CPS 150/Spring 2017 | Assignment 1 | Restaurant Check Calculator
Assigned: February 13. | Due: February 20, 11 pm. | Cutoff: February 23, 11 pm
Instructor: Tamisra H. Sanyal

**Objectives:** Basic programming involving arithmetic, input, and output, output formatting, named constants, modular programming using methods.

**Functionality**
A restaurant patron orders a drink, an appetizer, a main dish (entrée), and a dessert. This program would take the cost information for each item from the user and calculate the total amount to be paid by the customer, including the sales tax to be charged (at 7.25%) and tips. The program would also suggest how much the tip should be (at levels of 15%, 18%, 20% of total of food costs, *not including the tax*). The program would then ask how much tip the customer actually wishes to pay. This last value would then be used to display the total amount the customer should pay.

**Program input/output**
The program should ask for the prices of each item the customer ordered. It would then show the suggested tip amounts at various levels and ask how much tip the customer wants to pay. It would then show the final output (a restaurant check), itemizing total food cost, sales tax, tip and the total due from the customer. It would include the name of the restaurant (defined as a named constant). See sample output. Use `System.out.printf` for formatted output. Choose a nice name for your restaurant. The sales tax rate should be defined as a named constant.

The output from the program has two clear sections: The dialog with the user, and the displayed/printed check. These two sections should be separated by a horizontal line (a bunch of hyphens).

**Program Structure**
This is essentially a straight-line input-process-output (IPO) program, but you should break up the code into methods (similar to the sample program BasicIPOModular.java). All these methods will accept no parameters. The main method will do the following in sequence:

1. Print the output identification line – **CPS 150 Assignment 1 by *your name***
2. Call a method to get server's first name, prices for drink, appetizer, entrée and dessert
3. Calculate the total cost of the food items (no need to call a method, this is just one statement)
4. Call a method to calculate and print the tip suggestions at 15%, 18%, and 20%.
5. Call a method to prompt for and get the tip actually paid by the customer (you could reuse a method from *BasicIPOModular2.java*).
6. Call a method to calculate the Sales tax (7.25% of the total food price not including the tips) and the total due from customer (= total food price + sales tax + tips).
7. Call a method to print the restaurant check separated from the user interaction by a line of hyphens, the check should be nicely formatted, as in the sample output.
8. Print the sign-off line – **Assignment 1 complete**

You may use the methods that combine prompting and getting input values from *BasicIPOModulr2.java*.

**Program style requirements**
You should declare the variables and named constants at the class level, including the Scanner created on System.in when these are shared by methods. But any variables/constants used in only one method, such as for calculating the suggested tip amounts, should be declared as **local** identifiers within that method.
To get full credit for the assignment, you must do the following:

- Use variable/constant/method names that are meaningful
- Header block comments and sections comments should be given
- Program text should be easy to read and nicely formatted (NetBeans can help with **Source-> format**)
- Program output should be easy to read and well –organized.

**What to submit**
You will need to submit a Microsoft word document containing your source code and the screen shot. See the document "How to submit assignments" posted on Isidore.

**Note**
You can use the following method for rounding a value to two digits after decimal point. This should be done with all money calculations that may have a result with > 2 digits after decimal point (Sales tax and total due).

```
static double round2(double d) {
    return Math.round(d * 100) / 100.0;
}
```

**Sample results**