Project Four
The Cuban Cafe Problem

A new Cuban cafe is opening in Maryland and Virginia. Write a program for the cafe to display the bill for the cashier. The cafe offers two options: Chef's Menu (32.50) and Vegetarian Meal (28.99).

The program first requests the state abbreviation (MD or VA). The program then accepts the number of *Chef's Menu* meals and the number of *Vegetarian Meals*, performs the necessary calculations, computes the sales tax, and displays the bill with the suggested tip similar to the example below.

```
Enter cafe state:    md

Enter number of Chef's Menu (-1 to stop).....12
Enter number of Vegetarian Meals.............2

            Parkville Cuban Cafe
                              Price    Total
Chef's Menu         12      32.50    390.00
Vegetarian Meal      2      28.99     57.98
    SubTotal                          447.98
Tax (6%)                               26.88

      Total bill                      474.86

Suggested tip (20%)                    89.60


 Enter number of Chef's Menu (-1 to stop).....
```

Part I
Complete the UML diagram for the **Cafe** class.
Code and test the **Cafe** class.

Cafe has these *instance variables* in the *constructor*:

String **state** is assigned the value passed to it (the state abbreviation entered by the user in main). Convert the value passed to the constructor to uppercase.
String **name** is assigned null.
double **taxRate** is assigned 0.

Cafe contains these *methods*.

- **setName** method uses **state** to determine and assign the cafe name to **name**. Use a nested if or switch. Nothing is returned.

state	name
MD	Parkville Cuban Cafe
VA	Alexandria Cuban Cafe
Any other value	null

- **setTaxRate** method uses **state** to determine and assign the tax rate to **taxRate**. Use a nested if or switch. Nothing is returned.

state	taxRate
MD	.06
VA	.04
Any other value	0

- **getTaxRate** method returns the tax rate.

- **toString** method returns a string that displays the first line of the receipt. Here is an example of possible output

```
Alexandria Cuban Cafe
```

main should follow this sequence to test the **Cafe** class:

Request the state for the cafe.
Instantiate an object using the **Cafe** class.
Use the **Cafe** class to assign the cafe name (*setName*).
Use the **Cafe** class to assign the tax rate (*setTaxRate*).
Use these displays to see if the code is working:
Display cafe name using the **Cafe** class *toString* method.
Display the tax rate using the **Cafe** class *getTaxRate* method.

Part II - The **Bill** Class

Now that you have completed the **Cafe** class, you can plan and code the **Bill** class.

Set up a final for the tip like this:

static final double TIP = .20;

Bill has these *instance variables* in the *constructor*:

String **category** will be set to null.
int **mealQty** will be passed the number of meals ordered for the category.
double **mealPrice** will be initialized to 0.
double **mealCost** will be initialized to 0.

Bill contains these *methods*.

calcChefCost sets the category to "Chef's Menu", sets the mealPrice to 32.50, calculates the cost of the meals, and sets and returns the mealCost.

calcVegCost sets the category to "Vegetarian Meal", sets the mealPrice to 28.99, calculates the cost of the meals, and sets and returns the mealCost.

calcTip calculates the suggested tip using the final **TIP**. Return the calculated tip based on the total sent to this method.

Set up the method header like this:

static double calcTip(double subtotal)

display displays the category, mealQty, mealPrice, and mealCost. Use one printf.

The method header and body look like this:

```
public void display()
        System.out.printf( "\n    %-16s %3d    %6.2f   %7.2f", cat, mealQty, mealPrice, mealCost);
```

You can copy/paste the above code into the display method.

main follows this sequence to use the **Bill** class. Add the necessary statements to the main you coded to test the **Cafe** class.

Establishes the name and tax rate for the cafe once before the while loop. These values do not change during the run of the program.

Requests the number of meals for the Chef's Menu and the Vegetarian Meal.

Uses the **Bill** class to instantiate these objects:

chefBill
vegBill

Uses the **Bill** class to calculate the Chef's Menu and Vegetarian Meal costs.

Subtotals the costs.

Displays the cafe name created earlier using the **Cafe** class

– use the **toString** method.

Displays the two lines showing the Chef's Menu and the Vegetarian costs using the **Bill** class

– use the **display** method.

Displays the subtotal.

Displays the tax. Use the **getTaxRate** method from the **Cafe** class to determine the tax. Use the NumberFormat class and/or printf to format the percent and tax.

Displays the total cost.

Uses the **Bill** class to calculate the tip based on the subtotal. The call should look like this:

??? = **Bill.calcTip**(???)

Use a pretest while loop that loops until the user enters -1 for Chef's Menu.

Here are some examples from the *Payroll Program* that may be helpful in displaying the totals, tax, and tip.

```
System.out.printf("        Gross Pay = %8.2f\n", grossPay);

System.out.printf("        Tax = %3s  ", fmtPct.format(TAXRATE));
```

You will be submitting 3 java files for the final version. Any class, variable or method names in **bold** must be used in your program.

Place your name at the top of all files.

Project Points – Check *Schedule* for due dates.	What to include
Project 4 – Part I (20 points)	Submit a UML diagram for the Cafe class. Submit the Cafe class code. Do not submit your main code. A class should work with any main so I will test it with my main. You should, however, code main so you can test the Cafe class as well as the rest of the requirements.
Project 4 - Final Version (100 points)	Final Version of main, Cafe and Bill classes

*Place your name at the top of all the files.