Programming with Numbers and Strings; Controlling Selection and Iteration

Your Supermarket Coupons – The Grocery Bill

**Understand the Application**

Each supermarket wants your repeat business.  To learn exactly how much, you can start by looking at the value of coupons that stand to be awarded by your local market.  For example, say your grocery bill usually is between $10 and $60 dollars.  Let’s say for today that it is $20.  That means for spending $20 today your market will entice your return visit by offering you a discount coupon that can be applied to your next grocery shopping bill.  Of course, in order to encourage increased grocery spending the amount of the discount awarded will be based on the amount of groceries purchased.

**The Program Spec**

Your program will prompt the user for the cost of their groceries.

Then you are to calculate the value of the coupon to be awarded based on the amount of the grocery bill.  Use the table below to determine coupon amounts.

**Money Spent                                               Coupon Percentage**

Less than $10                                              No coupon

From $10 to less than $60                          8%

From $60 to less than $150                       10%

From $150 to less than $210                     12%

From $210 or more                                      14%

Display an appropriate result message to the user including both the coupon percentage awarded and the dollar amount of their discount coupon.

**Deliverable:  yournameLab2.py**  Your source code solution and a copy of the run pasted into your source submission file.  Be sure to comment out your run so that your **.py**file will still run in the grader test bed.

**Input Error Checking:**  You can assume that when it is time for the user to enter a number to represent the amount of his/her day’s grocery bill that a valid numeric value will be typed in.  You do not need to test for non-numeric nor negative values.  Note:  A numeric 0 input is a valid data entry.

**Iteration Control:**  Use a repetition structure to generate your test cases in one program run.

Here are some other requirements:

1. Create constants versus using literals in your source code.
2. Use a float to represent both the shopper’s grocery bill as well as the coupon amount.
3. Format the discount coupon amount to 2 decimal places (Example:  $19.20).
4. Show at least five different test cases, supplying five different sets of values that the user enters each time.  Have each test demonstrate one of the five coupon percentages awarded.

Here is a sample run (remember you’ll need at least five test cases):

…

Please enter the cost of your groceries: 160.01

You win a discount coupon of $19.20. (12% of your purchase)

…..