

Shopping Cart Program

[Start Assignment](#)

Due Monday by 11:59pm **Points** 100 **Submitting** a file upload

Implement a Shopping Cart program in Python. The program should allow users to add products to their shopping cart, remove products, update quantities, view the list of items, set a tax rate, and calculate the total cost and tax. The program should be menu-driven and use variables, assignments, data types, conditional statements, classes, loops, and lists.

Requirements:

- 1. Product Class:** Create a `Product` class to represent individual products. Each product should have the following attributes:
 - Name
 - Price per unit
 - Quantity
- 2. ShoppingCart Class:** Create a `ShoppingCart` class to manage the shopping cart. This class should have the following functionalities:
 - Add a product to the cart.
 - Remove a product from the cart by name.
 - Update the quantity of a product in the cart.
 - Calculate the subtotal (sum of product prices * quantities).
 - Set and store the tax rate for the cart.
 - Calculate the total tax (subtotal * tax rate).
 - Calculate the total cost (subtotal + total tax).
 - Display the list of products in the cart with their names, prices, and quantities.
- 3. Menu-Driven Interface:** Implement a menu-driven interface that allows users to perform the following actions:
 - Add a product to the cart (enter name, price, and quantity).
 - Remove a product from the cart by entering its name.
 - Update the quantity of a product in the cart (enter the name and new quantity).
 - Display all items in the cart.
 - Show the total cost of items in the cart.
 - Set the tax rate (prompt the user to enter the tax rate as a decimal, e.g., 0.08 for 8% tax).
 - Show the total tax amount.
 - Quit the program.

4. **Unit Test:** Write four new unit tests to verify the functionality of your `ShoppingCart` class. Test cases should cover adding, removing, updating products, and calculating totals. Your code should also pass the attached Unit Tests.

Submission Guidelines:

1. Submit your python code including unit test code and a screenshot of the unit test results

Use the following code template if needed

[ShoppingCartTemplate-1.py](https://uvu.instructure.com/courses/583149/files/118305849?wrap=1) (https://uvu.instructure.com/courses/583149/files/118305849?wrap=1) 
(https://uvu.instructure.com/courses/583149/files/118305849/download?download_frd=1)

[test_shopping_cart-1.py](https://uvu.instructure.com/courses/583149/files/118305848?wrap=1) (https://uvu.instructure.com/courses/583149/files/118305848?wrap=1) 
(https://uvu.instructure.com/courses/583149/files/118305848/download?download_frd=1)

Shopping Cart Program				
Criteria	Ratings			Pts
Code passes attached Unit Tests	80 pts Code passes all Unit Tests	50 pts Coded passes at least 50% of the Unit Tests	0 pts No Marks	80 pts
Code introduced at least 4 new Unit Tests	20 pts Full Marks	10 pts Code introduced 2 new Unit Tests	0 pts No Marks	20 pts
				Total Points: 100