
Building a Simple Shopping Cart

Objective:

By completing this assignment, you will:

- Learn to use variables and user input.
 - Understand type casting and exception handling.
 - Format output cleanly using Python's `f-strings`.
-

Step-by-Step Instructions

Part 1: Set Up Your File

1. Create a new Python file named `shopping_cart.py` in your IDE (PyCharm or Visual Studio Code).
2. At the top of your file, write a **block comment** to include:
 - Your name.
 - The date.
 - A brief explanation of what the program does. Example:

```
"""
Author: Your Name
Date: YYYY-MM-DD

This program simulates a simple shopping cart.
Users can add items, specify quantities, and see the total cost.
The program uses exception handling to manage invalid inputs.
"""
```

Part 2: The Base Code

Start with this skeleton code:

```
print("Welcome to the Shopping Cart Program!")

# Step 1: Ask the user for the number of items
```

Part 3: Expand the Program

1. Add More User Input:

- create variable name price and quantity for 3 items
- Inside a loop, ask the user for the name, price, and quantity of each item.

- Use `try..except` to handle invalid inputs.
- Example:

2. Calculate Total Cost:

- After collecting all the items, calculate the total cost:

Bonus: Additional Features

1. Add an **option to restart the program**:

- At the end of the program, ask if the user wants to start over or quit.

```
restart = input("\nWould you like to shop again? (yes/no): ").lower()
if restart == "yes":
    # Code to restart
else:
    print("Thank you for shopping with us!")
```

2. Add a **discount feature**:

- If the total cost exceeds \$100, apply a 10% discount:

```
if total_cost > 100:
    discount = total_cost * 0.1
    total_cost -= discount
    print(f"\nYou saved ${discount:.2f} with a 10% discount!")
    print(f"Discounted Total: ${total_cost:.2f}")
```

Part 6: Documentation and Submission

1. At the top of your program, include:
 - Your name.
 - The date.
 - A description of the program's functionality.
2. Save your file as `shopping_cart.py`.
3. Upload the file to a new GitHub repository.

Grading Rubric

Criteria	Points
Proper use of variables	10
User input and type casting	10
Exception handling with <code>try..except</code>	15
Clear and formatted output	10
Code documentation and GitHub upload	10

Submission

1. Push your `shopping_cart.py` file to a GitHub repository.
2. Share the repository link as part of your lab 3 submission.