Harika Jupaka

2403A51L31

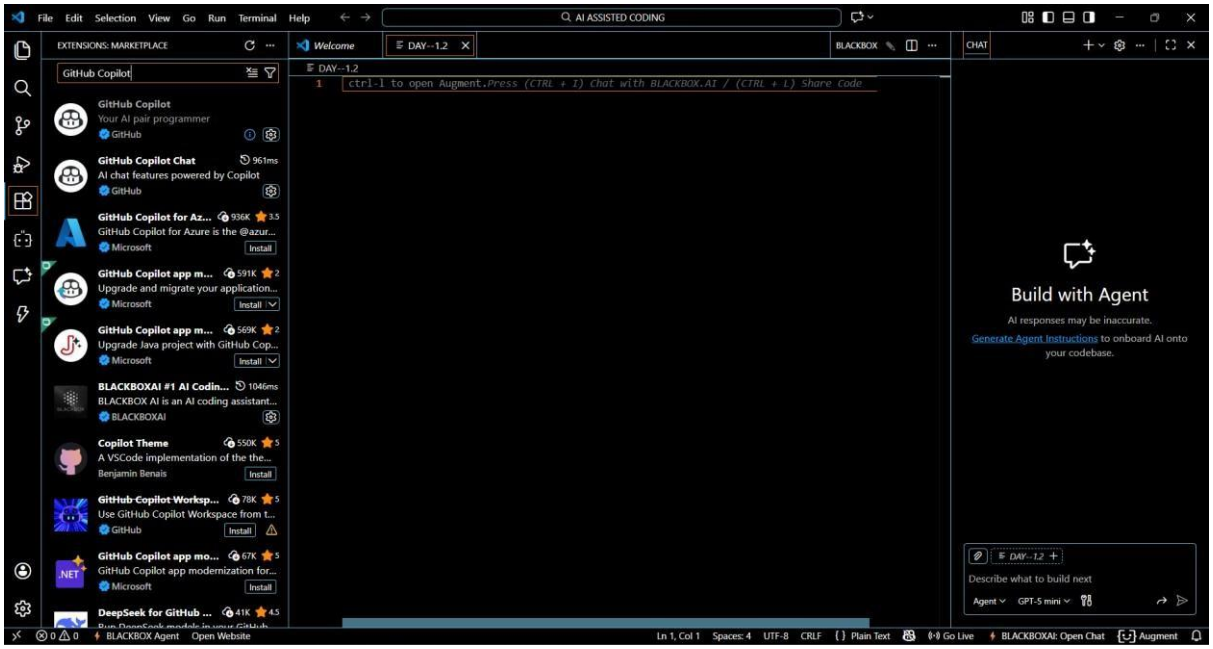
B-52

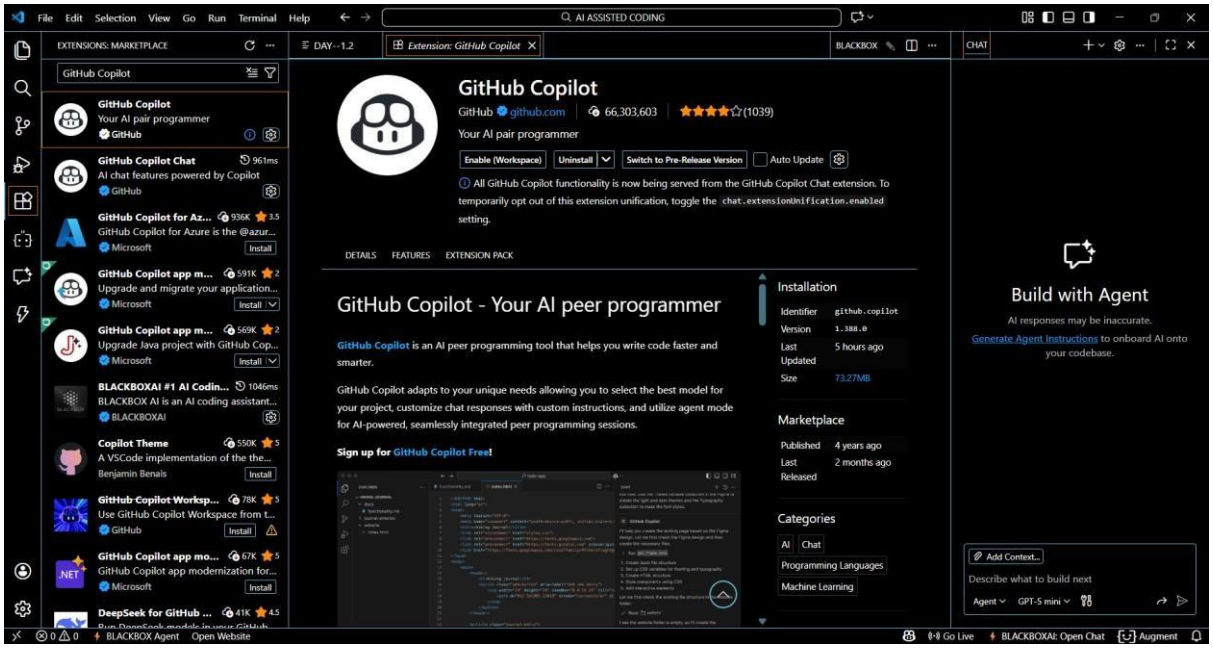
Lab 1.5: AI-Assisted Coding using GitHub Copilot

Task 0: Environment Setup

Steps:

1. Install **Visual Studio Code**
2. Open VS Code → Extensions
3. Search **GitHub Copilot**
4. Click **Install**
5. Sign in with GitHub account
6. Enable Copilot suggestions





Explanation: GitHub Copilot was installed and configured in Visual Studio Code by signing in with a GitHub account. This enables AI-based code suggestions directly inside the editor, helping developers write code faster and more efficiently.

Task 1: String Reversal Without Functions

Prompt:

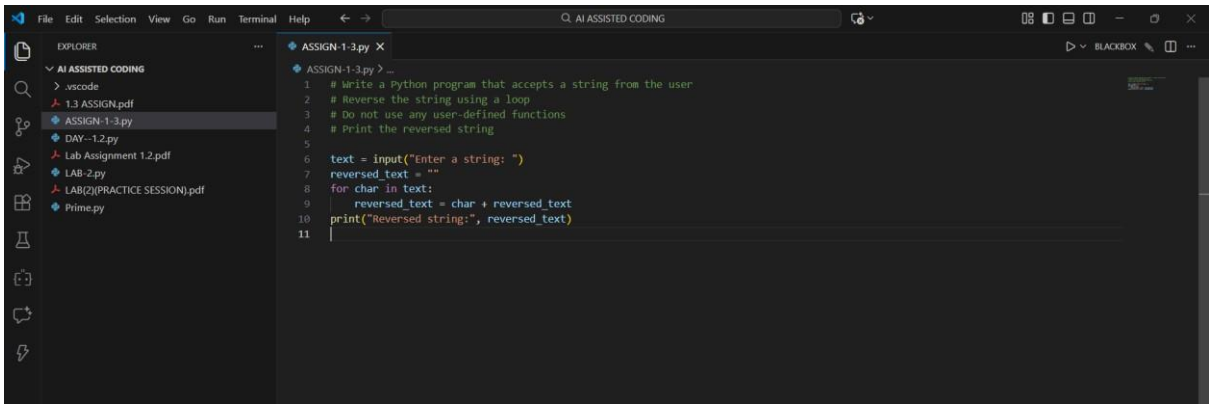
Write a Python program that accepts a string from the user

Reverse the string using a loop

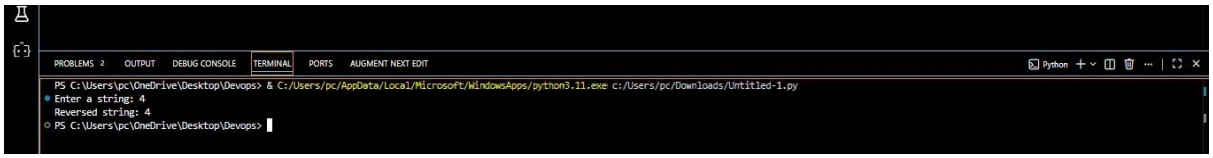
Do not use any user-defined functions

Print the reversed string

###CODE:



OUTPUT:



Explanation: In this task, GitHub Copilot generated Python code to reverse a string using a loop without defining any functions. The logic was written directly in the main program, demonstrating basic procedural programming.

Task 2: Code Optimization & Readability

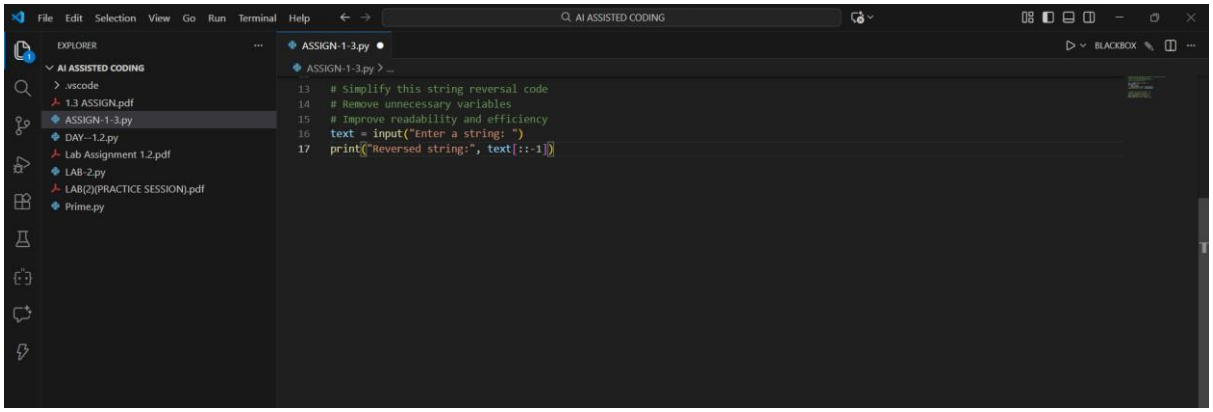
Prompt:

Simplify this string reversal code

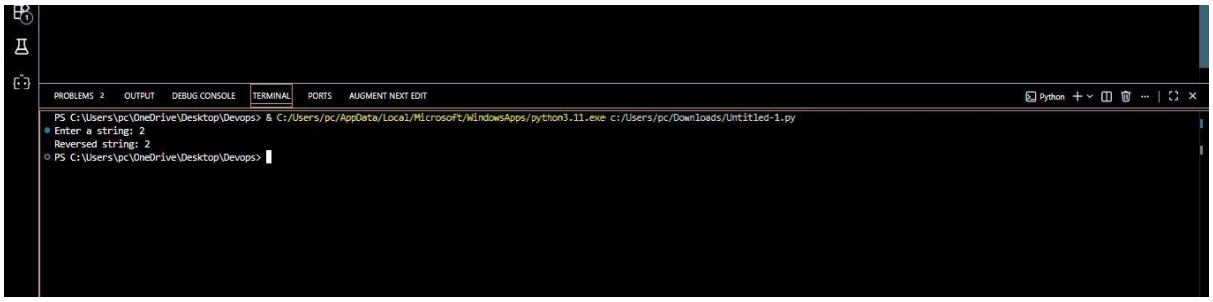
Remove unnecessary variables

Improve readability and efficiency

###CODE:



OUTPUT:



Explanation: The Copilot-generated code was optimized by simplifying the logic and removing unnecessary variables. The improved version produces the same output with better readability and reduced code complexity.

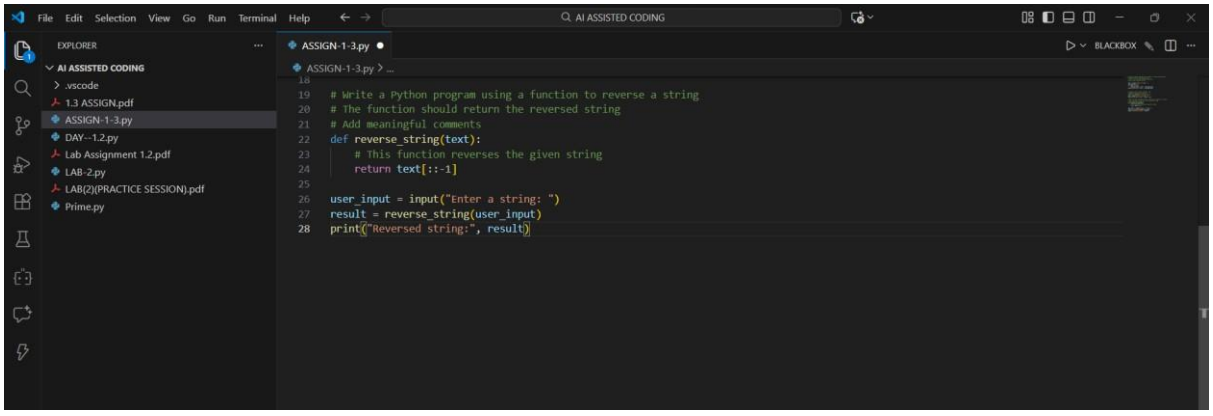
Task 3: String Reversal Using Functions

Prompt:

Write a Python program using a function to reverse a string

The function should return the reversed string

Add meaningful comments **###CODE:**



OUTPUT:

Explanation: GitHub Copilot was used to create a modular program using a user-defined function to reverse a string. This approach improves reusability, clarity, and makes the code easier to maintain.

Task 4: Procedural vs Modular Comparison

Step 1: Prompt:

Compare string reversal programs with and without functions

Discuss clarity, reusability, debugging, and scalability

Step 2: Answer (Analysis Output)

Feature	Without Function	With Function
Code clarity	Medium	High
Reusability	Low	High
Debugging	Difficult	Easy
Large applications	Not suitable	Suitable

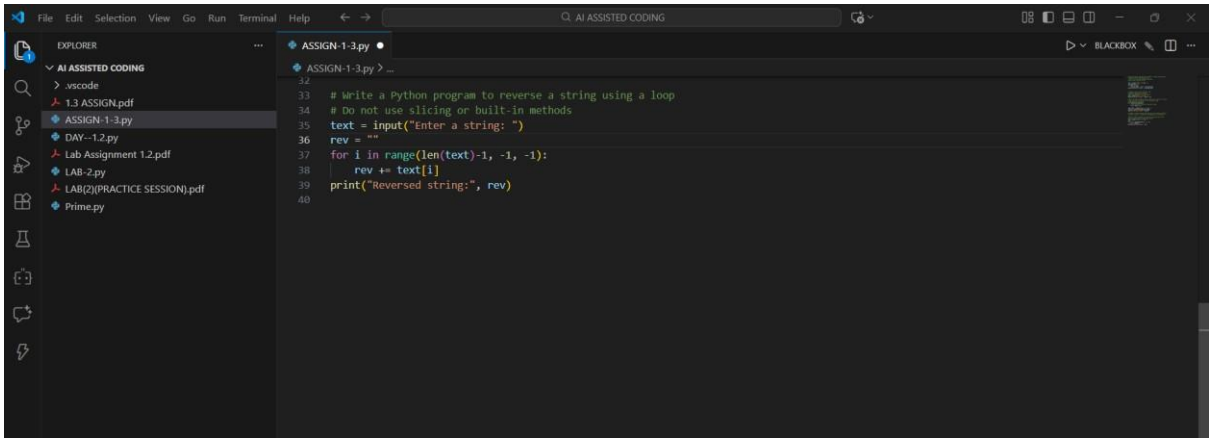
Explanation: A comparison was made between function-based and nonfunction-based programs. The analysis shows that modular code is more reusable, easier to debug, and better suited for large-scale applications.

Task 5: Loop vs Built-in Reversal

Step 1: Loop-Based Prompt

Write a Python program to reverse a string using a loop

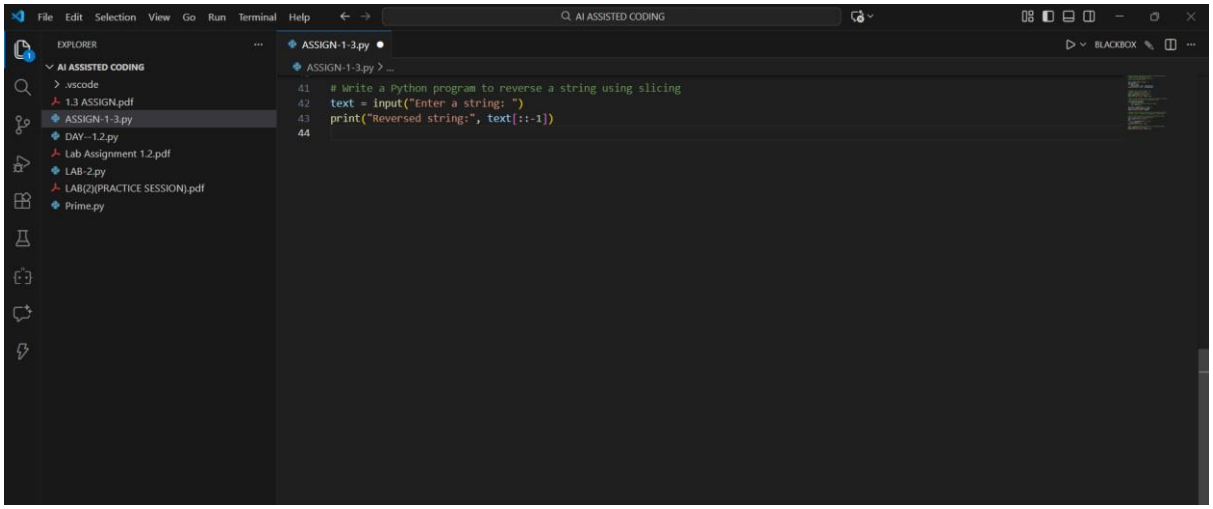
Do not use slicing or built-in methods **###CODE:**



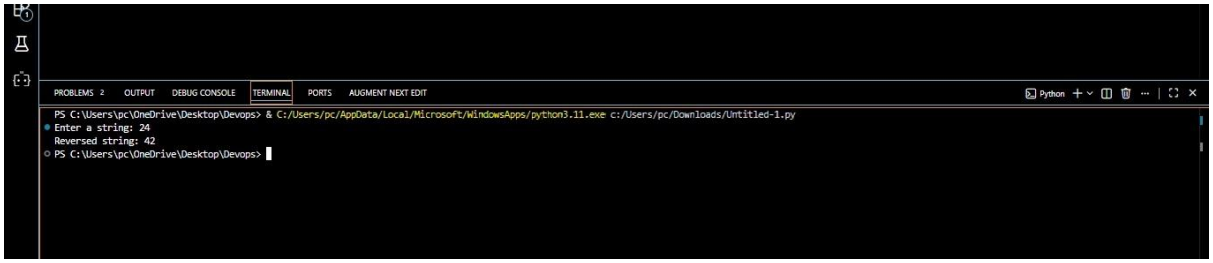
OUTPUT:

Step 2: Built-in Prompt

Write a Python program to reverse a string using slicing **###CODE:**



OUTPUT:



Explanation: Two different string reversal approaches were generated using Copilot: loop-based and built-in slicing. Both methods have the same time complexity, but the built-in approach is more concise and readable.