

## Computer Project #2

### Assignment Overview

This assignment focuses on the design, implementation and testing of a Python program which uses control structures to solve the problem described below.

It is worth 20 points (2% of course grade) and must be completed no later than 11:59 PM on Monday, September 16.

### Assignment Specifications

1. The program will compute and display information for a utility company which supplies water to its customers. For a specified customer, the program will compute and display the amount of money which the customer will be billed for water usage during the current billing period.
2. The program will prompt the user to enter three values (in the following order):
  - a. The customer's code (a character)
  - b. The customer's beginning meter reading (a positive integer value)
  - c. The customer's ending meter reading (a positive integer value)

It will then process that customer information and display the results. The program will continue to process customers until the user enters an invalid customer code.

3. The program will compute the gallons of water used by the customer during the current billing period.

The meter is read by a representative of the utility company at the start and at the end of the billing period, and the readings are taken from a meter which has nine digits and records tenths of a gallon.

4. The program will compute the amount of money that the customer will be billed, based on the customer's code and water usage, using the following information.

Code 'r' (residential):

\$5.00 plus \$0.0005 per gallon used

Code 'c' (commercial):

\$1000.00 for 4 million gallons or less, and \$0.00025 for each additional gallon used

Code 'i' (industrial):

\$1000.00 if usage does not exceed 4 million gallons; \$2000.00 if usage exceeds 4 million gallons but does not exceed 10 million gallons; and \$2000.00 plus \$0.00025 for each additional gallon if usage exceeds 10 million gallons.

5. For each customer, the program will display a summary with the following information:

- a. The customer's code
- b. The customer's beginning meter reading
- c. The customer's ending meter reading
- d. The gallons of water used by the customer
- e. The amount of money billed to the customer

All output will be appropriately labeled and formatted.

### Assignment Deliverables

The deliverable for this assignment is the following file:

proj02.py – the source code for your Python program

Be sure to use the specified file name (“proj02.py”) and to submit it for grading via the **handin** system before the project deadline.

### Assignment Notes

1. As stated above, the meter's dial has nine digits and records tenths of a gallon. For example, assuming that the beginning reading was 444400003 and the ending reading was 444400135, then the customer used 13.2 gallons of water during the billing period.
2. Since the meter's dial only has nine digits, the reading at the end of the billing period may be less than the reading at the beginning of the billing period. For example, assuming that the beginning reading was 999999997 and the ending reading was 000000005, then the customer used 0.8 gallons of water during the billing period.
3. The program is not required to check the user-supplied meter readings for errors. That is, you may assume that the user always enters valid meter readings (integer values between 0 and 999999999).
4. The amount of money billed to a customer should be displayed as a monetary value. That is, it should be displayed with a dollar sign and two fractional digits (for example, \$125.00 or \$43.87). Information about formatting output is on pages 201-204 of the textbook.

### Commentary

- To clarify the project specifications, we've provided sample output at the end of this document.
- Be sure to prompt the user for the three inputs in the correct order. And, your program cannot prompt the user for any other inputs. To grade your program, your TA will enter a series of inputs – and it will be the same series of inputs for each student's program.
- It isn't necessary for the user to type leading zeroes when entering meter readings.
- Similarly, it isn't necessary for the program to print leading zeroes when displaying meter readings.

## Suggestions

1. *Solve the problem using pencil and paper first.* You cannot write a program until you have figured out how to solve the problem. This first step is best done collaboratively with another student. (However, once the discussion turns to Python specifics and the subsequent writing of Python, you must work on your own.)
2. Use IDLE to create a new program:
  - a. Use the required file name (“proj02.py”).
  - b. If you are in a CSE lab, select the H: drive as the location to store your file.
3. Write a simple version of the program (perhaps one which repeatedly prompts the user for a customer code and displays it).
4. Run the program and track down any errors.
5. Use the **handin** system to turn in the first version of your program.
6. Cycle through the steps to incrementally develop your program:
  - a. Edit your program to add new capabilities.
  - b. Run the program and fix any errors.
7. Use the **handin** system to submit your final version.
8. Be sure to log out when you leave the room, if you’re working in a public lab.

*Be sure to save a copy of your completed program in your CSE file space (H: drive on the lab machines) **before** the project deadline. If you write your program at home and turn it in from home, you will need to copy it to your CSE file space **before** the deadline. In case of problems with electronic submission, an archived copy on the CSE disk is the only acceptable evidence of completion.*

## Sample Output

```
>>> ===== RESTART =====
>>>

Enter customer code: r
Enter beginning meter reading: 444400003
Enter ending meter reading:    444400135

Customer code: r
Beginning meter reading: 444400003
Ending meter reading:    444400135
Gallons of water used: 13.2
Amount billed: $5.01

Enter customer code: c
Enter beginning meter reading: 999999997
Enter ending meter reading:    000000005

Customer code: c
Beginning meter reading: 999999997
Ending meter reading:    5
Gallons of water used: 0.8
Amount billed: $1000.00

Enter customer code: z
>>>
```

## Optional Extensions

Your program is only required to meet the specifications given above. However, there are several optional extensions to the project that you should consider adding to your program:

1. Recognize upper case letters for customer codes ('R', 'C' and 'I').
2. Display all meter readings as nine-digit numbers (with leading zeroes).
3. Trap invalid user-supplied meter readings.

The last item deserves some discussion. Robust programs should detect, report and recover from invalid user inputs. That is, a robust program will check user-supplied inputs to determine if they are valid or not, will report any problems to the user, and will take the appropriate steps to continue with execution.

A robust version of this program would detect invalid meter readings (values outside the range 0 to 999999999), report them to the user, and allow the user to continue with the next customer. Please note that the user should always enter three items (code and two meter readings), even if the first meter reading is invalid.