ASSIGNMENT-6.4

Name : Sharanya Thanugula

Hall ticket No: 2403A51401

Batch:16

Task 1:

Start by creating a Python class named Student that includes three attributes: name, roll\_number, and marks. Then, prompt GitHub Copilot to complete two methods within the class. The first method, display\_details(), should print the student's name, roll number, and marks in a clean, readable format. The second method, is\_passed(), should use an if-else condition to check whether the student's marks are above average—assume the average threshold is 50—and return True if the student has passed, otherwise False and the input should be take in dynamic,take more than 3 students input at one time

Code:

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

Output:

A screenshot of a computer

AI-generated content may be incorrect.

Explanation:

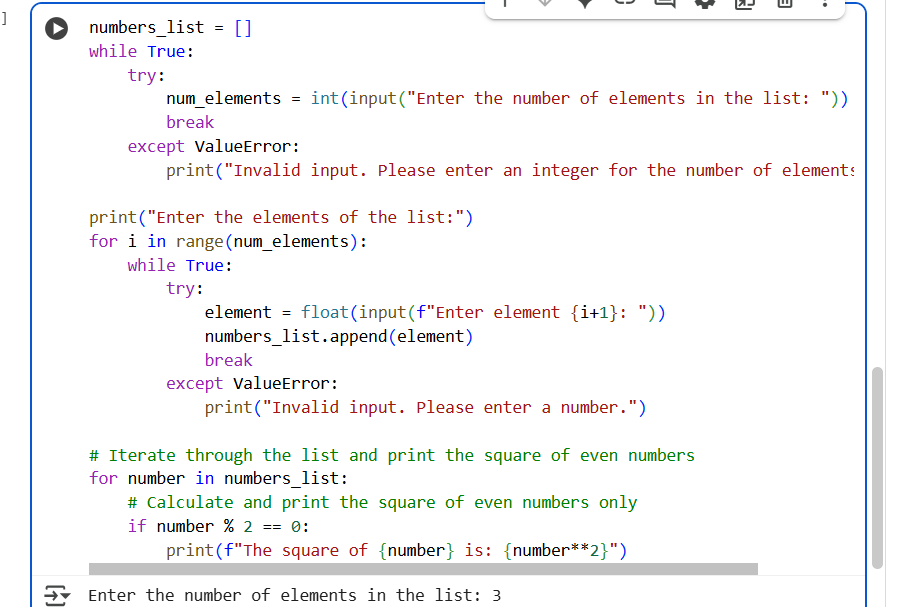
A close-up of a text

AI-generated content may be incorrect.

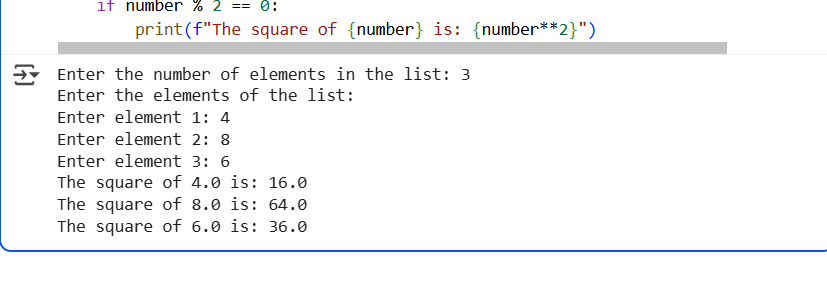
Task 2:

writing a Python for loop that iterates through a predefined list of numbers. Include the first two lines: one that defines the list and another that starts the loop using for number in list:. Then, add a comment prompt asking Copilot to complete the loop by calculating and printing the square of even numbers only. The expected result is a complete loop with conditional logic using if number % 2 == 0, and output that displays the squared value of each even number in the list , take the input in dynamic and take the inputs at ones more than 3

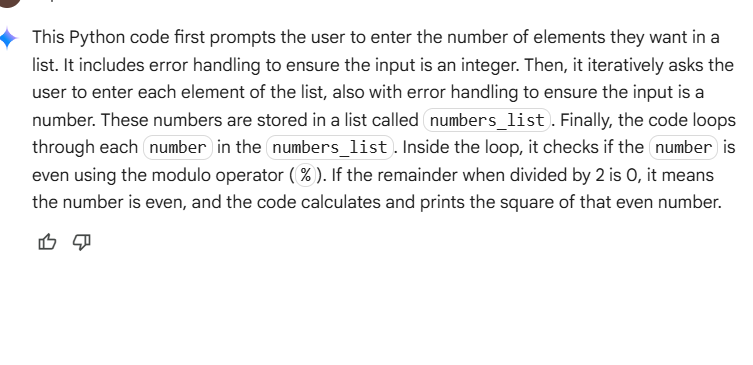
code:



Output:



Explanation:



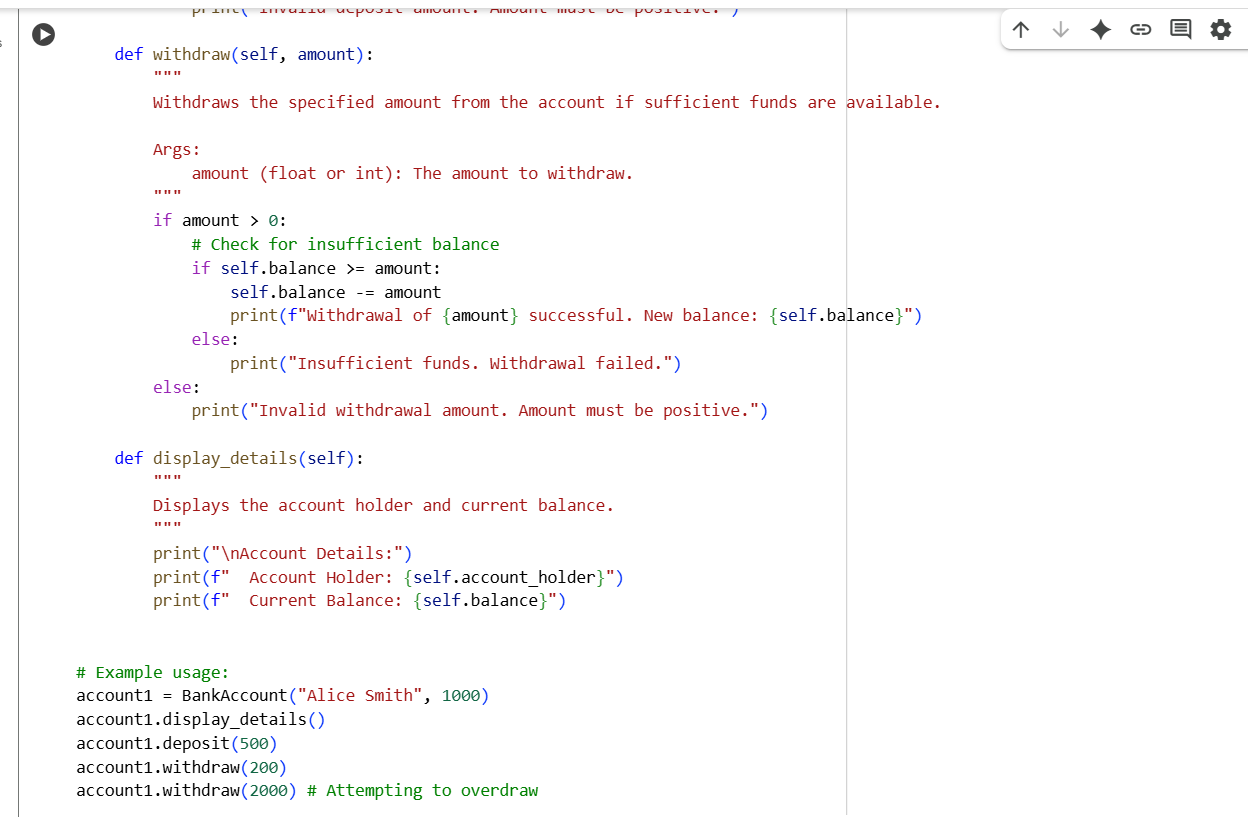
Task 3:

Create a Python class named BankAccount with two attributes: account\_holder (a string representing the name of the account owner) and balance (a float or integer representing the current account balance). Then, prompt Copilot to complete three methods within the class. The first method, deposit(amount), should add the specified amount to the balance. The second method, withdraw(amount), should subtract the amount from the balance only if sufficient funds are available—otherwise, it should print a warning or return an error message. The third method should check for insufficient balance using an if condition and prevent overdrawing. The expected outcome is a fully functional class with properly defined methods that use self attributes and conditional logic to manage deposits and withdrawals safely

Code:

A screenshot of a computer

AI-generated content may be incorrect.

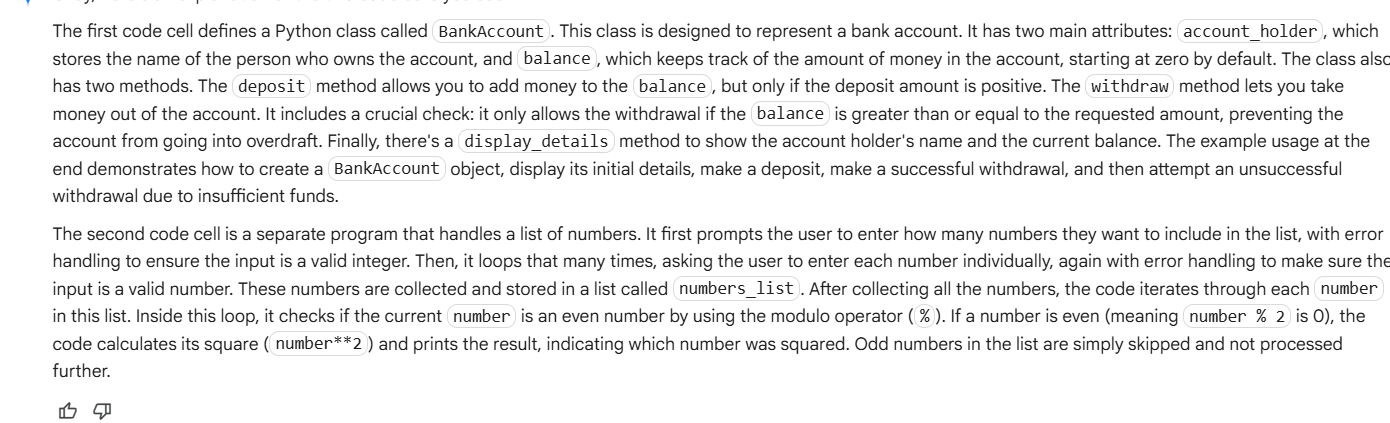


Output:

A screenshot of a computer

AI-generated content may be incorrect.

Explanation:



Task 4:

Define a list of student dictionaries where each dictionary contains two keys: name and score. Then begin writing a while loop that iterates through this list. Add a comment prompt asking Copilot to complete the loop by checking if each student's score is greater than 75, and if so, printing their name in a formatted output. The expected result is a complete while loop with proper condition checks using if student['score'] > 75, and output that clearly displays the names of students who meet the criteria take the input in dynamic once more than 3 inputs

Code:

A screenshot of a computer program

AI-generated content may be incorrect.

Output:

A screenshot of a computer program

AI-generated content may be incorrect.

Explanation:

A screenshot of a computer

AI-generated content may be incorrect.

Task 5:

writing a Python class named ShoppingCart that initializes with an empty list called items. Then prompt Copilot to generate methods for adding an item to the cart (add\_item(item, price)), removing an item (remove\_item(item)), and calculating the total bill. The total calculation should use a loop to iterate through the items and apply conditional discounts—for example, apply a 10% discount if the item price exceeds 500. The expected outcome is a fully implemented class with method definitions that use self attributes, loops, and if-else statements to manage the cart and apply discount logic appropriately take the input in dynamic structure

Code: A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

Output:

A screenshot of a computer

AI-generated content may be incorrect.

Explanation:

