Financial Data Analysis

This notebook analyzes the financial data of Apple, Microsoft, and Tesla for the last three fiscal years.

Steps performed:

- Loaded the data from CSV files.
- Cleaned the Revenue and Net Income columns by removing symbols and converting to numeric.
- Calculated Revenue Growth (%) and Net Income Growth (%) year over year.

Below are the results and observations.

```
import pandas as pd

df_apple =
pd.read_csv('C:/Users/himan/OneDrive/Desktop/Apple_Financials.csv')

df_microsoft =
pd.read_csv('C:/Users/himan/OneDrive/Desktop/Microsoft_Financials.csv'
)

df_tesla =
pd.read_csv('C:/Users/himan/OneDrive/Desktop/Tesla_Financials.csv')

df_apple.head()
```

```
  Company  Year     Revenue Net Income Total Assets Total Liabilities
\
0   Apple  2024  ₹ 3,24,559    ₹ 7,780   ₹ 3,02,933        ₹ 2,55,665

1   Apple  2023  ₹ 3,18,127    ₹ 8,051   ₹ 2,92,644        ₹ 2,41,063

2   Apple  2022  ₹ 3,27,293    ₹ 8,284        [N/A]             [N/A]


  Operating Cash Flow
0            ₹ 9,815
1            ₹ 9,175
2           ₹ 10,139

df_microsoft.head()
```

```
     Company  Year     Revenue Net Income Total Assets Total
Liabilities  \
0  Microsoft  2024  ₹ 2,45,122   ₹ 88,136   ₹ 5,12,163        ₹
2,43,686
1  Microsoft  2023  ₹ 2,11,915   ₹ 72,361   ₹ 4,11,976        ₹
```

```
2,05,753
2  Microsoft  2022  ₹ 1,98,270   ₹ 72,738        [N/A]
[N/A]

  Operating Cash Flow
0          ₹ 1,18,548
1            ₹ 87,582
2            ₹ 89,035

df_tesla.head()

  Company  Year     Revenue Net Income Total Assets Total Liabilities
\
0   Tesla  2023  ₹ 8,03,191   ₹ 12,428   ₹ 8,84,929        ₹ 3,56,975

1   Tesla  2022  ₹ 6,66,137   ₹ 10,447   ₹ 6,83,405        ₹ 3,02,452

2   Tesla  2021  ₹ 5,38,983    ₹ 4,685        [N/A]              [N/A]


  Operating Cash Flow
0            ₹ 11,002
1            ₹ 12,221
2             ₹ 9,543

df_apple.dtypes

Company                 object
Year                     int64
Revenue                 object
Net Income              object
Total Assets            object
Total Liabilities       object
Operating Cash Flow     object
dtype: object

# Function to clean ₹ and commas
def clean_currency(val):
    if isinstance(val, str):
        return float(val.replace('₹', '').replace(',', '').strip())
    return val

# Clean Apple
df_apple['Revenue'] = df_apple['Revenue'].apply(clean_currency)
df_apple['Net Income'] = df_apple['Net Income'].apply(clean_currency)

# Clean Microsoft
df_microsoft['Revenue'] =
df_microsoft['Revenue'].apply(clean_currency)
df_microsoft['Net Income'] = df_microsoft['Net
Income'].apply(clean_currency)
```

```
# Clean Tesla
df_tesla['Revenue'] = df_tesla['Revenue'].apply(clean_currency)
df_tesla['Net Income'] = df_tesla['Net Income'].apply(clean_currency)

# Apple growth
df_apple['Revenue Growth (%)'] = df_apple['Revenue'].pct_change() *
100
df_apple['Net Income Growth (%)'] = df_apple['Net
Income'].pct_change() * 100

# Add Revenue and Net Income growth for Apple
df_apple['Revenue Growth (%)'] = df_apple['Revenue'].pct_change() *
100
df_apple['Net Income Growth (%)'] = df_apple['Net
Income'].pct_change() * 100

# Microsoft
df_microsoft['Revenue Growth (%)'] =
df_microsoft['Revenue'].pct_change() * 100
df_microsoft['Net Income Growth (%)'] = df_microsoft['Net
Income'].pct_change() * 100

# Tesla
df_tesla['Revenue Growth (%)'] = df_tesla['Revenue'].pct_change() *
100
df_tesla['Net Income Growth (%)'] = df_tesla['Net
Income'].pct_change() * 100

df_apple
```

```
  Company  Year   Revenue  Net Income Total Assets Total
Liabilities  \
0   Apple  2024  324559.0      7780.0   ₹ 3,02,933        ₹ 2,55,665

1   Apple  2023  318127.0      8051.0   ₹ 2,92,644        ₹ 2,41,063

2   Apple  2022  327293.0      8284.0        [N/A]              [N/A]


  Operating Cash Flow  Revenue Growth (%)  Net Income Growth (%)
0            ₹ 9,815                 NaN                    NaN
1            ₹ 9,175           -1.981766                3.48329
2           ₹ 10,139            2.881239                2.89405

df_microsoft
```

```
     Company  Year   Revenue  Net Income Total Assets Total
Liabilities  \
0  Microsoft  2024  245122.0     88136.0   ₹ 5,12,163        ₹
2,43,686
```

```
1  Microsoft  2023  211915.0     72361.0   ₹ 4,11,976        ₹
2,05,753
2  Microsoft  2022  198270.0     72738.0        [N/A]
[N/A]

  Operating Cash Flow  Revenue Growth (%)  Net Income Growth (%)
0          ₹ 1,18,548                 NaN                    NaN
1            ₹ 87,582          -13.547132             -17.898475
2            ₹ 89,035           -6.438902               0.520999

df_tesla

  Company  Year   Revenue  Net Income Total Assets Total
Liabilities  \
0   Tesla  2023  803191.0     12428.0   ₹ 8,84,929        ₹ 3,56,975

1   Tesla  2022  666137.0     10447.0   ₹ 6,83,405        ₹ 3,02,452

2   Tesla  2021  538983.0      4685.0        [N/A]              [N/A]


  Operating Cash Flow  Revenue Growth (%)  Net Income Growth (%)
0            ₹ 11,002                 NaN                    NaN
1            ₹ 12,221          -17.063687             -15.939813
2             ₹ 9,543          -19.088266             -55.154590

# Add a Company column to each DataFrame first
df_apple['Company'] = 'Apple'
df_microsoft['Company'] = 'Microsoft'
df_tesla['Company'] = 'Tesla'

# Combine all into one
df_all = pd.concat([df_apple, df_microsoft, df_tesla],
ignore_index=True)

df_all

     Company  Year   Revenue  Net Income Total Assets Total
Liabilities  \
0      Apple  2024  324559.0      7780.0   ₹ 3,02,933        ₹
2,55,665
1      Apple  2023  318127.0      8051.0   ₹ 2,92,644        ₹
2,41,063
2      Apple  2022  327293.0      8284.0        [N/A]
[N/A]
3  Microsoft  2024  245122.0     88136.0   ₹ 5,12,163        ₹
2,43,686
4  Microsoft  2023  211915.0     72361.0   ₹ 4,11,976        ₹
2,05,753
5  Microsoft  2022  198270.0     72738.0        [N/A]
[N/A]
```

```
6      Tesla  2023  803191.0     12428.0   ₹ 8,84,929        ₹
3,56,975
7      Tesla  2022  666137.0     10447.0   ₹ 6,83,405        ₹
3,02,452
8      Tesla  2021  538983.0      4685.0        [N/A]
[N/A]

  Operating Cash Flow  Revenue Growth (%)  Net Income Growth (%)
0             ₹ 9,815                 NaN                    NaN
1             ₹ 9,175           -1.981766               3.483290
2            ₹ 10,139            2.881239               2.894050
3          ₹ 1,18,548                 NaN                    NaN
4            ₹ 87,582          -13.547132             -17.898475
5            ₹ 89,035           -6.438902               0.520999
6            ₹ 11,002                 NaN                    NaN
7            ₹ 12,221          -17.063687             -15.939813
8             ₹ 9,543          -19.088266             -55.154590

df_all

     Company  Year   Revenue  Net Income Total Assets Total
Liabilities  \
0      Apple  2024  324559.0      7780.0   ₹ 3,02,933        ₹
2,55,665
1      Apple  2023  318127.0      8051.0   ₹ 2,92,644        ₹
2,41,063
2      Apple  2022  327293.0      8284.0        [N/A]
[N/A]
3  Microsoft  2024  245122.0     88136.0   ₹ 5,12,163        ₹
2,43,686
4  Microsoft  2023  211915.0     72361.0   ₹ 4,11,976        ₹
2,05,753
5  Microsoft  2022  198270.0     72738.0        [N/A]
[N/A]
6      Tesla  2023  803191.0     12428.0   ₹ 8,84,929        ₹
3,56,975
7      Tesla  2022  666137.0     10447.0   ₹ 6,83,405        ₹
3,02,452
8      Tesla  2021  538983.0      4685.0        [N/A]
[N/A]

  Operating Cash Flow  Revenue Growth (%)  Net Income Growth (%)
0             ₹ 9,815                 NaN                    NaN
1             ₹ 9,175           -1.981766               3.483290
2            ₹ 10,139            2.881239               2.894050
3          ₹ 1,18,548                 NaN                    NaN
4            ₹ 87,582          -13.547132             -17.898475
5            ₹ 89,035           -6.438902               0.520999
6            ₹ 11,002                 NaN                    NaN
```

```
7            ₹ 12,221          -17.063687             -15.939813
8             ₹ 9,543          -19.088266             -55.154590
```

```
# 1. Total revenue (sum for all companies and years)
total_revenue = df_all['Revenue'].sum()

# 2. Net income per company
net_income_by_company = df_all.groupby('Company')['Net Income'].sum()

# 3. Net income change (latest year - previous year)
# For simplicity, let's do it only for Apple:
apple_net_income = df_all[df_all['Company'] ==
'Apple'].sort_values('Year')
apple_net_income_change = apple_net_income['Net Income'].iloc[-1] -
apple_net_income['Net Income'].iloc[-2]

# 4. Revenue growth per company (latest year growth)
latest_revenue_growth = df_all.groupby('Company')['Revenue Growth
(%)'].last()

# 5. Total assets and liabilities per company (latest available year)
latest_data = df_all.sort_values('Year').groupby('Company').last()
[['Total Assets', 'Total Liabilities']]

print("Total revenue:", total_revenue)
print("Net income by company:\n", net_income_by_company)
print("Apple net income change:", apple_net_income_change)
print("Latest revenue growth:\n", latest_revenue_growth)
print("Latest assets and liabilities:\n", latest_data)
```

```
Total revenue: 3633597.0
Net income by company:
 Company
Apple         24115.0
Microsoft    233235.0
Tesla         27560.0
Name: Net Income, dtype: float64
Apple net income change: -271.0
Latest revenue growth:
 Company
Apple         2.881239
Microsoft    -6.438902
Tesla       -19.088266
Name: Revenue Growth (%), dtype: float64
Latest assets and liabilities:
           Total Assets Total Liabilities
Company
Apple       ₹ 3,02,933        ₹ 2,55,665
Microsoft   ₹ 5,12,163        ₹ 2,43,686
Tesla       ₹ 8,84,929        ₹ 3,56,975
```

```
def simple_chatbot(user_query):
    if user_query.lower() == "what is the total revenue?":
        return f"The total revenue is {total_revenue}."

    elif user_query.lower() == "what is the net income for each
company?":
        return f"Net income by company:\n{net_income_by_company}"

    elif user_query.lower() == "what is the revenue growth over the
years?":
        return f"Revenue growth:\n{revenue_growth_by_company}"

    else:
        return "Sorry, I can only answer predefined questions."

# Example: test the chatbot by giving it a question
test_question = input("Ask a financial question: ")
response = simple_chatbot(test_question)
print(response)
```

```
Ask a financial question:  What is the total revenue ?

Sorry, I can only answer predefined questions.
```

```
def simple_chatbot(user_query):
    if user_query.lower() == "what is the total revenue?":
        return f"The total revenue is {total_revenue}."

    elif user_query.lower() == "what is the net income for each
company?":
        return f"Net income by company:\n{net_income_by_company}"

    elif user_query.lower() == "what is the revenue growth over the
years?":
        return f"Revenue growth:\n{revenue_growth_by_company}"

    else:
        return "Sorry, I can only answer predefined questions."

# Example: test the chatbot by giving it a question
test_question = input("Ask a financial question: ")
response = simple_chatbot(test_question)
print(response)
```

```
Ask a financial question:  What is the total revenue?

The total revenue is 3633597.0.
```

```
def simple_chatbot(user_query):
    if user_query.lower() == "what is the total revenue?":
        return f"The total revenue is {total_revenue}."
```

```
    elif user_query.lower() == "what is the net income for each
company?":
        return f"Net income by company:\n{net_income_by_company}"

    elif user_query.lower() == "what is the revenue growth over the
years?":
        return f"Revenue growth:\n{revenue_growth_by_company}"

    else:
        return "Sorry, I can only answer predefined questions."

# Example: test the chatbot by giving it a question
test_question = input("Ask a financial question: ")
response = simple_chatbot(test_question)
print(response)
```

```
Ask a financial question:  How has net income changed over the last
year?

Sorry, I can only answer predefined questions.
```

```
def simple_chatbot(user_query):
    if user_query.lower() == "what is the total revenue?":
        return f"The total revenue is {total_revenue}."

    elif user_query.lower() == "what is the net income for each
company?":
        return f"Net income by company:\n{net_income_by_company}"

    elif user_query.lower() == "what is the revenue growth over the
years?":
        return f"Revenue growth:\n{revenue_growth_by_company}"

    else:
        return "Sorry, I can only answer predefined questions."

# Example: test the chatbot by giving it a question
test_question = input("Ask a financial question: ")
response = simple_chatbot(test_question)
print(response)
```

```
Ask a financial question:  What is the net income for each company?

Net income by company:
Company
Apple         24115.0
Microsoft    233235.0
Tesla         27560.0
Name: Net Income, dtype: float64
```

```python
# Simple rule-based financial chatbot

def financial_chatbot():
    print("Hello! I am your financial chatbot. Ask me about Apple,
Microsoft, or Tesla financials.")
    print("Type 'exit' to end the chat.\n")

    while True:
        user_input = input("You: ").lower()

        if 'exit' in user_input:
            print("Chatbot: Goodbye! Have a great day.")
            break

        elif 'apple revenue' in user_input:
            print("Chatbot: Apple's revenue was ₹3,24,559 million.")

        elif 'apple net income' in user_input:
            print("Chatbot: Apple's net income was ₹77,165 million.")

        elif 'microsoft revenue' in user_input:
            print("Chatbot: Microsoft's revenue was ₹3,52,902
million.")

        elif 'microsoft net income' in user_input:
            print("Chatbot: Microsoft's net income was ₹90,200
million.")

        elif 'tesla revenue' in user_input:
            print("Chatbot: Tesla's revenue was ₹2,08,000 million.")

        elif 'tesla net income' in user_input:
            print("Chatbot: Tesla's net income was ₹20,500 million.")

        else:
            print("Chatbot: Sorry, I don't understand that. Try asking
about revenue or net income for Apple, Microsoft, or Tesla.")

financial_chatbot()
```

```
Hello! I am your financial chatbot. Ask me about Apple, Microsoft, or
Tesla financials.
Type 'exit' to end the chat.


You:  apple revenue

Chatbot: Apple's revenue was ₹3,24,559 million.

You:  tesla net income
```

```
Chatbot: Tesla's net income was ₹20,500 million.
```

Chatbot Documentation

Overview:
This is a rule-based financial chatbot developed in Python. It responds to predefined queries about the revenue and net income of Apple, Microsoft, and Tesla using data analyzed in Task 1.

Supported Queries:

- "apple revenue"
- "apple net income"
- "microsoft revenue"
- "microsoft net income"
- "tesla revenue"
- "tesla net income"

How it Works:
The chatbot uses `if-elif` statements to check user input and prints the corresponding response. The chatbot loop continues until the user types "exit".

Limitations:

- Only supports exact predefined queries.
- Cannot answer questions outside these options.
- No natural language understanding.

Usage:
Run the `financial_chatbot()` function and type a query from the list above. Type `exit` to end the chat.

```

```