1. Obstacles

I initially was not sure how to only output the first error encountered and end the program there. Then I realized that since the code executes top to bottom, if I checked for each error right after reading in that specific data, I could catch the errors in order. I also realized that I could end the program if I included a return statement if the error was true. Additionally, I found it hard to keep track of the arithmetic within the nested ifs when calculating the bill. I remembered it is good practice for myself and anyone else reading my code to make the different rates constants rather than typing the specific numbers.

1. Test Data

The initial meter reading is a negative integer (-50)

The final meter reading is less than the initial reading (0, -50)

An empty string is provided for customer name (0, 50, )

The month number is 0 (0, 50, Vi, 0)

The month number is a negative integer (0, 50, Vi, -3)

The month number is an integer larger than 12 (0, 50, Vi, 13)

High season, HCF is 0 (50, 50, Vi, 4) 0

High season, HCF is well within than first tier (50, 65, Vi, 5) 81.15

High season, HCF is one less than first tier (50, 72, Vi, 7) 119.02

High season, HCF is cutoff for first tier (50, 73, Vi, 9) 124.43

High season, HCF is one more than first tier (50, 74, Vi, 10) 134.22

High season, HCF is well within second tier (50, 110, Vi, 6) 486.66

Low season, HCF is 0 (50, 50, Vi, 11) 0

Low season, HCF is well within than first tier (50, 59, Vi, 12) 48.69

Low season, HCF is one less than first tier (50, 64, Vi, 1) 75.74

Low season, HCF is cutoff for first tier (50, 65, Vi, 2) 81.15

Low season, HCF is one more than first tier (50, 66, Vi, 3) 88.92

Low season, HCF is well within second tier (50, 97, Vi, 3) 329.79

The bill has 0 as second digit after decimal point (305, 315, Vi, 6) 54.10

Customer name has multiple words and spaces (0, 50, Caitlyn Kim Merman, 4)