Project 3

Due February 13, 2015 at 11:55 PM

You will be working alone for this project. **This specification may change at any time for additional clarification.** You should avoid using existing source code as a primer that is currently available on the Internet. You **must** specify in your comments of the file any sources of code that you have viewed to help you complete this project. All class projects will be submitted to MOSS to determine if students have excessively collaborated. Excessive collaboration, or failure to list external code sources will result in the matter being transferred to Student Judicial Affairs.

1. You will be writing a Python program to handle a restaurants orders, payments of patrons, total sales, etc. There are five tables in this restaurant and once a table has started an order, it remains unavailable until payment is received. The list of menu items and prices are as follows:
   Appetizer $5.00
   Salad $3.00
   Soup $3.50
   Entree 1 $10.00
   Entree 2 $12.50
   Dessert $4.25
   Name your file **restaurant.py**.
   a. Place a comment at the top of your file as follows:
      ```
      #
      # Name: Your Name
      # ID  : Your Student ID
      # Date: Today's Date
      #
      ```
   b. Prompt the user to select for starting a new order, edit an existing order, printing a bill for a table, receive payment for a table, print a manager's report, or quit the program. The user must select a value from the list (ignore case) in order to continue. If the use does not enter a valid choice the Enter Choice> prompt is shown again without printing the entire menu. Below is the menu as it must be displayed.
      ```
      S Start New Order
      E Edit Order
      P Print Bill
      R Receive Payment
      M Manager Report
      Q Quit
      -----------------
      Enter Choice>
      ```

c. Selecting Start New Order with ‘S’ will prompt the user to select a table from the available tables. Below shows an example of all tables being available. If no table is available, the message:

```
No available tables at this time!
```

must be printed and the program will return to the main menu.

```
1 Table 1
2 Table 2
3 Table 3
4 Table 4
5 Table 5
---------
Select a Table>
```

If a table is currently occupied it must not appear in the list. Below is an example of Table 3 & 4 being occupied:

```
1 Table 1
2 Table 2
5 Table 5
---------
Select a Table>
```

If the user selects a table number that does is not available, the Select a Table> prompt is shown again without printing the list of tables. After a valid table is selected the Add/Remove Item menu described in Edit Order section below is entered.

d. Selecting Edit Order with ‘E’ will prompt the user to select a table from the occupied tables. Below shows an example of all tables being occupied. If no table is occupied, the message:

```
No orders started yet!
```

must be printed and the program will return to the main menu.

```
1 Table 1
2 Table 2
3 Table 3
4 Table 4
5 Table 5
---------
Select a Table>
```

If a table is currently available it must not appear in the list. Below is an example of Table 3 & 4 being available:

```
1 Table 1
2 Table 2
5 Table 5
---------
Select a Table>
```

If the user selects a table number that does is not occupied, the Select a Table> prompt is shown again without printing the list of tables. After a valid table is selected the Add/Remove Item menu shown below is displayed.

```
A Add Items
R Remove Items
D Done
--------------
Enter Choice>
```

If the user selects an option that is invalid, the Enter Choice> prompt is shown again without printing the menu. If the user selects Add Items with ‘A’ the Add Item menu is shown, discussed below. If the user selects Remove Items with ‘R’ the Remove Items menu is shown, discussed below. If the user selects Done with ‘D’ the program returns to the main menu. The Add Item menu shown below prompts to user to select the food item to add to the tables order. If the user selects Done with ‘D’ The program returns to the Add/Remove Item menu. If the user selects an invalid choice the Enter Choice> prompt is displayed again without printing the menu.

```
1 Appetizer
2 Salad
3 Soup
4 Entree 1
5 Entree 2
6 Dessert
D Done
-----------
Enter Choice>
```

If the user selects a food item to add the user is prompted to enter how many of that item type to add. Below is an example of Appetizer being selected. If the user enters a value less than zero, the error message

```
Invalid must be at least 0.
```

must be printed out, and the prompt is shown again.

```
Add how many Appetizer's>
```

Once a valid number of items is entered, the Add Item menu is shown again.

The Remove Item menu shows a list of the food items currently ordered for the table. If no items have been ordered, the error message

```
No items to remove.
```

must be printed and the program returns to the Add/Remove Item menu. If the user selects an invalid choice the Enter Choice> prompt is displayed again without printing the menu. The number of dashes below the menu **must** be the same as the widest menu item. Below is an example of the Remove Item menu where Appetizers and Entrée 1's have been entered.

```
1 Appetizer
2 Entree 1
D Done
-----------
Enter Choice>
```

If Done is selected with 'D' the program returns to the Add/Remove Items menu. If a valid food item number is selected, the user is prompted to enter the number of items to remove and shown a valid range. Below is an example of a prompt for removing Appetizers when 2 have already been ordered.

```
Remove how many Appetizer's [0 - 2]>
```

If the user enters a number less than zero the error message

```
Invalid must be at least 0.
```

must be printed and the prompt is shown again. If the user enters a number greater than the max, the error message stating that the must that can be removed is printed. Below is an example of a max of 2.

```
Invalid must be at most 2.
```

Once a valid number is entered the specified number of items is removed from the order. If no items remain in the order, the program returns to the Add/Remove Item menu.

e. Selecting Print Bill with ‘P’ will prompt the user to select a table from the occupied tables. Below shows an example of all tables being occupied. If no table is occupied, the message:

```
No orders started yet!
```

must be printed and the program will return to the main menu.

```
1 Table 1
2 Table 2
3 Table 3
4 Table 4
5 Table 5
---------
Select a Table>
```

If a table is currently available it must not appear in the list. Below is an example of Table 3 & 4 being available:

```
1 Table 1
2 Table 2
5 Table 5
---------
Select a Table>
```

If the user selects a table number that does is not occupied, the Select a Table> prompt is shown again without printing the list of tables. Once a valid table is selected, the bill for the table is printed. It **MUST** follow the format below. The table number is printed followed by a line of 36 dashes, then the items ordered, the count if more than 1 and the price plus the item total is displayed one on each subsequent line. A line with only a colon is displayed, followed by the Subtotal, Tax 8% (this must be rounded down to the nearest cent), and the Total followed by another line of 36 dashes. Below that is the suggested tip of 10, 15, and 20% followed by 36 dashes. All names of food items must be left aligned, the number ordered must be left aligned with x at location 14, the @ symbol must be aligned at location 19, the first $ symbol must be aligned at location 21, the decimal point at location 24, the colon at location 27, the second $ symbol at location 29, and the final decimal point at location 34. Below is an example for Table 1 with 1 Appetizer, 1 Salad, 9 Entrée 1’s, and 3 Desserts.

```
Bill for Table 1
------------------------------------
1. Appetizer      @ $ 5.00: $   5.00
2. Salad          @ $ 3.00: $   3.00
3. Entree 1  x9   @ $10.00: $  90.00
4. Dessert   x3   @ $ 4.25: $  12.75
                          :
Subtotal                  : $ 110.75
Tax 8%                    : $   8.86
Total                     : $ 119.61
------------------------------------
Suggested Tip
Tip 10%                   : $  11.07
Tip 15%                   : $  16.61
Tip 20%                   : $  22.15
------------------------------------
```

If no items have been ordered the line item should state that no items have been ordered. Below is an example for Table 2 where no items have been ordered.

```
Bill for Table 2
------------------------------------
No Items Purchased        : $   0.00
                          :
Subtotal                  : $   0.00
Tax 8%                    : $   0.00
Total                     : $   0.00
------------------------------------
Suggested Tip
Tip 10%                   : $   0.00
Tip 15%                   : $   0.00
Tip 20%                   : $   0.00
------------------------------------
```

Once the bill has been printed out the program returns to the main menu.

f. Selecting Receive Payment with 'R' will prompt the user to select a table from the occupied tables. Below shows an example of all tables being occupied. If no table is occupied, the message:

```
No orders started yet!
```

must be printed and the program will return to the main menu.

```
1 Table 1
2 Table 2
3 Table 3
4 Table 4
5 Table 5
---------
Select a Table>
```

If a table is currently available it must not appear in the list. Below is an example of Table 3 & 4 being available:

```
1 Table 1
2 Table 2
5 Table 5
---------
Select a Table>
```

If the user selects a table number that does is not occupied, the Select a Table> prompt is shown again without printing the list of tables. Once a valid table is selected, the user is prompted to enter the amount of payment for the table. Below is an example of the payment prompt for Table 1.

```
Enter payment for Table 1>
```

If the user enters an amount less than the bill total, the error message specifying that the payment must be at least the amount of the total is displayed and the user is prompted again to enter the payment amount. Below is an example where the total is $119.61.

```
Invalid must be at least 119.61.
```

Once the payment is entered the receipt is printed out and the program returns to the main menu, all information about the sale is added to the restraint totals, and the table becomes available again. The receipt has a similar format as the bill, but shows payment and excludes suggested tip. Below is an example of $135.00 is paid for the bill in the previous example.

```
Receipt for Table 1
------------------------------------
1. Appetizer      @ $ 5.00: $   5.00
2. Salad          @ $ 3.00: $   3.00
3. Entree 1  x9   @ $10.00: $  90.00
4. Dessert   x3   @ $ 4.25: $  12.75
                          :
Subtotal                  : $ 110.75
Tax 8%                    : $   8.86
Total                     : $ 119.61
                          :
Payment                   : $ 135.00
Change                    : $  15.39
------------------------------------
```

g. Selecting Manager Report with ‘M’ will print out the manager’s report that shows the list totals for the restaurant. The formatting for the report is similar to the bill and receipt except the totals for the restaurant are shown and the number of items is shown even if only one is sold. Below is an example of the totals where 1 Appetizer, 1 Salad, 9 Entrée 1’s and 3 Desserts have been sold.

```
Daily Totals
------------------------------------
1. Appetizer x1   @ $ 5.00: $   5.00
2. Salad     x1   @ $ 3.00: $   3.00
3. Entree 1  x9   @ $10.00: $  90.00
4. Dessert   x3   @ $ 4.25: $  12.75
                          :
Gross                     : $ 110.75
Tax 8%                    : $   8.86
Register                  : $ 119.61
------------------------------------
```

If no items have been sold, the No Items Sold message is shown in the list. Below is an example of no items being sold.

```
Daily Totals
------------------------------------
No Items Sold             : $   0.00
                          :
Gross                     : $   0.00
Tax 8%                    : $   0.00
Register                  : $   0.00
------------------------------------
```

Once printed the program returns to the main menu.

h. Selecting Quit with ‘Q’ will exit the program.