SR UNIVERSITY

School of Computer Science and Artificial Intelligence

Lab Assignment # 1

Program	**: B. Tech (CSE)**
Specialization	**: CSE**
Course Title	**: AI Assisted coding**
Course Code	**: 23CS201PC302**
Semester	**: 3 SEM**
Academic Session	**: 2025-2026**
Name of Student	**: Eshwar**
Enrollment No.	**: 2403a51l26**
Batch No.	**: 51**
Date	**:20-01-2026**

Submission Instructions:

(All instructions should be followed strictly to avoid deduction of marks)

1. Use the same file to complete the assignment and don't change the settings.
2. Minimum 10 screen shots of your account should be taken to showcase your work.
3. **File Format:**
   - Submit your assignment as a PDF document (pdf). Ensure the file is named according to the following convention:

     `BNo_StudentName_ai.coding_A1.`

     `Sample`: **B10_Rohit_22A523421_A1**
4. Fill all the entries mentioned on top section.
5. Mention your AWS Academy Virtual Lab Account details as shown in the next page.
6. **Don't write on this page.**
7. All answers should be answered from next page only.

Lab assignment :1.2

Task1:

Prompt:

AI-Generated Logic Without Modularization (Factorial without Functions)

```python
# AI-Generated Logic Without Modularization (Factorial without Functions
# This code calculates the factorial of a number without using functions
number = 5
factorial = 1
for i in range(1, number + 1):
    factorial *= i
print(f"The factorial of {number} is {factorial}")
```

Output:

```
PS C:\Users\Eshwar\OneDrive\Desktop\python> & "C:\Program Files\Python314\python.exe"
The factorial of 5 is 120
PS C:\Users\Eshwar\OneDrive\Desktop\python>
```

- **This program calculates the factorial of a given number using a loop.**
- **The for loop multiplies all numbers from 1 to the given number.**
- **Finally, it prints the calculated factorial value.**

Task 2:

Prompt:

AI Code Optimization & Cleanup (Improving Efficiency)

ai_coding.py

```python
# AI Code Optimization & Cleanup (Improving Efficiency)
number = 5
factorial = 1

for i in range(1, number + 1):
    factorial = factorial * i

print("The factorial of", number, "is", factorial)
```

Output:

```
The factorial of 5 is 120
PS C:\Users\Eshwar\OneDrive\Desktop\python>
```

- **This program calculates the factorial of a given number.>**
- **The loop multiplies numbers from 1 to the given number and stores the result. This program calculates the factorial of a given number.**
- **Finally, it prints the factorial value of the number.**

Task 3:

Prompt:

#Modular Design Using AI Assistance (Factorial with Functions)

```python
#Modular Design Using AI Assistance (Factorial with Functions)

def calculate_factorial(number):
    factorial = 1
    for i in range(1, number + 1):
        factorial *= i

    return factorial
if __name__ == "__main__":
    num = 5
    result = calculate_factorial(num)
    print(f"The factorial of {num} is {result}")
```

Output:

```
PS C:\Users\Eshwar\OneDrive\Desktop\python> & "C:\Program Files\Python314\python.exe" c:/Users/Eshwar/OneDrive/Desktop/python/ai_coding.py
The factorial of 5 is 120
```

- **The program defines a function to calculate the factorial of a number.**
- **The function uses a loop to multiply numbers from 1 to the given value.**
- **The main block calls the function and prints the final result.**

Task 4:

Prompt:
Comparative Analysis – Procedural vs Modular AI Code (With vs
Without Functions)

ai_coding.py

```python
#Comparative Analysis – Procedural vs Modular AI Code (With vsWithout Functions)
# Program to find factorial of a number (Without Function)

num = int(input("Enter a number: "))

factorial = 1

if num < 0:
    print("Factorial does not exist for negative numbers")
else:
    for i in range(1, num + 1):
        factorial = factorial * i

    print("Factorial of", num, "is:", factorial)


```

Output:

```
PS C:\Users\Eshwar\OneDrive\Desktop\python> & "C:\Prog
Enter a number: 5
Factorial of 5 is: 120
PS C:\Users\Eshwar\OneDrive\Desktop\python>
```

Program to find factorial using function:

ai_coding.py > ...

```python
## Program to find factorial using function

def factorial(n):
    if n < 0:
        return "Factorial does not exist for negative numbers"

    result = 1
    for i in range(1, n + 1):
        result = result * i
    return result


# Main block
num = int(input("Enter a number: "))
print("Factorial of", num, "is:", factorial(num))

```

Output:

```
Enter a number: 8
Factorial of 8 is: 40320
PS C:\Users\Eshwar\OneDrive\Desktop\python>
```

- **Procedural Code:**
  The entire factorial logic is written in a single block, making it simple but less reusable and harder to maintain for large programs.
- **Modular Code:**
  The factorial logic is placed inside a function, improving readability, reusability, and ease of debugging.
- **Conclusion:**
  Using functions makes the program more structured, scalable, and suitable for real-world applications.

Task 5:

Prompt:

AI-Generated Iterative vs Recursive Thinking

```python
# Iterative factorial program

def factorial_iterative(n):
    result = 1
    for i in range(1, n + 1):
        result *= i
    return result

num = int(input("Enter a number: "))
print("Factorial (Iterative):", factorial_iterative(num))

```

Output:

```
PS C:\Users\Eshwar\OneDrive\Desktop\python> & "C:\Program Files\Python314\python.exe" c:/U
Enter a number: 9
Factorial (Iterative): 362880
PS C:\Users\Eshwar\OneDrive\Desktop\python>
```

- **Recursive Version (Function Calling Itself):**

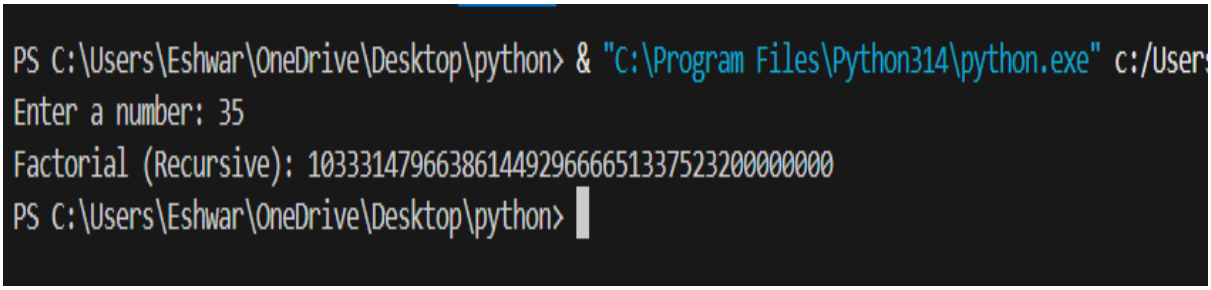
```python
# Recursive factorial program

def factorial_recursive(n):
    if n == 0 or n == 1:
        return 1
    return n * factorial_recursive(n - 1)

num = int(input("Enter a number: "))
print("Factorial (Recursive):", factorial_recursive(num))

```

Output:



- **The iterative version uses a loop to repeatedly multiply numbers until the factorial is calculated.**
- **The recursive version solves the problem by calling the same function with a smaller value until a base condition is me**
- **Both approaches produce the same output, but iteration is more memory-efficient while recursion shows better conceptual clarity.**