

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 - Tuesday	Time(s)	
Duration	2 Hours	Applicableto Batches	24CSBTB01 To 24CSBTB39
AssignmentNumber:1.2(Present assignment number)/24(Total number of assignments)			
Q.No.	Question	ExpectedTime to complete	
1	<p>Lab 1: Environment Setup – GitHub Copilot and VS Code Integration</p> <p><b>Lab Objectives:</b></p> <ul style="list-style-type: none"> <li>To install and configure GitHub Copilot in Visual Studio Code.</li> <li>To explore AI-assisted code generation using GitHub Copilot.</li> <li>To analyze the accuracy and effectiveness of Copilot's code suggestions.</li> <li>To understand prompt-based programming using comments and code context</li> </ul> <p><b>Lab Outcomes (LOs):</b> After completing this lab, students will be able to:</p>	Week1 - wednesday	

- 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

- Write a comment: # Function to check if a string is a valid palindrome (ignoring spaces and case) and allow Copilot to complete it.
- **Prompt :** write a python code for a function to check if a string is a valid palindrome

#### Expected Output#1

```

C:\Users\ navee > OneDrive > Desktop > MyFirstWebsite > Untitled-1.py > ...
1  # Function to check if a string is a valid palindrome (ignoring spaces and case)
2  def is_valid_palindrome(s: str) -> bool:
3      # Normalize the string by removing spaces and converting to lowercase
4      normalized_str = ''.join(s.split()).lower()
5
6      # Check if the normalized string is equal to its reverse
7      return normalized_str == normalized_str[::-1]
8
9  # Example usage
10 if __name__ == "__main__":
11     test_strings = [
12
13         "emordnilap",
14         "Go hang a salami I'm a lasagna hog",
15         "Mr. Owl ate my metal worm",
16         "Never odd or even",
17         "Yo banana boy",
18         "Eva, can I see bees in a cave?",
19         "Doc, note I dissent. A fast never prevents a fatness. I diet on cod.",
20         "Madam, in Eden, I'm Adam",
21     ]
22     for s in test_strings:
23         result = is_valid_palindrome(s)
24         print(f'{s} -> Palindrome: {result}')
25
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS C:\Users\ navee\AppData\Local\Programs\Microsoft VS Code> & C:\Users\ navee\AppData\Local\Programs
\Python\Python313\python.exe c:/Users/navee/OneDrive/Desktop/MyFirstWebsite/Untitled-1.py
'emordnilap' -> Palindrome: False
'Go hang a salami I'm a lasagna hog' -> Palindrome: False
'Mr. Owl ate my metal worm' -> Palindrome: False
'Never odd or even' -> Palindrome: True
'Yo banana boy' -> Palindrome: True
'Eva, can I see bees in a cave?' -> Palindrome: False
'Doc, note I dissent. A fast never prevents a fatness. I diet on cod.' -> Palindrome: False
'Madam, in Eden, I'm Adam' -> Palindrome: False
PS C:\Users\ navee\AppData\Local\Programs\Microsoft VS Code>

```

#### Task Description#2

- Generate a Python function that returns the Fibonacci sequence up to n terms. Prompt with only a function header and docstring
- **Prompt :** write a python code for a function that returns the Fibonacci sequence up to n terms

#### Expected Output#2

```
my1.rdp | index.html | Untitled-1.py x
C: > Users > navee > OneDrive > Desktop > MyFirstWebsite > Untitled-1.py > ...
1 def fibonacci_sequence(n):
2     """
3     Returns a list containing the Fibonacci sequence up to n terms.
4
5     Args:
6         n (int): The number of terms to generate.
7
8     Returns:
9         List[int]: The Fibonacci sequence as a list.
10    """
11    sequence = []
12    a, b = 0, 1
13    for _ in range(n):
14        sequence.append(a)
15        a, b = b, a + b
16    return sequence
17
18 # Example usage
19 n=int(input("Enter the number of terms :"))
20 print(fibonacci_sequence(n))

PROBLEMS | OUTPUT | DEBUG CONSOLE | TERMINAL | PORTS
PS C:\Users\navee\AppData\Local\Programs\Microsoft VS Code> & C:\Users\navee\AppData\Local\Programs\Python\Python313\python.exe c:/Users/navee/OneDrive/Desktop/MyFirstWebsite/Untitled-1.py
Enter the number of terms :12
[0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]
PS C:\Users\navee\AppData\Local\Programs\Microsoft VS Code> |
```

**Task Description#3**

- Write a comment like # Function to reverse a string and use Copilot to generate the function.
- **Prompt :** write a python code for Function to reverse a string

**Expected Output#3**

```
my1.rdp | index.html | Untitled-1.py X
C:\Users\ navee > OneDrive > Desktop > MyFirstWebsite > Untitled-1.py > ...
1  def reverse_string(s):
2      """
3      Returns the reversed version of the input string.
4
5      Args:
6      |   s (str): The string to reverse.
7
8      Returns:
9      |   str: The reversed string.
10     """
11     return s[::-1]
12
13 # Example usage
14 print(reverse_string("hello"))
15 print(reverse_string("Python123"))

PROBLEMS | OUTPUT | DEBUG CONSOLE | TERMINAL | PORTS
PS C:\Users\ navee\AppData\Local\Programs\Microsoft VS Code> & C:\Users\ navee\AppData\Local\Programs\Python\Python313\python.exe c:/Users/ navee/OneDrive/Desktop/MyFirstWebsite/Untitled-1.py
olleh
321nohtyP
PS C:\Users\ navee\AppData\Local\Programs\Microsoft VS Code> |
```

Task Description#4

- Generate a program that simulates a basic calculator (add, subtract, multiply, divide). Write the comment: # Simple calculator with 4 operations and let AI complete it.
- **Prompt :** python code only for a program that simulates a basic calculator (add, subtract, multiply, divide)

Expected Output#4

```
my1.rdp index.html Untitled-1.py X
C:\Users\navee> OneDrive > Desktop > MyFirstWebsite > Untitled-1.py > ...
1 # Simple calculator with 4 operation C:\Users\navee\OneDrive\Desktop\MyFirstWebsite
2 def calculator(a, b, operation):
3     """
4     Performs basic arithmetic operations on two numbers.
5
6     Args:
7         a (float): The first number.
8         b (float): The second number.
9         operation (str): The operation to perform ('add', 'subtract', 'multiply', 'divide').
10
11     Returns:
12         float: The result of the operation.
13     """
14     if operation == 'add':
15         return a + b
16     elif operation == 'subtract':
17         return a - b
18     elif operation == 'multiply':
19         return a * b
20     elif operation == 'divide':
21         if b != 0:
22             return a / b
23         else:
24             raise ValueError("Cannot divide by zero.")
25     else:
26         raise ValueError("Invalid operation. Choose from 'add', 'subtract', 'multiply', or 'divide'.")
27
28 # Example usage
29 print(calculator(10, 5, 'add'))
30 print(calculator(10, 5, 'subtract'))
31 print(calculator(10, 5, 'multiply'))
32 print(calculator(10, 5, 'divide'))

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\navee\AppData\Local\Programs\Microsoft VS Code> & C:\Users\navee\AppData\Local\Programs\Python\Python313\python.exe c:/Users/navee/OneDrive/Desktop/MyFirstWebsite/Untitled-1.py
15
5
50
2.0
PS C:\Users\navee\AppData\Local\Programs\Microsoft VS Code> 
```

#### Task Description#5

- Use a comment to instruct AI to write a function that reads a file and returns the number of lines..
- **Prompt :** python code only for a function that reads a file and returns the number of lines..
- 

#### Expected Output#5

```
index.html Untitled-1.py X hi.txt
C:\Users\navee> OneDrive > Desktop > MyFirstWebsite > Untitled-1.py > ...
1 def count_lines_in_file(filename):
2     """
3     Return the number of lines in the given file.
4
5     Example:
6         >>> count_lines_in_file("sample.txt")
7         5
8     """
9     with open(filename, 'r') as file:
10         return sum(1 for _ in file)
11
12
13 # Example usage
14 filename = "sample.txt"
15
16 # Create a sample file for demonstration
17 with open(filename, 'w') as f:
18     f.write("Line 1\nLine 2\nLine 3\nLine 4\nLine 5\n")
19
20 print(count_lines_in_file(filename)) # Output: 5
21

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\navee\AppData\Local\Programs\Microsoft VS Code> & C:\Users\navee\AppData\Local\Programs\Python\Python313\python.exe c:/Users/navee/OneDrive/Desktop/MyFirstWebsite/Untitled-1.py
5
PS C:\Users\navee\AppData\Local\Programs\Microsoft VS Code> 
```

**Note:** Report should be submitted a word document for all tasks in a single document with prompts, comments & code explanation, and output and if required, screenshots

**Evaluation Criteria:**

	Criteria	Max Marks		
	Task #1	0.5		
	Task #2	0.5		
	Task #3	0.5		
	Task #4	0.5		
	Task #5	0.5		
	<b>Total</b>	<b>2.5 Marks</b>		