

Assignment - 5.4

P.SIDDARTHA 2303A51693

Prompt to Copilot

Generate a Python script that collects user name, age, and email. Add comments on how to anonymize or protect this data.

The screenshot shows a code editor interface with a dark theme. On the left is the code editor pane displaying a Python script. On the right are several tabs and toolbars.

```
1 import hashlib
2
3 # Collect user data
4 name = input("Enter your name: ")
5 age = input("Enter your age: ")
6 email = input("Enter your email: ")
7
8 # Hash email to anonymize personal identity
9 hashed_email = hashlib.sha256(email.encode()).hexdigest()
10
11 # NOTE: Never store raw personal data in plain text.
12 # Use hashing or encryption before storing in databases.
13
14 user_data = {
15     "name": name,
16     "age": age,
17     "email_hash": hashed_email
18 }
19
20 print("Stored User Data:", user_data)
21
```

Below the code editor are several tabs: PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS. The TERMINAL tab is active, showing a command-line session:

```
/usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A51693_5.4.py
● (base) ramcharan@Rams-MacBook-Air-2 AI_LAB % /usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A51693_5.4.py
Enter your name: ramcharan
Enter your age: 21
Enter your email: 3423@gmail.com
Stored User Data: {'name': 'ramcharan', 'age': '21', 'email_hash': 'f3bc4be2e24ca721f6cff8e6b8aca3110604f3ce7c27c2ce7719a2e29bcae74f'}
```

On the far right, there are two Python-related dropdown menus:

- Python (highlighted)
- Python

Prompt to Copilot

Generate a Python function for sentiment analysis and handle potential bias in data.

```
1 def analyze_sentiment(text):
2     # Simple word-based sentiment check (demo purpose)
3     positive_words = ["good", "happy", "great", "excellent"]
4     negative_words = ["bad", "sad", "terrible", "poor"]
5
6     text = text.lower()
7
8     score = 0
9     for word in positive_words:
10        if word in text:
11            score += 1
12        for word in negative_words:
13            if word in text:
14                score -= 1
15
16    # Ethical consideration:
17    # Ensure training data is diverse and balanced
18    # Remove offensive or culturally biased terms
19    # Avoid making decisions only based on limited keywords
20
21    if score > 0:
```

The screenshot shows a terminal window with the following content:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS + ⌂ ⌄ ⌁ ⌂ ×
/usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A51569_5.4.py
● (base) ramcharan@Rams-MacBook-Air-2 AI_LAB % /usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A51569_5.4.py
Enter your name: ramcharan
Enter your age: 21
Enter your email: 3423@gmail.com
Stored User Data: {'name': 'ramcharan', 'age': '21', 'email_hash': 'f3bc4be2e24ca721f6cff8e6b8aca3110604f3ce7c27c2ce7719a2e29bcae74f'}
● (base) ramcharan@Rams-MacBook-Air-2 AI_LAB % /usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A51569_5.4.py
Positive
✧ (base) ramcharan@Rams-MacBook-Air-2 AI_LAB %
```

The terminal shows the execution of a Python script named `2303A51569_5.4.py`. It prompts the user for their name, age, and email. The user inputs `ramcharan`, `21`, and `3423@gmail.com` respectively. The script then stores this data in a dictionary and prints the stored user data. Finally, it outputs the word `Positive`.

Prompt to Copilot

Generate product recommendation code that follows fairness and transparency.

```
1 def recommend_products(user_history, all_products):
2     recommendations = []
3
4     for product in all_products:
5         if product["category"] in user_history:
6             recommendations.append(product)
7
8     # Ethical Guidelines:
9     # - Do not promote only paid or sponsored products
10    # - Ensure equal visibility to small sellers
11    # - Explain why items are recommended
12
13    return recommendations
14
15
16 user_history = ["electronics", "books"]
17
18 products = [
19     {"name": "Laptop", "category": "electronics"},
20     {"name": "Novel", "category": "books"},
21     {"name": "Shoes", "category": "fashion"}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS + ⌂ ⌂ ⌂ ⌂ ⌂ ⌂

● (base) ramcharan@Rams-MacBook-Air-2 AI_LAB % /usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A5
1569_5.4.py

Enter your name: ramcharan

Enter your age: 21

Enter your email: 3423@gmail.com

Stored User Data: {'name': 'ramcharan', 'age': '21', 'email_hash': 'f3bc4be2e24ca721f6cff8e6b8aca3110604f3ce
7c27c2ce7719a2e29bcae74f'}

● (base) ramcharan@Rams-MacBook-Air-2 AI_LAB % /usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A5
1569_5.4.py

Positive

● (base) ramcharan@Rams-MacBook-Air-2 AI_LAB % /usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A5
1569_5.4.py

Recommended Products: [{name: 'Laptop', category: 'electronics'}, {name: 'Novel', category: 'books'}]

Prompt to Copilot

Generate logging functionality that avoids sensitive data.

```
1 import logging
2
3 logging.basicConfig(filename="app.log", level=logging.INFO)
4
5 def login_user(username, password):
6     # DO NOT log passwords or emails
7
8     logging.info(f"Login attempt for user: {username}")
9
10    # Simulated authentication
11    if password == "admin123":
12        logging.info("Login successful")
13        return True
14    else:
15        logging.warning("Login failed")
16        return False
17
18
19 login_user("|_user", "test123")
20
21 # Ethical Logging:
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

(base) ramcharan@Rams-MacBook-Air-2 AI_LAB % /usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A5
1569_5.4.py
7c27c2ce7719a2e29bcae74f'}

- (base) ramcharan@Rams-MacBook-Air-2 AI_LAB % /usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A5
1569_5.4.py
Positive
- (base) ramcharan@Rams-MacBook-Air-2 AI_LAB % /usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A5
1569_5.4.py
Recommended Products: [{'name': 'Laptop', 'category': 'electronics'}, {'name': 'Novel', 'category': 'books'}]
- (base) ramcharan@Rams-MacBook-Air-2 AI_LAB % /usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A5
1569_5.4.py
- (base) ramcharan@Rams-MacBook-Air-2 AI_LAB % /usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A5
1569_5.4.py

Python

Python

Prompt to Copilot

Generate ML model code and add responsible usage documentation.

```
1  from sklearn.linear_model import LinearRegression
2  import numpy as np
3
4  # Sample training data
5  X = np.array([[1], [2], [3], [4]])
6  y = np.array([100, 200, 300, 400])
7
8  model = LinearRegression()
9  model.fit(X, y)
10
11 prediction = model.predict([[5]])
12 print("Predicted value:", prediction)
13
14 """
15 Responsible AI Guidelines:
16 - This model is trained on very small data; predictions may be inaccurate.
17 - Do not use for financial or medical decisions.
18 - Always test with real-world datasets.
19 - Monitor model bias and performance regularly.
20 - Provide explanation of predictions to users when possible.
21 """
```

The screenshot shows a code editor interface with a dark theme. At the top, there are tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS. Below these tabs is a terminal window displaying the execution of a Python script named '1569_5.4.py'. The terminal output includes several lines of text indicating the model's training data, its prediction for an input of 5 (outputting [500]), and a block of text titled 'Responsible AI Guidelines' with five bullet points. To the right of the terminal, there is a sidebar with two entries under the heading 'Python': a green square icon followed by 'Python' and a grey square icon followed by 'Python'.

```
(base) ramcharan@Rams-MacBook-Air-2 AI_LAB % /usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A5
1569_5.4.py
Positive
(base) ramcharan@Rams-MacBook-Air-2 AI_LAB % /usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A5
1569_5.4.py
Recommended Products: [{'name': 'Laptop', 'category': 'electronics'}, {'name': 'Novel', 'category': 'books'}]
(base) ramcharan@Rams-MacBook-Air-2 AI_LAB % /usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A5
1569_5.4.py
(base) ramcharan@Rams-MacBook-Air-2 AI_LAB % /usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A5
1569_5.4.py
(base) ramcharan@Rams-MacBook-Air-2 AI_LAB % /usr/local/bin/python3 /Users/ramcharan/Documents/AI_LAB/2303A5
1569_5.4.py
Predicted value: [500.]
```