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	Week4 - Wednesday	**Time(s)**	
Duration	2 Hours	**Applicableto Batches**	

AssignmentNumber:9.3(Present assignment number)/**24**(Total number of assignments)

Q.No.	**Question**	***ExpectedTime to complete***
1	Lab 9: Documentation Generation: Automatic documentation and code comments **Lab Objectives:** • To understand the importance of documentation and code comments in software development. • To explore how AI-assisted coding tools can generate meaningful documentation and	Week4 - Wednesday

inline comments.

- To practice generating function-level and module-level docstrings automatically.
- To evaluate the quality, accuracy, and limitations of AI-generated documentation.
- To develop a small automated tool for documentation generation in Python..

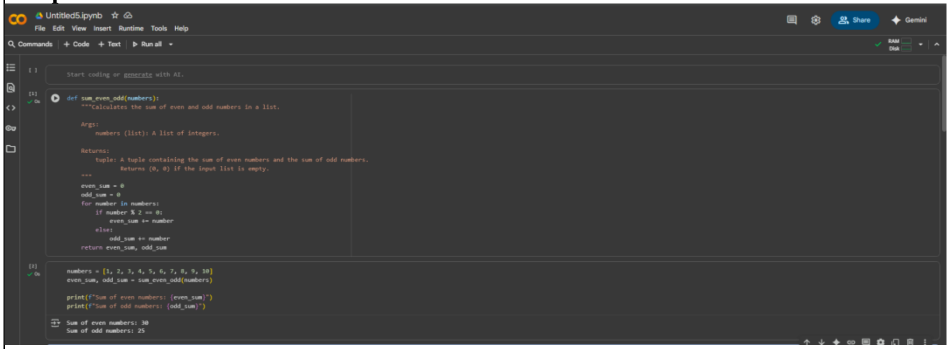
Lab Outcomes (LOs):
After completing this lab, students will be able to:

- Apply AI-assisted coding tools to generate docstrings and inline comments for Python code.
- Critically analyze AI-generated documentation for correctness, completeness, and readability.
- Create structured documentation (function-level, module-level) following standard formats.
- Design and implement a mini documentation generator tool to automate code commenting and docstring creation.

Task Description#1 Basic Docstring Generation

- Write python function to return sum of even and odd numbers in the given list.
- Incorporate manual **docstring** in code with Google Style
- Use an AI-assisted tool (e.g., Gemini, Copilot, Cursor AI) to generate a docstring describing the function.
- Compare the AI-generated docstring with your manually written one.

Expected Outcome#1:


Untitled5.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text Run all
Share
Gemini
Start coding or generate with AI.

```
def sum_even_odd(numbers):
    """Calculates the sum of even and odd numbers in a list.

    Args:
        numbers (list): A list of integers.

    Returns:
        tuple: A tuple containing the sum of even numbers and the sum of odd numbers.
               Returns (0, 0) if the input list is empty.
    """
    even_sum = 0
    odd_sum = 0
    for number in numbers:
        if number % 2 == 0:
            even_sum += number
        else:
            odd_sum += number
    return even_sum, odd_sum
```

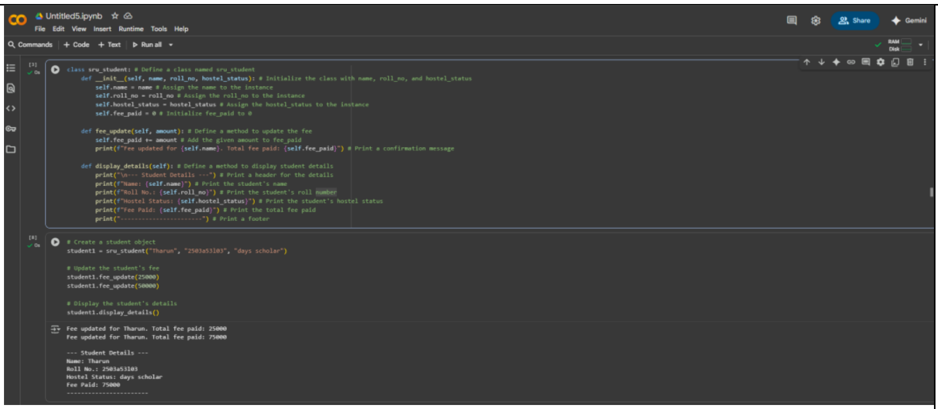
```
numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
even_sum, odd_sum = sum_even_odd(numbers)

print(f"Sum of even numbers: {even_sum}")
print(f"Sum of odd numbers: {odd_sum}")
```

```
Sum of even numbers: 30
Sum of odd numbers: 25
```

Task Description#2 Automatic Inline Comments

- Write python program for **sru_student** class with attributes like name, roll no., hostel_status and **fee_update** method and **display_details** method.
- Write comments manually for each line/code block
- Ask an AI tool to add inline comments explaining each line/step.
- Compare the AI-generated comments with your manually written one.

Expected Output#2:



```
class sru_student: # Define a class named sru_student
    def __init__(self, name, roll_no, hostel_status): # Initialize the class with name, roll_no, and hostel_status
        self.name = name # Assign the name to the instance
        self.roll_no = roll_no # Assign the roll_no to the instance
        self.hostel_status = hostel_status # Assign the hostel_status to the instance
        self.fee_paid = 0 # Initialize fee_paid to 0

    def fee_update(self, amount): # Define a method to update the fee
        self.fee_paid += amount # Add the given amount to fee_paid
        print(f"Fee updated for {self.name}. Total fee paid: {self.fee_paid}") # Print a confirmation message

    def display_details(self): # Define a method to display student details
        print("\n--- Student Details ---") # Print a header for the details
        print(f"Name: {self.name}") # Print the student's name
        print(f"Roll No.: {self.roll_no}") # Print the student's roll number
        print(f"Hostel Status: {self.hostel_status}") # Print the student's hostel status
        print(f"Fee Paid: {self.fee_paid}") # Print the total fee paid
        print("-----------------------") # Print a footer
```

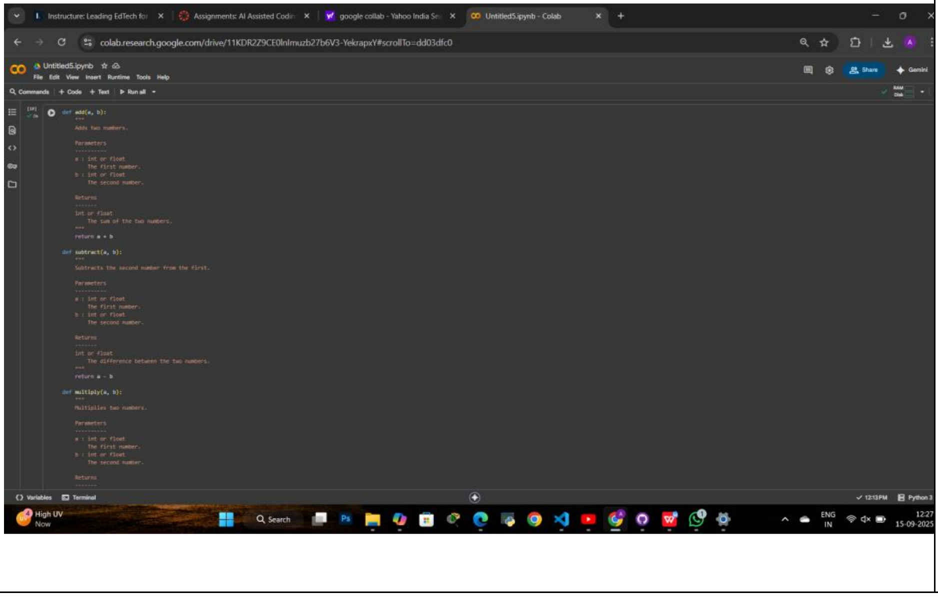
```
# Create a student object
student1 = sru_student("Tharun", "2503a53103", "days scholar")

# Update the student's fee
student1.fee_update(25000)
student1.fee_update(50000)

# Display the student's details
student1.display_details()
```

```
Fee updated for Tharun. Total fee paid: 25000
Fee updated for Tharun. Total fee paid: 75000

--- Student Details ---
Name: Tharun
Roll No.: 2503a53103
Hostel Status: days scholar
Fee Paid: 75000
-----------------------
```

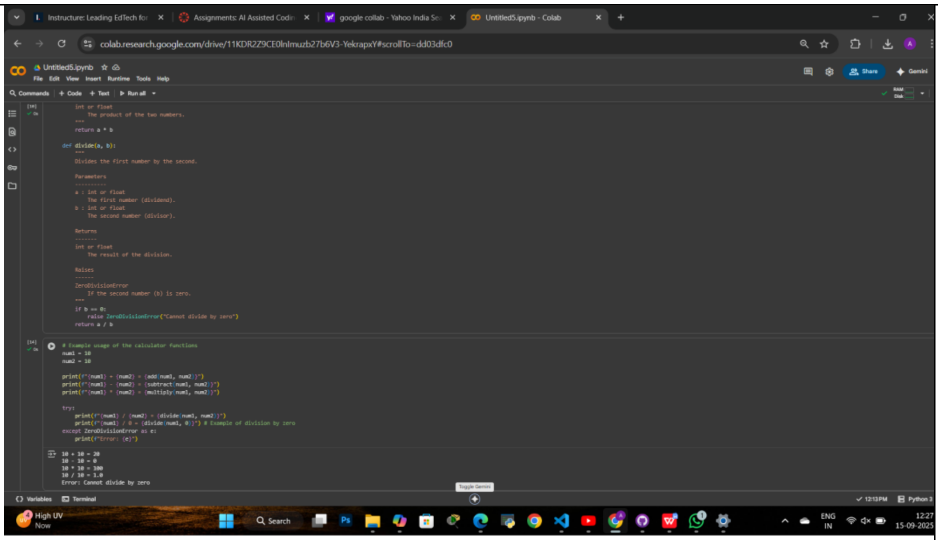
Task Description#3

- Write a Python script with 3–4 functions (e.g., calculator: add, subtract, multiply, divide).
- Incorporate manual **docstring** in code with NumPy Style
- Use AI assistance to generate a module-level docstring + individual function docstrings.
- Compare the AI-generated docstring with your manually written one.

Expected Output#3:


Instructure: Leading EdTech for
Assignments: AI Assisted Codin
google collab - Yahoo India Se
Untitled5.ipynb - Colab
colab.research.google.com/drive/11KDR2Z9CE0lnlmuzb27b6V3-YekrapxY#scrollTo=dd03dfc0
Untitled5.ipynb
File Edit View Insert Runtime Tools Help
Commands
+ Code
+ Text
Run all
Share
Gemini
Variables
Terminal
12:13 PM
Python 3
High UV
Now
Search
ENG
IN
12:27
15-09-2025

```
def add(a, b):
    """
    Adds two numbers.

    Parameters
    ----------
    a : int or float
        The first number.
    b : int or float
        The second number.

    Returns
    -------
    int or float
        The sum of the two numbers.
    """
    return a + b

def subtract(a, b):
    """
    Subtracts the second number from the first.

    Parameters
    ----------
    a : int or float
        The first number.
    b : int or float
        The second number.

    Returns
    -------
    int or float
        The difference between the two numbers.
    """
    return a - b

def multiply(a, b):
    """
    Multiplies two numbers.

    Parameters
    ----------
    a : int or float
        The first number.
    b : int or float
        The second number.

    Returns
    -------
```

	
	colab.research.google.com/drive/11KDR2Z9CE0lnImuzb27b6V3-YekrapxY#scrollTo=dd03dfc0
	Untitled5.ipynb
	```
	    int or float
	        The product of the two numbers.
	    """
	    return a * b

	def divide(a, b):
	    """
	    Divides the first number by the second.

	    Parameters
	    ----------
	    a : int or float
	        The first number (dividend).
	    b : int or float
	        The second number (divisor).

	    Returns
	    -------
	    int or float
	        The result of the division.

	    Raises
	    ------
	    ZeroDivisionError
	        If the second number (b) is zero.
	    """
	    if b == 0:
	        raise ZeroDivisionError("Cannot divide by zero")
	    return a / b
	```
	```
	# Example usage of the calculator functions
	num1 = 10
	num2 = 10

	print(f"{num1} + {num2} = {add(num1, num2)}")
	print(f"{num1} - {num2} = {subtract(num1, num2)}")
	print(f"{num1} * {num2} = {multiply(num1, num2)}")

	try:
	    print(f"{num1} / {num2} = {divide(num1, num2)}")
	    print(f"{num1} / 0 = {divide(num1, 0)}") # Example of division by zero
	except ZeroDivisionError as e:
	    print(f"Error: {e}")
	```
	```
	10 + 10 = 20
	10 - 10 = 0
	10 * 10 = 100
	10 / 10 = 1.0
	Error: Cannot divide by zero
	```
	 Instructure: Leading EdTech for
	Assignments: AI Assisted Codin
	google collab - Yahoo India Sea
	Untitled5.ipynb - Colab
	File Edit View Insert Runtime Tools Help
	Commands + Code + Text Run all
	Share
	Gemini
	Toggle Gemini
	Variables Terminal
	12:13 PM
	Python 3
	High UV
	Now
	Search
	ENG
	IN
	12:27
	15-09-2025
	
	Push documentation whole workspace as .md file in GitHub Repository
	
	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	