</b> Riyals

Electricity Bill	
customer number: <b>765443</b>	
Consumed Kilowatts : <b>8000</b> kilowatts	Electricity charge: <b>1000</b> Riyals
Note: you have exceeded more than 6000. For that your bill is more charged	

Write in Java the class **Electricity\_Bill**

**Answer:**

Name:	Section:	Fall 2008
ID:	CSC111 MidTerm-2	Dr.

**Question 3:** (continue of question 2).

<b>TestCar</b>
<b>+ main(String[] args): void</b>

We assume that the class Car ( in question 2) is written in Java.

The question is: write in Java the class **TestCar** that processes the following tasks:

- a. Read information of two cars.
- b. Display a message telling us which car is more economic.
- c. Display information of the car, which has the minimum distance.
- d. Display the remaining gallons of the car, which has minimum distance.
- e. Read n the number of cars to be processed.
- f. Read the information of all the cars.
- g. Display the id of the car, which has the maximum value of remaining gallons.

---

**Answer:**

Name:	Section:	Fall 2008
ID:	CSC111 Final	Dr. Mohamed

- Question 4 (8 marks): Give the output of the following program

```

public class Sayara {
    private int id;
    private String owner;
    private double milage, fuel;
    private double consuming;
    private static int nbSayarat = 0;
    public Sayara(String o, double m, double f, double c ) {
        nbSayarat ++;
        id = nbSayarat;
        owner = o;
        milage = m;
        fuel = f;
        consuming=c;
        display( );
    }
    public void go(double d) {
        double x = fuel / consuming * 100;
        System.out.print(owner + "'s car :::> ");
        System.out.print("requested distance : " + d );
        System.out.print(" , consuming rate : "+consuming);
        System.out.print(" , available fuel : " + fuel );
        System.out.println( " , possibly distance = " + x);
        if (x < d) {
            fuel  = fuel - consuming * x / 100.0 ;
            milage += x;
        }
        else {
            fuel  = fuel - consuming * d / 100.0 ;
            milage += d;
        }
    }
    public void display(){
        System.out.print(owner + " 's Car :::> id : " + id);
        System.out.println(" , milage : " + milage + " , fuel : "+ fuel);
    }
}

```

TestSayara.Java

//----- Give the output of each line -----

```

public class TestSayara {
    public static void main(String[] args) {
        line 1. Sayara fathersCar = new Sayara("Mohammad", 200.0, 25.00, 5.0);
        line 2. Sayara sonsCar  = new Sayara("Ibrahim",100.0, 40.00, 10.0);
        line 3. fathersCar.go(800.0);
        line 4. sonsCar.go(200.0);
        line 5. fathersCar.display();
        line 6. sonsCar.display();
    }
}

```

## Output

```
Mohammad 's car :::> id : 1 , milage : 200.0 ,fuel : 25.0
Ibrahim 's car :::> id : 2 , milage : 100.0 ,fuel : 40.0
Mohammad's car :::>requested distance :800.0 , consuming rate :
5.0 , available fuel : 25.0 , possibly distance = 500.0
Ibrahim's car :::>requested distance :200.0 , consuming rate :
10.0 , available fuel : 40.0 , possibly distance = 400.0
Mohammad 's car :::> id : 1 , milage : 700.0 ,fuel : 0.0
Ibrahim 's car :::> id : 2 , milage : 300.0 ,fuel : 20.0
```