In [28]:
```
import pandas as pd
```

In [29]:
```
df = pd.read_csv(r'C:\Users\malav\OneDrive\Desktop\10K_Filings.csv')
```

In [30]:
```
cols_to_convert = ['Total Revenue', 'Net Income', 'Total Assets', 'Total Liabilities', 'Cash Flow']
for col in cols_to_convert:
    df[col] = df[col].replace('[\$,]', '', regex=True).astype(float)
```

In [31]:
```
df['Revenue Growth (%)'] = df.groupby('Company')['Total Revenue'].pct_change() * 100
df['Income Growth (%)'] = df.groupby('Company')['Net Income'].pct_change() * 100
df.fillna(0, inplace=True)
print(df)
```

```
     Company  Year  Total Revenue  Net Income  Total Assets  \
0  Microsoft  2022       198270.0     72738.0      364840.0
1  Microsoft  2023       211915.0     72361.0      411976.0
2  Microsoft  2024       245122.0     88136.0      512163.0
3      Tesla  2022        81462.0     12587.0       82338.0
4      Tesla  2023        96773.0     14974.0      106618.0
5      Tesla  2024        97690.0      7153.0      122070.0
6      Apple  2022       394328.0     99803.0      352755.0
7      Apple  2023       383285.0     96995.0      352583.0
8      Apple  2024       391035.0     93736.0      364980.0

   Total Liabilities  Cash Flow  Revenue Growth (%)  Income Growth (%)
0           198298.0    89035.0            0.000000           0.000000
1           205753.0    87582.0            6.882030          -0.518299
2           243686.0   118548.0           15.669962          21.800417
3            36440.0    14724.0            0.000000           0.000000
4            43009.0    13256.0           18.795267          18.964010
5            48390.0    14923.0            0.947578         -52.230533
6           302083.0   118254.0            0.000000           0.000000
7           290437.0   110543.0           -2.800461          -2.813543
8           308030.0   118254.0            2.021994          -3.359967
```

In [32]:
```
summary = df.groupby('Company').agg({
    'Revenue Growth (%)': 'mean',
    'Income Growth (%)': 'mean',
}).reset_index()
print(summary)
```

	Company	Revenue Growth (%)	Income Growth (%)
0	Apple	-0.259489	-2.057837
1	Microsoft	7.517331	7.094040
2	Tesla	6.580948	-11.088841

In [22]:
```
!pip install pandas flask
```

```
Requirement already satisfied: pandas in c:\users\malav\anaconda3\lib\site-packages (1.4.4)
Requirement already satisfied: flask in c:\users\malav\anaconda3\lib\site-packages (1.1.2)
Requirement already satisfied: python-dateutil>=2.8.1 in c:\users\malav\anaconda3\lib\site-packages (from pandas) (2.8.
2)
Requirement already satisfied: pytz>=2020.1 in c:\users\malav\anaconda3\lib\site-packages (from pandas) (2022.1)
Requirement already satisfied: numpy>=1.18.5 in c:\users\malav\anaconda3\lib\site-packages (from pandas) (1.21.5)
Requirement already satisfied: itsdangerous>=0.24 in c:\users\malav\anaconda3\lib\site-packages (from flask) (2.0.1)
Requirement already satisfied: Jinja2>=2.10.1 in c:\users\malav\anaconda3\lib\site-packages (from flask) (2.11.3)
Requirement already satisfied: click>=5.1 in c:\users\malav\anaconda3\lib\site-packages (from flask) (8.0.4)
Requirement already satisfied: Werkzeug>=0.15 in c:\users\malav\anaconda3\lib\site-packages (from flask) (2.0.3)
Requirement already satisfied: colorama in c:\users\malav\anaconda3\lib\site-packages (from click>=5.1->flask) (0.4.5)
Requirement already satisfied: MarkupSafe>=0.23 in c:\users\malav\anaconda3\lib\site-packages (from Jinja2>=2.10.1->fla
sk) (2.0.1)
Requirement already satisfied: six>=1.5 in c:\users\malav\anaconda3\lib\site-packages (from python-dateutil>=2.8.1->pan
das) (1.16.0)
```

In [52]:
```
import re

def simple_chatbot(user_query, df, summary):
    user_query = user_query.lower()

    # Define financial metrics and their corresponding column names
    financial_metrics = {
        "total revenue": "Total Revenue",
        "net income": "Net Income",
        "total assets": "Total Assets",
        "total liabilities": "Total Liabilities",
        "cash flow": "Cash Flow",
        "revenue growth": "Revenue Growth (%)",
        "income growth": "Income Growth (%)",
    }

    # Extract the year if mentioned in the query
    year_match = re.search(r'\b(20\d{2})\b', user_query)
    year = int(year_match.group(0)) if year_match else None

    # Check if the query is asking about a change in financial metrics
    if "changed over the last year" in user_query or "compare" in user_query:
```

```
        # Identify if a company is mentioned in the query
        matched_company = None
        for company in summary['Company'].values:
            if company.lower() in user_query:
                matched_company = company
                break

        # Process the query for the change in financial metrics
        for metric_key, column_name in financial_metrics.items():
            if metric_key in user_query:
                if matched_company:
                    # Get data for the company
                    company_data = df[df['Company'] == matched_company]
                    last_year_data = company_data[company_data['Year'] == company_data['Year'].max() - 1]
                    current_year_data = company_data[company_data['Year'] == company_data['Year'].max()]
                    if last_year_data.empty or current_year_data.empty:
                        return f"No data found for {metric_key} of {matched_company} for the last two years."

                    last_year_value = last_year_data[column_name].sum()
                    current_year_value = current_year_data[column_name].sum()
                    change = current_year_value - last_year_value
                    change_message = "increased" if change > 0 else "decreased"
                    return f"The {metric_key} for {matched_company} has {change_message} by {abs(change):,.2f} million

                else:
                    # Sum for all companies
                    total_last_year = df[df['Year'] == df['Year'].max() - 1][column_name].sum()
                    total_current_year = df[df['Year'] == df['Year'].max()][column_name].sum()
                    change = total_current_year - total_last_year
                    change_message = "increased" if change > 0 else "decreased"
                    return f"The {metric_key} has {change_message} by {abs(change):,.2f} million over the last year."

    # Rest of your code for other queries like highest/lowest, total revenue, etc.
    for metric_key, column_name in financial_metrics.items():
        if any(keyword in user_query for keyword in [f"highest {metric_key}", f"largest {metric_key}", f"most {metric_k
            find_max = True
        elif any(keyword in user_query for keyword in [f"lowest {metric_key}", f"smallest {metric_key}", f"least {metri
            find_max = False
        else:
            continue  # If neither, move to the next metric

        # Filter data for the requested year or use the latest available
        if year:
            year_data = df[df["Year"] == year]
            if year_data.empty:
```

```
                return f"No data found for {metric_key} in {year}."
        else:
            year_data = df[df["Year"] == df["Year"].max()]  # Latest available year

        # Find the company with the highest or lowest value for the metric
        top_company = year_data.loc[year_data[column_name].idxmax()] if find_max else year_data.loc[year_data[column_na
        comparison_word = "highest" if find_max else "lowest"

        return f"The company with the {comparison_word} {metric_key} in {year if year else top_company['Year']} is {top

    # Identify the company in the query
    matched_company = None
    for company in summary['Company'].values:
        if company.lower() in user_query:
            matched_company = company
            break

    if matched_company:
        # Get the most recent data for the matched company
        latest_data = df[df['Company'] == matched_company].sort_values('Year', ascending=False).iloc[0]

        # Check if the user is asking for a specific metric for a company
        for metric_key, column_name in financial_metrics.items():
            if metric_key in user_query:
                # Fetch data for the requested year or the latest available
                if year:
                    year_data = df[(df['Company'] == matched_company) & (df['Year'] == year)]
                    if year_data.empty:
                        return f"No data found for {metric_key} of {matched_company} in {year}."
                    value = year_data.iloc[0][column_name]
                else:
                    value = latest_data[column_name]

                return f"The {metric_key} for {matched_company} in {year} is {value:,.2f} million." if year else f"The
    else:
        # If no company is mentioned, calculate the sum of all companies for the requested metric
        for metric_key, column_name in financial_metrics.items():
            if metric_key in user_query:
                # Sum for all companies, optionally filtered by year
                if year:
                    year_data = df[df["Year"] == year]
                    total_value = year_data[column_name].sum()
                    return f"The total {metric_key} for all companies in {year} is {total_value:,.2f} million."
                else:
                    total_value = df[column_name].sum()
```

```
                    return f"The total {metric_key} for all companies is {total_value:,.2f} million."


# Get user input for the number of questions
n = int(input('Enter how many questions you want to ask: '))

for i in range(n):
    user_query = input(f"Ask financial question {i+1}: ")
    response = simple_chatbot(user_query, df, summary)
    print(response)
```

```
Enter how many questions you want to ask: 10
Ask financial question 1: what is the total net income?
The total net income for all companies is 558,483.00 million.
Ask financial question 2: what is the total net income in 2024?
The total net income for all companies in 2024 is 189,025.00 million.
Ask financial question 3: what is the total net income for microsoft in 2024?
The net income for Microsoft in 2024 is 88,136.00 million.
Ask financial question 4: what is the total net income for microsoft in 2023?
The net income for Microsoft in 2023 is 72,361.00 million.
Ask financial question 5: How has net income changed over the last year?
The net income has increased by 4,695.00 million over the last year.
Ask financial question 6: How has net income for microsoft changed over the last year?
The net income for Microsoft has increased by 15,775.00 million over the last year.
Ask financial question 7: total revenue?
The total total revenue for all companies is 2,099,880.00 million.
Ask financial question 8: which company has highest total revenue?
The company with the highest total revenue in 2024 is Apple with 391,035.00 million.
Ask financial question 9: which company has lowest total revenue?
The company with the lowest total revenue in 2024 is Tesla with 97,690.00 million.
Ask financial question 10: total revenue in 2024?
The total total revenue for all companies in 2024 is 733,847.00 million.
```