Assignment 1.1

# Lab 1: Environment Setup – GitHub Copilot and VS Code Integration

* Name – G.Sanjansah ,
* Date –
* Subject – AI Assisted Coding
* Hall Ticket Number – 2503a52l20
* Student Mail Id – 2503a52l20@sru.edu.in

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

Task :

● Install and configure GitHub Copilot in VS Code. Take screenshots of  
each step.  
Expected Output :

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

Task 1 - Factorial without Functions :

● Description:

Use GitHub Copilot to generate a Python program that calculates the  
factorial of a number without defining any functions (using loops  
directly in the main code).

● Expected Output:

o A working program that correctly calculates the factorial for  
user-provided input.  
o Screenshots of the code generation process.

Task 2 - Improving Efficiency :

● Description:

Examine the Copilot-generated code from Task 1 and demonstrate  
how its efficiency can be improved (e.g., removing unnecessary

variables, optimizing loops).

● Expected Output:

o Original and improved versions of the code.  
o Explanation of how the improvements enhance performance.

Task 3 - Factorial with Functions :

● Description:

Use GitHub Copilot to generate a Python program that calculates the  
factorial of a number using a user-defined function.

● Expected Output:

o Correctly working factorial function with sample outputs.  
o Documentation of the steps Copilot followed to generate the  
function.

Task 4 - Comparative Analysis – With vs Without Functions :

● Description:

Differentiate between the Copilot-generated factorial program with  
functions and without functions in terms of logic, reusability, and  
execution.

● Expected Output:

o A comparison table or short report explaining the differences.

Task 5: Iterative vs Recursive Factorial :

● Description:

Prompt GitHub Copilot to generate both iterative and recursive  
versions of the factorial function.

● Expected Output:

o Two correct implementations.  
o A documented comparison of logic, performance, and  
execution flow between iterative and recursive approaches.