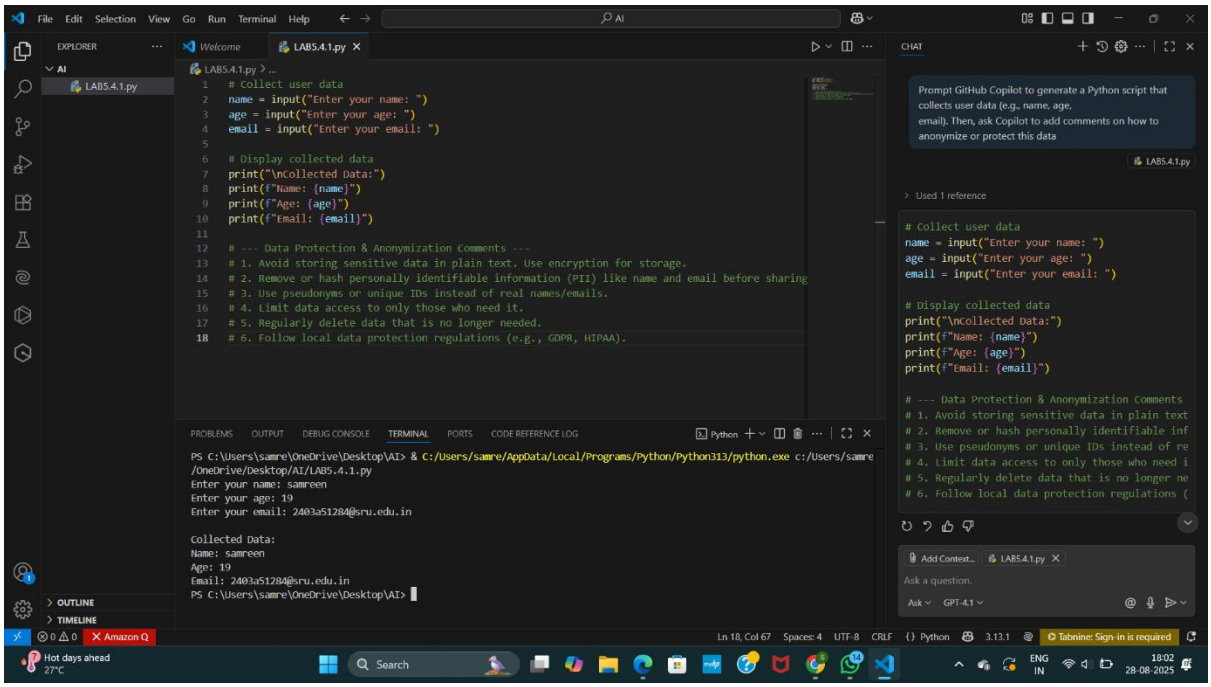
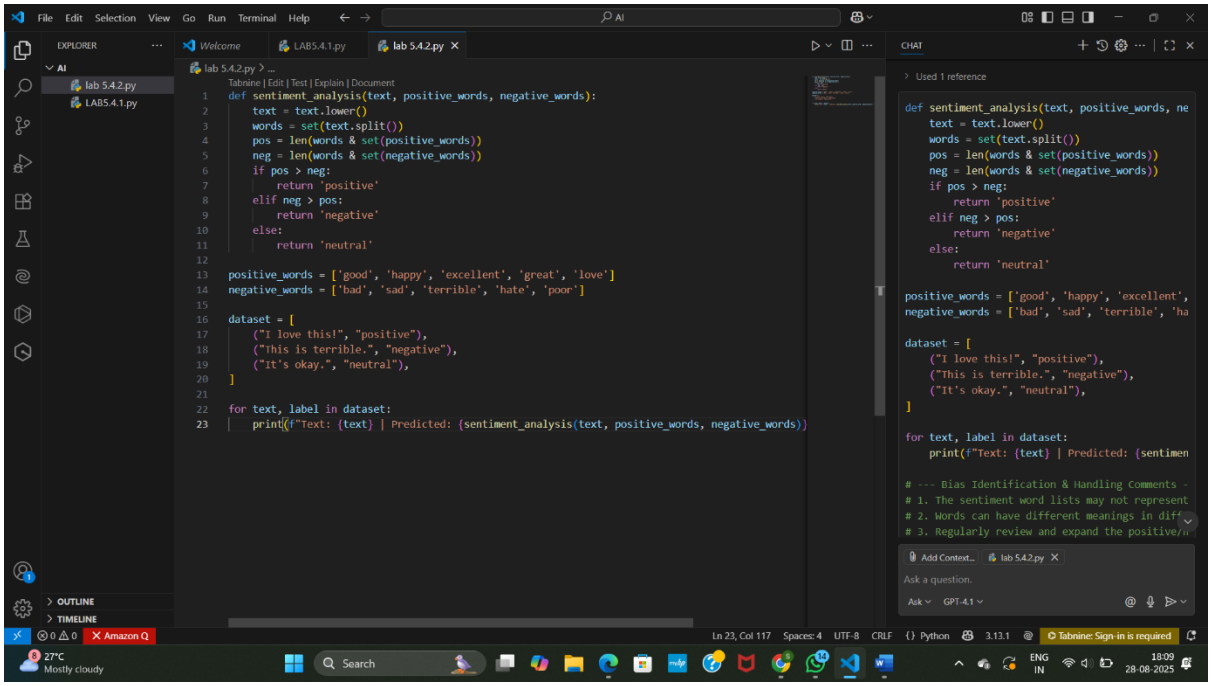
ASSIGNMENT:1

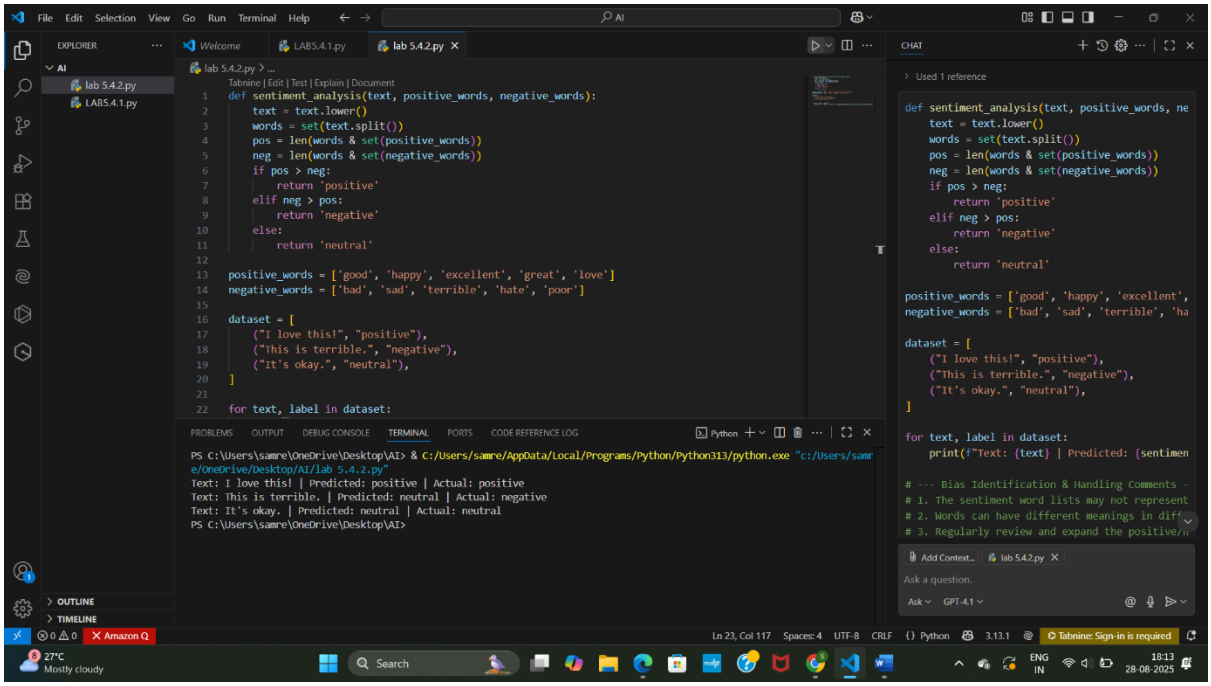
Task:1



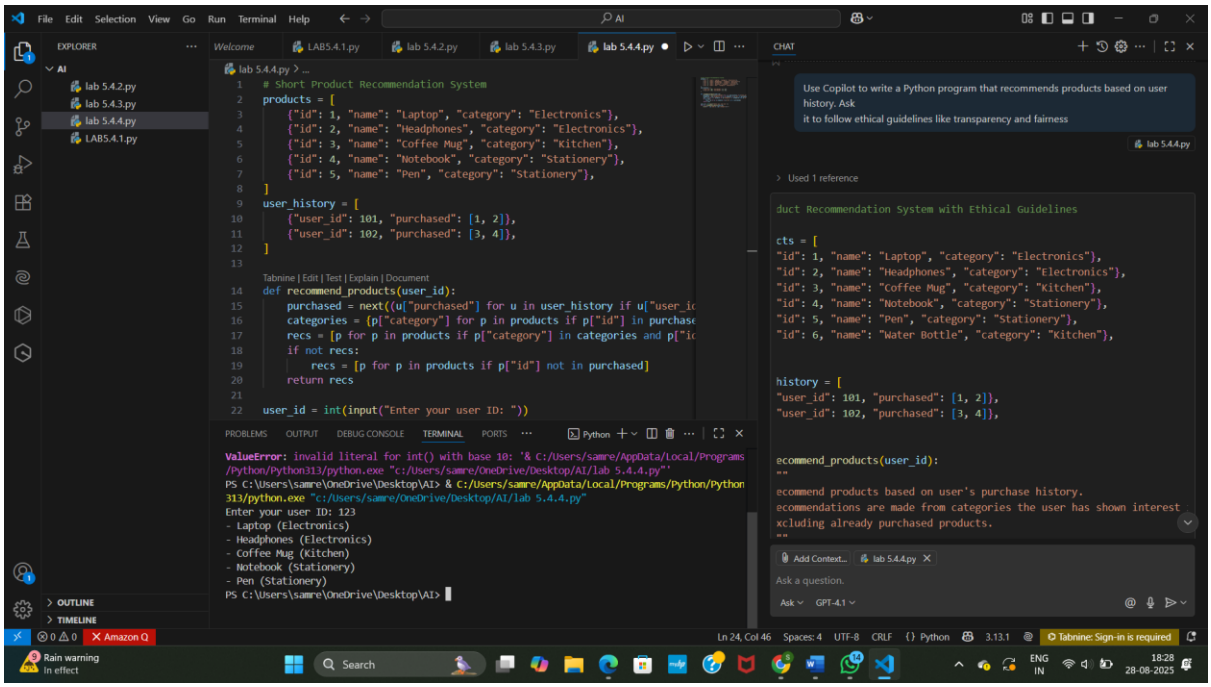
Task:2



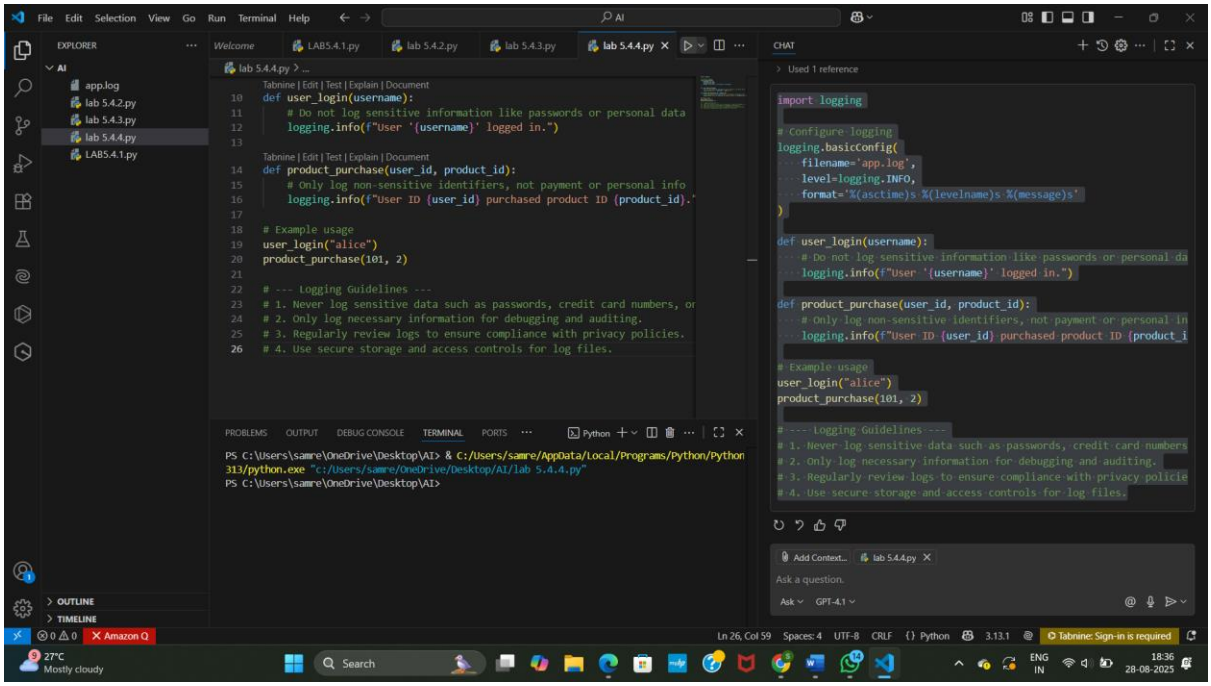
Output:


def sentiment_analysis(text, positive_words, negative_words):
text = text.lower()
words = set(text.split())
pos = len(words & set(positive_words))
neg = len(words & set(negative_words))
if pos > neg:
return 'positive'
elif neg > pos:
return 'negative'
else:
return 'neutral'
positive_words = ['good', 'happy', 'excellent', 'great', 'love']
negative_words = ['bad', 'sad', 'terrible', 'hate', 'poor']
dataset = [
("I love this!", "positive"),
("This is terrible.", "negative"),
("It's okay.", "neutral"),
]
for text, label in dataset:
PS C:\Users\samre\OneDrive\Desktop\AI> & C:/Users/samre/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/samre/OneDrive/Desktop/AI/lab 5.4.2.py"
Text: I love this! | Predicted: positive | Actual: positive
Text: This is terrible. | Predicted: negative | Actual: negative
Text: It's okay. | Predicted: neutral | Actual: neutral
PS C:\Users\samre\OneDrive\Desktop\AI>

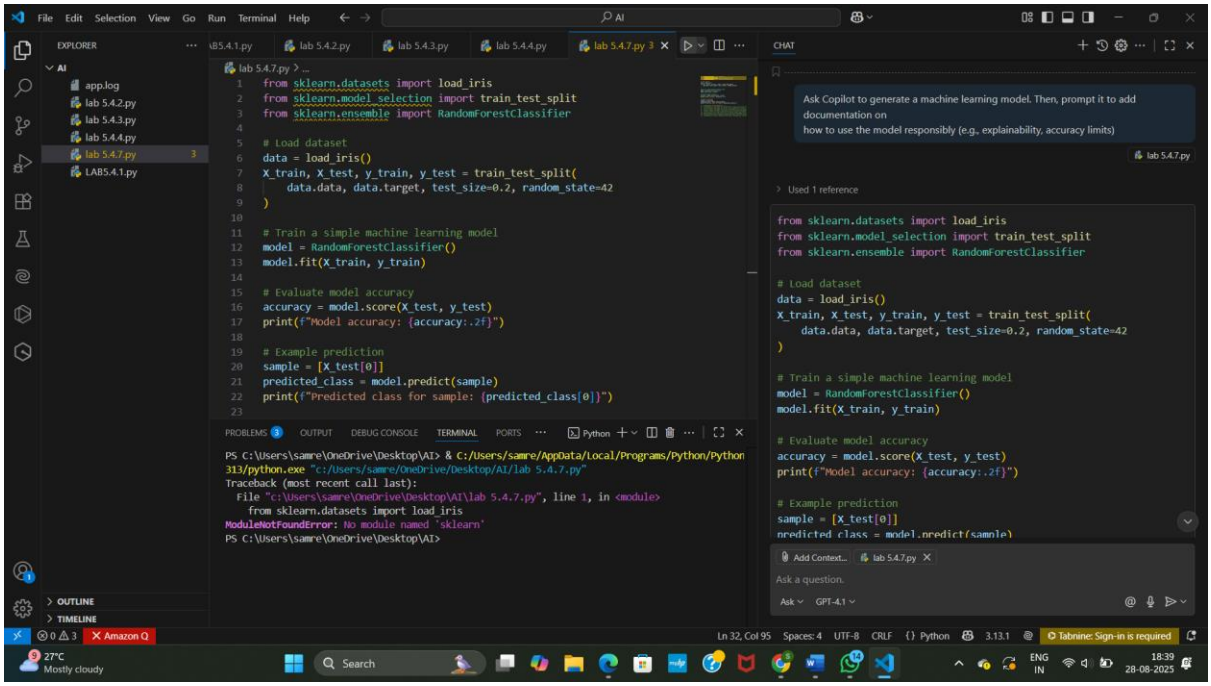
Task:3

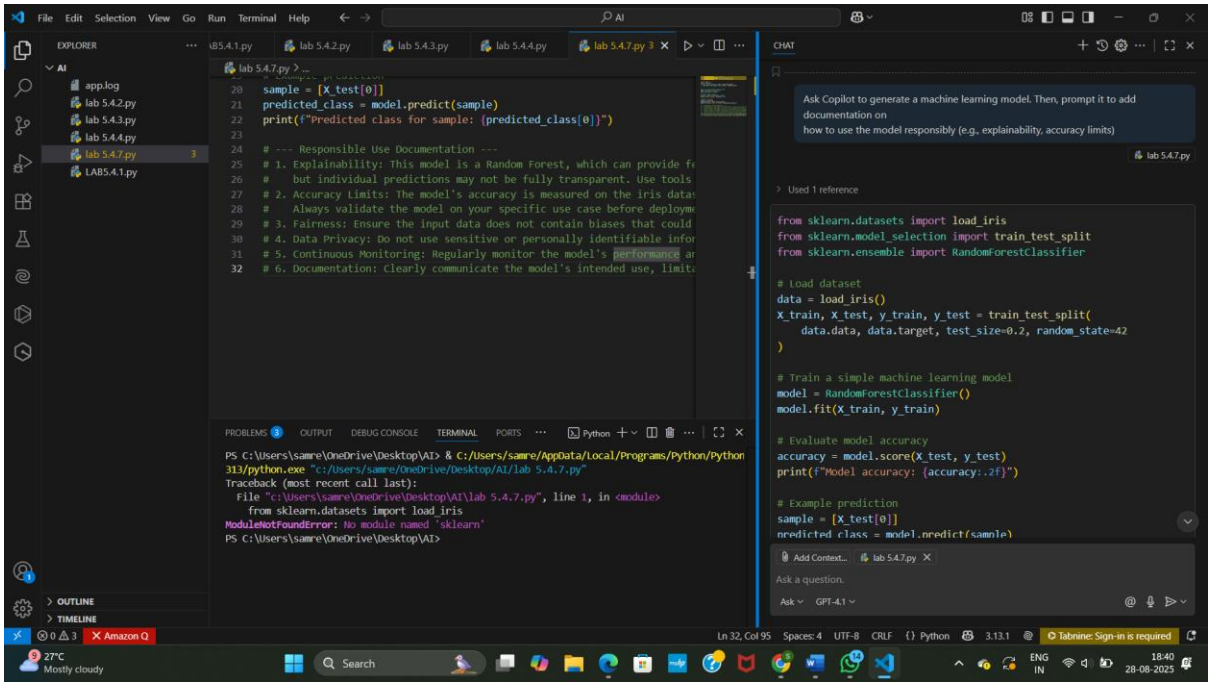

products = [
{"id": 1, "name": "Laptop", "category": "Electronics"},
{"id": 2, "name": "Headphones", "category": "Electronics"},
{"id": 3, "name": "Coffee Mug", "category": "Kitchen"},
{"id": 4, "name": "Notebook", "category": "Stationery"},
{"id": 5, "name": "Pen", "category": "Stationery"},
]
user_history = [
{"user_id": 101, "purchased": [1, 2]},
{"user_id": 102, "purchased": [3, 4]},
]
def recommend_products(user_id):
user_id = int(input("Enter your user ID: "))
ValueError: invalid literal for int() with base 10: '& C:/Users/samre/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/samre/OneDrive/Desktop/AI/lab 5.4.4.py"'
PS C:\Users\samre\OneDrive\Desktop\AI> & C:/Users/samre/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/samre/OneDrive/Desktop/AI/lab 5.4.4.py"
Enter your user ID: 123
- Laptop (Electronics)
- Headphones (Electronics)
- Coffee Mug (Kitchen)
- Notebook (Stationery)
- Pen (Stationery)
PS C:\Users\samre\OneDrive\Desktop\AI>

Task:4


File Edit Selection View Go Run Terminal Help
AI
EXPLORER
AI
app.log
lab 5.4.2.py
lab 5.4.3.py
lab 5.4.4.py
LAB5.4.1.py
Welcome
LAB5.4.1.py
lab 5.4.2.py
lab 5.4.3.py
lab 5.4.4.py
lab 5.4.4.py > ...
Tabnine | Edit | Test | Explain | Document
def user_login(username):
# Do not log sensitive information like passwords or personal data
logging.info(f"User '{username}' logged in.")
Tabnine | Edit | Test | Explain | Document
def product_purchase(user_id, product_id):
# Only log non-sensitive identifiers, not payment or personal info
logging.info(f"User ID {user_id} purchased product ID {product_id}.")
# Example usage
user_login("alice")
product_purchase(101, 2)
# --- Logging Guidelines ---
# 1. Never log sensitive data such as passwords, credit card numbers, or
# 2. Only log necessary information for debugging and auditing.
# 3. Regularly review logs to ensure compliance with privacy policies.
# 4. Use secure storage and access controls for log files.
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python
PS C:\Users\samre\OneDrive\Desktop\AI> & C:/Users/samre/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/samre/OneDrive/Desktop/AI/lab 5.4.4.py"
PS C:\Users\samre\OneDrive\Desktop\AI>
OUTLINE
TIMELINE
Amazon Q
Ln 26, Col 59 Spaces: 4 UTF-8 CRLF Python 3.13.1 Tabnine: Sign-in is required
CHAT
Used 1 reference
import logging
# Configure logging
logging.basicConfig(
filename='app.log',
level=logging.INFO,
format='%(asctime)s %(levelname)s %(message)s'
)
def user_login(username):
# Do not log sensitive information like passwords or personal da
logging.info(f"User '{username}' logged in.")
def product_purchase(user_id, product_id):
# Only log non-sensitive identifiers, not payment or personal in
logging.info(f"User ID {user_id} purchased product ID {product_i
# Example usage
user_login("alice")
product_purchase(101, 2)
# --- Logging Guidelines ---
# 1. Never log sensitive data such as passwords, credit card numbers
# 2. Only log necessary information for debugging and auditing.
# 3. Regularly review logs to ensure compliance with privacy policie
# 4. Use secure storage and access controls for log files.
Add Context...
lab 5.4.4.py
Ask a question.
Ask
GPT-4.1
27°C
Mostly cloudy
Search
ENG
IN
18:36
28-08-2025

Task:5


File Edit Selection View Go Run Terminal Help
AI
EXPLORER
AI
app.log
lab 5.4.2.py
lab 5.4.3.py
lab 5.4.4.py
lab 5.4.7.py 3
LAB5.4.1.py
lab 5.4.2.py
lab 5.4.3.py
lab 5.4.4.py
lab 5.4.7.py 3
lab 5.4.7.py > ...
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
# Load dataset
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
)
# Train a simple machine learning model
model = RandomForestClassifier()
model.fit(X_train, y_train)
# Evaluate model accuracy
accuracy = model.score(X_test, y_test)
print(f"Model accuracy: {accuracy:.2f}")
# Example prediction
sample = [X_test[0]]
predicted_class = model.predict(sample)
print(f"Predicted class for sample: {predicted_class[0]}")
PROBLEMS 3 OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python
PS C:\Users\samre\OneDrive\Desktop\AI> & C:/Users/samre/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/samre/OneDrive/Desktop/AI/lab 5.4.7.py"
Traceback (most recent call last):
File "c:\Users\samre\OneDrive\Desktop\AI\lab 5.4.7.py", line 1, in <module>
from sklearn.datasets import load_iris
ModuleNotFoundError: No module named 'sklearn'
PS C:\Users\samre\OneDrive\Desktop\AI>
OUTLINE
TIMELINE
Amazon Q
Ln 32, Col 95 Spaces: 4 UTF-8 CRLF Python 3.13.1 Tabnine: Sign-in is required
CHAT
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)
lab 5.4.7.py
Used 1 reference
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
# Load dataset
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
)
# Train a simple machine learning model
model = RandomForestClassifier()
model.fit(X_train, y_train)
# Evaluate model accuracy
accuracy = model.score(X_test, y_test)
print(f"Model accuracy: {accuracy:.2f}")
# Example prediction
sample = [X_test[0]]
predicted_class = model.predict(sample)
Add Context...
lab 5.4.7.py
Ask a question.
Ask
GPT-4.1
27°C
Mostly cloudy
Search
ENG
IN
18:39
28-08-2025


File Edit Selection View Go Run Terminal Help
AI
EXPLORER
AI
app.log
lab 5.4.2.py
lab 5.4.3.py
lab 5.4.4.py
lab 5.4.7.py 3
LAB5.4.1.py
lab 5.4.2.py
lab 5.4.3.py
lab 5.4.4.py
lab 5.4.7.py 3
sample = [X_test[0]]
predicted_class = model.predict(sample)
print(f"Predicted class for sample: {predicted_class[0]}")
# --- Responsible Use Documentation ---
# 1. Explainability: This model is a Random Forest, which can provide fe
# but individual predictions may not be fully transparent. Use tools
# 2. Accuracy Limits: The model's accuracy is measured on the iris datas
# Always validate the model on your specific use case before deploym
# 3. Fairness: Ensure the input data does not contain biases that could
# 4. Data Privacy: Do not use sensitive or personally identifiable infor
# 5. Continuous Monitoring: Regularly monitor the model's performance ar
# 6. Documentation: Clearly communicate the model's intended use, limit
PROBLEMS 3
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Python
PS C:\Users\samre\OneDrive\Desktop\AI> & C:/Users/samre/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/samre/OneDrive/Desktop/AI/lab 5.4.7.py"
Traceback (most recent call last):
File "c:\Users\samre\OneDrive\Desktop\AI\lab 5.4.7.py", line 1, in <module>
from sklearn.datasets import load_iris
ModuleNotFoundError: No module named 'sklearn'
PS C:\Users\samre\OneDrive\Desktop\AI>
OUTLINE
TIMELINE
CHAT
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)
lab 5.4.7.py
Used 1 reference
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
# Load dataset
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
)
# Train a simple machine learning model
model = RandomForestClassifier()
model.fit(X_train, y_train)
# Evaluate model accuracy
accuracy = model.score(X_test, y_test)
print(f"Model accuracy: {accuracy:.2f}")
# Example prediction
sample = [X_test[0]]
predicted_class = model.predict(sample)
Add Context...
lab 5.4.7.py
Ask a question.
Ask
GPT-4.1
Ln 32, Col 95
Spaces: 4
UTF-8
CRLF
Python
3.13.1
Tabnine: Sign-in is required
Amazon Q
27°C
Mostly cloudy
Search
ENG
IN
18:40
28-08-2025