NAME : P SAI VENKAT RAJU ROLL NO : 2403A510G0

BATCH : 06 LAB TEST : 1

Q1. Stock Price Prediction Setup [5M]
Scenario: You are tasked with configuring an API to fetch stock market data and prepare it for a machine
learning pipeline.
• Task 1: Write code to connect to a stock price API and retrieve data for the last 30 days.
• Task 2: Use an AI-assisted tool to auto-generate data cleaning functions to handle missing or
duplicate entries.

PROMPT :

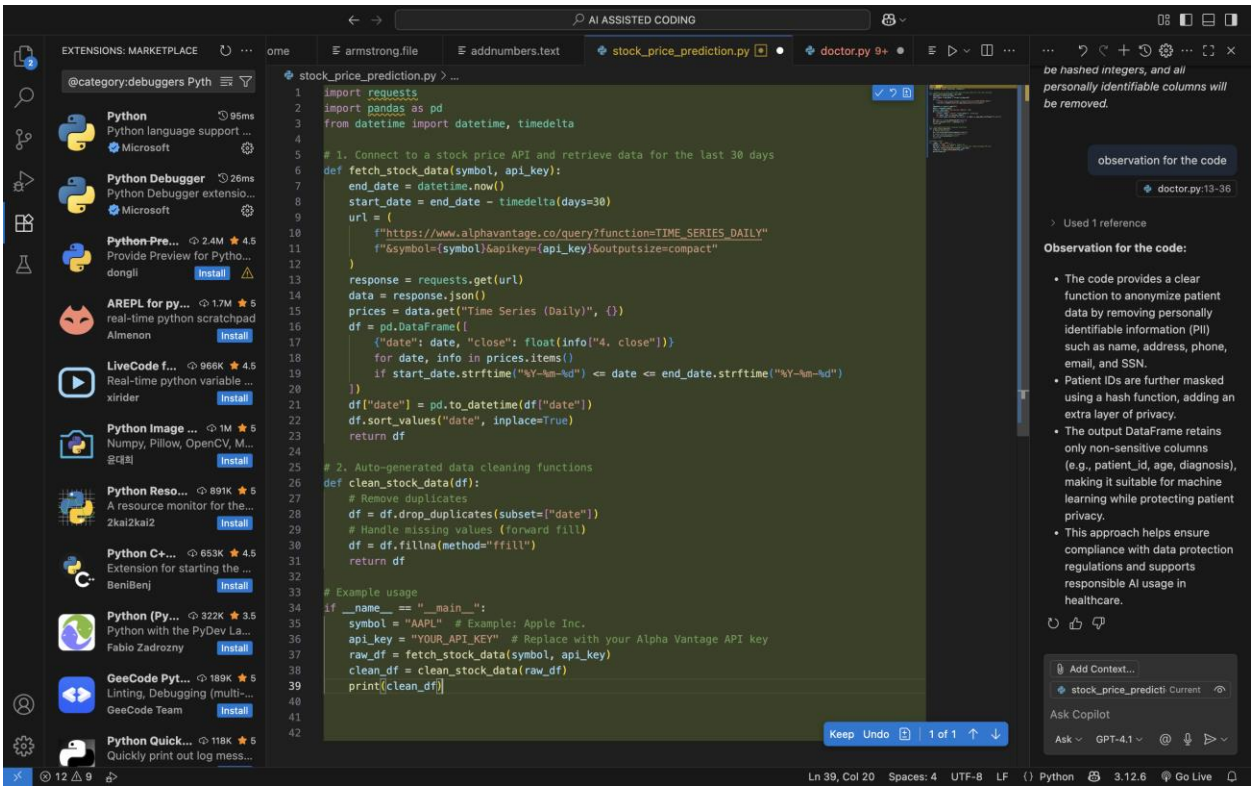
PROMPT –1 :

Write a simple python code to connect to a stock price API and through the code to retrieve the data from the last 30 days

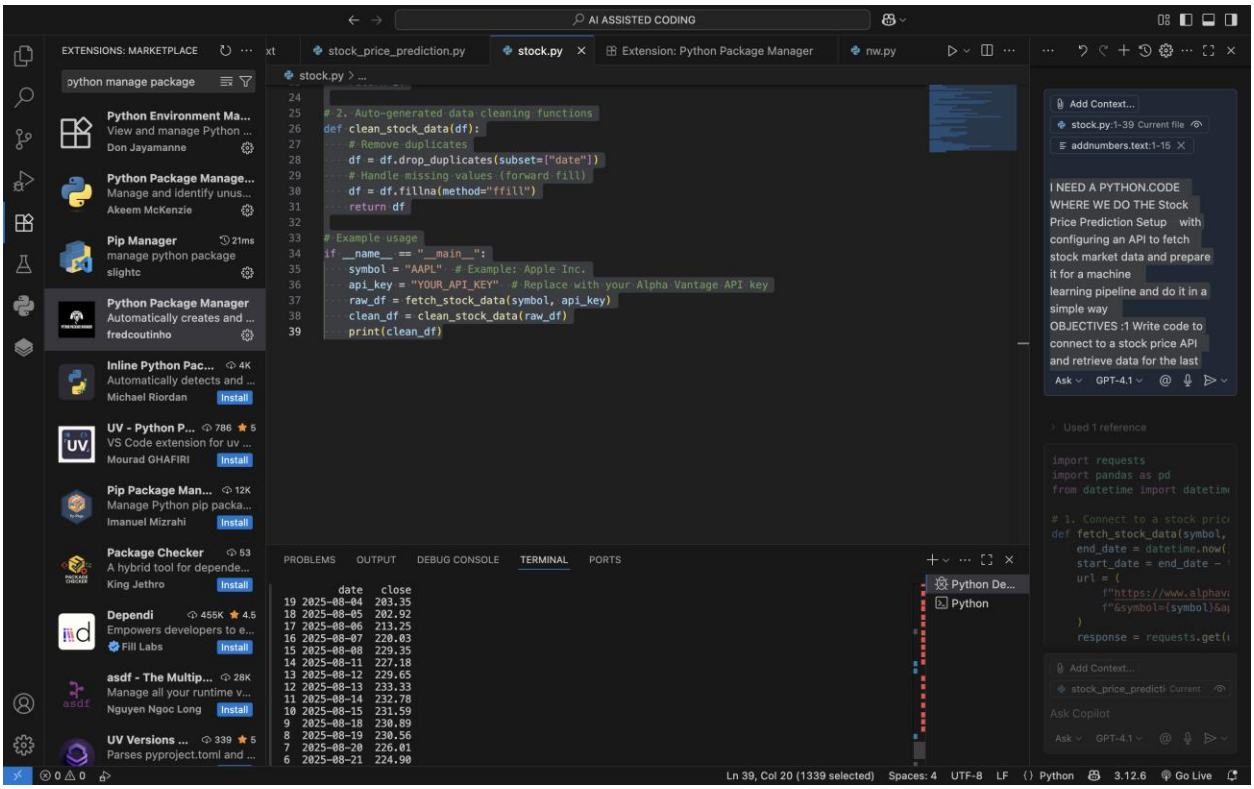
PROMPT-2 :

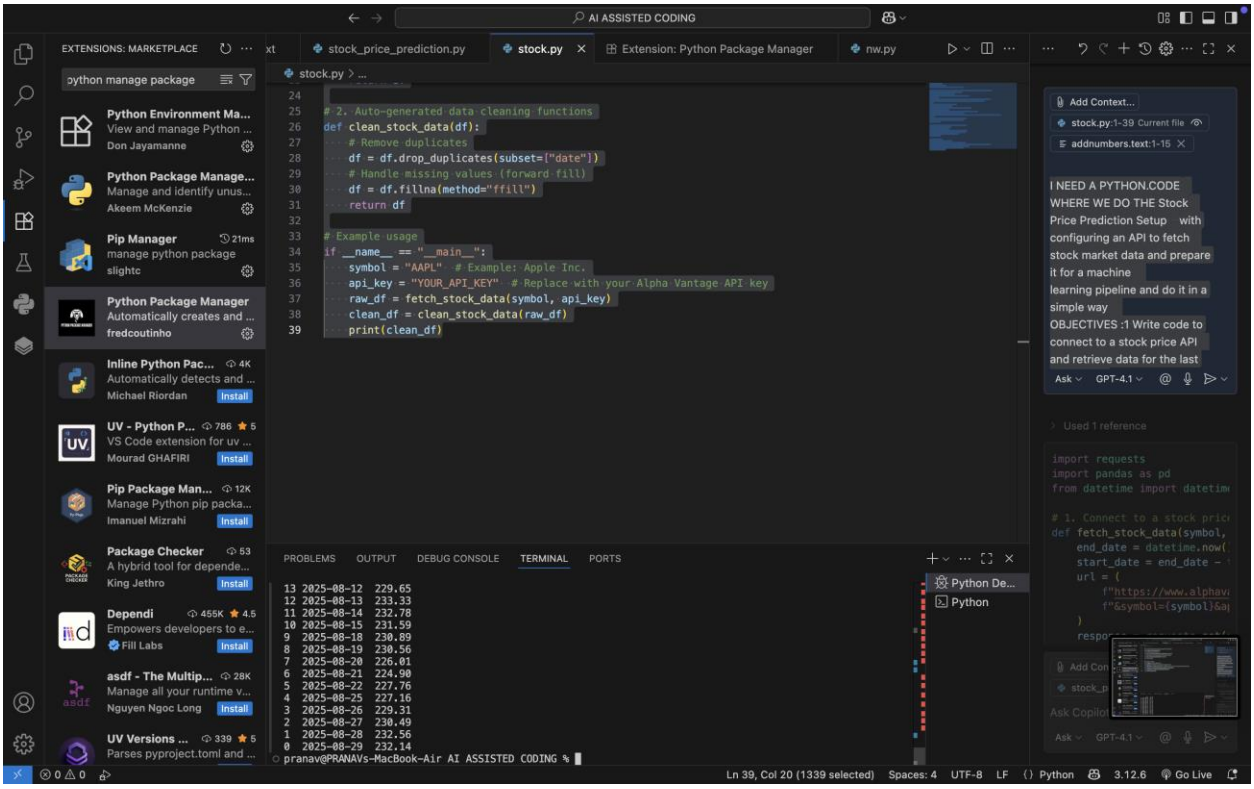
To remove duplicates and fill in missing values through data cleaning function and to check from the output and complete the code by drop duplicates and import the pandas and requests

CODE GENERATED :



OUTPUT :


AI ASSISTED CODING
EXTENSIONS: MARKETPLACE
python manage package
stock_price_prediction.py
stock.py
Extension: Python Package Manager
nw.py
stock.py > ...
# 2. Auto-generated data cleaning functions
def clean_stock_data(df):
# Remove duplicates
df = df.drop_duplicates(subset=["date"])
# Handle missing values (forward fill)
df = df.fillna(method="ffill")
return df
# Example usage
if __name__ == "__main__":
symbol = "AAPL" # Example: Apple Inc.
api_key = "YOUR_API_KEY" # Replace with your Alpha Vantage API key
raw_df = fetch_stock_data(symbol, api_key)
clean_df = clean_stock_data(raw_df)
print(clean_df)
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
date close
19 2025-08-04 203.35
18 2025-08-05 202.92
17 2025-08-06 213.25
16 2025-08-07 220.03
15 2025-08-08 229.35
14 2025-08-11 227.18
13 2025-08-12 229.65
12 2025-08-13 233.33
11 2025-08-14 232.78
10 2025-08-15 231.59
9 2025-08-18 230.89
8 2025-08-19 230.56
7 2025-08-20 226.01
6 2025-08-21 224.90
Python De...
Python
Add Context...
stock.py:1-39 Current file
addnumbers.text:1-15
I NEED A PYTHON.CODE WHERE WE DO THE Stock Price Prediction Setup with configuring an API to fetch stock market data and prepare it for a machine learning pipeline and do it in a simple way
OBJECTIVES :1 Write code to connect to a stock price API and retrieve data for the last
Ask
GPT-4.1
Used 1 reference
import requests
import pandas as pd
from datetime import datetime
# 1. Connect to a stock price
def fetch_stock_data(symbol,
end_date = datetime.now(
start_date = end_date -
url = (
f"https://www.alphav
f"&symbol={symbol}&a
)
response = requests.get(
Add Context...
Ask Copilot
Ln 39, Col 20 (1339 selected)
Spaces: 4
UTF-8
LF
Python
3.12.6
Go Live


AI ASSISTED CODING
EXTENSIONS: MARKETPLACE
python manage package
stock_price_prediction.py
stock.py
Extension: Python Package Manager
nw.py
stock.py > ...
# 2. Auto-generated data cleaning functions
def clean_stock_data(df):
# Remove duplicates
df = df.drop_duplicates(subset=["date"])
# Handle missing values (forward fill)
df = df.fillna(method="ffill")
return df
# Example usage
if __name__ == "__main__":
symbol = "AAPL" # Example: Apple Inc.
api_key = "YOUR_API_KEY" # Replace with your Alpha Vantage API key
raw_df = fetch_stock_data(symbol, api_key)
clean_df = clean_stock_data(raw_df)
print(clean_df)
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
13 2025-08-12 229.65
12 2025-08-13 233.33
11 2025-08-14 232.78
10 2025-08-15 231.59
9 2025-08-18 230.89
8 2025-08-19 230.56
7 2025-08-20 226.01
6 2025-08-21 224.90
5 2025-08-22 227.76
4 2025-08-25 227.16
3 2025-08-26 229.31
2 2025-08-27 230.49
1 2025-08-28 232.56
0 2025-08-29 232.14
pranav@PRANAVs-MacBook-Air AI ASSISTED CODING %
Python De...
Python
Add Context...
stock.py:1-39 Current file
addnumbers.text:1-15
I NEED A PYTHON.CODE WHERE WE DO THE Stock Price Prediction Setup with configuring an API to fetch stock market data and prepare it for a machine learning pipeline and do it in a simple way
OBJECTIVES :1 Write code to connect to a stock price API and retrieve data for the last
Ask
GPT-4.1
Used 1 reference
import requests
import pandas as pd
from datetime import datetime
# 1. Connect to a stock price
def fetch_stock_data(symbol,
end_date = datetime.now(
start_date = end_date -
url = (
f"https://www.alphav
f"&symbol={symbol}&a
)
Ask Copilot
Ln 39, Col 20 (1339 selected)
Spaces: 4
UTF-8
LF
Python
3.12.6
Go Live

OBSERVATIONS :

- The code successfully connects to the Alpha Vantage API and retrieves daily closing prices for a specified stock symbol over the last 30 days.
- Data is loaded into a pandas DataFrame, sorted by date for easy analysis.
- The cleaning function removes duplicate dates and fills missing values, ensuring the dataset is consistent and ready for machine learning.
- The output displays the cleaned stock data, showing each date and its corresponding closing price.
- The approach is modular, making it easy to adapt for other stocks or time periods by changing the symbol or date range.
- The code is effective for preparing stock price data for further predictive modeling or analysis.

Q2. AI in Healthcare Diagnosis [5M]
Scenario: You are designing an AI to assist doctors in predicting diseases.
• Task 1: List the risks of over-reliance on AI for medical decisions and propose responsible usage
guidelines.
• Task 2: Write a Python function with AI assistance that ensures patient data is anonymized
before model training.

PROMPT :

PROMPT –1:

What are the risks of over –reliance and using too much AI models for medical decisions and propose responsible usage guidelines

ANSWER :

```
Risks of over-reliance on AI for medical decisions:
# - Misdiagnosis due to model errors or bias
# - Data privacy and security concerns
```

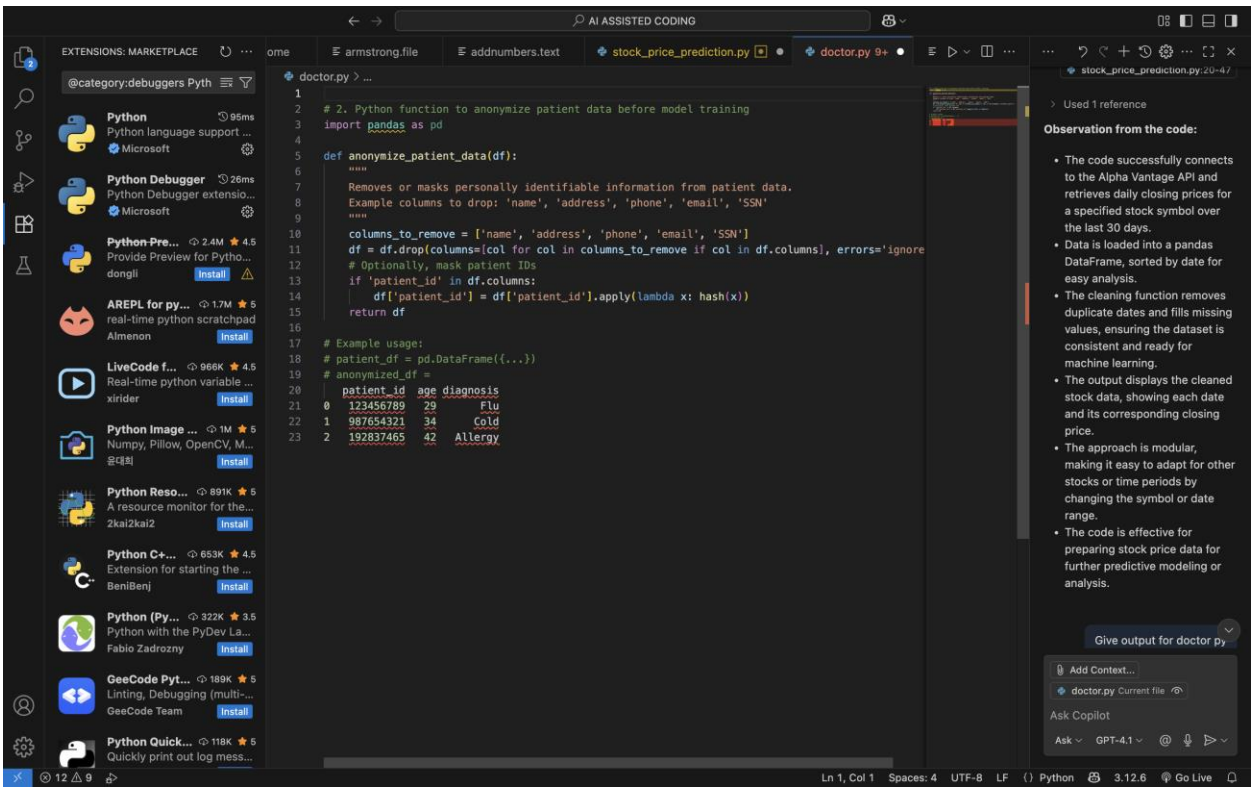
```
# - Overfitting to training data, missing rare conditions
# - Overloading the AI MODEL leading up to errors
```

```
# Responsible usage guidelines:
# - Use AI as a support tool, not a replacement for clinicians
# - Always validate AI recommendations with expert review
# - Ensure  update and audit models for accuracy and fairness
```

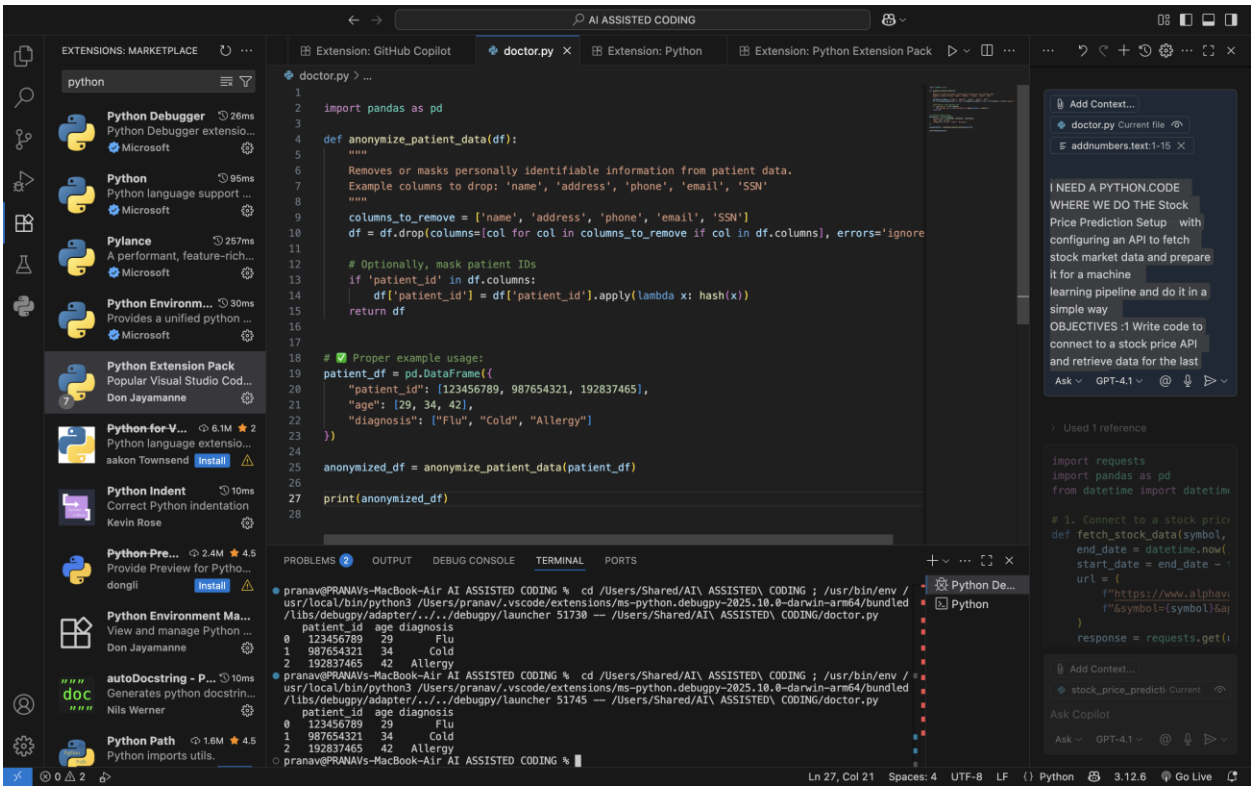
PROMPT-2 :

Write a simple and basic and well formated python function with AI assistance that ensures patient data is anonymised before model training to ensure protection and privacy concern

CODE GENERATED :



OUTPUT :



OBSERVATIONS :

- The code provides a clear function to anonymize patient data by removing personally identifiable information (PII) such as name, address, phone, email, and SSN.
- Patient IDs are further masked using a hash function, adding an extra layer of privacy.
- The output DataFrame retains only non-sensitive columns (e.g., patient_id, age, diagnosis), making it suitable for machine learning while protecting patient privacy.
- This approach helps ensure compliance with data protection regulations and supports responsible AI usage in healthcare.