

Assignment 4

12 November 2012

Due Monday 12 November by class time, but via email.

For this assignment, you will write and test a short Python program that computes and displays a store receipt according to this set of rules:

1. The customer purchases exactly 3 items.
2. Any single item that costs over \$10 is subject to a 5% sales tax.
3. If the total cost of the 3 items (including any taxes) is over \$25, then the store offers a 10% discount.
4. The receipt must display the taxable and non-taxable subtotals, the tax amount, and the discount amount, if any.
5. Amounts on the receipt do **not** need to be rounded to the nearest cent, padded to two decimal places, or have decimal points aligned.

Below are some sample runs of the program. Your solution should work for these inputs, but also for others according to the rules above. User input is underlined.

Sample 1

Sample 2

Sample 3

Notes and hints

1. Start with the simplest pieces of the functionality you can, and get them to work before adding more. For example, just ask for the price of *one* item, and determine whether or not it is taxable. Run and test that program before moving on.
2. We need the user to input numbers with decimals, which are called ‘floats’ in most programming languages. Recall the syntax of the input statement. The *prompt* is inside the quotes. The variable `price1` will then contain whatever number the user typed.

```
price1 = float(raw_input("Enter price of item 1: "))
```

3. The problem description refers to percentages, such as a 5% tax rate, and a 10% discount. You cannot use the percent sign directly in Python, instead we have to divide by 100 and use that decimal number. So 5% is written as 0.05, as in:

```
tax = taxable_sum * 0.05
```

4. Try to use sensible variable names, such as `tax`, `total`, `discount`, etc. Names cannot have spaces in them – instead we use the ‘underscore’ character, as in `nontax_sum`.
5. Your program will use several conditional statements. The syntax is shown by example below. Don’t forget the colon at the end of the `if` line, and use indentation (tab key) to show which statements are controlled by the condition:

```
if price1 > 10:
    taxable_sum = taxable_sum + price1
else:
    print " Item 1 $", price1
```

Email your `.py` file to me as an attachment by class time on Monday 12 November.