

Name:Guggilla Anuja Batch=06 2403A51101

SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE		DEPARTMENT OF COMPUTER SCIENCE ENGINEERING	
ProgramName: B. Tech		Assignment Type: Lab	AcademicYear:2025-2026
CourseCoordinatorName		Venkataramana Veeramsetty	
Instructor(s)Name		Dr. V. Venkataramana (Co-ordinator)	
		Dr. T. Sampath Kumar	
		Dr. Pramoda Patro	
		Dr. Brij Kishor Tiwari	
		Dr.J.Ravichander	
		Dr. Mohammand Ali Shaik	
		Dr. Anirodh Kumar	
		Mr. S.Naresh Kumar	
		Dr. RAJESH VELPULA	
		Mr. Kundhan Kumar	
		Ms. Ch.Rajitha	
		Mr. M Prakash	
		Mr. B.Raju	
		Intern 1 (Dharma teja)	
		Intern 2 (Sai Prasad)	
		Intern 3 (Sowmya)	
NS_2 (Mounika)			
CourseCode	24CS002PC215	CourseTitle	AI Assisted Coding
Year/Sem	II/I	Regulation	R24
Date and Day of Assignment	Week1 - Wednesday	Time(s)	
Duration	2 Hours	Applicable to Batches	24CSBTB01 To 24CSBTB39
AssignmentNumber: 1.3(Present assignment number)/24(Total number of assignments)			
Q.No.	Question	Expected Time to complete	
1	Lab 1: Environment Setup – GitHub Copilot and VS Code Integration Lab Objectives: <ul style="list-style-type: none"> To install and configure GitHub Copilot in Visual Studio Code. 	Week1 - Wednesday	

- To explore AI-assisted code generation using GitHub Copilot.
- To analyze the accuracy and effectiveness of Copilot's code suggestions.
- To understand prompt-based programming using comments and code context

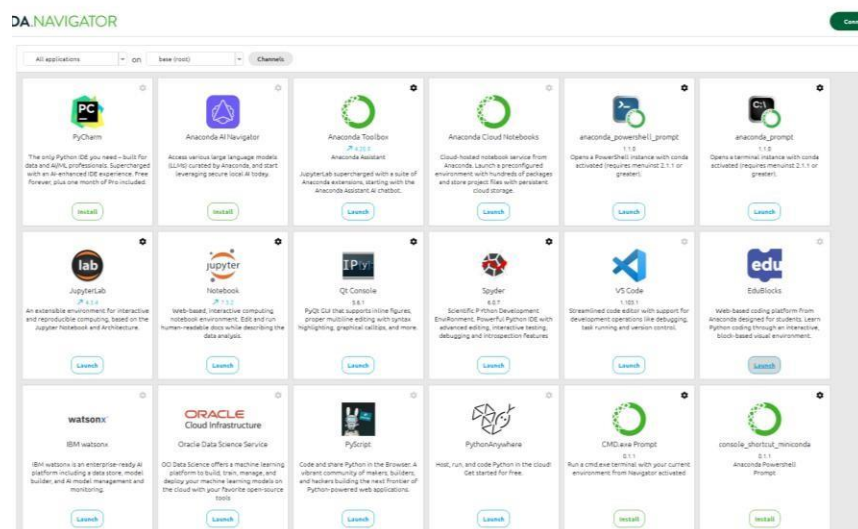
Lab Outcomes (LOs):


After completing this lab, students will be able to:

- Set up GitHub Copilot in VS Code successfully.
- Use inline comments and context to generate code with Copilot.
- Evaluate AI-generated code for correctness and readability.
- Compare code suggestions based on different prompts and programming styles.

Task Description#1

- Install and configure GitHub Copilot in VS Code. Take screenshots of each step.
-






GitHub Copilot

GitHub github.com | 46,697,127 | ★★★★★ (984)

Your AI pair programmer

[Disable](#) [Uninstall](#) [Switch to Pre-Release Version](#) ☒ Auto Update 

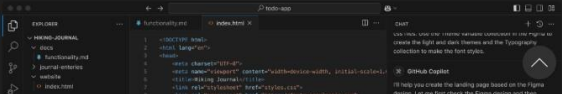
[DETAILS](#) [FEATURES](#) [EXTENSION PACK](#)

GitHub Copilot - Your AI peer programmer

GitHub Copilot is an AI peer programming tool that helps you write code faster and smarter.

GitHub Copilot adapts to your unique needs allowing you to select the best model for your project, customize chat responses with custom instructions, and utilize agent mode for AI-powered, seamlessly integrated peer programming sessions.

Sign up for GitHub Copilot Free!



Installation

Identifier	github.copilot
Version	1.350.0
Last Updated	2025-07-30, 22:31:11
Size	63.33MB

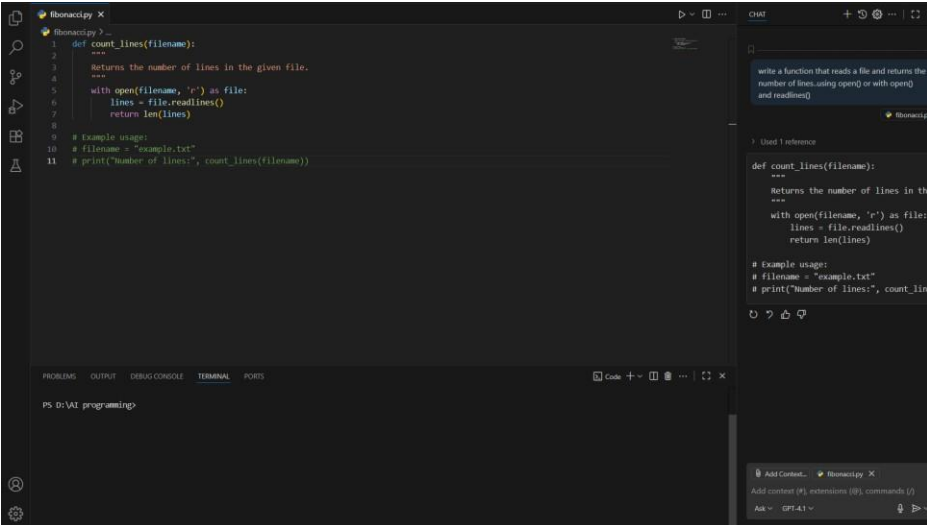
Marketplace

Published	2021-06-29, 19:56:17
Last Released	2025-08-14, 07:18:30

Categories

Expected Output#1

- Install and configure GitHub Copilot in VS Code. Take screenshots of each step.



```
def count_lines(filename):  
    """  
    Returns the number of lines in the given file.  
    """  
    with open(filename, 'r') as file:  
        lines = file.readlines()  
        return len(lines)  
  
# Example usage:  
# filename = "example.txt"  
# print("Number of lines:", count_lines(filename))
```

write a function that reads a file and returns the number of lines using open() or with open() and readlines()

```
def count_lines(filename):  
    """  
    Returns the number of lines in the  
    """  
    with open(filename, 'r') as file:  
        lines = file.readlines()  
        return len(lines)  
  
# Example usage:  
# filename = "example.txt"  
# print("Number of lines:", count_lines(filename))
```

Task Description#2

- Use Copilot to generate a `is_prime()` Python function.

```

1 def is_prime(n):
2     """
3     Returns True if n is a prime number, otherwise False.
4     """
5     if n <= 1:
6         return False
7     if n == 2:
8         return True
9     if n % 2 == 0:
10        return False
11    for i in range(3, int(n ** 0.5) + 1, 2):
12        if n % i == 0:
13            return False
14    return True
15
16 # Example usage:
17 num = int(input("Enter a number: "))
18 print("Is prime:", is_prime(num))

```

Expected Output#2

- Function to check primality with correct logic.

Task Description#3

- Write a comment like `# Function to reverse a string` and use Copilot to generate the function.

```

1 def reverse_string(s):
2     """
3     Returns the reversed version of the input string.
4     """
5     return s[::-1]
6
7 s = input("Enter a string to reverse: ")
8 print("Reversed string:", reverse_string(s))

```

Expected Output#3

- Auto-completed reverse function

Task Description#4

- Generate both recursive and iterative versions of a factorial function using comments..

Expected Output#4

- Two working factorial implementations

Criteria	Max Marks
Successful Setup of Copilot (Task #1)	0.5
is_prime() Python function (Task #2)	0.5
Reverse a string function (Task #3)	0.5
Factorial Function (Task #4)	0.5
Find the largest number (Task #5)	0.5
Total	2.5 Marks