

Assignment-5.4 (10F5)

Task-1: Prompt GitHub Copilot to generate a Python script that collects user data (e.g., name, age, email). Then, ask Copilot to add comments on how to anonymize or protect this data.

The screenshot shows the GitHub Copilot interface. On the left, a Python script named `Task1.py` is displayed. The script includes a header with a title and purpose, followed by imports for `hashlib`, `re`, `datetime`, and `json`. It defines a `collect_user_data()` function that prompts the user for their name, age, and email, with validation for each. The script also includes comments about data protection and anonymization. On the right, the Copilot chat window shows a prompt: "generate a Python script that collects user data (e.g., name, age, email). Try to add comments on how to anonymize or protect the data". The response includes a confirmation that the script was created, a list of key features, and a summary of the script's capabilities.

```
1 """
2 User Data Collection Script with Data Protection and Anonymization Guidelines
3 This script demonstrates how to collect user data while implementing security
4 and privacy best practices.
5 """
6
7 import hashlib
8 import re
9 from datetime import datetime
10 import json
11
12
13 # =====
14 # DATA COLLECTION FUNCTIONS
15 # =====
16
17 def collect_user_data():
18     """
19     Collect user information with input validation.
20
21     Data Protection Note: Always validate and sanitize user inputs to prevent
22     injection attacks and ensure data quality.
23     """
24     print("\n=== User Data Collection Form ===\n")
25
26     # Collect Name
27     while True:
28         name = input("Enter your full name: ").strip()
29         # Validation: Check if name is not empty and contains only letters and sp
30         if name and all(c.isalpha() or c.isspace() for c in name):
31             break
32         print("Invalid input. Please enter a valid name (letters and spaces only)")
33
34     # Collect Age
35     while True:
36         try:
37             age = int(input("Enter your age: "))
```

USER DATA COLLECTION SCRIPT WITH ANONYMIZATION TIPS

generate a Python script that collects user data (e.g., name, age, email). Try to add comments on how to anonymize or protect the data

✓ Created Python script for user data collection

I'll create a Python script that collects user data with comprehensive comments on data protection and anonymization practices.

✓ Created `Task1.py`

I've created a comprehensive Python script that collects user data with extensive comments on data protection and anonymization. Here's what's included:

Key Features:

- Data Collection** - Validates name, age, and email inputs with proper error handling
- Data Protection Comments** covering:
 - Input validation and sanitization
 - Email verification best practices
 - Password hashing strategies (bcrypt, argon2)
 - Encryption recommendations (AES-256)
 - Database security measures
 - GDPR/HIPAA compliance notes
- Anonymization Techniques:**
 - Email hashing (SHA-256) for creating unique, irreversible IDs
 - Age binning to prevent re-identification

Describe what to build next

Agent v Auto v

OUTPUT:

The screenshot shows a terminal window with the command to run `Task1.py`. The output displays the user data collection form, the collected data, and the anonymized data. The script also includes warnings about PII and provides instructions on how to handle the data.

```
PS C:\Users\tloke\OneDrive\Documents\AI_79_Assignments> & C:/Users/tloke/AppData/Local/Python/pythoncore-3.14-64/python.exe c:/Users/tloke/OneDrive/Documents/AI_79_Assignments/Task1.py

=====
USER DATA COLLECTION AND PRIVACY DEMONSTRATION
=====

=== User Data Collection Form ===

Enter your full name: LOKESH NAIDU
Enter your age: 20
Enter your email address: lokesh@gmail.com

--- Original User Data (with PII) ---
Name: LOKESH NAIDU
Age: 20
Email: lokesh@gmail.com
Email (masked): l****h@gmail.com

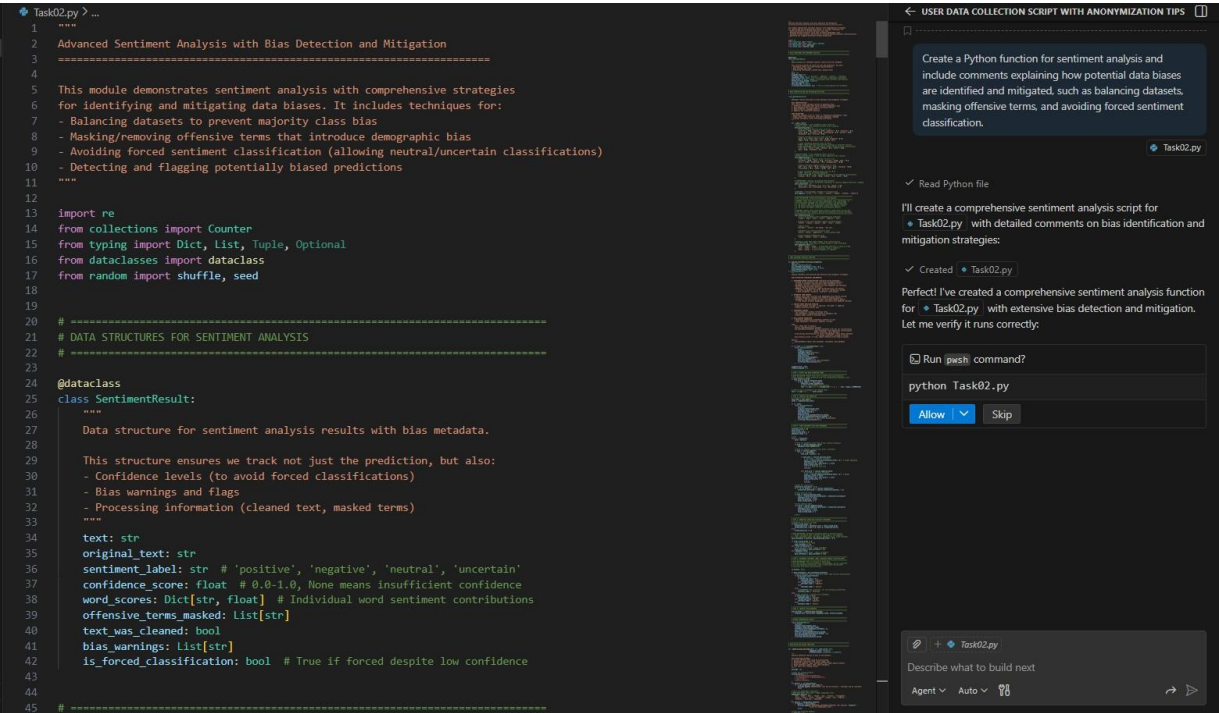
--- Anonymized User Data (privacy-safe) ---
User ID: 4954da2894a2e1d6... (unique anonymous hash)
Age Range: 18-25
Timestamp: 2026-01-22T13:00:00 (rounded to hour)

✓ Original data saved to: user_data_original.json
WARNING: This file contains PII and should be encrypted!
✓ Anonymized data saved to: user_data_anonymized.json
This file is privacy-safe for analytics and sharing.

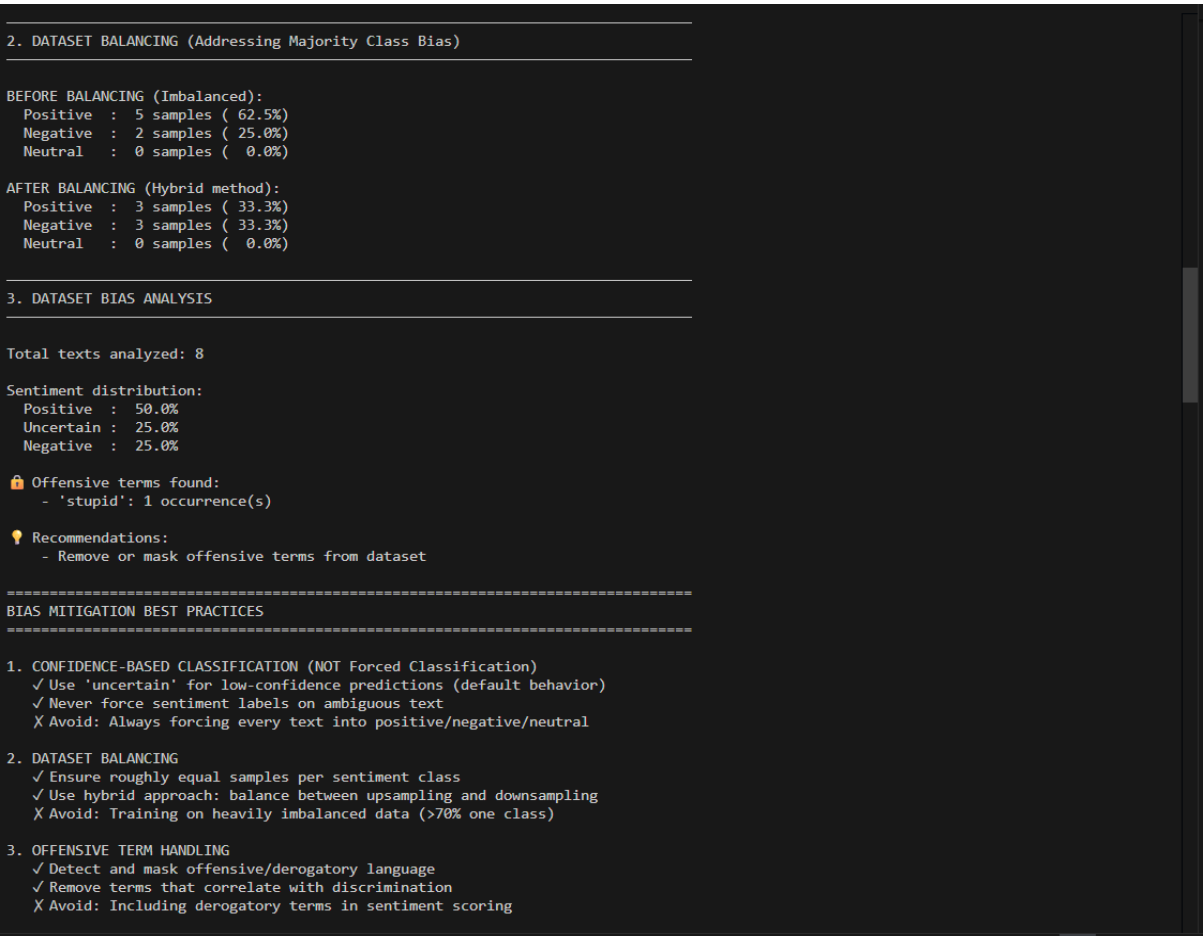
=====
Data collection and anonymization complete!
=====

PS C:\Users\tloke\OneDrive\Documents\AI_79_Assignments>
```

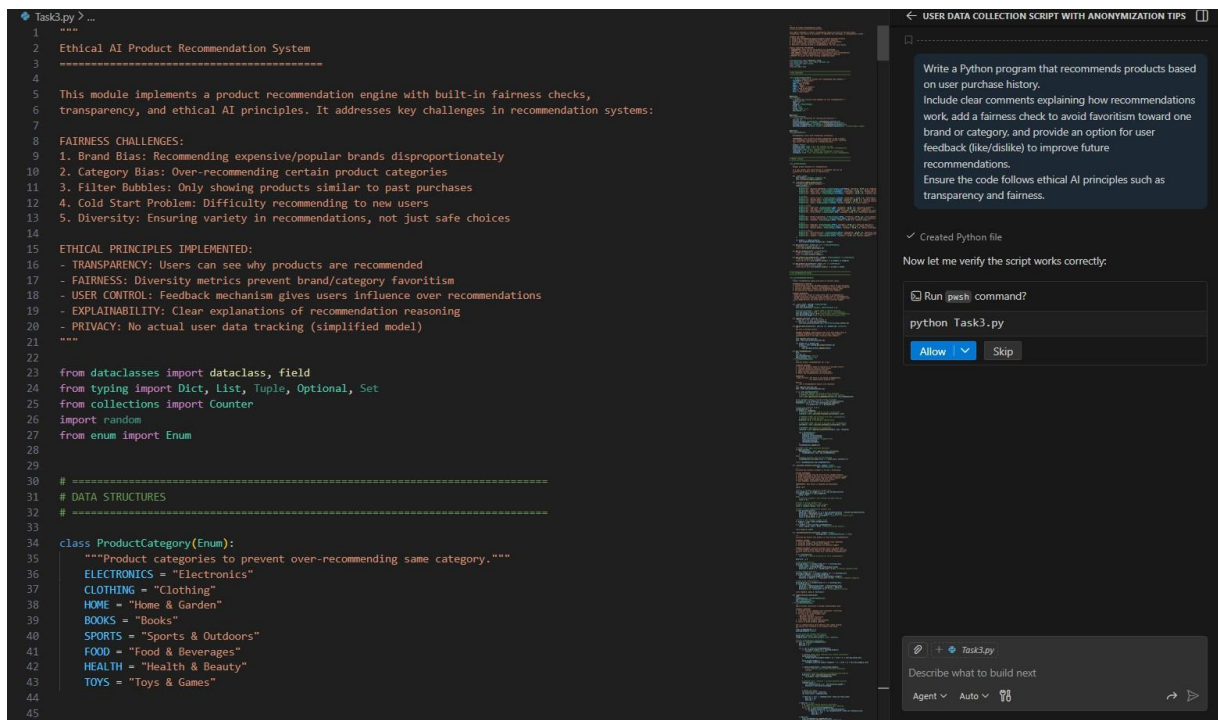
Task2: Ask Copilot to generate a Python function for sentiment analysis. Then prompt Copilot to identify and handle potential biases in the data.



OUTPUT:



Task3: Use Copilot to write a Python program that recommends products based on user history. Ask it to follow ethical guidelines like transparency and fairness.



OUTPUT:

```
1. USER PROFILE AND PURCHASE HISTORY

User ID: user_123
Purchase history (5 items):
  • Wireless Headphones (Electronics) - $79.99 - Rating: 4.5/5
  • Running Shoes (Clothing) - $99.99 - Rating: 4.7/5
  • Vitamin C Serum (Health & Beauty) - $39.99 - Rating: 4.6/5
  • Python Programming (Books) - $49.99 - Rating: 4.8/5
  • Yoga Mat (Sports & Outdoors) - $24.99 - Rating: 4.7/5

2. FAIR RECOMMENDATIONS (WITH FAIRNESS CONSTRAINTS)

Top 5 Recommendations:

1. Bicycle Helmet
  Brand: SafeRide | Category: Sports & Outdoors
  Price: $59.99 | Rating: 4.6/5
  Relevance Score: 0.73
  Diversity Score: 0.50
  Final Score: 0.73
  Confidence: 0.78
  Why? Popular category in your purchase history • Highly rated (4.6/5 stars) • Similar price to your typical purchases

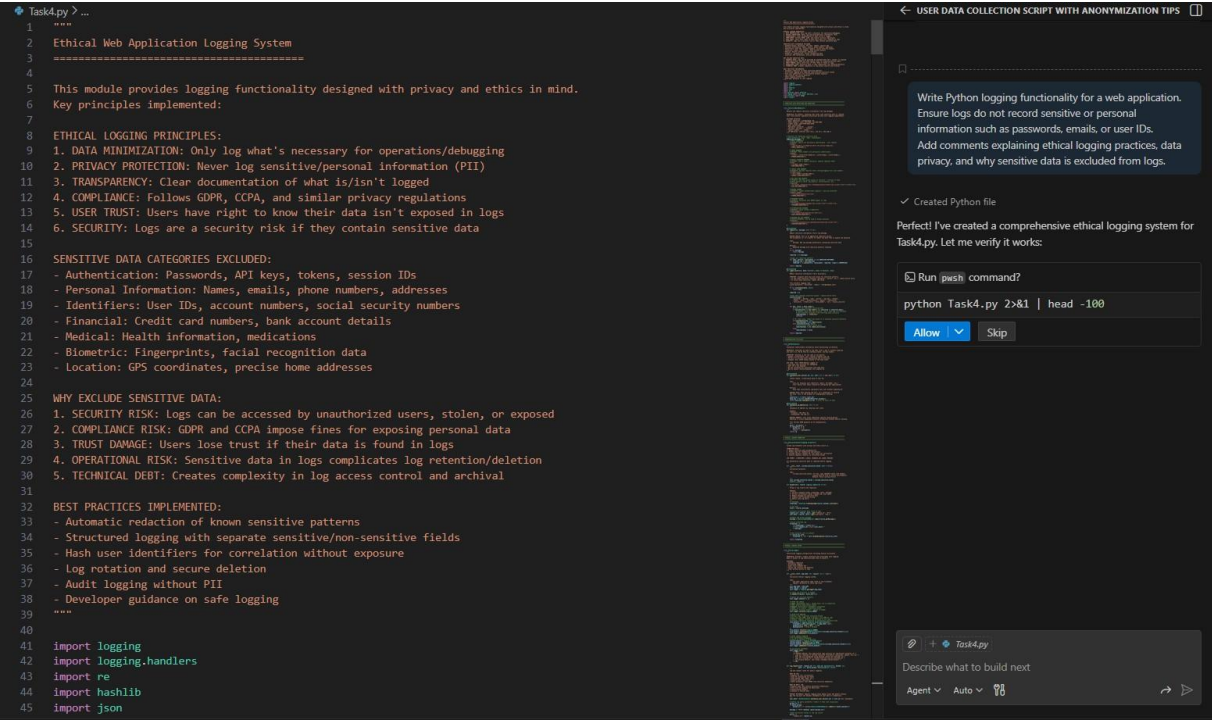
2. Jeans
  Brand: DenimCo | Category: Clothing
  Price: $59.99 | Rating: 4.3/5
  Relevance Score: 0.71
  Diversity Score: 0.50
  Final Score: 0.71
  Confidence: 0.76
  Why? Popular category in your purchase history • Similar price to your typical purchases

3. Laptop Stand
  Brand: ErgoWorks | Category: Electronics
  Price: $39.99 | Rating: 4.6/5
  Relevance Score: 0.67
  Diversity Score: 0.50
  Final Score: 0.67
  Confidence: 0.78
  Why? Popular category in your purchase history • Highly rated (4.6/5 stars) • Similar price to your typical purchases

4. Face Moisturizer
  Brand: BeautyGlow | Category: Health & Beauty
  Price: $34.99 | Rating: 4.5/5
  Relevance Score: 0.64
  Diversity Score: 0.93
  Final Score: 0.73
  Confidence: 0.77
  Why? Popular category in your purchase history • Highly rated (4.5/5 stars) • Similar price to your typical purchases

5. Cookbook
  Brand: FoodBiz | Category: Books
  Price: $29.99 | Rating: 4.4/5
  Relevance Score: 0.62
  Diversity Score: 0.92
  Final Score: 0.71
  Confidence: 0.76
```

Task4: Prompt Copilot to generate logging functionality in a Python web application. Then, ask it to ensure the logs do not record sensitive information.



OUTPUT:

1. SENSITIVE DATA REDACTION

Before and After Redaction:

Original: User logged in with email: john.doe@example.com
Redacted: User logged in with email: [EMAIL_REDACTED]

Original: Password reset requested for user: 123-45-6789
Redacted: Password reset requested for user: [SSN_REDACTED]

Original: API call with key: sk-1234567890abcdef
Redacted: API call with key: sk-1234567890abcdef

Original: Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9
Redacted: [AUTH_HEADER_REDACTED]

Original: Payment processed: 4532-1234-5678-9012
Redacted: Payment processed: [CREDIT_CARD_REDACTED]

Original: User phone: (555) 123-4567
Redacted: User phone: [PHONE_REDACTED]

2. DICTIONARY/FORM DATA REDACTION

Original form data:

```
{
  "username": "john_doe",
  "email": "john@example.com",
  "password": "SecurePass123!",
  "api_key": "sk_test_12345",
  "phone": "555-123-4567",
  "address": "123 Main St"
}
```

Redacted form data:

```
{
  "username": "john_doe",
  "email": "[EMAIL_REDACTED]",
  "password": "[REDACTED]",
  "api_key": "[REDACTED]",
  "phone": "[PHONE_REDACTED]",
  "address": "123 Main St"
}
```

3. USER ANONYMIZATION (Consistent Hashing)

Task5: 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).

The screenshot shows a code editor with a Python script named `Task5.py`. The script defines a `LoanApplicant` dataclass and a `ModelPrediction` dataclass. The `LoanApplicant` class includes fields for `age`, `annual_income`, `credit_score`, `years_employed`, `existing_debt`, and `employment_industry`. It also includes a `demographic_group` field with a note about fairness. The `ModelPrediction` class includes fields for `applicant_id`, `prediction`, `confidence`, and `risk_score`.

The sidebar on the right shows the Copilot-generated documentation for the `Task5.py` file. The documentation includes a title "USER DATA COLLECTION SCRIPT WITH ANONYMIZATION TIPS", a description "Generate a simple machine learning model in Python. Include clear documentation (inline comments or a README) explaining how the model works, its accuracy limitations, and how to use it responsibly. Add notes on explainability, potential bias, and fairness considerations.", and a list of tasks: "Read Python file", "I'll create a comprehensive machine learning model with extensive documentation on responsible AI", "Created Task5.py", and "Perfect! I've created a comprehensive, responsible machine learning model. Let me verify it runs correctly:". There is also a section for "Run pshell command?" with the command `python Task5.py 2>&1 | head -150` and buttons for "Allow" and "Skip".

OUTPUT:

```
=====
RESPONSIBLE MACHINE LEARNING MODEL - CREDIT RISK CLASSIFICATION
=====

1. DATASET CREATION

Total applicants: 9
Sample applicants:
- app_001: Age 35, Income $80000k, Credit Score 750
- app_002: Age 42, Income $95000k, Credit Score 760
- app_003: Age 28, Income $60000k, Credit Score 720

2. MODEL TRAINING

✓ Model trained successfully
✓ Training on 9 applicants
✓ Features used: Age, Income, Credit Score, Years Employed, Existing Debt
✓ No sensitive features (gender, race, etc.) used in model

3. PREDICTIONS WITH EXPLAINABILITY

Applicant: app_001
Prediction: LOW_RISK
Risk Score: 12.15%
Confidence: 88%
Uncertainty: 76%
Explanation: Model predicts low risk (confidence: 76%) based on: Credit Score decreases risk, Income decreases risk, Age decreases risk
Feature Contributions:
- Credit Score: -1.00 ↓ risk
- Income: -0.48 ↓ risk
- Age: -0.28 ↓ risk
- Years Employed: -0.23 ↓ risk
- Existing Debt: +0.04 ↑ risk

Applicant: app_002
Prediction: LOW_RISK
Risk Score: 8.69%
Confidence: 91%
Uncertainty: 83%
Explanation: Model predicts low risk (confidence: 83%) based on: Credit Score decreases risk, Income decreases risk, Age decreases risk
Feature Contributions:
```