This is the first program we are going to look at. The next three are about Faculty and are located in loopscomboetc.zip rather than in a subfolder.
February 17, 2015

First was 3.80, so 3.80 + 5.70 = 9.50 and since there were two orders the average is 4.75.

Loading into the CheckBoxList when the form is loaded.

Looking at btnCalc_Click we see a For loop that establishes ct as an index to be used to see what muffins are checked.

cbMuffins.Items.Count is 5. We want ct to go from 0 to 4 because indexes start with 0 and I am using ct as an index. So since there are 5 things and I am using 0,1,2,3,4 as the index, I need to subtract 1 from the count to get 4.

Inside the loop we find out whether the muffin was checked. If it was I do the calc and add to wkAmtOwe.
I did not set up a collection, instead in this example I assigned the data to the CheckedListBox in the form_load.

Collection is an alternative approach.
To verify that the count is 5.
If Blueberry and Cranberry are checked and I want 2 of each, then for Blueberry I have 2 times .95 and I add that to wkAmtOwe. And then Cranberry has 2 times .95 and then add that to wkAmtOwe.

Once I have wkAmtOwe I check to see if the discount radio button is checked which will reduce wkAmtOwe.
If `wkAmtOwe` is greater than `wkHighestSale` then it will replace `wkHighestSale`. On the first sale of the day `wkAmtOwe` is going to replace the 0 in `wkHighestSale`. If `wkAmtOwe` is not greater than `wkHighestSale` it will stay as it is.

Add `wkAmtOwe` to `wkTotal` and add 1 to `wkCustCt`.

- `wkTotal`: 3.80
- `wkCustCt`: 1
And the next one is 1.90 so I will add that.
And the first one was 3.80 so 15.96 but apparently I messed up and clicked again because the average comes out wrong. I redid later and it worked right!
Which gives me an average of 5.32 which is correct!
February 17, 2015

I have no idea what I clicked to get this average. In a few slides you will see it corrected.
The code!
I forgot to mention the for loop in class. I am checking to see if something is checked and if it is I am changing the state to Unchecked.

If wkCustCt > 0 which means there was a customer, I will calculate the average, and then format it. Then I put together a message including literals, edited data and the wkNewLine. Note the _ because I wrote on multiple lines.

Once the message has been set up I will write it using an OKOnly style (earlier version of msgbox). If there were no customers the no data to summarize message is written.
The calculation for the first order was 3.80, for the second order 10.26 and for the third order 1.90 (just like I thought I did before) and this time I am getting the correct average of 5.32.

As I process I keep track of and display wkTotal and wkCustCt.

As I do the summary, I display the wkTotal and wkCustCt to make sure that I am following the data. Great way to debug!
Now I am starting to look at the Faculty programs. This is FacCrsSetup.

Add courses to the list box.

When a different semester is clicked a line is written in the list box and the Add button is enabled.

I am adding the items to the two combo boxes. This is done in the load and note the Add button is not enabled.

Resetting the combo boxes.
This ComboBox is of the DropDownStyle.
The Combo box is simple style.
After I select a semester.
I selected from the Combo box and then clicked Add Course to get it added to the List Box.
Now I am going to use Items collection to load the courses and the semesters. So the only thing I did in the Load is set the enabled of the add button to false. Notice the ResetText()
The third version uses InputBox to let the user change the data each time the program is run.

There is an initializing input and then the loop to add to the combo boxes. The last statement in the loop takes in another course to add or the word Done to end the loop.

There are separate loops for semester and course combo boxes.
This shows the while and the until with the condition being asked before entering the loop meaning the loop may not be entered or the condition being asked at the end of the loop which means the loop will always be run at least once.

While tests what keeps you in and until tests what gets you out.
The For loop.