Import openai

Import pandas as pd

# Load your OpenAI API key from a secure location

Openai.api\_key = ‘YOUR\_OPENAI\_API\_KEY’

# Load datasets from the provided sources

Transaction\_data = pd.read\_csv(‘customer\_transactions.csv’) # Replace with the actual file path

Loan\_data = pd.read\_csv(‘loan\_payment\_history.csv’) # Replace with the actual file path

Market\_data = pd.read\_csv(‘nifty\_historical\_data.csv’) # Replace with the actual file path

Financial\_transactions = pd.read\_csv(‘financial\_transactions.csv’) # Replace with the actual file path

# Function to take inputs from the user for real-time data

Def get\_customer\_input():

Customer\_id = input(“Enter Customer ID: “)

Name = input(“Enter Customer Name: “)

Age = int(input(“Enter Age: “))

Income = float(input(“Enter Monthly Income: “))

Cibil\_score = int(input(“Enter CIBIL Score: “))

Savings\_per\_year = float(input(“Enter Amount Saved Last Year: “))

Return {

‘customer\_id’: customer\_id,

‘name’: name,

‘age’: age,

‘income’: income,

‘cibil\_score’: cibil\_score,

‘savings\_per\_year’: savings\_per\_year

}

# Preprocess customer transaction data for advisory generation

Def preprocess\_transaction\_data(customer\_id):

Customer\_transactions = transaction\_data[transaction\_data[‘CustomerID’] == customer\_id]

Total\_amount = customer\_transactions[‘TransactionAmount (INR)’].sum()

Return {

‘total\_amount’: total\_amount

}

# Preprocess customer loan data for credit insights

Def preprocess\_loan\_data(customer\_id):

Customer\_loans = loan\_data[loan\_data[‘Loan\_ID’] == customer\_id]

Loan\_info = []

For \_, loan in customer\_loans.iterrows():

Loan\_info.append({

‘loan\_status’: loan[‘loan\_status’],

‘principal’: loan[‘Principal’],

‘terms’: loan[‘terms’],

‘effective\_date’: loan[‘effective\_date’],

‘due\_date’: loan[‘due\_date’],

‘paid\_off\_time’: loan[‘paid\_off\_time’],

‘past\_due\_days’: loan[‘past\_due\_days’]

})

Return loan\_info

# Preprocess market data for trends and insights

Def preprocess\_market\_data():

Recent\_market\_trends = market\_data.tail(5)[[‘Date’, ‘Open’, ‘High’, ‘Low’, ‘Close’, ‘Volume’]].to\_dict(orient=’records’)

Return recent\_market\_trends

# Preprocess financial transactions data for additional insights

Def preprocess\_financial\_transactions():

Transaction\_summary = financial\_transactions[[‘Transaction ID’, ‘Date’, ‘Currency’, ‘Sender’, ‘Receiver’, ‘Amount’, ‘Fee’, ‘Type’]].to\_dict(orient=’records’)

Return transaction\_summary

# Function to generate financial advisory based on customer info and datasets

Def generate\_financial\_advisory(personal\_info, transaction\_info, loan\_info, market\_trends, financial\_trans):

Prompt = f”””

Financial Advisory:

- Customer Name: {personal\_info[‘name’]}

- Age: {personal\_info[‘age’]}

- Monthly Income: {personal\_info[‘income’]}

- CIBIL Score: {personal\_info[‘cibil\_score’]}

- Savings Last Year: {personal\_info[‘savings\_per\_year’]}

Transaction Summary:

* Total Transaction Amount: {transaction\_info[‘total\_amount’]}

Loan Summary:

“””

For loan in loan\_info:

Prompt += f”””

- Loan Status: {loan[‘loan\_status’]}

- Principal: {loan[‘principal’]}

- Terms: {loan[‘terms’]}

- Effective Date: {loan[‘effective\_date’]}

- Due Date: {loan[‘due\_date’]}

- Paid Off Time: {loan[‘paid\_off\_time’]}

- Past Due Days: {loan[‘past\_due\_days’]}

“””

Prompt += f”””

Recent Market Trends (Nifty Index):

{market\_trends}

Financial Transactions Summary:

{financial\_trans}

Provide a financial advisory for this customer focusing on saving strategies, credit management, debt reduction, and potential investment opportunities considering the recent market trends.

“””

Response = openai.ChatCompletion.create(

Model=”gpt-4”,

Messages=[

{“role”: “system”, “content”: “You are a financial advisor providing guidance based on customer data and current market trends.”},

{“role”: “user”, “content”: prompt}

],

Max\_tokens=150,

Temperature=0.7

)

Advisory = response[‘choices’][0][‘message’][‘content’]

Return advisory

# Main code to get inputs, process data, and generate advisory

Personal\_info = get\_customer\_input()

Transaction\_info = preprocess\_transaction\_data(personal\_info[‘customer\_id’])

Loan\_info = preprocess\_loan\_data(personal\_info[‘customer\_id’])

Market\_trends = preprocess\_market\_data()

Financial\_trans = preprocess\_financial\_transactions()

# Generate advisory

Advisory\_text = generate\_financial\_advisory(personal\_info, transaction\_info, loan\_info, market\_trends, financial\_trans)

# Display advisory

Print(“\nGenerated Financial Advisory:\n”)

Print(advisory\_text)