

Lab Assignment 1.2

AI Assisted Coding

Name: Rohan beri

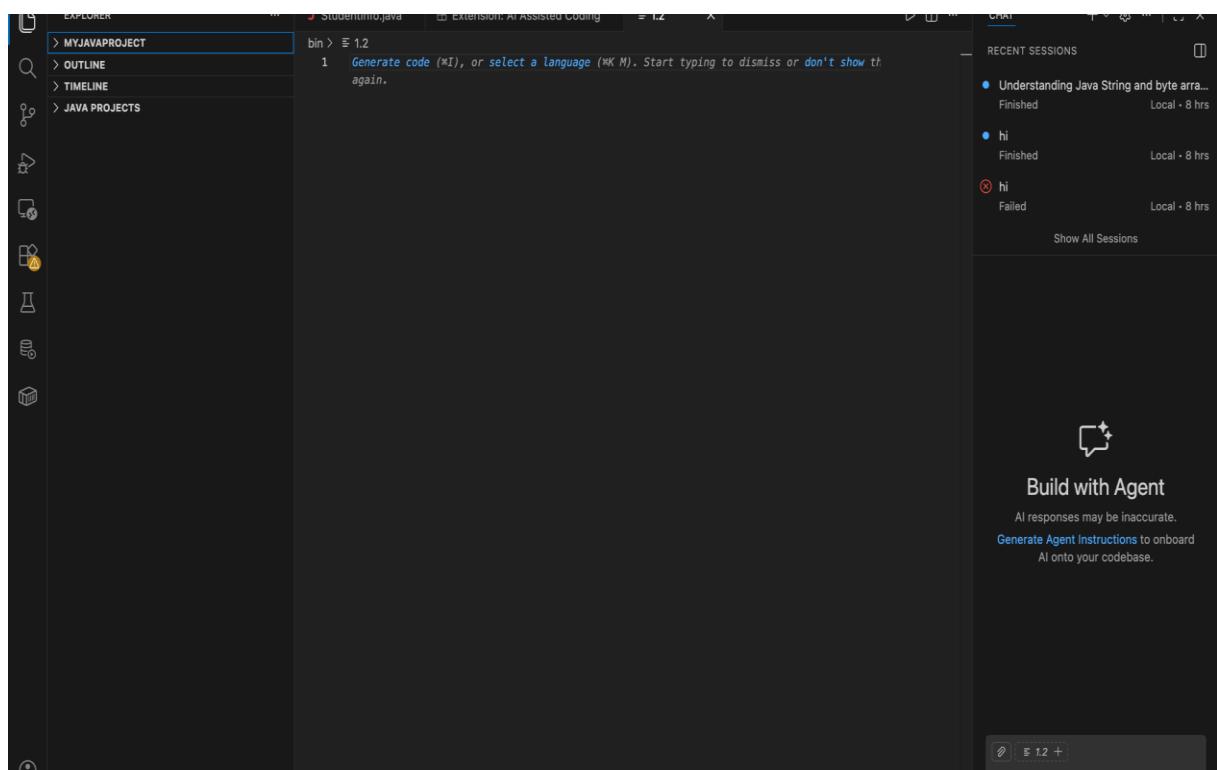
EnrollmentNumber:2403A51L09

Batch:51

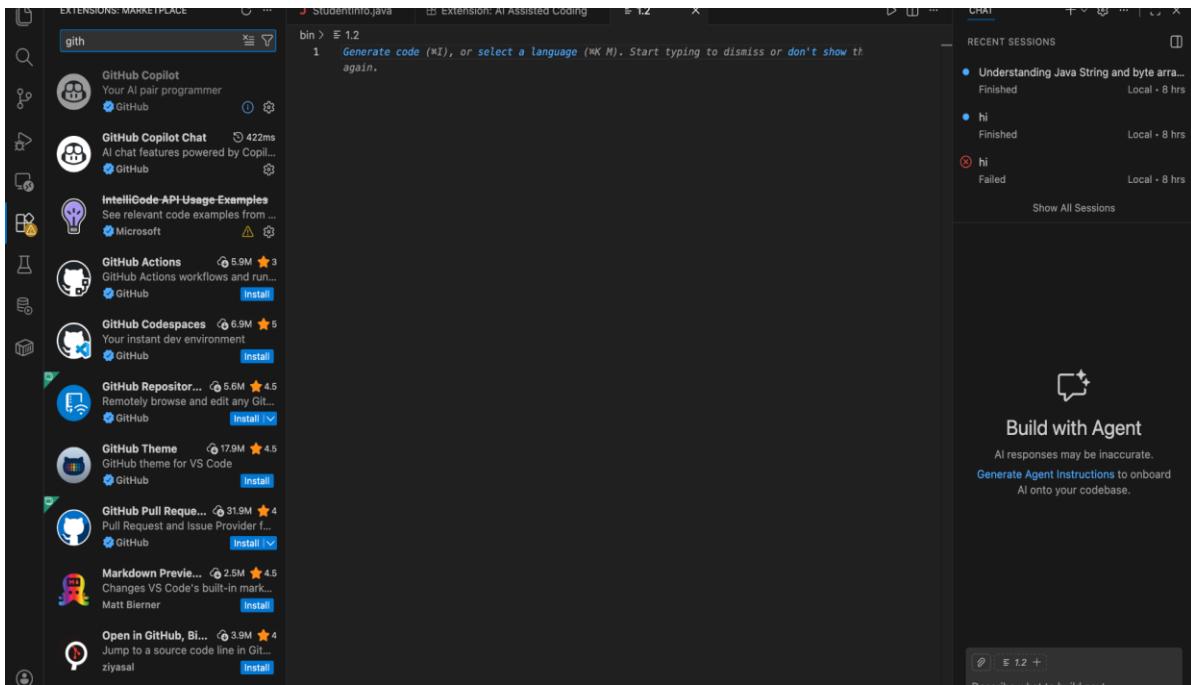
Task 0: GitHub Copilot Installation & Configuration

Steps Followed:

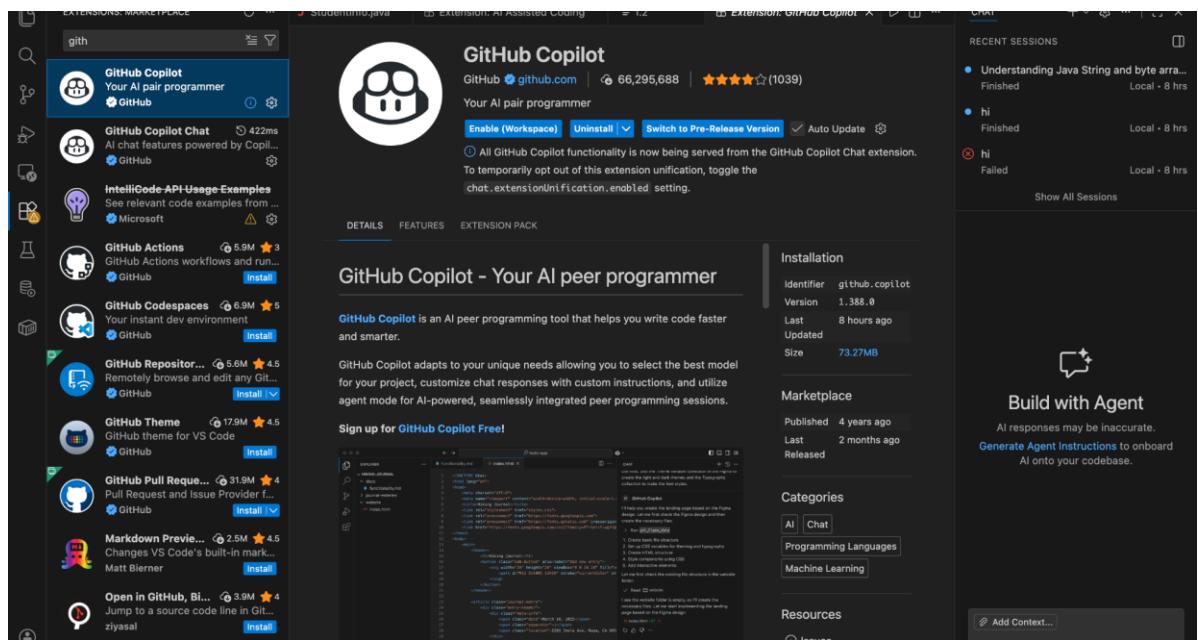
1. Installed Visual Studio Code
2. Opened Extensions Marketplace



3. Searched for GitHub Copilot



4. Clicked Install



5. Signed in with GitHub Account

6. Enabled Copilot suggestions

7. Verified Copilot inline suggestions in Python file.

```
1 """ write a python program o calculate factorial of number using loops only, without defining any function """
2 number = int(input("Enter a number to calculate its factorial: "))
3 factorial = 1
4 for i in range(1, number + 1):
5     factorial *= i
6 print(f"The factorial of {number} is {factorial}")
```

The operation couldn't be completed. Unable to locate a Java Runtime.
Please visit <http://www.java.com> for information on installing Java.

- saikoushik@sais-MacBook-Air-3 Ai Assisted coding % /opt/homebrew/bin/python3 "/Users/saikoushik/Desktop/AI Assisted coding/Day-1.2"
- Enter a number to calculate its factorial: 2
- saikoushik@sais-MacBook-Air-3 Ai Assisted coding % /opt/homebrew/bin/python3 "/Users/saikoushik/Desktop/AI Assisted coding/Day-1.2"
 Enter a number to calculate its factorial: 4
 The factorial of 4 is 24
- saikoushik@sais-MacBook-Air-3 Ai Assisted coding %

a

Task 1: AI-Generated Logic Without Modularization (Factorial without Functions)

Prompt Used: “Write a Python program to calculate factorial of a number using loops only, without defining any function.”

```
1 """ write a python program o calculate factorial of number using loops only, without defining any function """
2 n = int(input("Enter a number : "))
3 result = 1
4
5 for i in range(1, n + 1):
6     result *= i
7 print(f"The factorial of {n} is {result}")
```

The operation couldn't be completed. Unable to locate a Java Runtime.
Please visit <http://www.java.com> for information on installing Java.

- saikoushik@sais-MacBook-Air-3 Ai Assisted coding % /opt/homebrew/bin/python3 "/Users/saikoushik/Desktop/AI Assisted coding/Day-1.2"
- Enter a number to calculate its factorial: 2
- saikoushik@sais-MacBook-Air-3 Ai Assisted coding % /opt/homebrew/bin/python3 "/Users/saikoushik/Desktop/AI Assisted coding/Day-1.2"
 Enter a number to calculate its factorial: 4
 The factorial of 4 is 24
- saikoushik@sais-MacBook-Air-3 Ai Assisted coding % /opt/homebrew/bin/python3 "/Users/saikoushik/Desktop/AI Assisted coding/Day-1.2"
 Enter a number to calculate its factorial: 7
 The factorial of 7 is 5040
- saikoushik@sais-MacBook-Air-3 Ai Assisted coding %

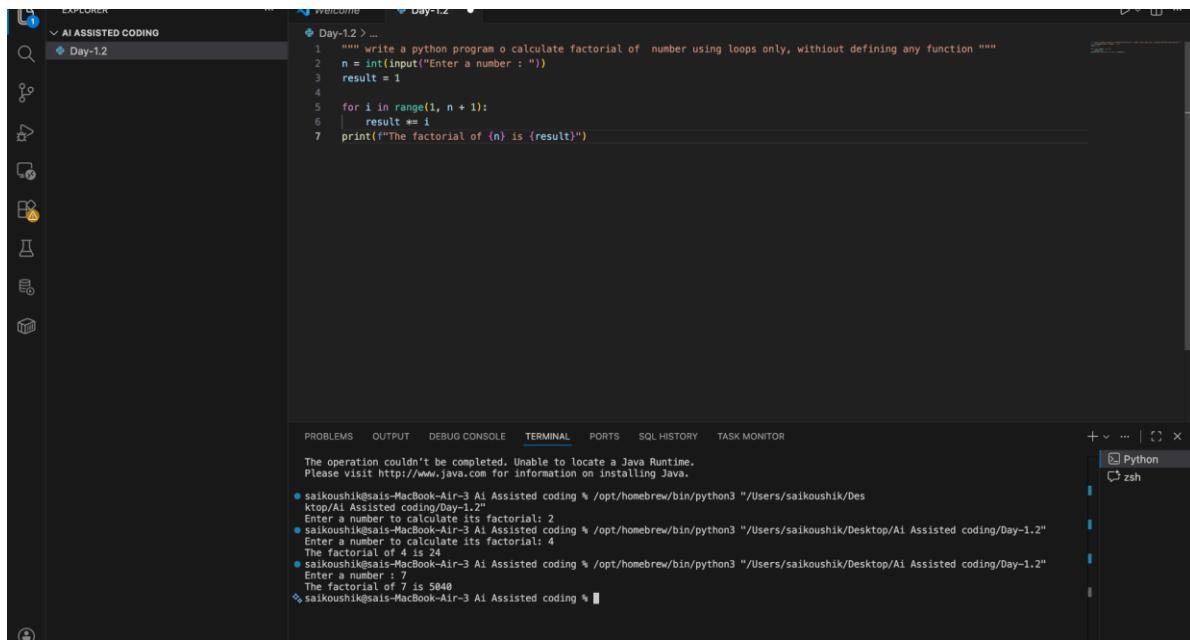
GitHub Copilot was very helpful for a beginner as it generated correct logic instantly.

It followed basic Python syntax and loop structure accurately.

The code was readable and easy to understand.
However, it did not include input validation automatically.
Best practices like modular design were not applied unless explicitly prompted.

Task 2: AI Code Optimization & Cleanup

Original Code:



A screenshot of the Visual Studio Code (VS Code) interface. The left sidebar shows the 'EXPLORER' view with a folder named 'AI ASSISTED CODING' containing a file 'Day-1.2'. The main editor area contains the following Python code:

```
1 """ write a python program o calculate factorial of number using loops only, without defining any function """
2 n = int(input("Enter a number : "))
3 result = 1
4
5 for i in range(1, n + 1):
6     result *= i
7 print("The factorial of {} is {}".format(result))
```

The 'TERMINAL' tab at the bottom shows the output of running the script. It starts with a Java runtime error message, followed by three successful runs of the script for inputs 2, 4, and 7, each printing the correct factorial value (2, 24, and 5040 respectively).

Prompt Used: “Optimize this code and make it more readable”

The screenshot shows the VS Code interface with the Explorer sidebar open, displaying 'AI ASSISTED CODING' and a file named 'Day-1.2'. The main editor area contains two versions of a Python script to calculate factorial. The first version is a simple loop, and the second is an optimized version using a for loop with range(2, n+1). The terminal below shows the execution of the code and its output.

```

1 """ write a python program o calculate factorial of number using loops only, without defining any function """
2 n = int(input("Enter a number : "))
3 result = 1
4
5 for i in range(1, n + 1):
6     result = result * i
7 print("Factorial is :", result)
8
9 """ Optimize this code and make it more durable """
10
11 n = int(input("Enter a number : "))
12 factorial = 1
13 for i in range(2, n + 1):
14     factorial *= i
15 print("Factorial of {} is: {}".format(n, factorial))

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SQL HISTORY TASK MONITOR

Please visit <http://www.java.com> for information on installing Java.

- saikoushik@saikoushik-MacBook-Air-3 ~ % /opt/homebrew/bin/python3 "/Users/saikoushik/Desktop/Ai Assisted coding/Day-1.2"
 ktop/Ai Assisted coding/Day-1.2
 Enter a number to calculate its factorial: 2
 saikoushik@saikoushik-MacBook-Air-3 ~ % /opt/homebrew/bin/python3 "/Users/saikoushik/Desktop/Ai Assisted coding/Day-1.2"
 Enter a number to calculate its factorial: 4
 The factorial of 4 is 24
 saikoushik@saikoushik-MacBook-Air-3 ~ % /opt/homebrew/bin/python3 "/Users/saikoushik/Desktop/Ai Assisted coding/Day-1.2"
 Enter a number to calculate its factorial: 7
 The factorial of 7 is 5040
 saikoushik@saikoushik-MacBook-Air-3 ~ % /opt/homebrew/bin/python3 "/Users/saikoushik/Desktop/Ai Assisted coding/Day-1.2"
 File "/Users/saikoushik/Desktop/Ai Assisted coding/Day-1.2", line 11
 n = int(input("Enter a number : "))
 ^
 IndentationError: unexpected indent
 saikoushik@saikoushik-MacBook-Air-3 ~ % /opt/homebrew/bin/python3 "/Users/saikoushik/Desktop/Ai Assisted coding/Day-1.2"

The optimized version improves clarity, maintainability, and readability without affecting performance.

Task 3: Modular Design Using AI Assistance (Factorial with Functions)

Prompt Used: “Create a Python function to calculate factorial and call it from main block”

The screenshot shows the VS Code interface with the Explorer sidebar open, displaying 'AI ASSISTED CODING' and a file named 'Day-1.2'. The main editor area contains a Python script using a function to calculate factorial. The terminal below shows the execution of the code and its output.

```

1 """ write a python function o calculate factorial and call it from.main block """
2 def calculate_factorial(num):
3     """Returns factorial of a number"""
4     result = 1
5     for i in range(1, num + 1):
6         result *= i
7     return result
8
9 number = int (input ("Enter a number: "))
10 print("Factorial is:", calculate_factorial (number))

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SQL HISTORY TASK MONITOR

/opt/homebrew/bin/python3 "/Users/saikoushik/Desktop/Ai Assisted coding/Day-1.2"
The operation couldn't be completed. Unable to locate a Java Runtime.
Please visit <http://www.java.com> for information on installing Java.

- saikoushik@saikoushik-MacBook-Air-3 ~ % /opt/homebrew/bin/python3 "/Users/saikoushik/Desktop/Ai Assisted coding/Day-1.2"
 Enter a number: 14
 Factorial is: 87178291200
 saikoushik@saikoushik-MacBook-Air-3 ~ %

Modularity improves reusability by allowing the same function to be used across multiple programs. It also simplifies testing and debugging.

Task 4: Comparative Analysis

Procedural vs Modular AI Code

Criteria	Without Function	With Function
Logic Clarity	Moderate	High
Reusability	No	Yes
Debugging Ease	Difficult	Easy
Large Project Suitability	Poor	Excellent
AI Dependency Risk	Higher	Lower

Conclusion:

Function-based design is more scalable and suitable for real-world applications.

Task 5: Iterative vs Recursive AI Code

Prompt Used: “Generate iterative and recursive factorial programs in Python”

```

23
24
25     """Generate iterative and recursive factorial programs in Python"""
26     def iterative_factorial(n):
27         result = 1
28         for i in range(1, n + 1):
29             result *= i
30         return result
31
32     """Recursive factorial function"""
33     def factorial_recursive(n):
34         if n == 0 or n == 1:
35             return 1
36         return n * factorial_recursive(n - 1)
37     print("Iterative factorial is:", iterative_factorial(number))
38     print("Recursive factorial is:", factorial_recursive(number))
39
40

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SQL HISTORY TASK MONITOR

/opt/homebrew/bin/python3 "/Users/saikoushik/Desktop/Ai Assisted coding/Day-1.2"
The operation couldn't be completed. Unable to locate a Java Runtime.
Please visit <http://www.java.com> for information on installing Java.

```

● saikoushik@saikoushik-MacBook-Air-3 ~ Ai Assisted coding % /opt/homebrew/bin/python3 "/Users/saikoushik/Desktop/Ai Assisted coding/Day-1.2"
Enter a number: 4
Factorial is: 24
● saikoushik@saikoushik-MacBook-Air-3 ~ Ai Assisted coding % /opt/homebrew/bin/python3 "/Users/saikoushik/Desktop/Ai Assisted coding/Day-1.2"
Enter a number: 4
Factorial is: 24
Iterative factorial is: 24
Recursive factorial is: 24
● saikoushik@saikoushik-MacBook-Air-3 ~ Ai Assisted coding %

```

Execution Flow Explanation:

- Iterative version uses a loop and constant memory.
- Recursive version uses function calls and stack memory.

Comparison:

Aspect	Iterative	Recursive
Readability	Simple	Elegant
Stack Usage	No	Yes
Performance	Faster	Slower
Risk	Low	Stack Overflow
Recommendation	Preferred	Avoid for large inputs