Assignment P3

Due Date: March 20

Purpose
The Python, as well as the dooflingie, adventure continues! In this assignment, you will use Python’s selection statements in an almost real kind of program. You will also continue to use basic Python statements and work on your coding style.

Problem
The manager of the local Rinky Dinky Dooflingie outlet store must implement a new coding scheme to price dooflingies. Because you did such a fine job with the first project, she has hired you to write a program to simulate the workings of her new cash registers so she can train her staff before the holiday rush. Dooflingies are expected to be the hit of the holiday season!

Input
Your program should prompt the user for the following items:

- The character price code of the dooflingie (see section on Specifics, below).
- Whether or not there is a discount (a character; 'y' or 'Y' means yes; anything else means no).
- If there is a discount, the discount percentage (an integer; i.e., if the discount is 20%, the user would type in 20).
- The amount of money tendered by the customer (a float; assume that it will always be enough). Note that this will be input before you see the total (it’s only a simulation, after all).

Output
Your program should produce a neat, aligned sales slip similar to the following (price code C and 20% discount entered):

```
Price (Code C)................. 20.99
Less discount (20%)........... 4.20
MA Tax (6.25%)................ 1.05

TOTAL......................... 17.84

Amount tendered............... 20.00
Change due..................... 2.16

Thank You! Have a Dooflingie Day!
```

If there is no discount, the discount line should not be printed. If the price code is incorrect, an error message should be printed instead of the sales slip. Although the form does not have to be identical, you must display all of the items shown in the sales slip above; the price code as well as the discount percentage must be shown within a line. Good intra- and inter-line spacing is important; the neater it is, the better. All dollar values should be right justified, have two decimal places, and be aligned by the decimal point. Note that the tax must also be computed.
Specifics

- You must use both `else` and `elif` forms of selection.

- Both upper- and lower-case character input is acceptable. However, the sales slip should contain only upper-case price codes.

- Valid price codes are as follows (with associated prices and dooflingie type):

<table>
<thead>
<tr>
<th>Code</th>
<th>Price</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>24.99</td>
<td>Big Blue Dinkie Doo</td>
</tr>
<tr>
<td>B</td>
<td>22.99</td>
<td>Red Rinkie</td>
</tr>
<tr>
<td>C</td>
<td>20.99</td>
<td>Little Doo</td>
</tr>
<tr>
<td>D</td>
<td>10.99</td>
<td>Rinkie Dinkie Doohinkie</td>
</tr>
</tbody>
</table>

- You must worry about round-off error. Every multiplication or division computation involving floating point values can give you a result with many decimal places. You must round off to the nearest penny at each stage of your computations. You may use the `round()` function or you can use your own code for this. If you do use the `round()` function, note that it isn’t quite right and you have to add a little more code for it to work properly. Do not rely on output formatting, because that makes the output only look pretty; the computations will still have all of the decimal places and your final answer may not be correct.

- Use named constants for relevant items.

- Be sure to follow all style guidelines.

Notes

As with the last program, writing an algorithm before attempting to code the program will show you where some of the problems will arise (remember, however, that many details are left out of an algorithm). Or you may like to try out the idea of a flow chart as shown in the text; this can help you see where the selection statements will fit into your program. I also suggest that you code one portion of the program at a time, making sure that the newest piece you just typed in works properly before continuing.

As usual, send me an electronic copy of your source code (mgousie@wheatoncollege.edu) following the same file naming conventions as the previous projects. Submit by 11:59:59 PM on the due date for the project to be on time. Hand in a printout of your source code in class the next day.

*I love deadlines. I like the whooshing sound they make as they fly by.*

– Douglas Adams (1952-2001)