**Python Case Studies**

**1️. Case Study: ATM Simulation System**

**Problem Statement**

**Develop an ATM simulation that allows users to:**

**• Check balance**

**• Deposit money**

**• Withdraw money**

**• Exit**

**Steps to Solve**

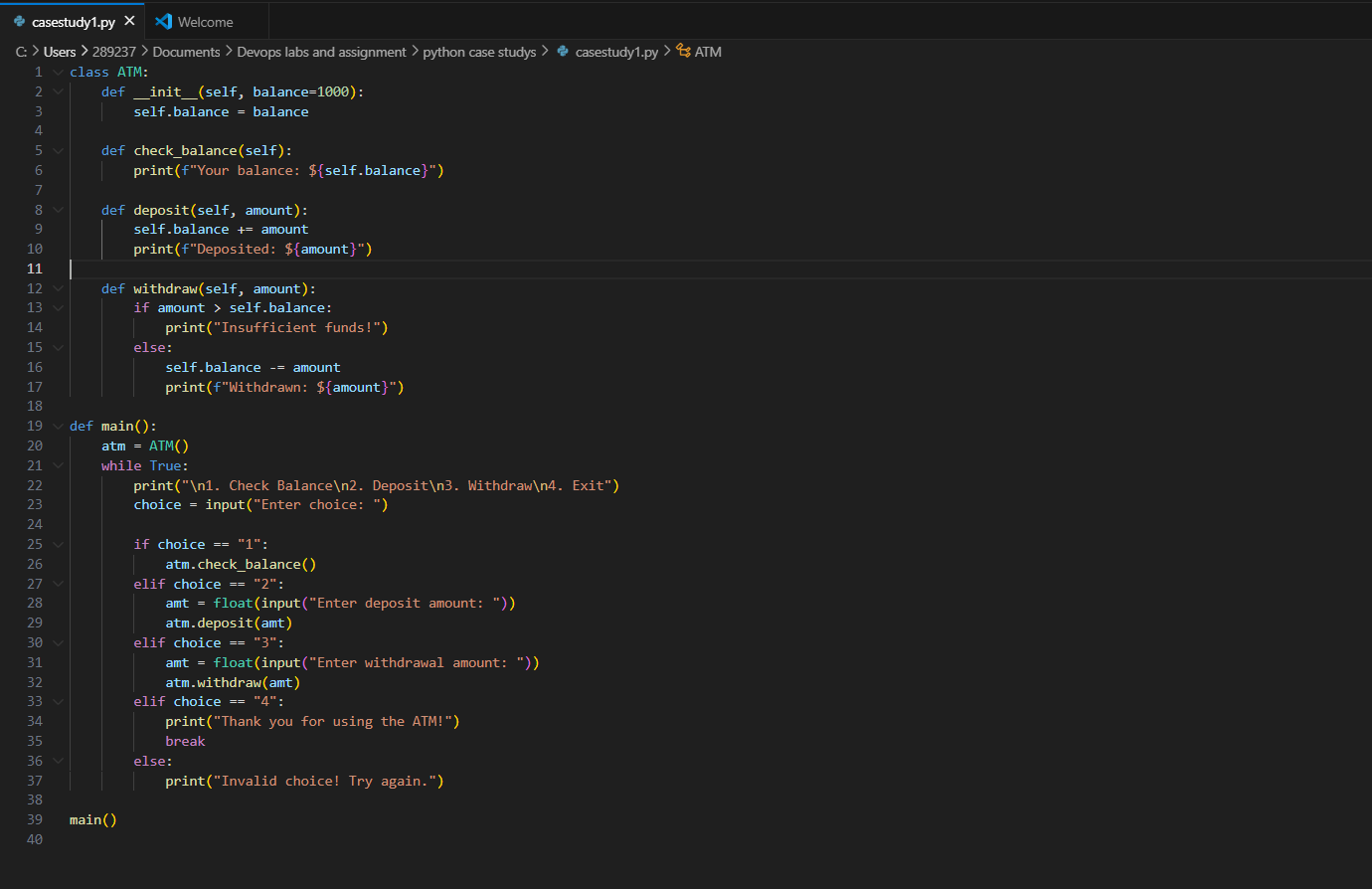
**1. Define an initial balance.**

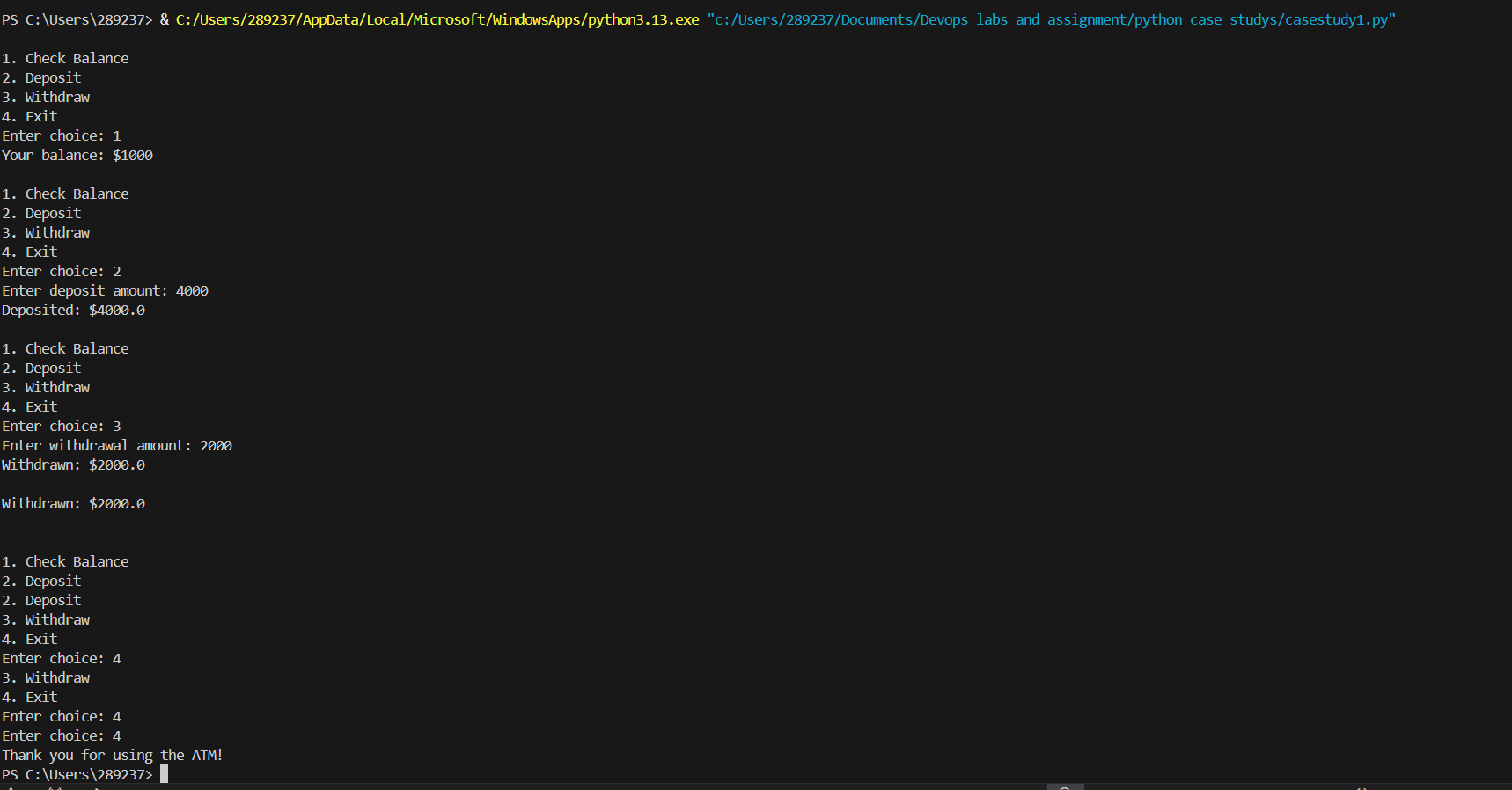
**2. Create a menu-driven system to perform transactions.**

**3. Ensure withdrawal does not exceed balance.**

**4. Exit the program when the user chooses.**

**Code :**



**Output:** 

**2️. Case Study: E-commerce Order Management**

**Problem Statement**

**Create an Order Management System for an e-commerce platform. The system should**

**allow:**

**• Adding products to a cart**

**• Viewing the cart**

**• Checking out (calculating total price)**

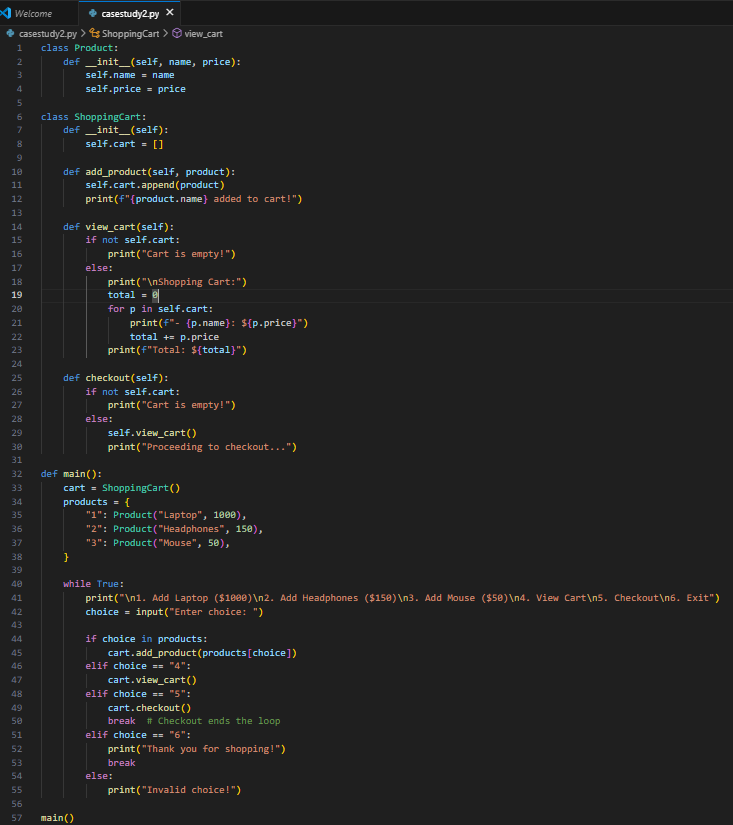
**Steps to Solve**

**1. Define a Product class.**

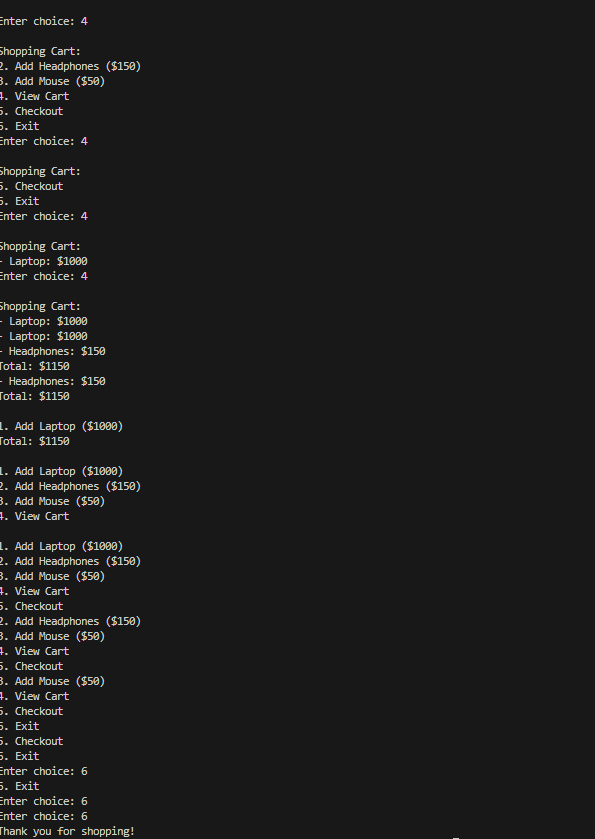
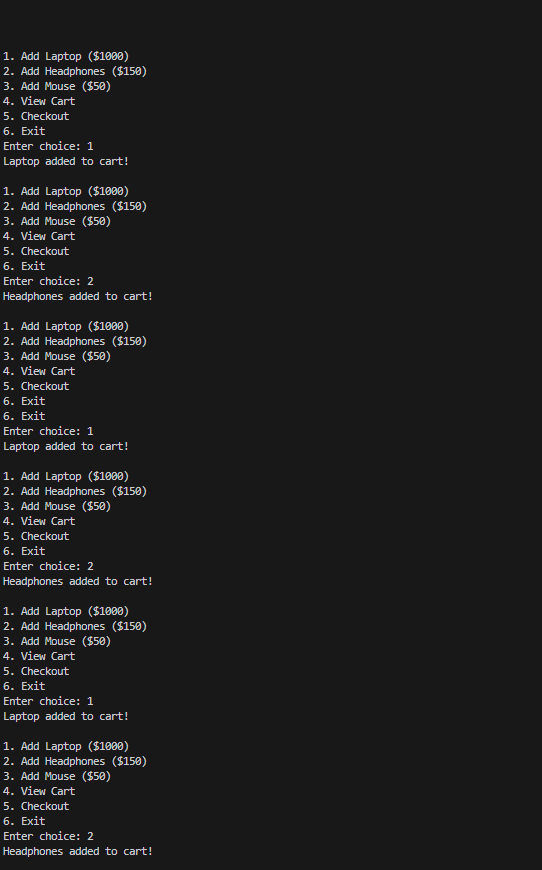
**2. Create a shopping cart to store items.**

**3. Provide options to add/view/checkout.**

**Code :**



**Output:**



**3️. Case Study: Student Grade Management System**

**Problem Statement**

**Develop a system to manage student grades:**

**• Add student grades**

**• View student grades**

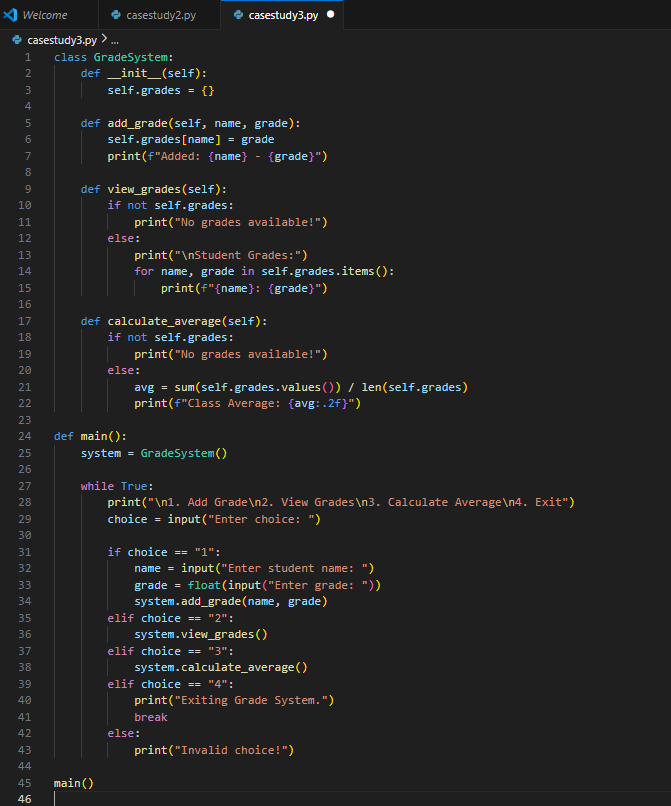
**• Calculate the average grade**

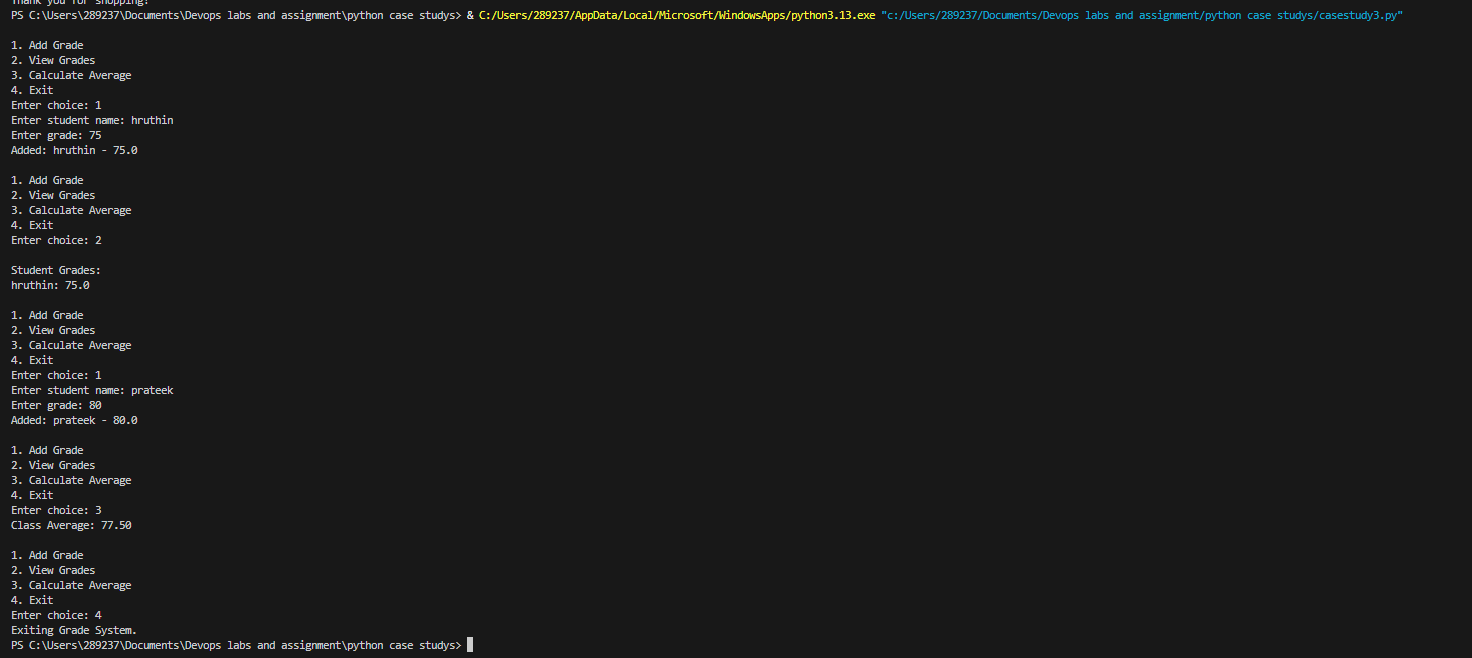
**Steps to Solve**

**1. Create a dictionary to store student grades.**

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

**3. Use a loop for interaction.**

**Code:** 

**Output:** 

**4️. Case Study: Hospital Patient Management**

**Problem Statement**

**Create a hospital management system that:**

**• Adds new patients**

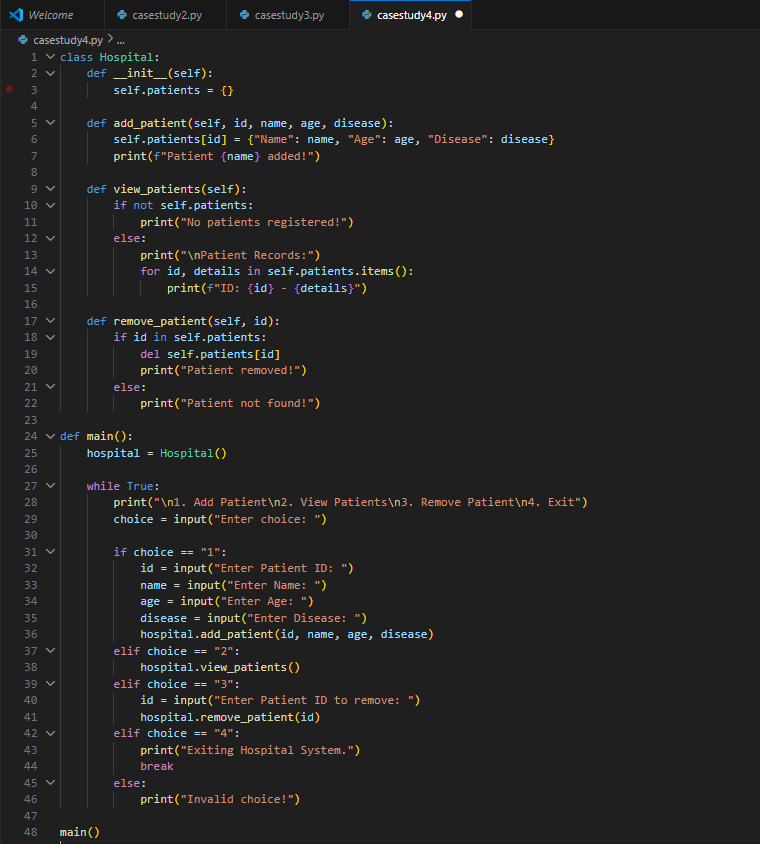
**• Displays patient details**

**• Deletes patients**

**Steps to Solve**

**1. Use a dictionary to store patient records.**

**2. Implement add, view, and delete functions.**

**Code:** ****

**Output**: 