Assignment 5.1

# Lab 5: Ethical Foundations – Responsible AI Coding Practices

* 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 Description - 1 (Privacy in API Usage) :

Task: Use an AI tool to generate a Python program that connects to a weather API.

Prompt:  
"Generate code to fetch weather data securely without exposing API keys in the code."

Expected Output:  
• Original AI code (check if keys are hardcoded).  
• Secure version using environment variables.

Task Description - 2 (Privacy & Security in File Handling) :

Task: Use an AI tool to generate a Python script that stores user data (name, email, password) in a file.

Analyse: Check if the AI stores sensitive data in plain text or without encryption.

Expected Output:  
• Identified privacy risks.  
• Revised version with encrypted password storage (e.g., hashing).

Task Description - 3 (Transparency in Algorithm Design) :

Objective: Use AI to generate an Armstrong number checking function with comments and explanations.

Instructions:  
1. Ask AI to explain the code line-by-line.  
2. Compare the explanation with code functionality.

Expected Output:  
• Transparent, commented code.  
• Correct, easy-to-understand explanation

Task Description - 4 (Transparency in Algorithm Comparison):

Task: Use AI to implement two sorting algorithms (e.g., QuickSort and BubbleSort).

Prompt:  
"Generate Python code for QuickSort and BubbleSort, and include comments explaining step-by-step how each works and where they  
differ."

Expected Output:  
• Code for both algorithms.  
• Transparent, comparative explanation of their logic and efficiency

Task Description - 5 (Transparency in AI Recommendations):

Task: Use AI to create a product recommendation system.

Prompt:  
"Generate a recommendation system that also provides reasons for each suggestion."

Expected Output:  
• Code with explainable recommendations.  
• Evaluation of whether explanations are understandable.

Task Description - 6 (Transparent Code Generation):

Task: Ask AI to generate a Python function for calculating factorial using recursion.

Prompt:  
"Generate a recursive factorial function with comments that explain each line and a final summary of the algorithm’s flow."

Expected Output:  
• Fully commented code.  
• Clear documentation of how recursion works.

Task Description - 7 (Inclusiveness in Customer Support):

Code Snippet:

Task:  
Regenerate the code so that support messages use neutral language (e.g.“Dear {name}”) and optionally accept preferred titles.

Expected Output:  
• Neutral, user-friendly support responses