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

Q.No.	Question	*ExpectedTime to complete*
1	Lab 7: AI-Error Debugging with AI: Systematic approaches to finding and	Week4 - Wednesday

fixing bugs

Lab Objectives:

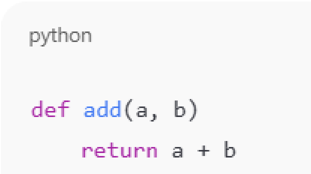
- To identify and correct syntax, logic, and runtime errors in Python programs using AI tools.
- To understand common programming bugs and AI-assisted debugging suggestions.
- To evaluate how AI explains, detects, and fixes different types of coding errors.
- To build confidence in using AI to perform structured debugging practices.

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

- Use AI tools to detect and correct syntax, logic, and runtime errors.
- Interpret AI-suggested bug fixes and explanations.
- Apply systematic debugging strategies supported by AI-generated insights.
- Refactor buggy code using responsible and reliable programming patterns.

Task Description#1

- Paste a function with a missing colon (add(a, b)), and let AI fix the syntax error.



python

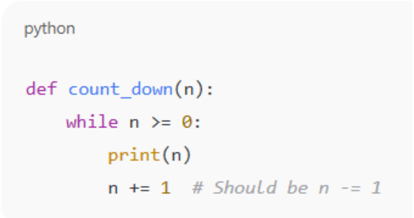
```python
def add(a, b)
    return a + b
```

Expected Output#1

- Corrected function with syntax fix

Task Description#2 (Loops)

- Identify and fix a logic error in a loop that causes infinite iteration.



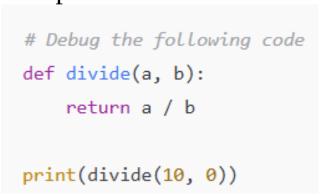
```python
def count_down(n):
    while n >= 0:
        print(n)
        n += 1  # Should be n -= 1
```

Expected Output#2

- AI fixes increment/decrement error

Task Description#3

- Debug a runtime error caused by division by zero. Let AI insert try-except.



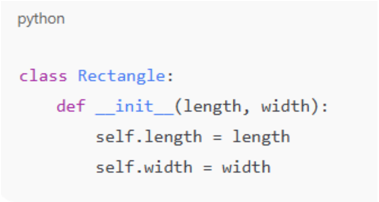
```python
# Debug the following code
def divide(a, b):
    return a / b

print(divide(10, 0))
```

Expected Output#3

- Corrected function with safe error handling

Task Description#4

- Provide a faulty class definition (missing self in parameters). Let AI fix it



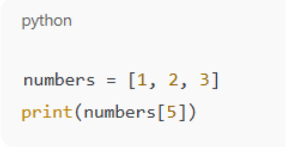
```python
class Rectangle:
    def __init__(length, width):
        self.length = length
        self.width = width
```

Expected Output#4

- Correct __init__() method and explanation

Task Description#5

- Access an invalid list index and use AI to resolve the Index Error.



```python
numbers = [1, 2, 3]
print(numbers[5])
```

Expected Output#5

- AI suggests checking length or using safe access logic

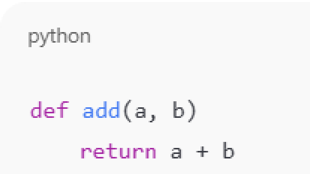
Note: Report should be submitted a word document for all tasks in a single document with prompts, comments & code explanation, and output and if required, screenshots

Evaluation Criteria:

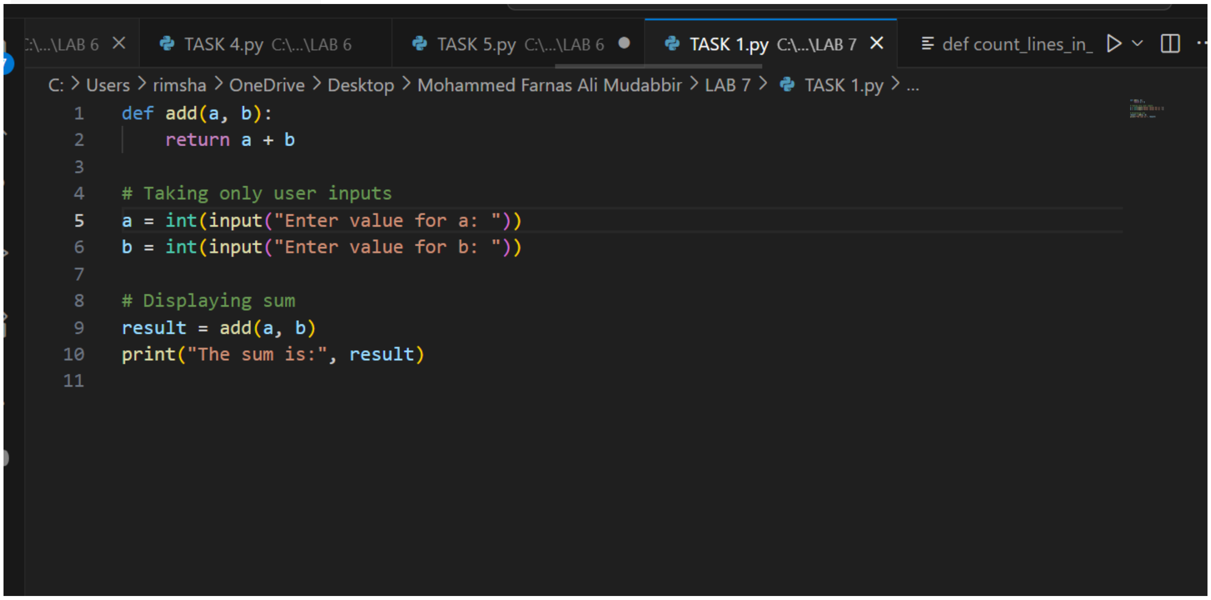
Criteria	Max Marks
Identification of bugs	0.5
Application of AI-suggested fixes	0.5
Explanation and understanding of errors	0.5
Corrected code functionality	0.5
Report structure and reflection	0.5
Total	**2.5 Marks**

Task Description#1

- Paste a function with a missing colon (add(a, b)), and let AI fix the syntax error.



```python
def add(a, b)
    return a + b
```



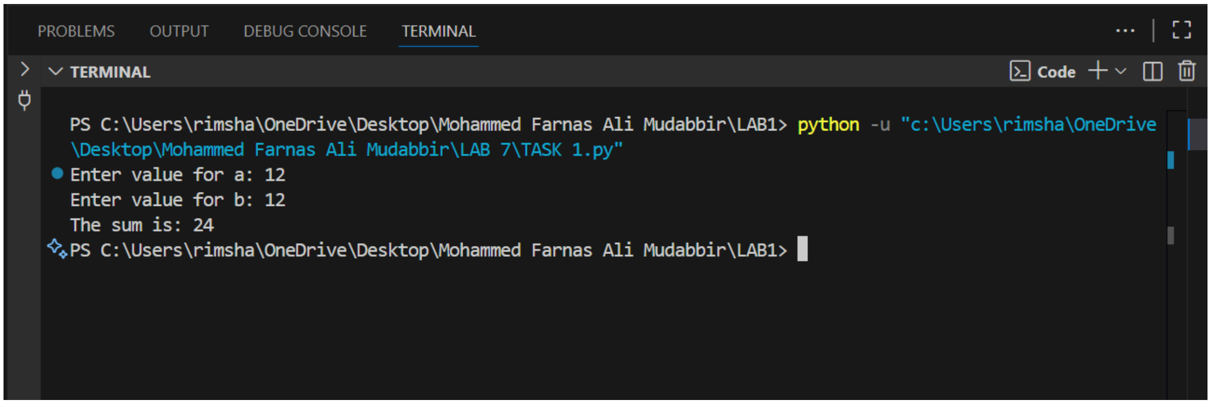
C: > Users > rimsha > OneDrive > Desktop > Mohammed Farnas Ali Mudabbir > LAB 7 > TASK 1.py > ...

```python
def add(a, b):
    return a + b

# Taking only user inputs
a = int(input("Enter value for a: "))
b = int(input("Enter value for b: "))

# Displaying sum
result = add(a, b)
print("The sum is:", result)

```

Expected Output#1

- Corrected function with syntax fix

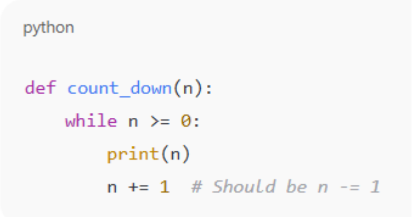
Practical output:



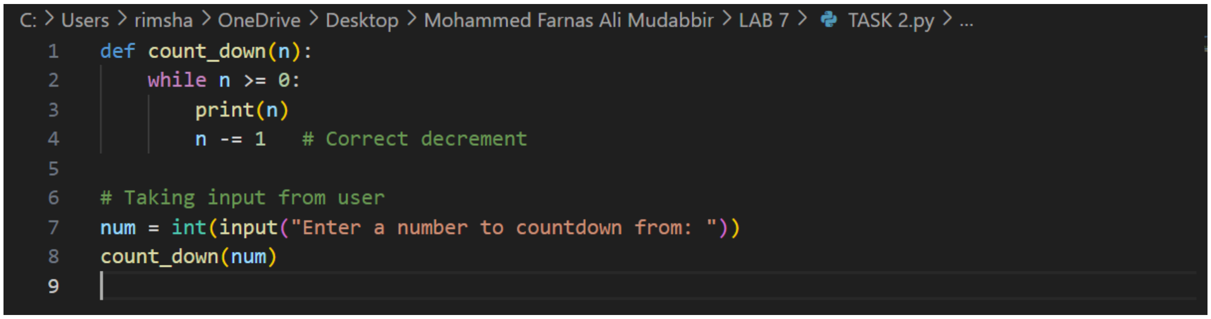
```
PS C:\Users\rimsha\OneDrive\Desktop\Mohammed Farnas Ali Mudabbir\LAB1> python -u "c:\Users\rimsha\OneDrive
\Desktop\Mohammed Farnas Ali Mudabbir\LAB 7\TASK 1.py"
Enter value for a: 12
Enter value for b: 12
The sum is: 24
PS C:\Users\rimsha\OneDrive\Desktop\Mohammed Farnas Ali Mudabbir\LAB1>
```

Task Description#2 (Loops)

- Identify and fix a logic error in a loop that causes infinite iteration.



```python
def count_down(n):
    while n >= 0:
        print(n)
        n += 1  # Should be n -= 1
```

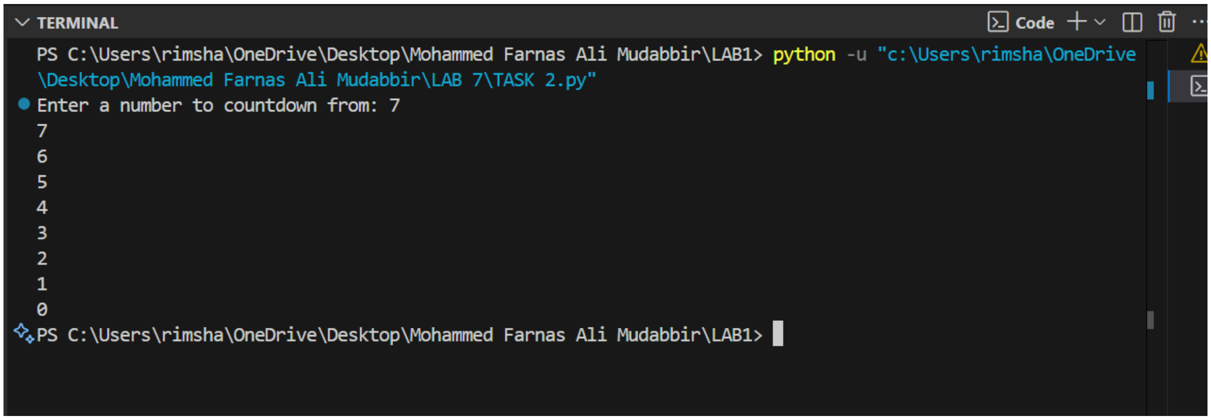

C: > Users > rimsha > OneDrive > Desktop > Mohammed Farnas Ali Mudabbir > LAB 7 > TASK 2.py > ...

```python
def count_down(n):
    while n >= 0:
        print(n)
        n -= 1   # Correct decrement

# Taking input from user
num = int(input("Enter a number to countdown from: "))
count_down(num)

```

Expected Output#2

- AI fixes increment/decrement error

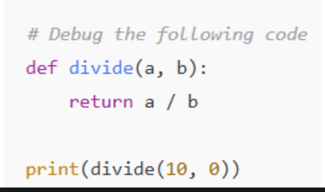
Practical output:



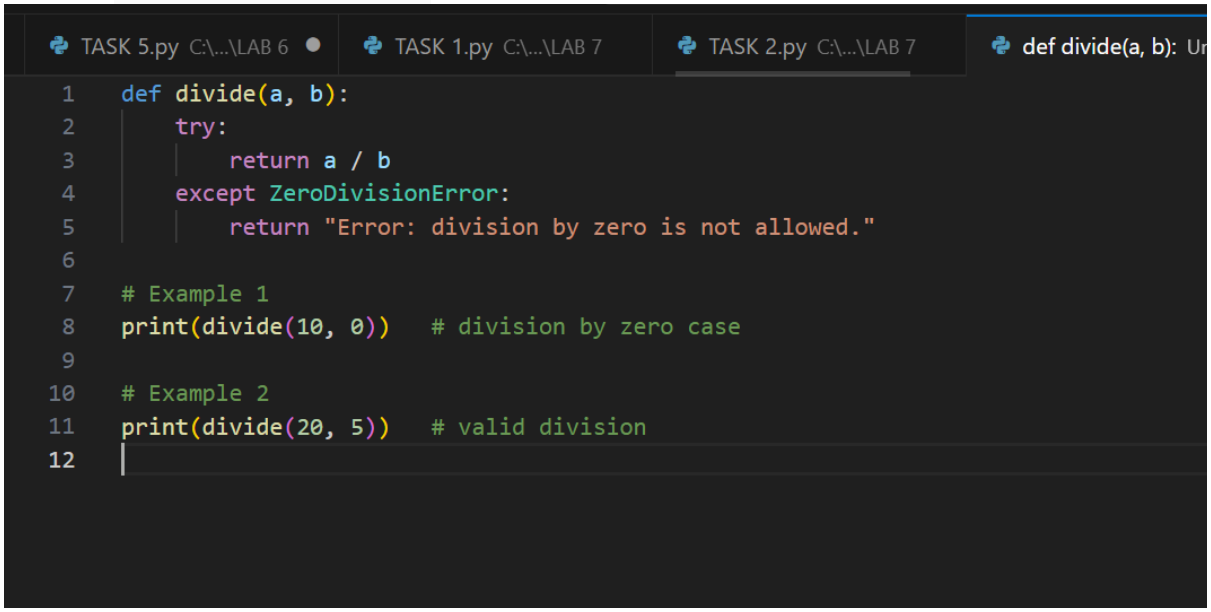
```
PS C:\Users\rimsha\OneDrive\Desktop\Mohammed Farnas Ali Mudabbir\LAB1> python -u "c:\Users\rimsha\OneDrive
\Desktop\Mohammed Farnas Ali Mudabbir\LAB 7\TASK 2.py"
Enter a number to countdown from: 7
7
6
5
4
3
2
1
0
PS C:\Users\rimsha\OneDrive\Desktop\Mohammed Farnas Ali Mudabbir\LAB1>
```

Task Description#3

- Debug a runtime error caused by division by zero. Let AI insert try-except.



```python
# Debug the following code
def divide(a, b):
    return a / b

print(divide(10, 0))
```

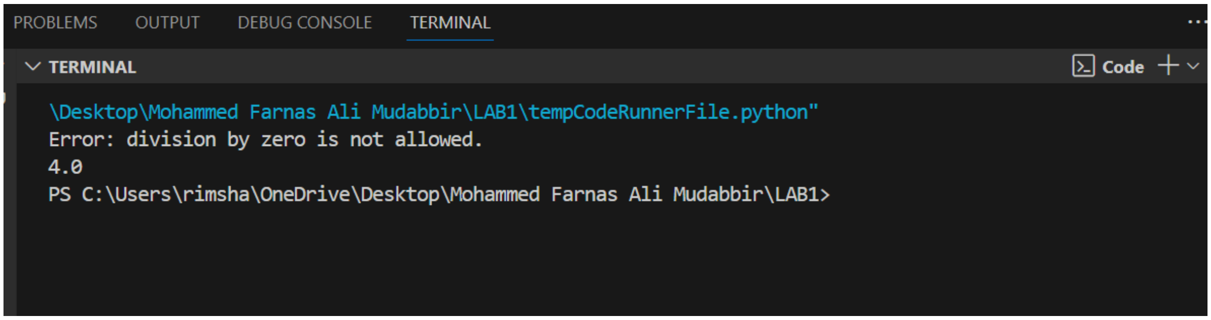


```python
def divide(a, b):
    try:
        return a / b
    except ZeroDivisionError:
        return "Error: division by zero is not allowed."

# Example 1
print(divide(10, 0))   # division by zero case

# Example 2
print(divide(20, 5))   # valid division

```

Expected Output#3

- Corrected function with safe error handling

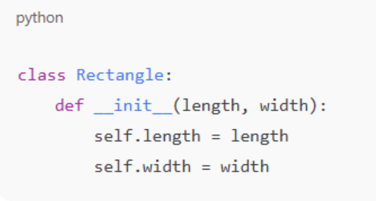
Practical output:



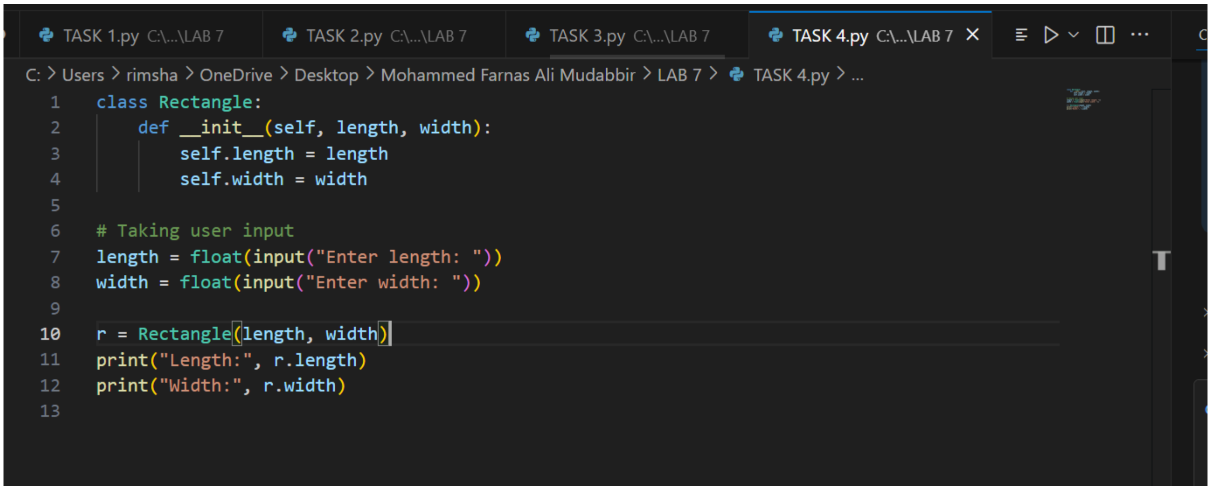
```
\Desktop\Mohammed Farnas Ali Mudabbir\LAB1\tempCodeRunnerFile.python"
Error: division by zero is not allowed.
4.0
PS C:\Users\rimsha\OneDrive\Desktop\Mohammed Farnas Ali Mudabbir\LAB1>
```

Task Description#4

- Provide a faulty class definition (missing self in parameters). Let AI fix it



```python
class Rectangle:
    def __init__(length, width):
        self.length = length
        self.width = width
```

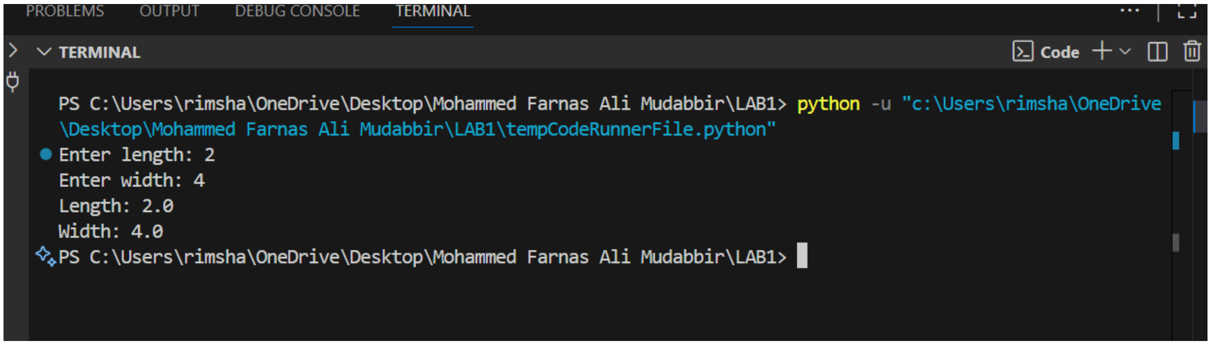

TASK 1.py C:\...\LAB 7
TASK 2.py C:\...\LAB 7
TASK 3.py C:\...\LAB 7
TASK 4.py C:\...\LAB 7
C: > Users > rimsha > OneDrive > Desktop > Mohammed Farnas Ali Mudabbir > LAB 7 > TASK 4.py > ...

```python
class Rectangle:
    def __init__(self, length, width):
        self.length = length
        self.width = width

# Taking user input
length = float(input("Enter length: "))
width = float(input("Enter width: "))

r = Rectangle(length, width)
print("Length:", r.length)
print("Width:", r.width)
```

Expected Output#4

- Correct __init__() method and explanation

Practical output:



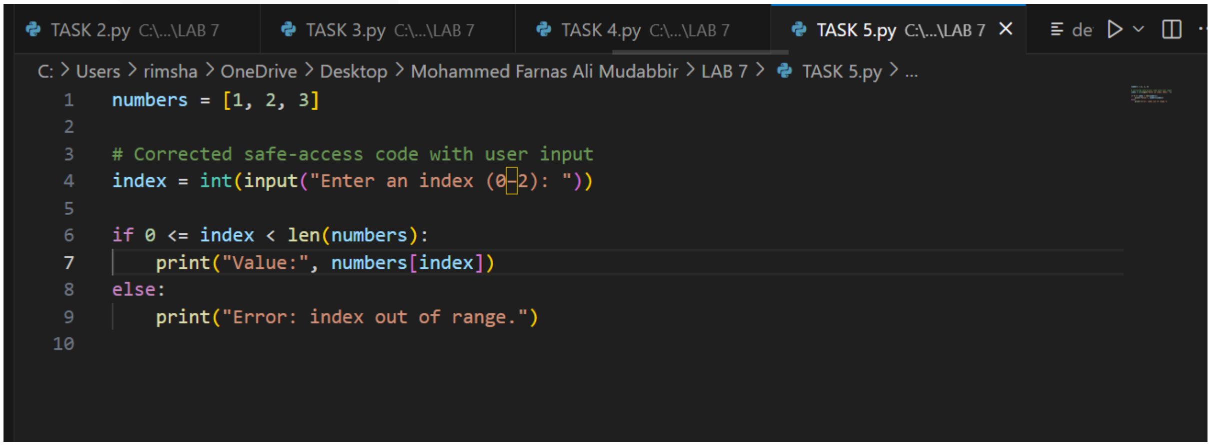
```
PS C:\Users\rimsha\OneDrive\Desktop\Mohammed Farnas Ali Mudabbir\LAB1> python -u "c:\Users\rimsha\OneDrive
\Desktop\Mohammed Farnas Ali Mudabbir\LAB1\tempCodeRunnerFile.python"
Enter length: 2
Enter width: 4
Length: 2.0
Width: 4.0
PS C:\Users\rimsha\OneDrive\Desktop\Mohammed Farnas Ali Mudabbir\LAB1>
```

Task Description#5

- Access an invalid list index and use AI to resolve the Index Error.

python

```python
numbers = [1, 2, 3]
print(numbers[5])
```

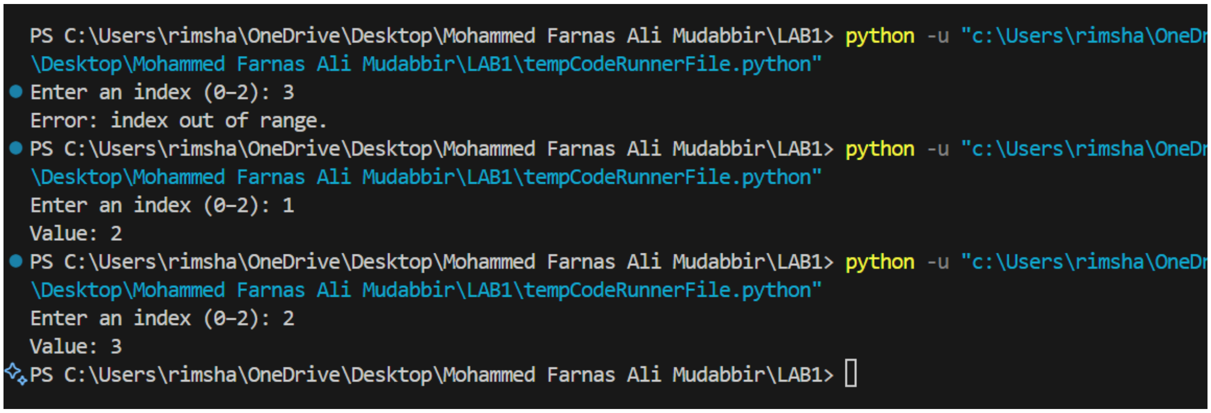

TASK 2.py C:\...\LAB 7
TASK 3.py C:\...\LAB 7
TASK 4.py C:\...\LAB 7
TASK 5.py C:\...\LAB 7
C: > Users > rimsha > OneDrive > Desktop > Mohammed Farnas Ali Mudabbir > LAB 7 > TASK 5.py > ...

```python
numbers = [1, 2, 3]

# Corrected safe-access code with user input
index = int(input("Enter an index (0-2): "))

if 0 <= index < len(numbers):
    print("Value:", numbers[index])
else:
    print("Error: index out of range.")

```

Expected Output#5

AI suggests checking length or using safe access logic

Practical output:



```
PS C:\Users\rimsha\OneDrive\Desktop\Mohammed Farnas Ali Mudabbir\LAB1> python -u "c:\Users\rimsha\OneDr
\Desktop\Mohammed Farnas Ali Mudabbir\LAB1\tempCodeRunnerFile.python"
Enter an index (0-2): 3
Error: index out of range.
PS C:\Users\rimsha\OneDrive\Desktop\Mohammed Farnas Ali Mudabbir\LAB1> python -u "c:\Users\rimsha\OneDr
\Desktop\Mohammed Farnas Ali Mudabbir\LAB1\tempCodeRunnerFile.python"
Enter an index (0-2): 1
Value: 2
PS C:\Users\rimsha\OneDrive\Desktop\Mohammed Farnas Ali Mudabbir\LAB1> python -u "c:\Users\rimsha\OneDr
\Desktop\Mohammed Farnas Ali Mudabbir\LAB1\tempCodeRunnerFile.python"
Enter an index (0-2): 2
Value: 3
PS C:\Users\rimsha\OneDrive\Desktop\Mohammed Farnas Ali Mudabbir\LAB1> 
```