Creating a ratio analysis calculator in Python can be a great way to analyze financial data. Below is a simple implementation of a ratio analysis calculator that allows you to compute some of the most common financial ratios, such as profitability, liquidity, efficiency, and leverage ratios. This is a basic example, and you can expand it by adding more ratios as needed.

In [1]:
```python
class RatioAnalysisCalculator:
    def __init__(self):
        pass

    def current_ratio(self, current_assets, current_liabilities):
        """Calculate Current Ratio"""
        return current_assets / current_liabilities

    def quick_ratio(self, current_assets, inventory, current_liabilities):
        """Calculate Quick Ratio"""
        return (current_assets - inventory) / current_liabilities

    def debt_to_equity_ratio(self, total_liabilities, total_equity):
        """Calculate Debt to Equity Ratio"""
        return total_liabilities / total_equity

    def gross_margin_ratio(self, gross_profit, revenue):
        """Calculate Gross Margin Ratio"""
        return gross_profit / revenue

    def net_profit_margin(self, net_profit, revenue):
        """Calculate Net Profit Margin"""
        return net_profit / revenue

    def return_on_assets(self, net_income, total_assets):
        """Calculate Return on Assets"""
        return net_income / total_assets

    def return_on_equity(self, net_income, total_equity):
```

```
        """Calculate Return on Equity"""
        return net_income / total_equity

def main():
    calculator = RatioAnalysisCalculator()

    print("Ratio Analysis Calculator")
    print("-------------------------")

    # Example input for Current Ratio
    current_assets = float(input("Enter Current Assets: "))
    current_liabilities = float(input("Enter Current Liabilities: "))
    print("Current Ratio: {:.2f}".format(calculator.current_ratio(current_assets, current_liabilities)))

    # Example input for Quick Ratio
    inventory = float(input("Enter Inventory: "))
    print("Quick Ratio: {:.2f}".format(calculator.quick_ratio(current_assets, inventory, current_liabilities)))

    # Example input for Debt to Equity Ratio
    total_liabilities = float(input("Enter Total Liabilities: "))
    total_equity = float(input("Enter Total Equity: "))
    print("Debt to Equity Ratio: {:.2f}".format(calculator.debt_to_equity_ratio(total_liabilities, total_equity)))

    # Example input for Gross Margin Ratio
    gross_profit = float(input("Enter Gross Profit: "))
    revenue = float(input("Enter Revenue: "))
    print("Gross Margin Ratio: {:.2f}".format(calculator.gross_margin_ratio(gross_profit, revenue)))

    # Example input for Net Profit Margin
    net_profit = float(input("Enter Net Profit: "))
    print("Net Profit Margin: {:.2f}".format(calculator.net_profit_margin(net_profit, revenue)))

    # Example input for Return on Assets
    total_assets = float(input("Enter Total Assets: "))
    print("Return on Assets: {:.2f}".format(calculator.return_on_assets(net_profit, total_assets)))

    # Example input for Return on Equity
    print("Return on Equity: {:.2f}".format(calculator.return_on_equity(net_profit, total_equity)))

if __name__ == "__main__":
    main()
```

```
Ratio Analysis Calculator
-------------------------
Enter Current Assets: 1500000
Enter Current Liabilities: 600000
Current Ratio: 2.50
Enter Inventory: 3500000
Quick Ratio: -3.33
Enter Total Liabilities: 6500000
Enter Total Equity: 7500000
Debt to Equity Ratio: 0.87
Enter Gross Profit: 2500000
Enter Revenue: 3000000
Gross Margin Ratio: 0.83
Enter Net Profit: 2800000
Net Profit Margin: 0.93
Enter Total Assets: 75000000
Return on Assets: 0.04
Return on Equity: 0.37
```

Explanation of the Code: RatioAnalysisCalculator Class: This class contains methods to calculate different financial ratios. Main Function: The main() function orchestrates user input and outputs calculated ratios. User Input: The script prompts the user to enter values for various financial parameters based on the ratios being calculated. How to Run This Code: Copy the code into a Python script file (e.g., ratio_analysis_calculator.py). Run the script using a Python interpreter. Follow the prompts to input financial data, and the calculator will output the respective ratios. Customization: You can add more ratios by implementing additional methods in the RatioAnalysisCalculator class. You can also improve the user interface or implement error handling where necessary.

In [ ]: