

Name:P.HARSHITHA Batch=05 2403A510C9

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		<table border="1"> <tr><td>Dr. V. Venkataramana (Co-ordinator)</td></tr> <tr><td>Dr. T. Sampath Kumar</td></tr> <tr><td>Dr. Pramoda Patro</td></tr> <tr><td>Dr. Brij Kishor Tiwari</td></tr> <tr><td>Dr.J.Ravichander</td></tr> <tr><td>Dr. Mohammand Ali Shaik</td></tr> <tr><td>Dr. Anirodh Kumar</td></tr> <tr><td>Mr. S.Naresh Kumar</td></tr> <tr><td>Dr. RAJESH VELPULA</td></tr> <tr><td>Mr. Kundhan Kumar</td></tr> <tr><td>Ms. Ch.Rajitha</td></tr> <tr><td>Mr. M Prakash</td></tr> <tr><td>Mr. B.Raju</td></tr> <tr><td>Intern 1 (Dharma teja)</td></tr> <tr><td>Intern 2 (Sai Prasad)</td></tr> <tr><td>Intern 3 (Sowmya)</td></tr> <tr><td>NS_2 (Mounika)</td></tr> </table>		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)
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

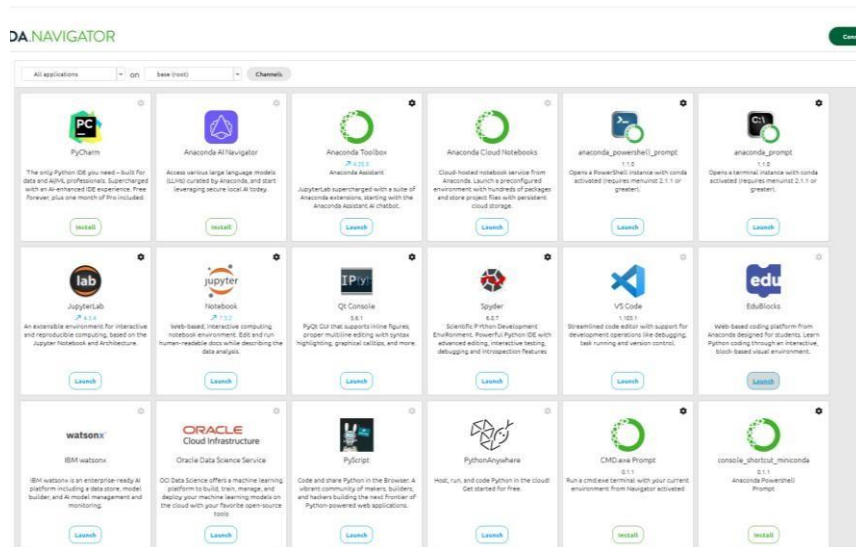
Lab Outcomes (LOs):

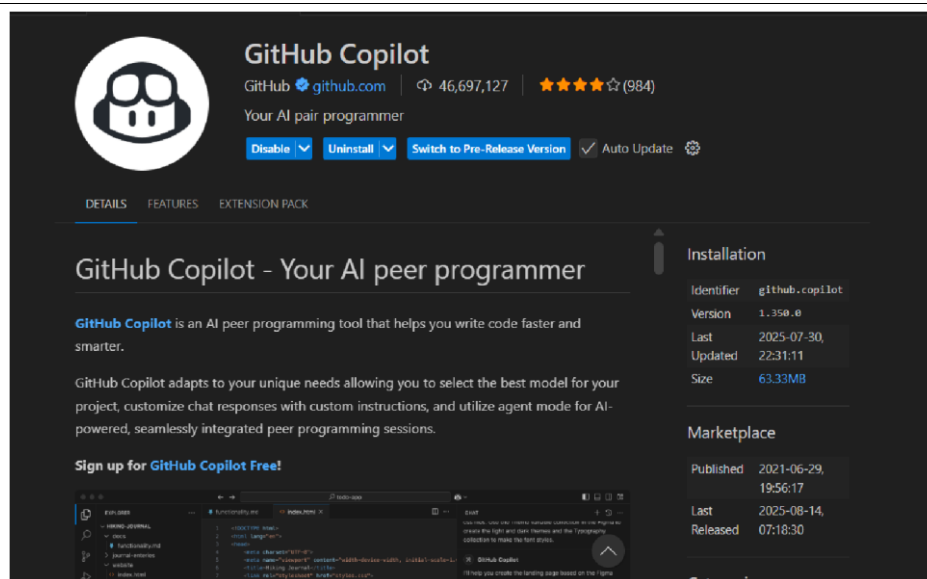
After completing this lab, students will be able to:

- 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

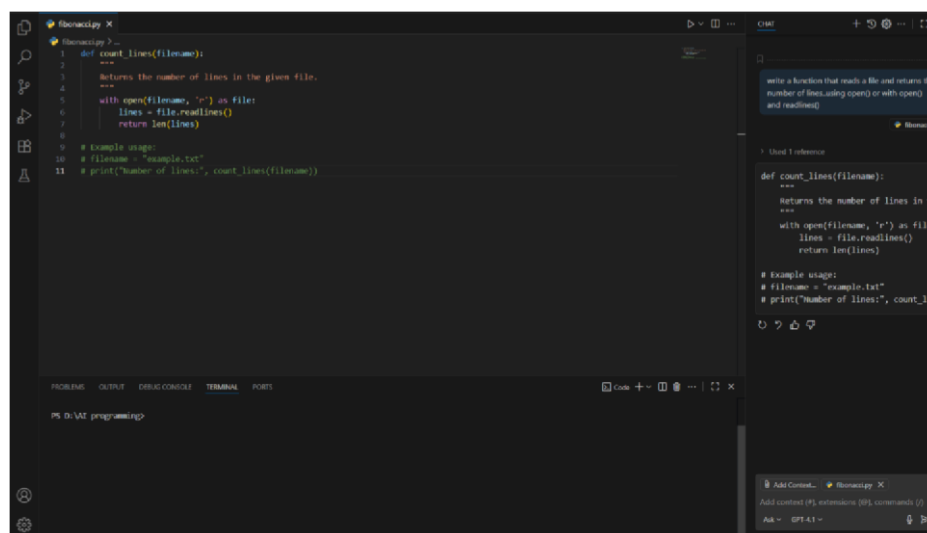
- Install and configure GitHub Copilot in VS Code. Take screenshots of each step.
-





Expected Output#1

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



- **Task Description#2**
- Use Copilot to generate a `is_prime()` Python function.

The screenshot shows a VS Code editor with a Python file named `fibonacci.py`. The code defines a function `is_prime(n)` that checks if a number is prime. It includes comments and a docstring. The function returns `True` if `n` is a prime number, otherwise `False`. The code also includes an example usage where a user is prompted to enter a number, and the function is called to check if it's prime. The terminal output shows the execution of the code, where the user enters '2' and the output is 'is prime: True'.

```

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))

```

Terminal output:

```

PS D:\AI programming> python -u "d:\AI programming\fibonacci.py"
Enter a number: 2
is prime: True
PS D:\AI programming>

```

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.

The screenshot shows a VS Code editor with a Python file named `fibonacci.py`. The code defines a function `reverse_string(s)` that returns the reversed version of the input string. It includes comments and a docstring. The function returns `s[::-1]`. The code also includes an example usage where a user is prompted to enter a string, and the function is called to reverse it. The terminal output shows the execution of the code, where the user enters 'abc' and the output is 'Reversed string: cba'.

```

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

```

Terminal output:

```

PS D:\AI programming> python -u "d:\AI programming\fibonacci.py"
Enter a string to reverse: abc
Reversed string: cba
PS D:\AI programming>

```

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

	Find the largest number (Task #5)	0.5	
	Total	2.5 Marks	