09 Prove: Assignment Milestone

Regarding Milestones and Assignments: Just as with the previous lessons, the Prove assignments for Lessons 09 and 10 will be working toward completing the same assignment. At the end of Lesson 09, you will complete part of the overall program, and then finish it in Lesson 10.

Read over the full requirements first. Then, at the bottom of this page, you'll see which features are required for the milestone at the end of Lesson 09.

Overview

Many apps use some type of shopping cart, where users can add items to the cart, remove items, view the contents of the cart, and see the total. Storing this collection of items requires keeping track of a list of items and updating it in various ways.

Project Description

For this project you will create a program that stores a list of products in a shopping cart along with their prices. The user should have the ability to add items to the list, remove them, and see the total price of the cart.

For the milestone deliverable, you only need to worry about storing a list of the names of the items (not the prices yet), and only need to be able to add new items and display the list. Then, for the complete project, you'll add the ability to store the prices, remove items, and compute the total.

Assignment

For this project, the user will be given a menu and have the ability to choose items from the menu. The options in the menu include the following:

01. Add a new item
02. Display the contents of the shopping cart
03. Remove an item (only needed for the final project deliverable)
04. Compute the total (only needed for the final project deliverable)
05. Quit

When the user chooses one of these options, the program should perform the appropriate action. Then the program should show them the menu again, and allow them to choose another option. It should continue running until the user choose the option to quit.

The following sections outline the expected behavior of each option:

I. ADD A NEW ITEM

The program asks the user for the name of the item. For the final project deliverable it should also ask the user for the price of the item.

The program stores these values in a list. When you are working on the milestone deliverable, you will only need one list (to store the names). For the final project, when you are storing both names and prices, you should use two lists, one for the names and one for the prices.

An example of the functionality of this item could look as follows:

```
What item would you like to add? socks
What is the price of 'socks'? 5.99
'socks' has been added to the cart.
```

After the user provides these values, the item and its price are stored in the list.

Hint: You should create new blank lists at the beginning of the program, so that when you get to this point, you'll have a list already in place in which you can store the name or price.

II. DISPLAY THE CONTENTS OF THE SHOPPING CART

The program should display all of the items in the shopping cart, one per line. For the milestone deliverable, you only need to display the names of the items. For the final project deliverable, the price of each item should be displayed next to the item.

Also, for the final project, the program should also display the number associated with each item in the list, beginning with 1. This is different than how Python will store them, because it will start counting at 0, but to make it more natural for the user, you should start the numbers at 1.

An example of the functionality of this item could look as follows:

```
The contents of the shopping cart are:
1. bed - $100.00
2. chair - $24.99
3. blanket - $48.50
```

III. REMOVE AN ITEM

The user types in the number of the item they want to remove and the item is removed from the list.

In order to do this correctly, your program will need to do each of the following:

01. Convert the number the user enters to a 0-based index. For example, if they want to remove the second item in the list they will type "2" (because they will see the numbers starting at 1), but this item will be stored in Python at index 1 (because Python will start counting at 0).

02. Make sure that the index is within the bounds of the current list (for example, you cannot remove item 10 if there are only 5 items in the list). If the index is outside the bounds of the list, you should not attempt to remove it, but rather, display a message informing the user that they have made an invalid choice.

03. Remove the item at that spot in the list. Don't forget that if you are storing names and prices, you'll need to remove both the name and the price from their respective lists.

The following shows an example of this functionality assuming there are 5 items in the list:

```
Which item would you like to remove? 3
Item removed.
```

```
Which item would you like to remove? 13
Sorry, that is not a valid item number.
```

IV. COMPUTE THE TOTAL

The program should iterate through each item in the list and add up the prices and then display the total amount to the user.

The following shows the expected result, assuming the total amount of the items in the shopping cart is $35.68

```
The total price of the items in the shopping cart is $35.68
```

Milestone Requirements

At the end of Lesson 09, to help make sure you are on track to finish the assignment, you need to complete the following:

01. Have menu system that repeats until the user chooses quit.
02. Create a list that will store the names of the items in the shopping cart.
03. Complete the option to add the name of the item to the list.
04. Complete the option to display the names of the items in the list.

The following demonstrates the functionality of the program at this point:

```
Welcome to the Shopping Cart Program!

Please select one of the following:
1. Add item
2. View cart
3. Remove item
4. Compute total
5. Quit
Please enter an action: 1
What item would you like to add? bed
'bed' has been added to the cart.

Please select one of the following:
1. Add item
2. View cart
3. Remove item
4. Compute total
5. Quit
Please enter an action: 1
What item would you like to add? chair
'chair' has been added to the cart.

Please select one of the following:
1. Add item
2. View cart
3. Remove item
4. Compute total
5. Quit
Please enter an action: 1
What item would you like to add? blanket
'blanket' has been added to the cart.

Please select one of the following:
```

```
1. Add item
2. View cart
3. Remove item
4. Compute total
5. Quit
Please enter an action: 2
The contents of the shopping cart are:
bed
chair
blanket

Please select one of the following:
1. Add item
2. View cart
3. Remove item
4. Compute total
5. Quit
Please enter an action: 5
Thank you. Goodbye.
```

Final Requirements

In addition to the functionality above, for the final project, you also need to complete the following:

01. Store prices as well as names.

02. Change the add functionality to also ask for and store the price of the item.

03. Change the display functionality to also display the prices of the items.

04. When displaying the items, display a number in front of each item. The numbers should start with 1.

05. Complete the option to display the total amount of the prices of all the items in the shopping cart.

06. Whenever prices are displayed, they should be shown to two decimal places.

07. Complete the option to remove an item from the shopping cart.

08. When removing an item, you should verify the following:

    » Both the item name and price are removed from their respective lists.

> » The number the user enters should be converted to a 0-based index and checked to make sure it is within the bounds of the list.
>
> » The program allows you to remove any item from the list including the first one and the last one. (Sometimes, if you have a bug in your project you might not allow removing the last item as you should.)

The following shows an example of the complete program:

```
Welcome to the Shopping Cart Program!

Please select one of the following:
1. Add item
2. View cart
3. Remove item
4. Compute total
5. Quit
Please enter an action: 1
What item would you like to add? Milk
What is the price of 'Milk'? 3.49
'Milk' has been added to the cart.

Please select one of the following:
1. Add item
2. View cart
3. Remove item
4. Compute total
5. Quit
Please enter an action: 1
What item would you like to add? Bread
What is the price of 'Bread'? 2.50
'Bread' has been added to the cart.

Please select one of the following:
1. Add item
2. View cart
3. Remove item
4. Compute total
5. Quit
Please enter an action: 1
What item would you like to add? Butter
What is the price of 'Butter'? 4.00
'Butter' has been added to the cart.

Please select one of the following:
1. Add item
2. View cart
3. Remove item
4. Compute total
```

```
5. Quit
Please enter an action: 2
The contents of the shopping cart are:
1. Milk - $3.49
2. Bread - $2.50
3. Butter - $4.00

Please select one of the following:
1. Add item
2. View cart
3. Remove item
4. Compute total
5. Quit
Please enter an action: 3
Which item would you like to remove? 2
Item removed.
Please select one of the following:
1. Add item
2. View cart
3. Remove item
4. Compute total
5. Quit
Please enter an action: 2
The contents of the shopping cart are:
1. Milk - $3.49
2. Butter - $4.00

Please select one of the following:
1. Add item
2. View cart
3. Remove item
4. Compute total
5. Quit
Please enter an action: 4
The total price of the items in the shopping cart is $7.49

Please select one of the following:
1. Add item
2. View cart
3. Remove item
4. Compute total
5. Quit
Please enter an action: 5
Thank you. Goodbye.
```

Make it your own

In addition to the functionality listed above, add something of your own creativity to the assignment. You could consider adding better formatting (for example, aligning the prices), storing extra information, such as the quantity of the items in the list, or you can add anything else you think of to make the shopping cart better.

Submission

When the the Milestone is complete, answer the questions in the associated quiz in I-learn.

When the final project is complete, upload your code to I-learn.