AI ASSISTED CODING

NAME: B.NANDINI

ENROLL NUMBER: 2403A52390

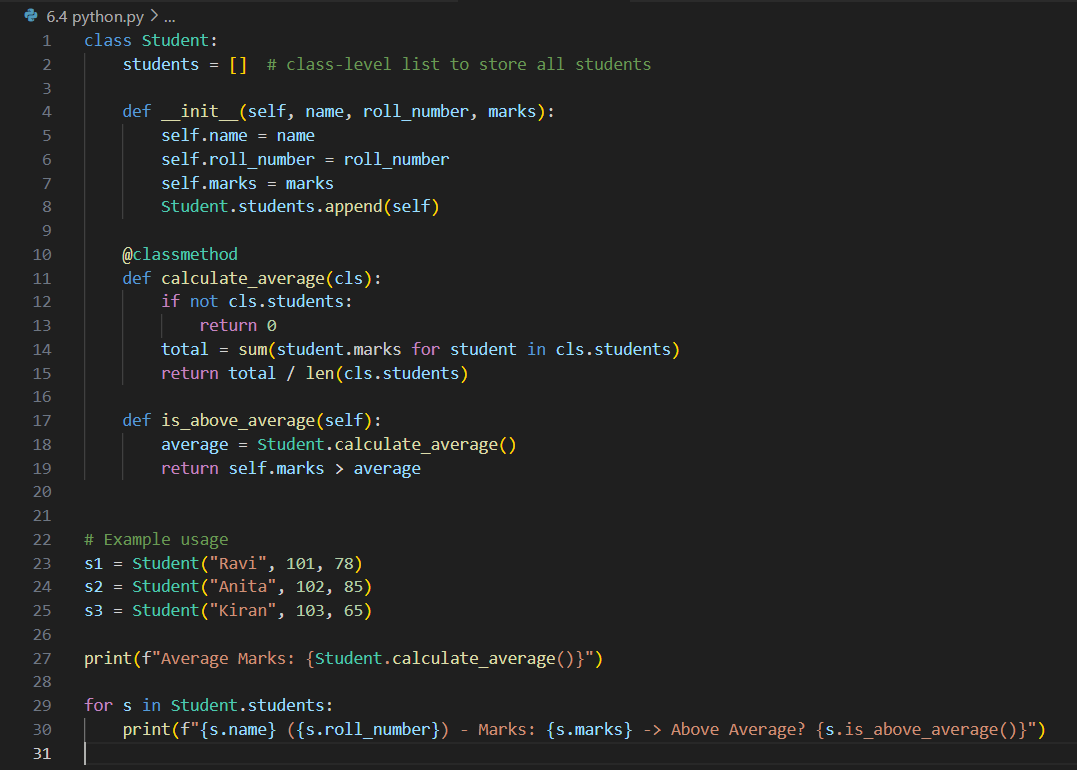
BATCH NUMBER :14

Lab assignment-6.4

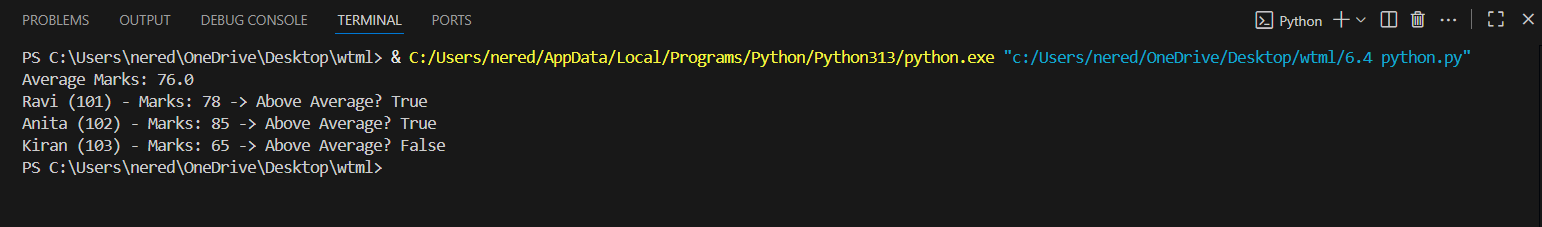
Promte 1:

generate a Python class named Student with attributes name, roll\_number, and marks. Prompt GitHub Copilot to complete methods for displaying details and checking if marks are above average

code:



Output:



Code explanation:

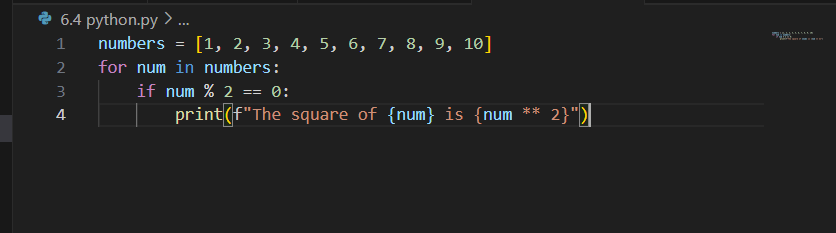
This code defines a Student class for managing student records and checking if a student's marks are above average:

* **Class Attribute:**  
  students is a class-level list that stores all student instances.
* **Constructor (\_\_init\_\_):**  
  Initializes each student's name, roll\_number, and marks.  
  Adds the new student to the students list.
* **Class Method (calculate\_average):**  
  Calculates the average marks of all students in the list.  
  Returns 0 if there are no students.
* **Instance Method (is\_above\_average):**  
  Checks if the student's marks are greater than the average marks of all students.
* **Example Usage:**  
  Creates three student objects.  
  Prints the average marks.  
  Iterates through all students, printing their details and whether their marks are above average.

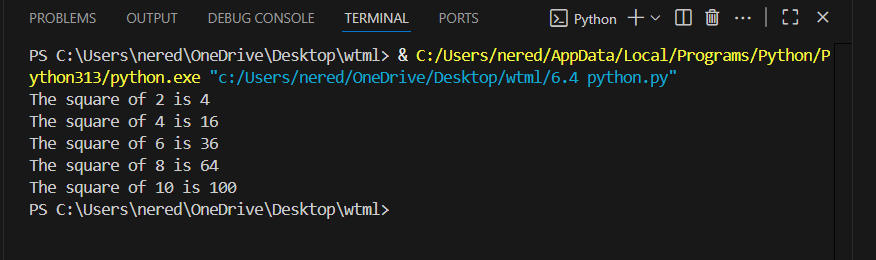
Promote 2:

• generate a python code for first two lines of a for loop to iterate through a list of numbers. Use a comment prompt to let Copilot suggest how to calculate and print the square of even numbers only.

Code:



Output:



Code explanation:

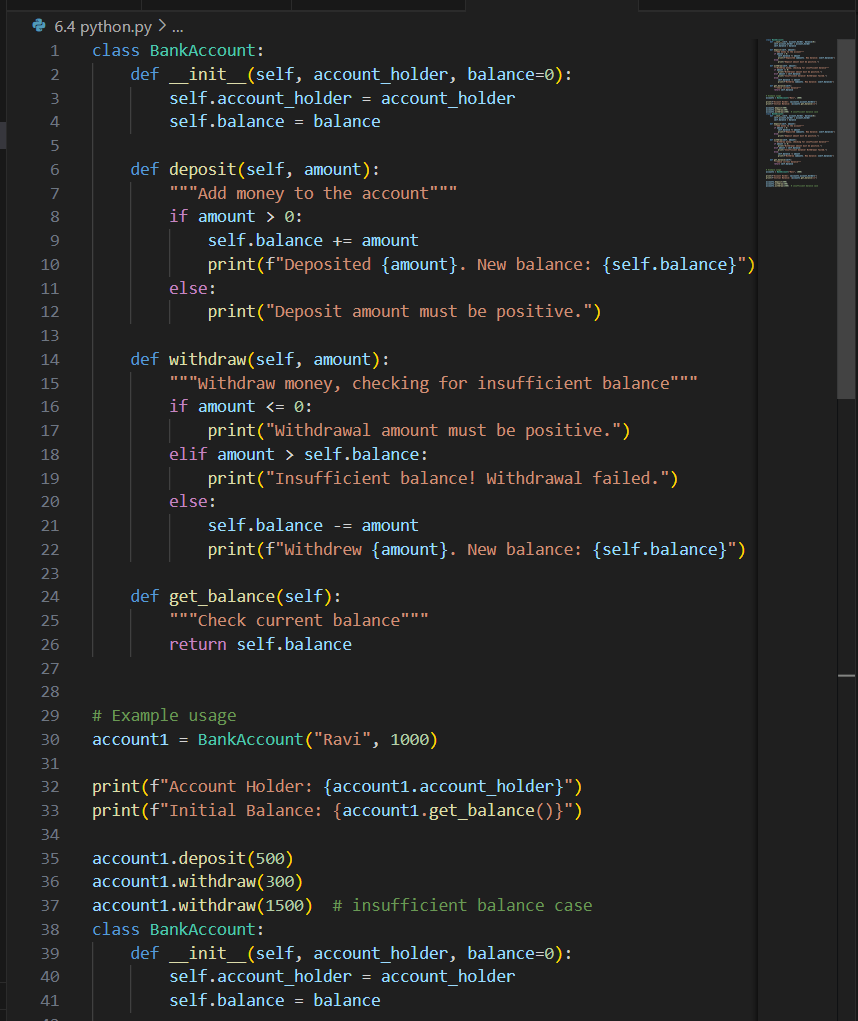
This code iterates through a list of numbers from 1 to 10:

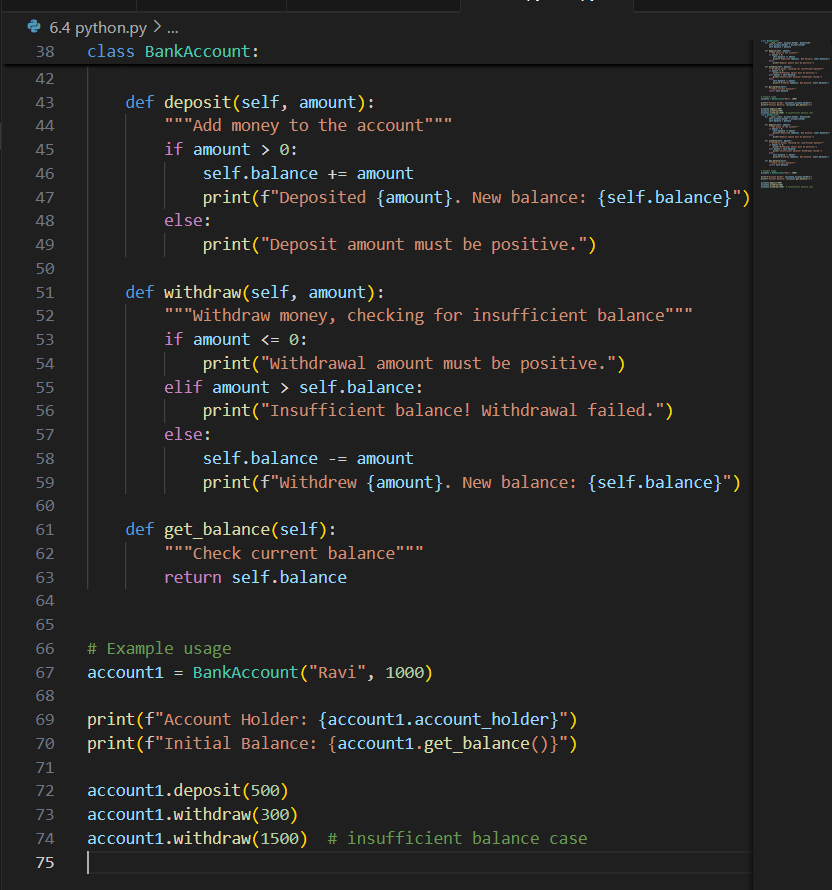
* [numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]](vscode-file://vscode-app/c:/Users/nered/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)  
  Defines a list of numbers.
* [for num in numbers:](vscode-file://vscode-app/c:/Users/nered/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)  
  Loops through each number in the list.
* [if num % 2 == 0:](vscode-file://vscode-app/c:/Users/nered/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)  
  Checks if the current number is even (divisible by 2).
* [print(f"The square of {num} is {num \*\* 2}")](vscode-file://vscode-app/c:/Users/nered/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)  
  If the number is even, prints its square using an f-string for formatting.

Promote 3:

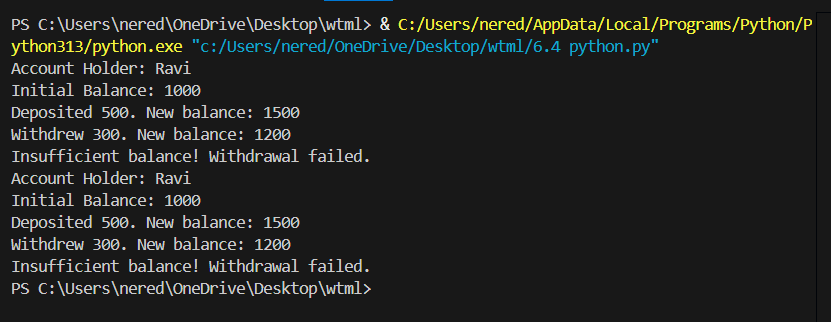
• Generate a python code to Create a class called BankAccount with attributes account\_holder and balance. Use Copilot to complete methods for deposit(), withdraw(), and check for insufficient balance

Code:





Output:



Code explanation:

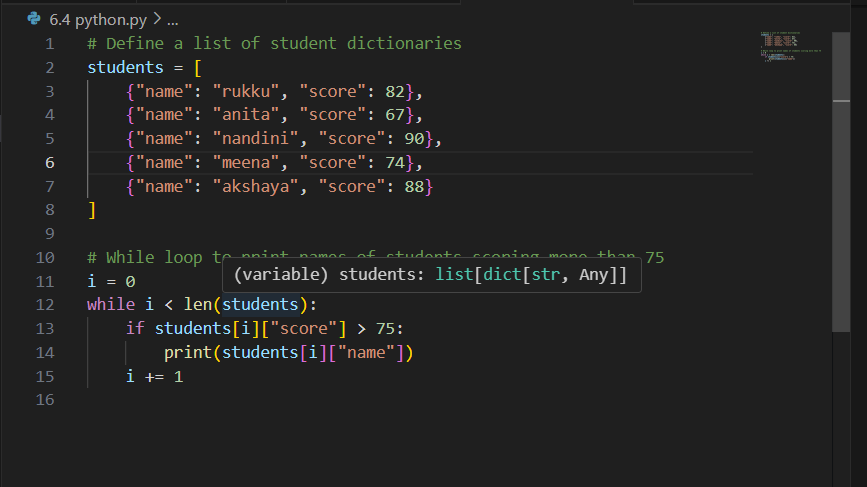
This code defines a [BankAccount](vscode-file://vscode-app/c:/Users/nered/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html) class to simulate basic banking operations:

* [\_\_init\_\_](vscode-file://vscode-app/c:/Users/nered/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)**method:**  
  Initializes the account with the holder's name and an optional starting balance (default is 0).
* [deposit](vscode-file://vscode-app/c:/Users/nered/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)**method:**  
  Adds money to the account if the amount is positive.  
  Prints the deposited amount and new balance.  
  If the amount is not positive, prints an error message.
* [withdraw](vscode-file://vscode-app/c:/Users/nered/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)**method:**  
  Withdraws money if the amount is positive and does not exceed the current balance.  
  Prints the withdrawn amount and new balance.  
  If the amount is not positive or exceeds the balance, prints an error message.
* [get\_balance](vscode-file://vscode-app/c:/Users/nered/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)**method:**  
  Returns the current account balance.
* **Example usage:**
  + Creates an account for "Ravi" with an initial balance of 1000.
  + Prints the account holder and initial balance.
  + Deposits 500, withdraws 300, and attempts to withdraw 1500 (which fails due to insufficient balance).
  + All actions print relevant messages to the console.

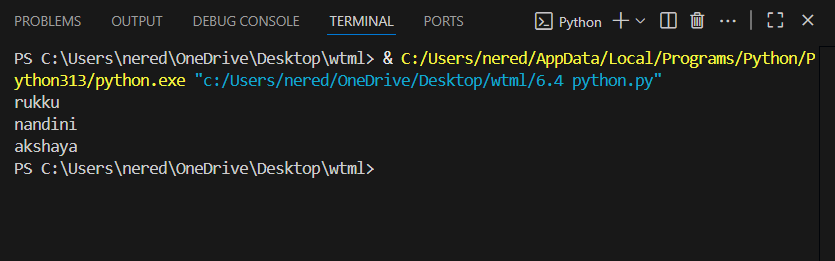
Promote 4:

• generate a python code and Define a list of student dictionaries with keys name and score. Ask Copilot to write a while loop to print the names of students who scored more than 75.

Code:



Output:



Code explaination:

This code defines a list of student dictionaries, each with a "name" and "score" key.  
It then uses a while loop to iterate through the list:

* [i = 0](vscode-file://vscode-app/c:/Users/nered/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html) initializes the loop counter.
* [while i < len(students):](vscode-file://vscode-app/c:/Users/nered/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html) loops through each student in the list.
* [if students[i]["score"] > 75:](vscode-file://vscode-app/c:/Users/nered/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html) checks if the student's score is greater than 75.
* If true, it prints the student's name.
* [i += 1](vscode-file://vscode-app/c:/Users/nered/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html) increments the counter to move to the next student.

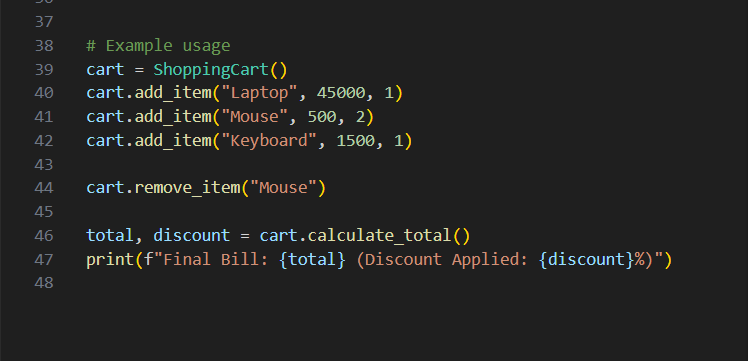
**Result:**  
It prints the names of all students who scored more than 75.

Promote 5:

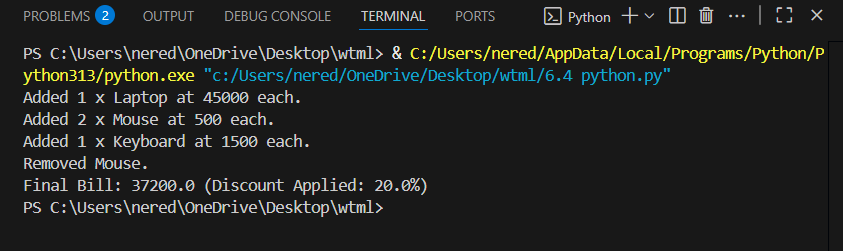
•generate a python code for Begin writing a class ShoppingCart with an empty items list. Prompt Copilot to generate methods to add\_item, remove\_item, and use a loop to calculate the total bill using conditional discounts.

Code:





Output:



Code explanation:

This code iterates through a list of numbers and prints the square of each even number.

numbers = [1, 2, 3, 4, 5, 6] creates a list of numbers.

The for loop goes through each number in the list.

Inside the loop, if number % 2 == 0: checks if the number is even.

If the number is even, print(f"Square of {number} is {number\*\* 12}") prints the result of raising the number to the 12th power (not the square).

Note:

To print the square, use number \*\* 2 instead of number \*\* 12.

The current code prints the 12th power, not the square.

GPT