# **Python Case Studies**

1. **ATM Simulation System**

An ATM simulation that allows users to:

• Check balance

• Deposit money

• Withdraw money

• Exit

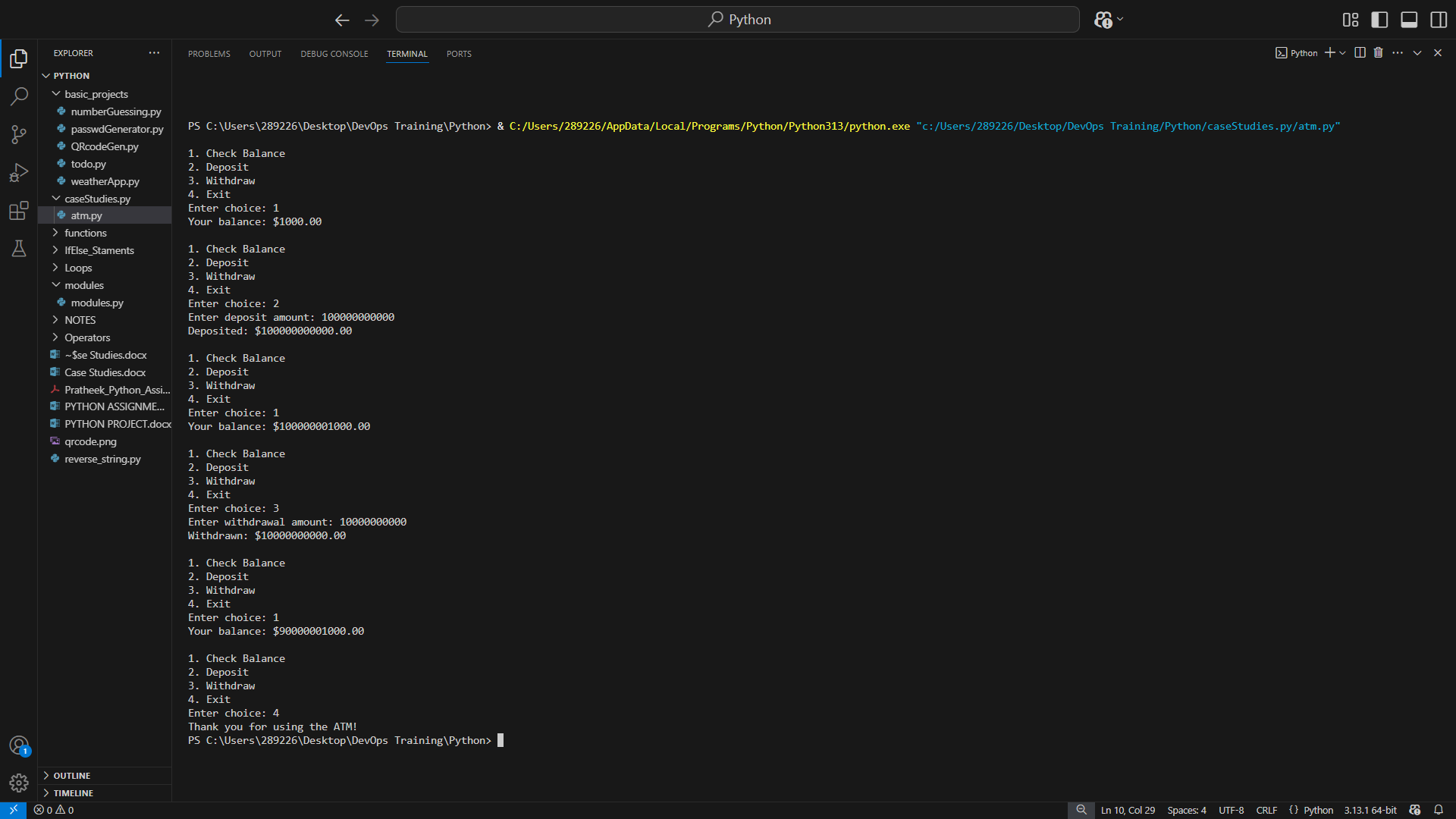
Steps to Solve

1. Define an initial balance.

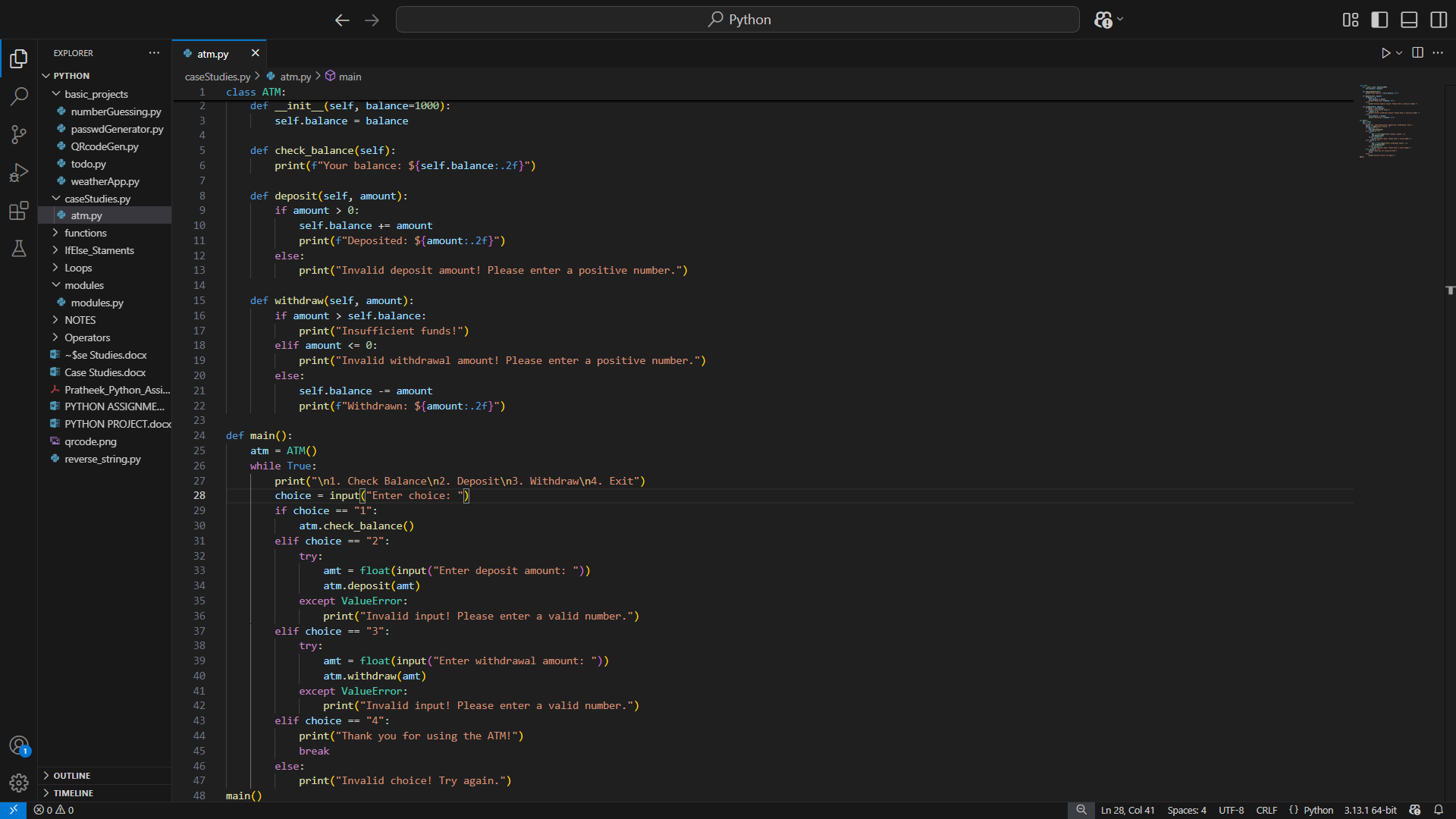
2. Create a menu-driven system to perform transactions.

3. Ensure withdrawal does not exceed balance.

4. Exit the program when the user chooses.

****

**Output :**



1. **E-commerce Order Management**

Order ManagementSystem for an e-commerce platform. The system should

allow:

• Adding products to a cart

• Viewing the cart

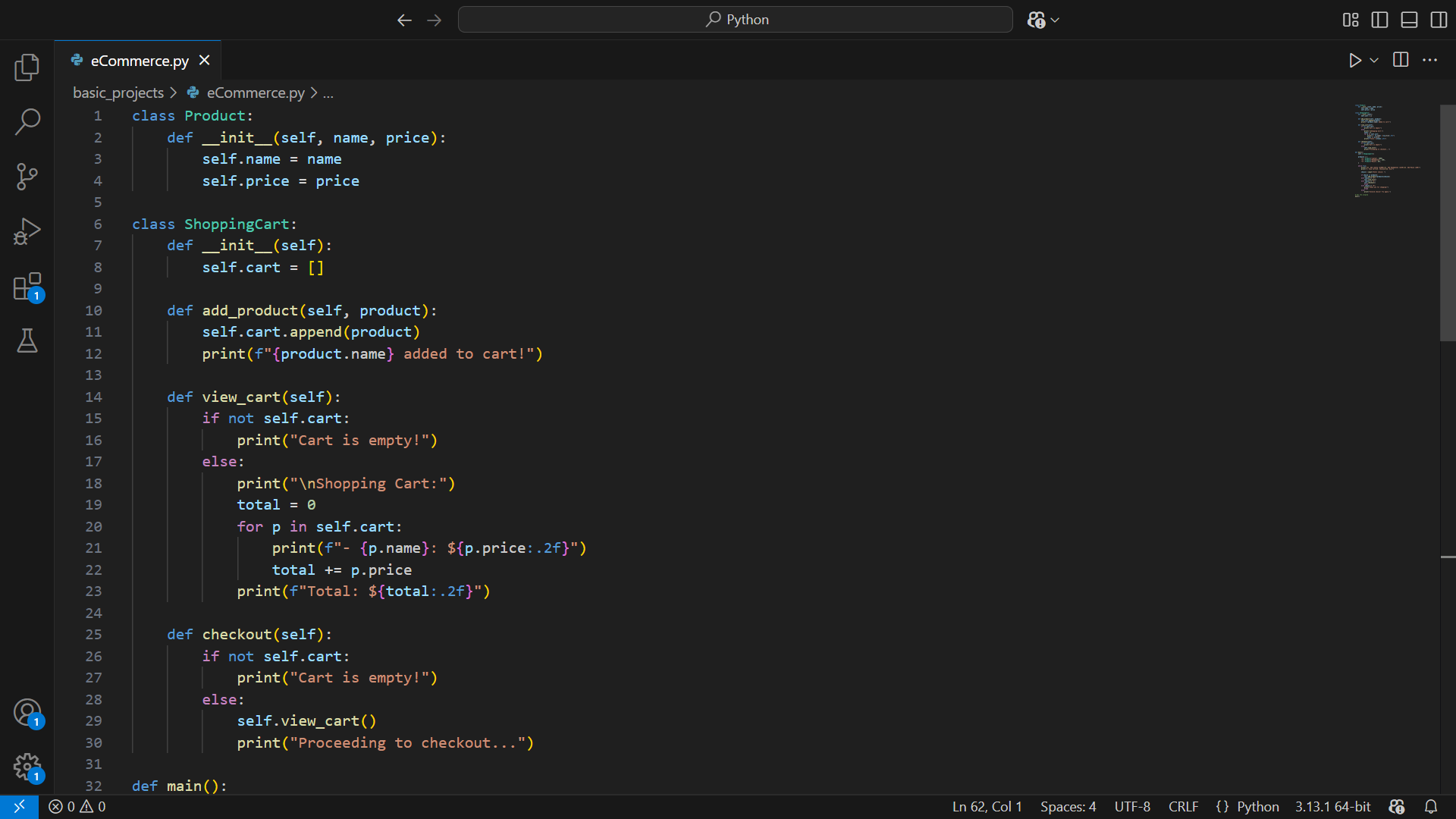
• Checking out (calculating total price)

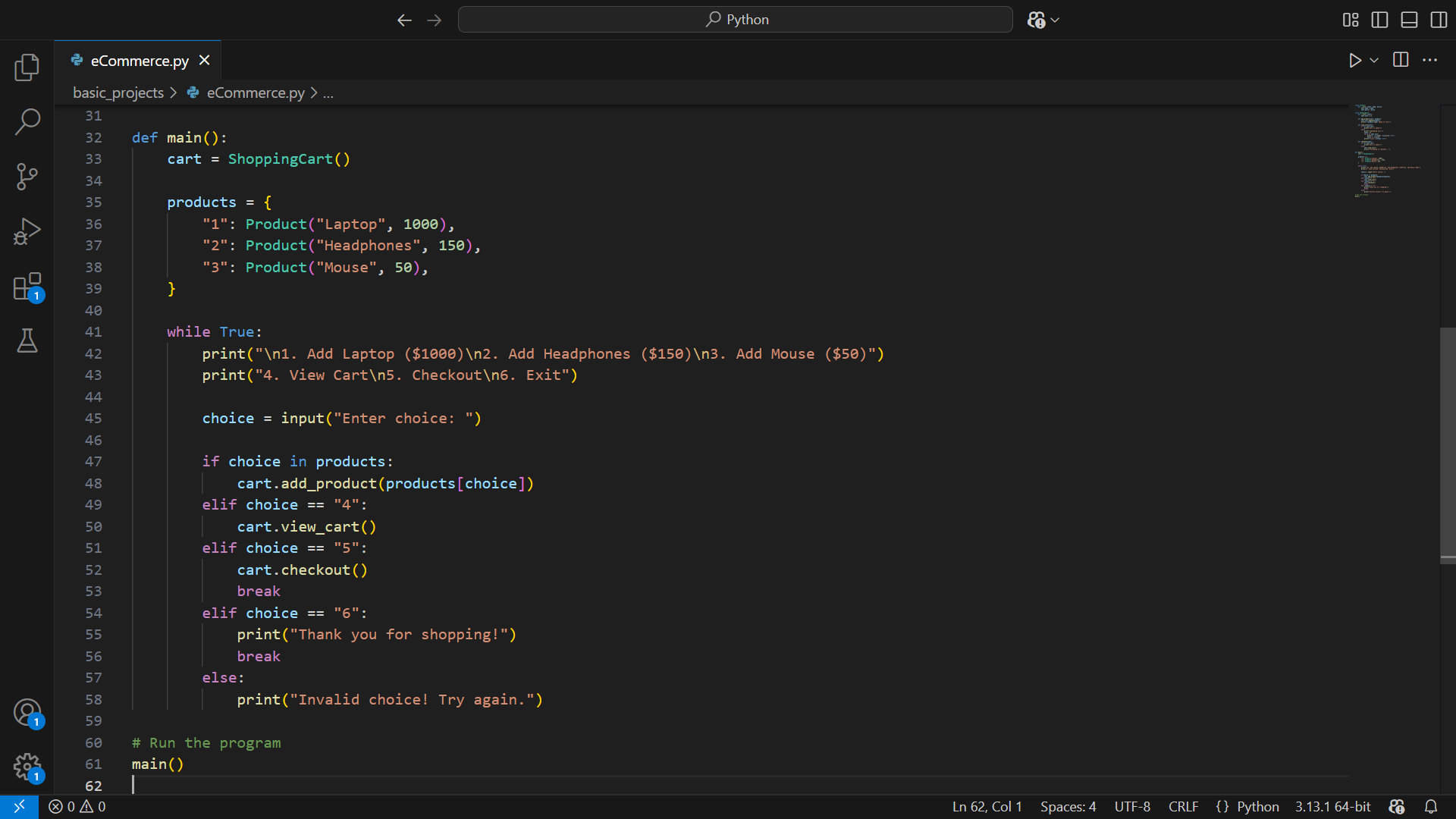
Steps to Solve

1. Define a Product class.

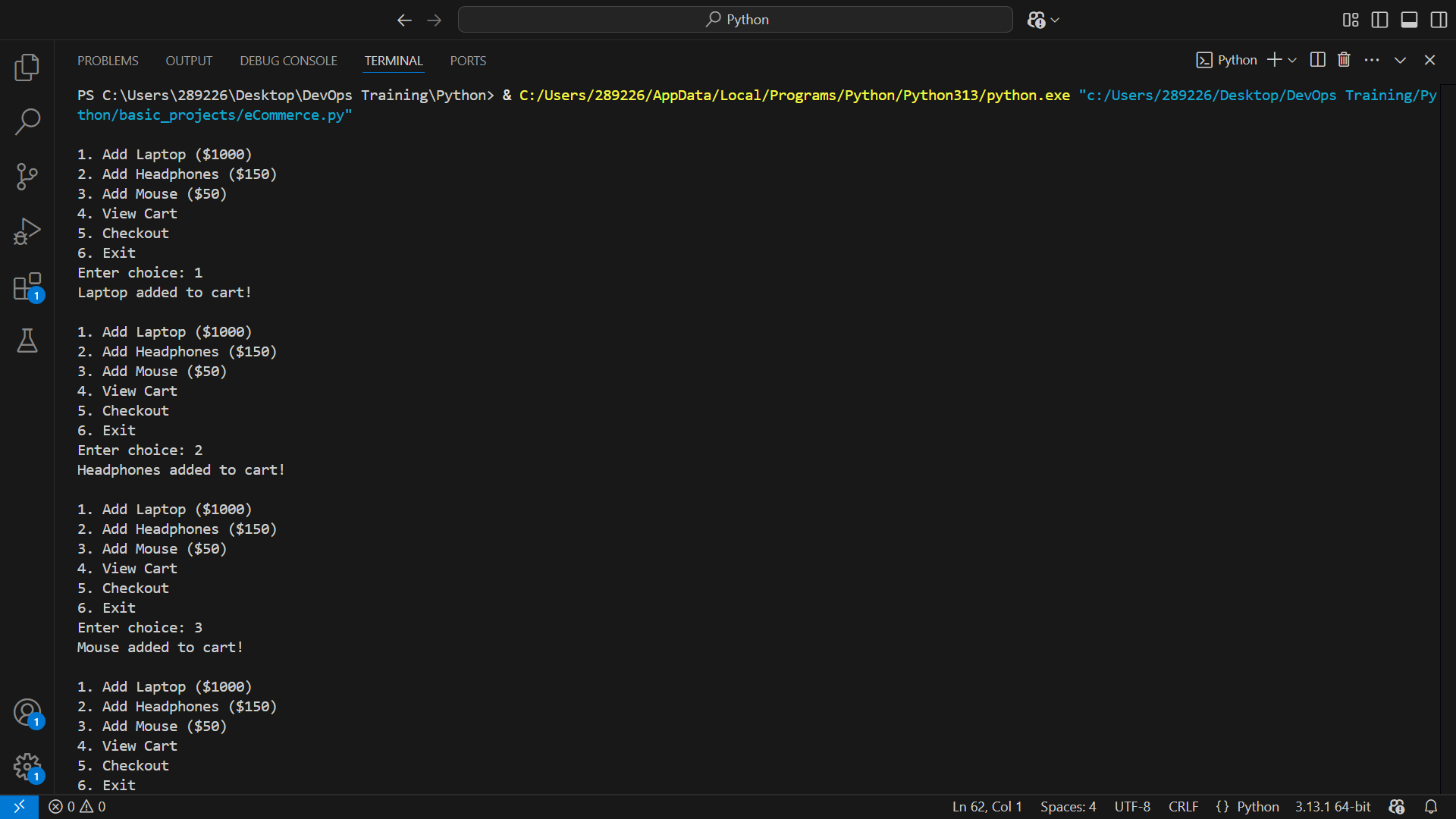
2. Create a shopping cart to store items.

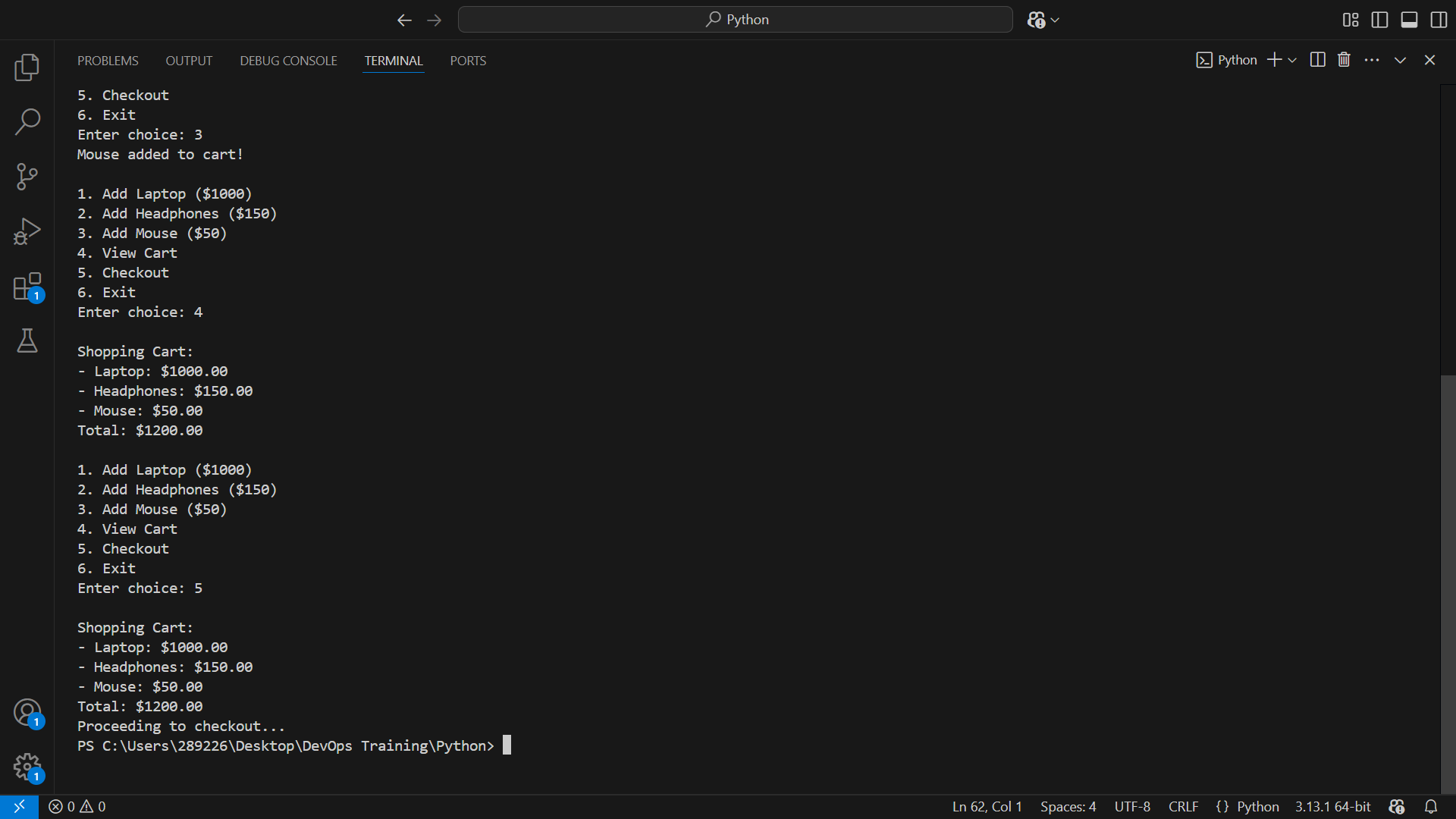
3. Provide options to add/view/checkout.





**Output :**





1. **Student Grade Management System**

A system to manage student grades:

• Add student grades

• View student grades

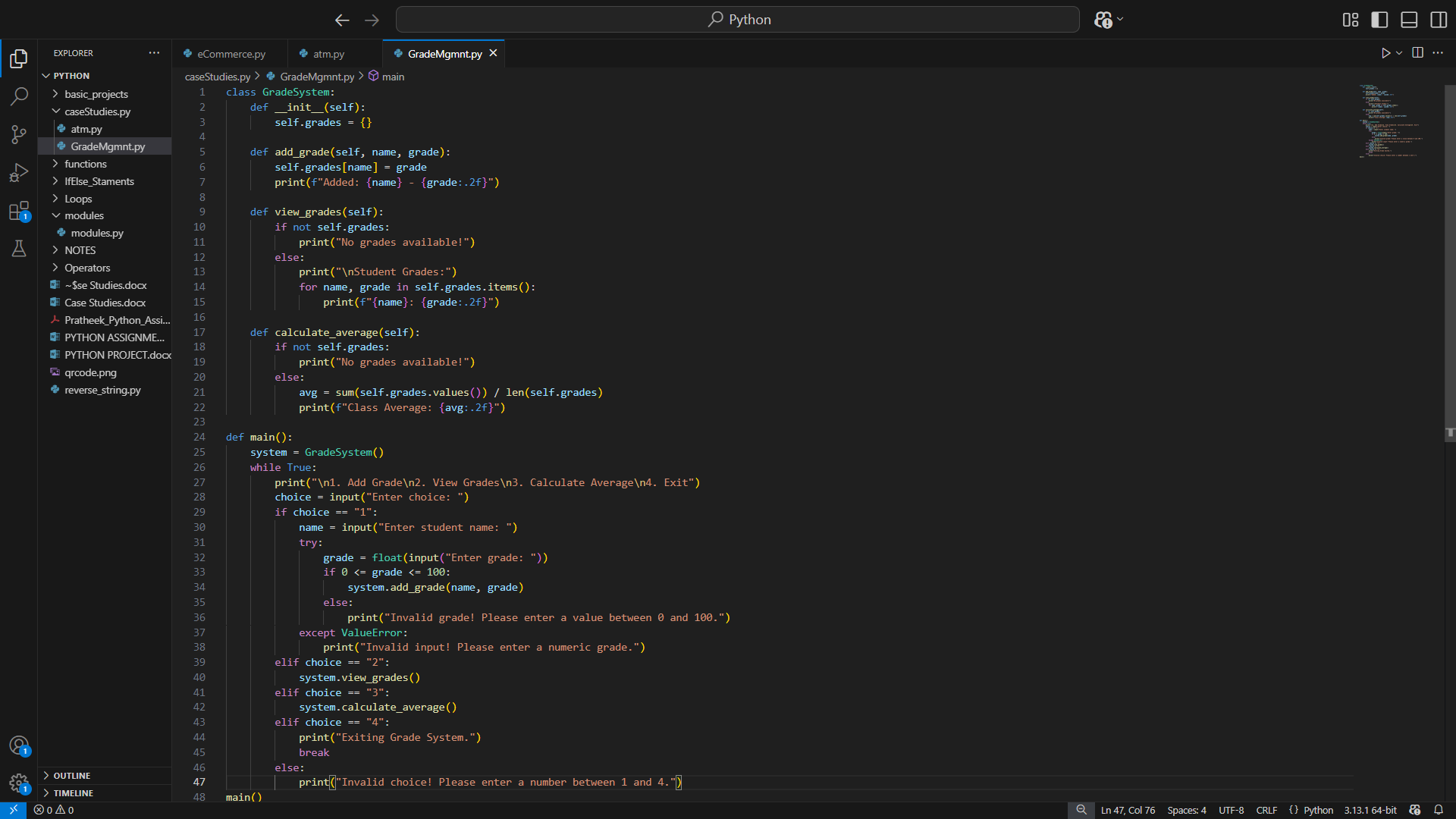
• Calculate the average grade

Steps to Solve

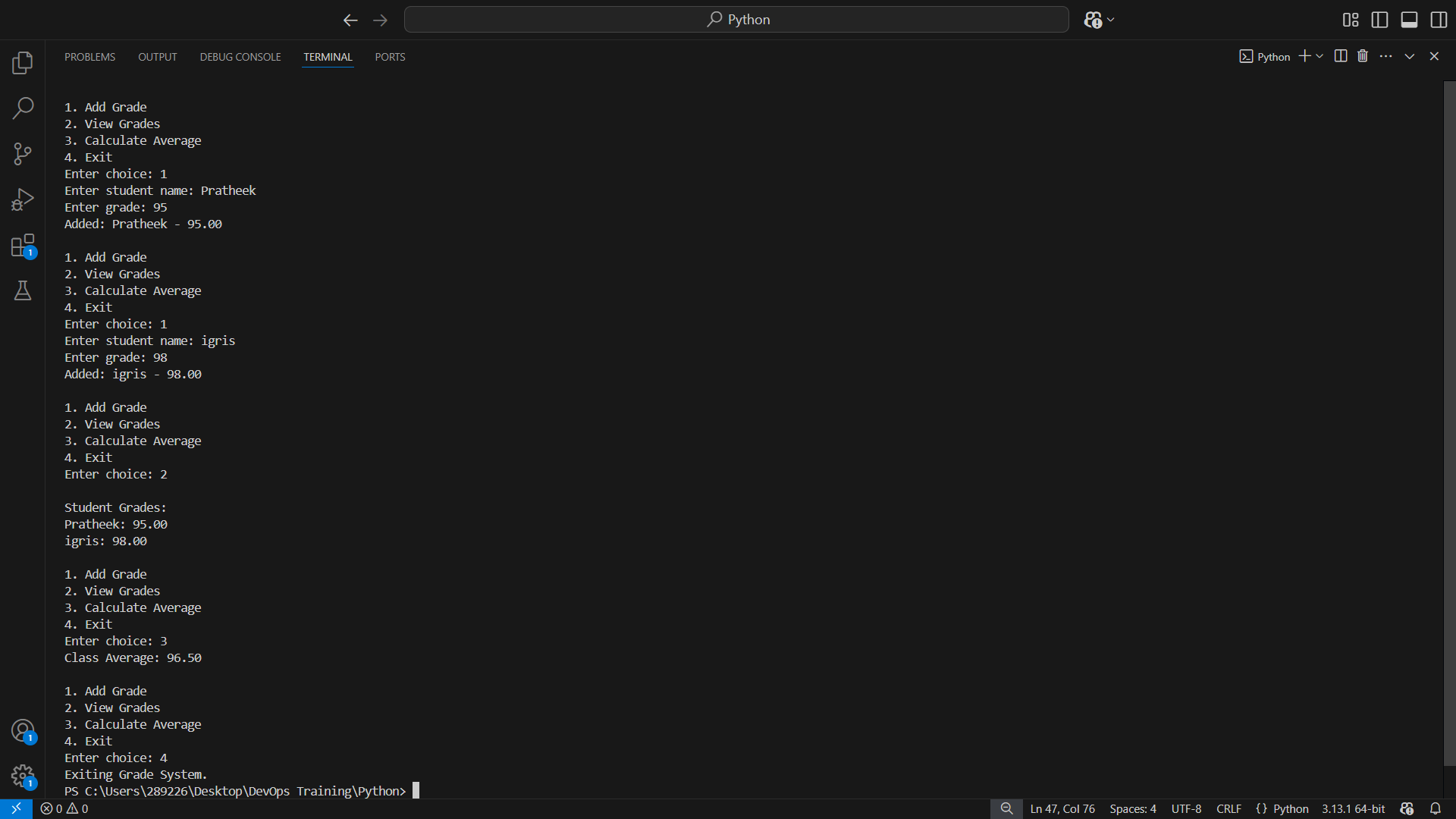
1. Create a dictionary to store student grades.

2. Provide options to add, view, and calculate average.

3. Use a loop for interaction.



**Output :**



1. **Hospital Patient Management**

A hospital management system that:

• Adds new patients

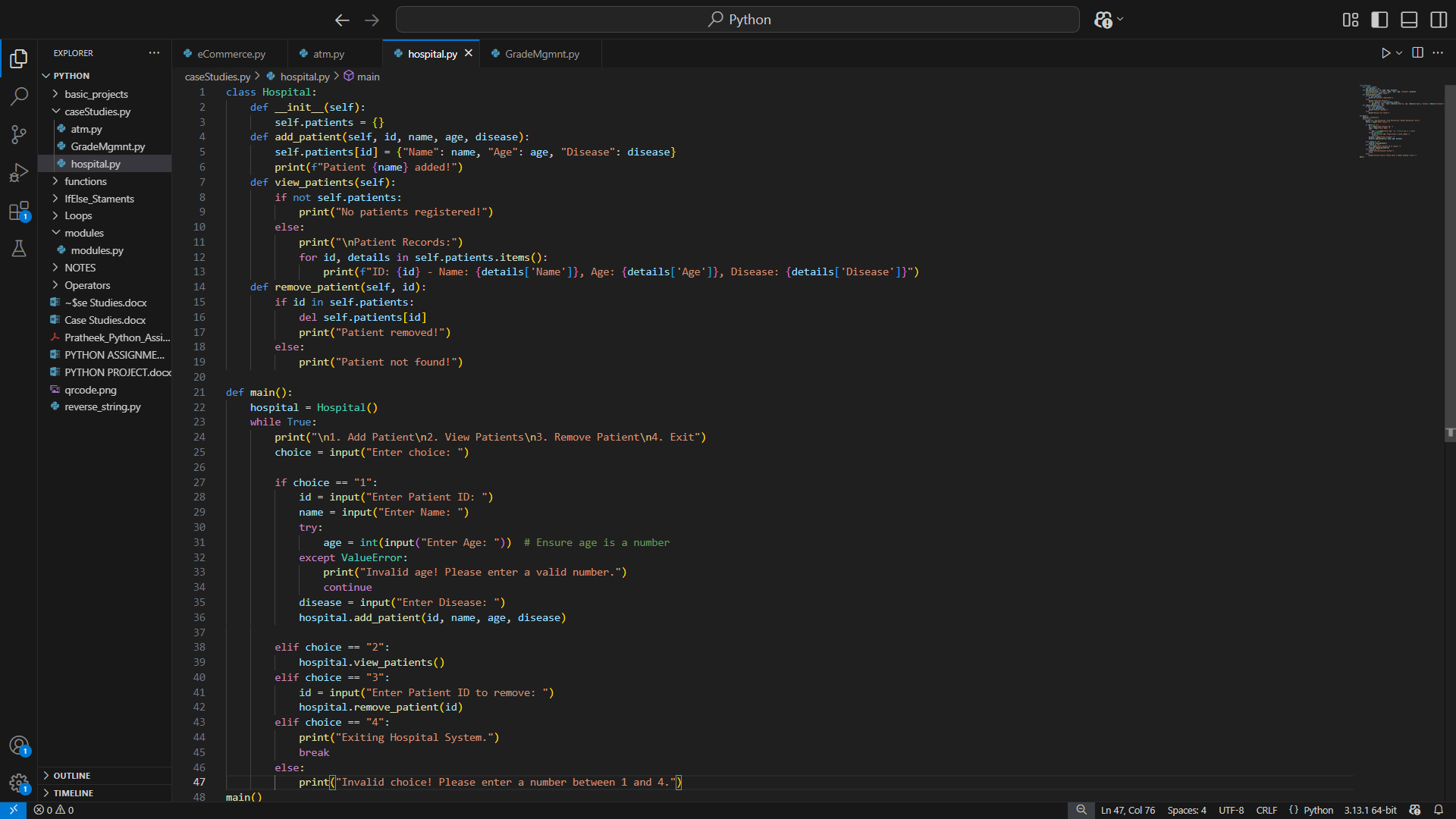
• Displays patient details

• Deletes patients

Steps to Solve

1. Use a dictionary to store patient records.

2. Implement add, view, and delete functions.



**Output :**

