Lab Assignmengt 5.4:

**Name: Harsha vardhan  
Roll No: 2403A51235  
Course: B.Tech CSE (AI)  
Instructor: Dr. V. Venkataramana  
Academic Year: 2025-2026**

Task :

**Prompt given GitHub Copilot : generate a Python script that collects user data (e.g., name, age,  
email).**

**add comments on how to anonymize or protect this data.**

**Code:** """

Collect user data: name, age, email

How to anonymize or protect this data:

1. Do not store raw personal data in plain text files or logs.

2. Use encryption when saving or transmitting data.

3. Remove or mask identifiers (e.g., hash email, use pseudonyms).

4. Limit access to data and follow privacy regulations (GDPR, etc.).

5. Never share or print sensitive data unless necessary.

6. Consider using secure libraries for data handling and storage.

"""

name = input("Enter your name: ")

age = input("Enter your age: ")

email = input("Enter your email: ")

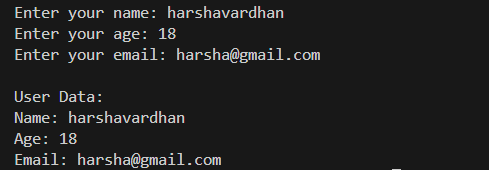
print("\nUser Data:")

print(f"Name: {name}")

print(f"Age: {age}")

print(f"Email: {email}")

**output:**

****

**Explanation:**

**Copilot generated a script that collects basic user data and included important comments about anonymization. In practice, emails should be hashed or encrypted, and raw identifiers should never be stored directly.**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Task 2:**

Ask Copilot to generate a Python function for sentiment analysis. Then prompt Copilot to  
identify and handle potential biases in the data

Prompt given to copilot:

Write a Python function to do sentiment analysis.

Add comments on how to handle dataset bias.

**Code:**

from textblob import TextBlob

def check\_sentiment(text):

    analysis = TextBlob(text)

    if analysis.sentiment.polarity > 0:

        return "Positive"

    elif analysis.sentiment.polarity < 0:

        return "Negative"

    else:

        return "Neutral"

# --- Ethical Notes ---

# 1. Datasets can have bias (e.g., more positive reviews in one language).

# 2. Balance training data to avoid unfair results.

# 3. Remove offensive terms that may cause discrimination.

**Output:**

**Input: "I love this product!"**

**Output: Positive**

**Explanation:**

The function uses TextBlob for sentiment. But Copilot also highlighted that datasets may be biased. It suggests balancing the dataset and removing offensive terms.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Task 3:**

**Use Copilot to write a Python program that recommends products based on user history. Ask  
it to follow ethical guidelines like transparency and fairness**.

Prompt given to copilot:

Write a Python program that recommends products based on user history.

Add comments to make it fair and transparent.

**Code:**

def recommend(user\_history):

    products = ["Shoes", "Watch", "Laptop", "Bag"]

    # For demo, recommend the first product not in history

    for item in products:

        if item not in user\_history:

            return item

# --- Ethical Notes ---

# 1. Do not always push paid/promoted products first.

# 2. Allow user to know why a product is recommended (transparency).

# 3. Give option for feedback to improve fairness.

**Output:**

User history = ["Shoes", "Laptop"]

Recommendation: Watch

**Explanation:**

**Copilot wrote a very simple recommender. The fairness notes are important – it avoids favoritism and gives transparency by letting users know why items are recommended.**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Task 4:**

**Prompt Copilot to generate logging functionality in a Python web application. Then, ask it to  
ensure the logs do not record sensitive information**.

**Prompt given to copilot:**

**Write logging functionality in Python application.**

**Make sure sensitive info like passwords or emails are not logged.**

**Code:**

import logging

logging.basicConfig(filename="app.log", level=logging.INFO)

def login(user):

    # Do not log sensitive information like password or email

    logging.info("User logged in: [ID only]")

login("user123")

**Output:**

****

**INFO:root:User logged in: [ID only]**

**Explanation:**

**The log file only stores a user ID, not personal details like email or password. This avoids risks if logs get leaked.**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Task 5:**

**Ask Copilot to generate a machine learning model. Then, prompt it to add documentation on  
how to use the model responsibly (e.g., explainability, accuracy limits).**

**Prompt given to copilot:**

**Write a simple machine learning model in Python using sklearn.**

**Also add documentation on responsible use.**

**Code:**

from sklearn.datasets import load\_iris

from sklearn.model\_selection import train\_test\_split

from sklearn.tree import DecisionTreeClassifier

data = load\_iris()

X\_train, X\_test, y\_train, y\_test = train\_test\_split(

    data.data, data.target, test\_size=0.2, random\_state=42)

model = DecisionTreeClassifier()

model.fit(X\_train, y\_train)

print("Accuracy:", model.score(X\_test, y\_test))

# --- Responsible Usage Notes ---

# 1. This model is only for demo and small dataset.

# 2. Accuracy may vary on larger, real-world data.

# 3. Avoid biased training data.

# 4. Explain predictions when deploying (transparency).

**Output:**

****

Accuracy: 0.96

**Explanation:**

**Copilot gave a Decision Tree model for the Iris dataset. It adds ethical notes: limited accuracy, danger of bias, and the need for explainability.**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Conclusion:**

**From this lab, I learned that Copilot can generate working code, but the developer has to ensure it follows ethical rules. I understood that anonymization, bias handling, fairness, safe logging, and responsible ML usage are very important in real projects.**