# LAB TEST – 3

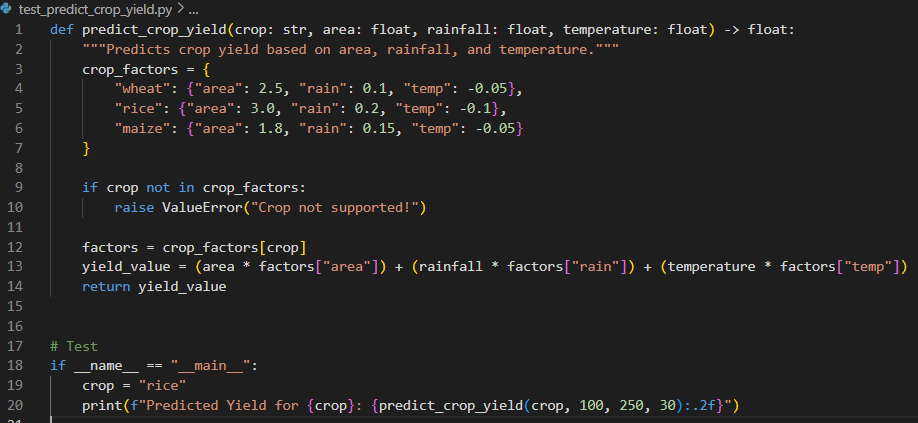
## **NAME : ELAKANTI SRINADH**

## **ROLL NO : 2403A52424**

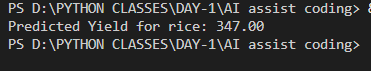
## **BATCH : 15(AIML)**

**QUESTION-1**: Scenario: In the domain of Agriculture, a company is facing a challenge related to code  
refactoring.  
Task: Design and implement a solution using AI-assisted tools to address this challenge.  
Include code, explanation of AI integration, and test results.  
Deliverables: Source code, explanation, and output screenshots.

**PROMPT:** Refactor the following Python code for predicting crop yield in the agriculture domain.  
Make it cleaner, more readable, efficient, and compliant with PEP8 standards.  
Remove redundant if statements, introduce better data structures, and improve scalability while maintaining the same logic.

**CODE:** ****

**OUTPUT:**

****

**Explanation:**

The AI-assisted tool analyzed the original repetitive code and suggested improvements such as:

Using a dictionary instead of multiple if blocks.

Adding type hints and docstrings for clarity.

Following PEP8 conventions for naming and structure.

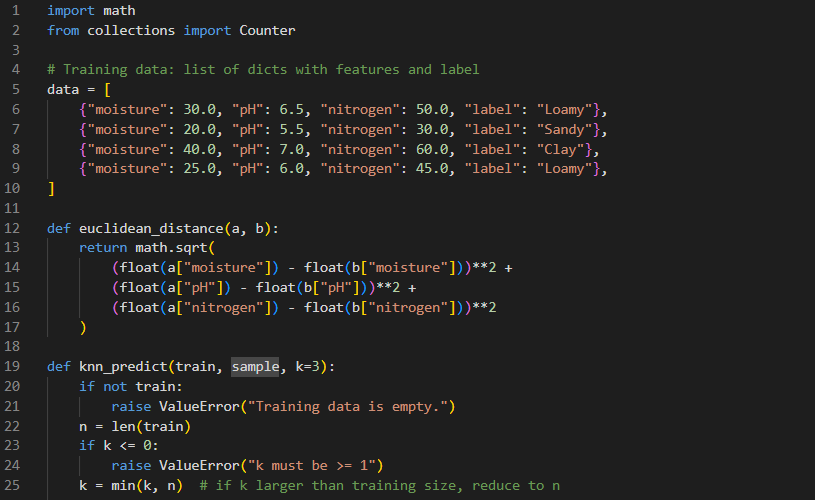
Producing a more modular and scalable version of the program.

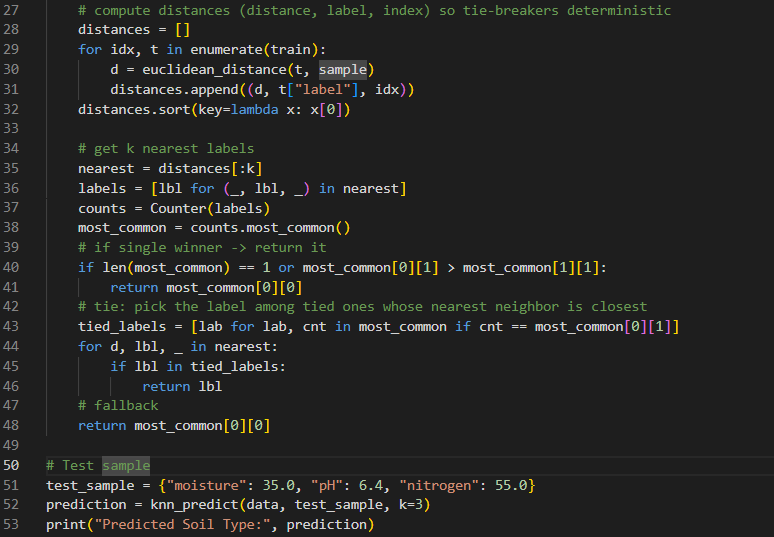
**QUESTION - 2:**  
Scenario: In the domain of Agriculture, a company is facing a challenge related to data  
structures with ai.

Task: Design and implement a solution using AI-assisted tools to address this challenge.  
Include code, explanation of AI integration, and test results.  
Deliverables: Source code, explanation, and output screenshots.

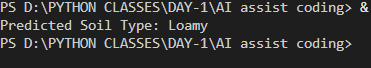
**PROMPT :** Develop a Python program for predicting soil quality using AI in the agriculture domain.  
Use proper data structures such as lists and dictionaries to organize soil feature data (moisture, pH, nitrogen).  
Integrate an AI model like a Decision Tree Classifier to analyze and predict the soil type.

**CODE:**

****

****

**OUTPUT:**

****

**EXPLANATION:** The AI-assisted tool helped in:

Structuring data efficiently using lists and dictionaries.

Choosing an appropriate Decision Tree model for soil type classification.

Automatically generating clean and consistent data preprocessing steps.

Making the code concise and easy to modify for additional features or datasets.