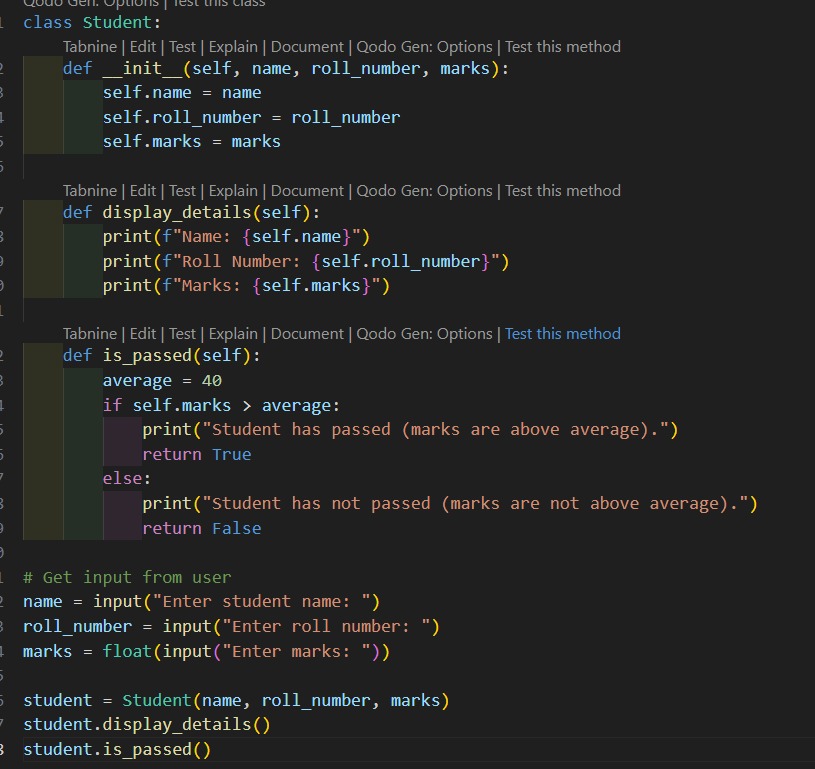
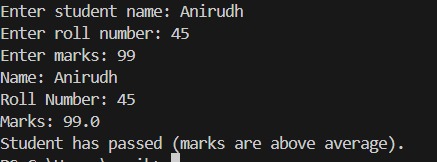
Assignment – 6.4

Task Description #1:  
• Start 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.  
Expected Outcome #1:  
• Completed class with Copilot-generated methods like display\_details() and is\_passed(),  
demonstrating use of if-else conditions

Prompt: Task Description #1:  
• Start 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.  
Expected Outcome #1:  
• Completed class with Copilot-generated methods like display\_details() and is\_passed(),  
demonstrating use of if-else conditions also print the Output

Code and Output:





Observation:In this Code We can Observe that the code is calculating the marks of the student to check whether the student is pass or not

Task Description #2:  
• Write the 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.  
Expected Outcome #2:  
• A complete loop generated by Copilot with conditional logic (if number % 2 == 0) and  
appropriate output

Prompt:

Task Description #2:  
• Write the 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.  
Expected Outcome #2:  
• A complete loop generated by Copilot with conditional logic (if number % 2 == 0) and  
appropriate output also print the Output

Code and Output:

A black background with white text and yellow letters

Description automatically generated

A black screen with white text

Description automatically generated

Observation:In tnis code we can Observe that the code is printing the square of even numbers given in the list

Task Description #3:  
• Create a class called BankAccount with attributes account\_holder and balance. Use Copilot to  
complete methods for deposit(), withdraw(), and check for insufficient balance.  
Expected Outcome #3:  
• Functional class with complete method definitions using if conditions and self attributes. Code  
should prevent overdrawing

Prompt: • Create a class called BankAccount with attributes account\_holder and balance. Use Copilot to  
complete methods for deposit(), withdraw(), and check for insufficient balance.  
Expected Outcome #3:  
• Functional class with complete method definitions using if conditions and self attributes. Code  
should prevent overdrawing also take input in the code and Print the Output

Code and Output:

A screen shot of a computer program

Description automatically generated

A screen shot of a computer

Description automatically generated

Observation:

In this Code we can observe that the code Is about banking and we can check balance,deposite,withdraw .

Task Description #4:  
• 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.  
Expected Outcome #4:  
• A complete while loop generated by Copilot with proper condition checks and formatted  
output

Prompt: • 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.  
Expected Outcome #4:  
• A complete while loop generated by Copilot with proper condition checks and formatted  
output also print the Output.

Code and Output:

A computer screen shot of text

Description automatically generated

A black screen with white text

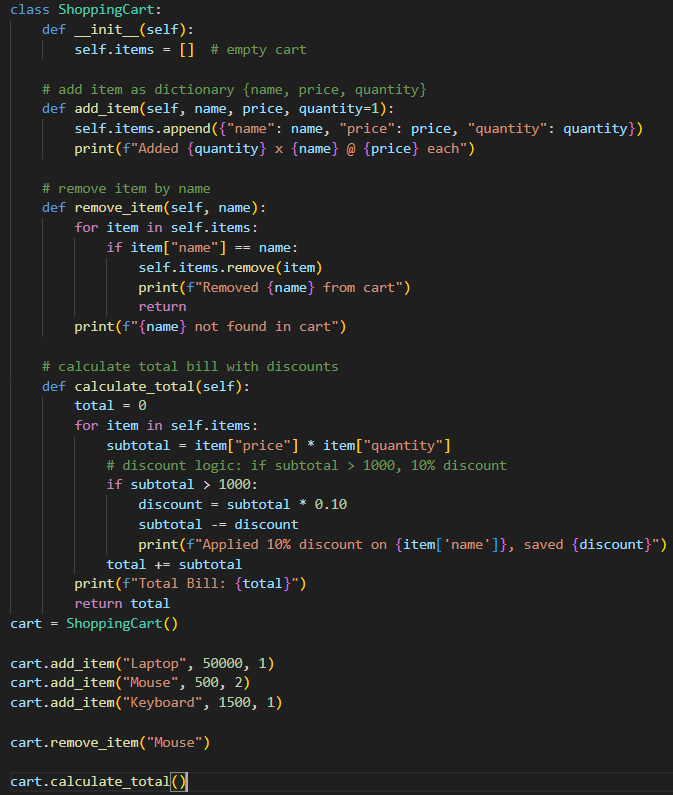
Description automatically generated

Observation: In this code we can observe that the code is checking that the students has scored more than 75 or not

Task Description #5:  
• 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.  
Expected Outcome #5:  
• A fully implemented ShoppingCart class with Copilot-generated loops and if-else statements  
handling item management and discount logic

Prompt: • 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.  
Expected Outcome #5:  
• A fully implemented ShoppingCart class with Copilot-generated loops and if-else statements  
handling item management and discount logic also print the Output

Code and Output:



A computer screen with white text

Description automatically generated

Observation:In this code we can see the code about the Shopping bill code consists of adding item, remove item, discount , total bill.