

## Assignment – 6

### AI Assistance in coding

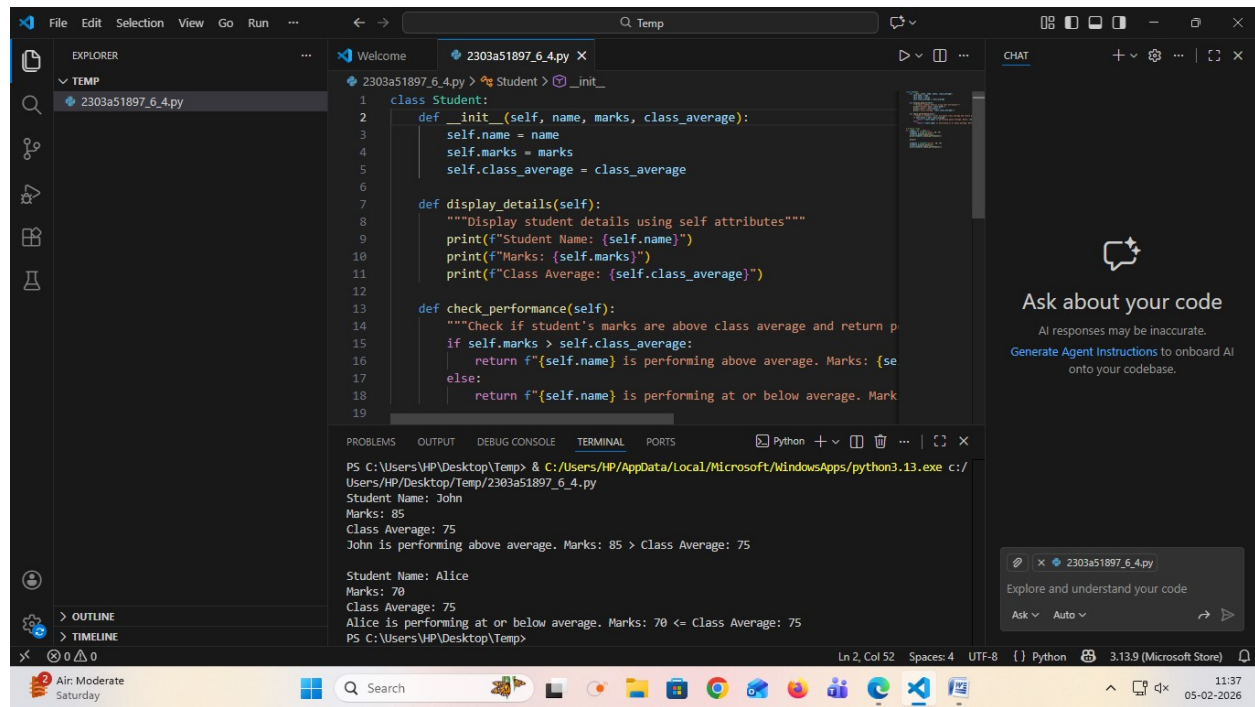
Name:K.Raghavendra, Ht.no:2303a51897 , Batch:09

## Task 1 – Student Performance Evaluation

Create a method to display student details using self attributes

Create a method that checks whether the student's marks are above the class average using if-else

Return an appropriate performance message



```
1 class Student:
2     def __init__(self, name, marks, class_average):
3         self.name = name
4         self.marks = marks
5         self.class_average = class_average
6
7     def display_details(self):
8         """Display student details using self attributes"""
9         print(f"Student Name: {self.name}")
10        print(f"Marks: {self.marks}")
11        print(f"Class Average: {self.class_average}")
12
13    def check_performance(self):
14        """Check if student's marks are above class average and return p
15        if self.marks > self.class_average:
16            return f"{self.name} is performing above average. Marks: {se
17        else:
18            return f"{self.name} is performing at or below average. Mark
19
```

```
PS C:\Users\VP\Desktop\Temp> & C:/Users/HP/AppData/Local/Microsoft/WindowsApps/python3.13.exe c:/Users/HP/Desktop/Temp/2303a51897_6_4.py
Student Name: John
Marks: 85
Class Average: 75
John is performing above average. Marks: 85 > Class Average: 75

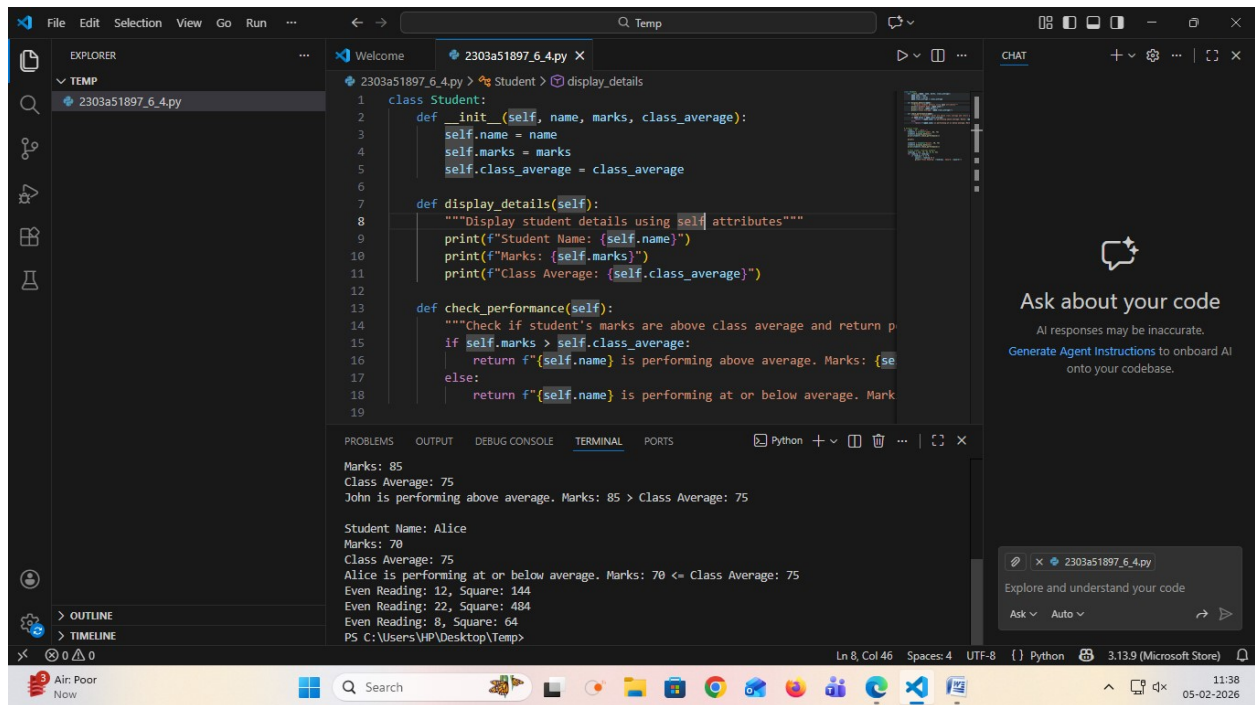
Student Name: Alice
Marks: 70
Class Average: 75
Alice is performing at or below average. Marks: 70 <= Class Average: 75
PS C:\Users\VP\Desktop\Temp>
```

## Task 2 – Data Processing in Monitoring System

Inside the loop, identify even numbers using modulus operator

Calculate the square of even readings

Print the reading and its square in a readable format



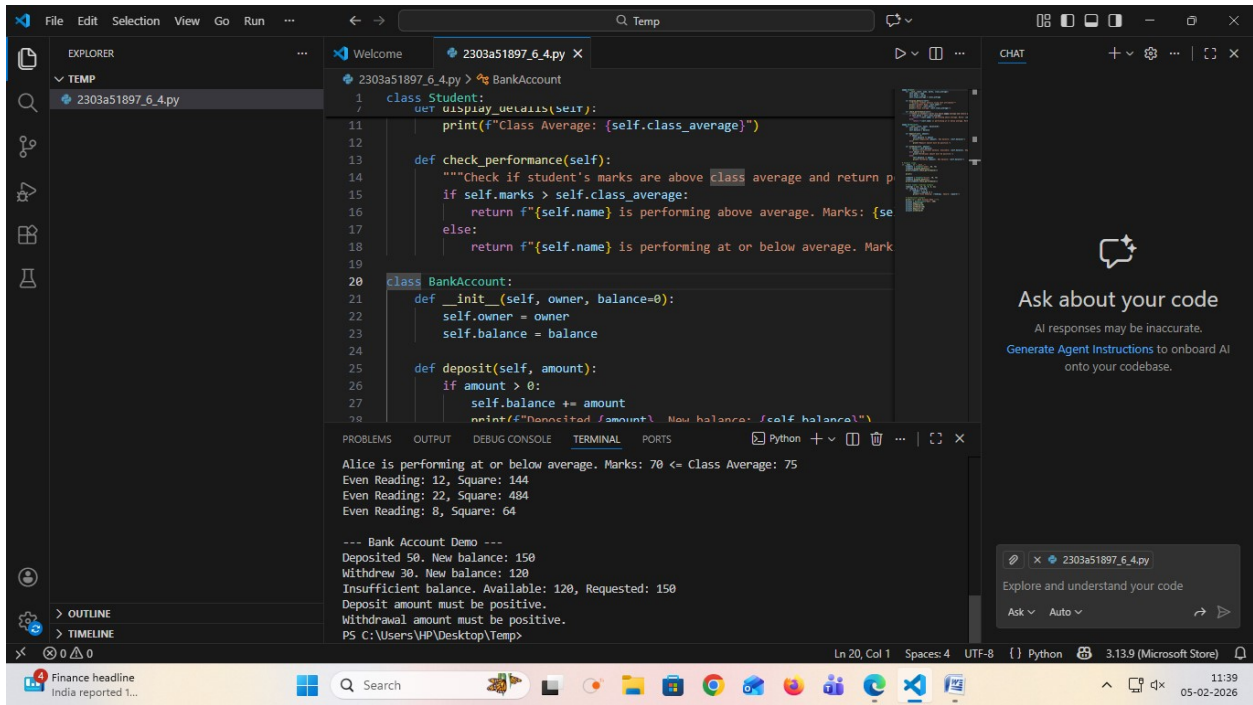
## Task 3 – Banking Transaction Simulation

Create a deposit method that adds money to balance

Create a withdraw method

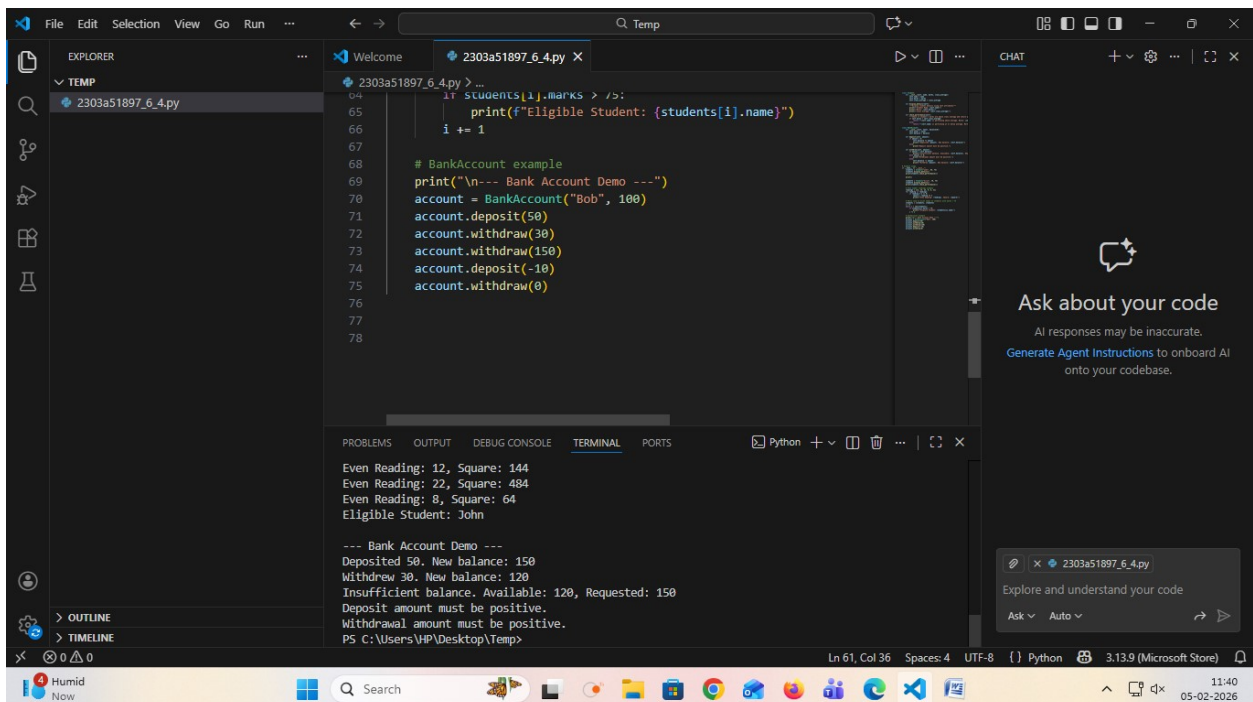
Use if-else to prevent withdrawal when balance is insufficient

Display user-friendly messages



## Task 4 – Student Scholarship Eligibility

Use a while loop to iterate through the student list  
Check if the score is greater than 75  
Print the name of eligible students



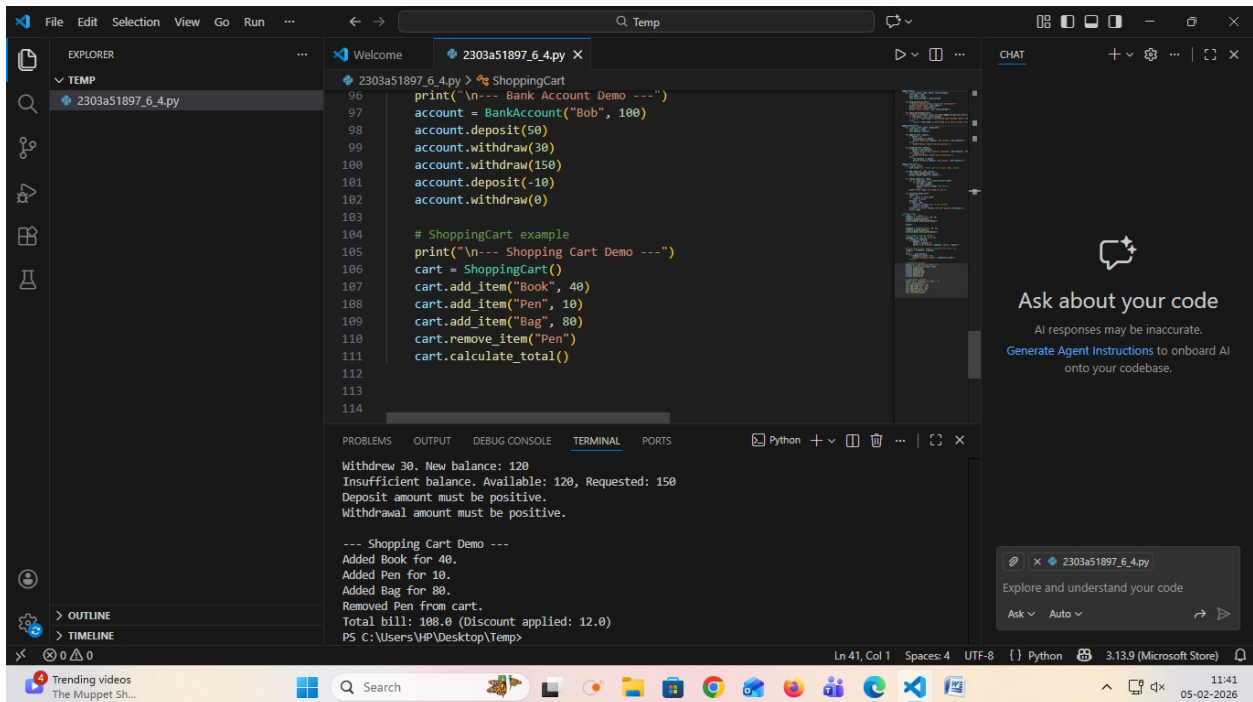
## Task 5 – Online Shopping Cart Module

Create a method to add items to the shopping cart

Create a method to remove items from the cart

Create a method to calculate total bill using a loop

Apply discount if total exceeds a certain amount



The screenshot shows a Visual Studio Code editor window with a Python file named `2303a51897_6_4.py`. The code defines a `BankAccount` class and a `ShoppingCart` class. The `BankAccount` class has methods for deposit, withdrawal, and balance checking. The `ShoppingCart` class has methods for adding and removing items, calculating the total bill, and applying a discount. The terminal output shows the execution of the code, including the creation of a bank account, the addition of items to the cart, and the calculation of the total bill with a discount applied.

```
96 print("\n--- Bank Account Demo ---")
97 account = BankAccount("Bob", 100)
98 account.deposit(50)
99 account.withdraw(30)
100 account.withdraw(150)
101 account.deposit(-10)
102 account.withdraw(0)
103
104 # ShoppingCart example
105 print("\n--- Shopping Cart Demo ---")
106 cart = ShoppingCart()
107 cart.add_item("Book", 40)
108 cart.add_item("Pen", 10)
109 cart.add_item("Bag", 80)
110 cart.remove_item("Pen")
111 cart.calculate_total()
112
113
114
```

Terminal Output:

```
Withdraw 30. New balance: 120
Insufficient balance. Available: 120, Requested: 150
Deposit amount must be positive.
Withdrawal amount must be positive.

--- Shopping Cart Demo ---
Added Book for 40.
Added Pen for 10.
Added Bag for 80.
Removed Pen from cart.
Total bill: 108.0 (Discount applied: 12.0)
PS C:\Users\JHP\Desktop\Temp>
```

## Analysis:

This experiment helped the practical use of Python concepts such as classes, loops, and conditional statements. By working on real-world scenarios like student evaluation, banking, and shopping systems, improved problem-solving skills and learned how GitHub Copilot can assist in faster and more efficient code development.