***Financial Analytics***

**DOMAIN : Finance**

# *Introduction*

Analyzing competition helps businesses identify their strengths and weaknesses, uncover opportunities for growth, and devise strategic plans to enhance their market position. This report focuses on analyzing the competition among the top 500 companies in India based on their market capitalization and quarterly sales. The dataset provides comprehensive information on these companies, including their market capitalization in crores and their quarterly sales in crores.

# *Data Description*

df.describe()



# *Python coding on Financial Analytics*

**Import the packages** import numpy as np import pandas as pd import seaborn as sns import matplotlib.pyplot as plt **Set the path for better visibility** import os

for dirname, \_, filenames in os.walk(r"C:\Users\sabir\Downloads\Financial Analytics data.csv"):

for filename in filenames: print(os.path.join(dirname, filename))

## Setting the style for Seaborn plots

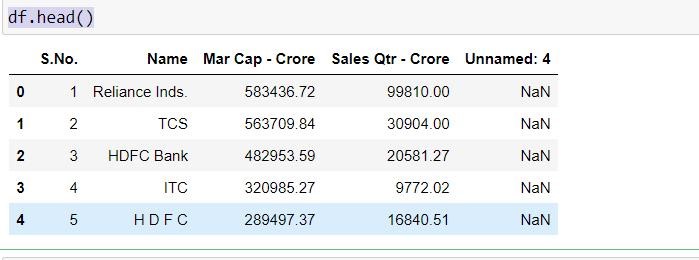
sns.set\_style('darkgrid')

***Reading the file***

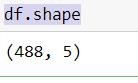
df = pd.read\_csv(r"C:\Users\sabir\Downloads\Financial Analytics data.csv")

## Reading the top 5 files

df.head()

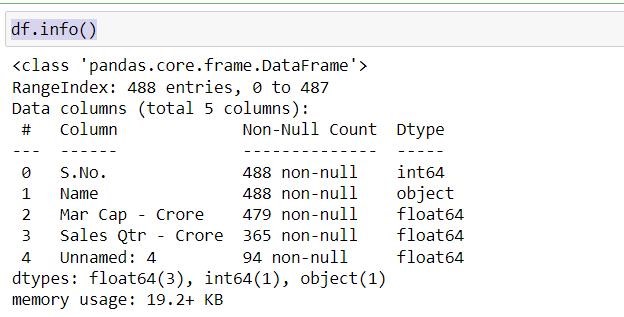


***Shape of the file*** df.shape



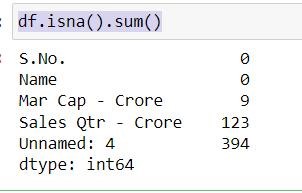
## *Information of the data set*

df.info()



## *Finding out the missing values*

df.isna().sum()

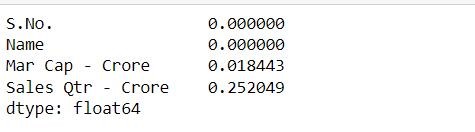


***Drop the unwanted column***

df=df.drop(['Unnamed: 4'],axis = 1)

## *Finding the proportion of missing values*

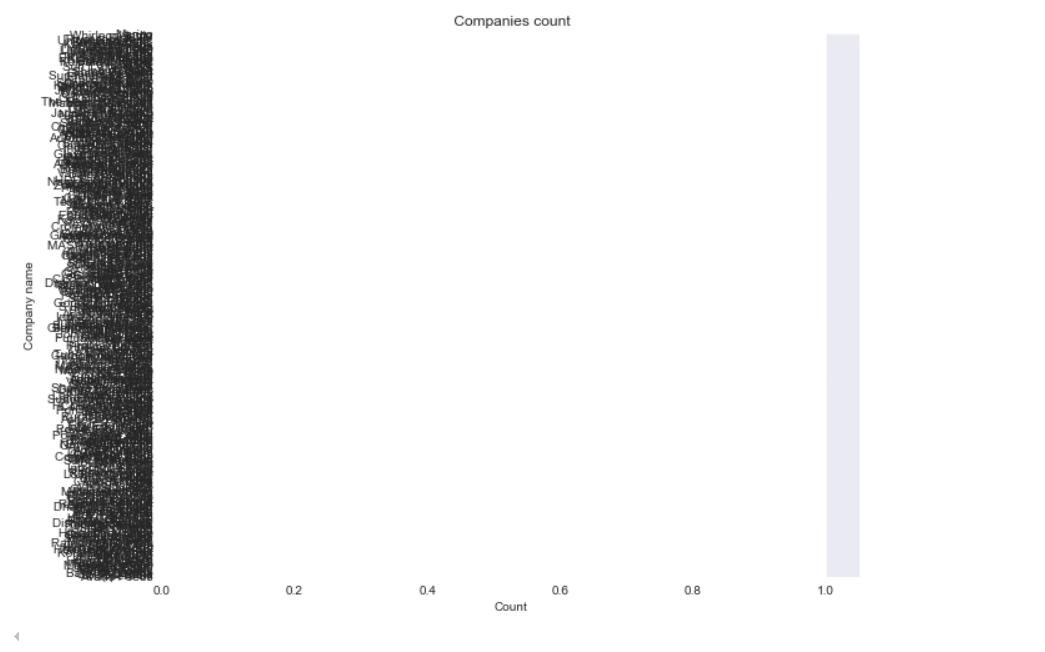
df.isna().mean()



## *Replacing the missing values with nan*

df=df.replace(r'^\s\*$',np.nan,regex = True) **Set the plot size for better visibility** plt.figure(figsize=(10, 8))

***Create a countplot for the 'Name' column*** sns.countplot(y=data['Name'], order=data['Name'].value\_counts().index) plt.title(' Companies count') plt.xlabel('Count') plt.ylabel('Company name') plt.show()



**Select the top 50 companies**

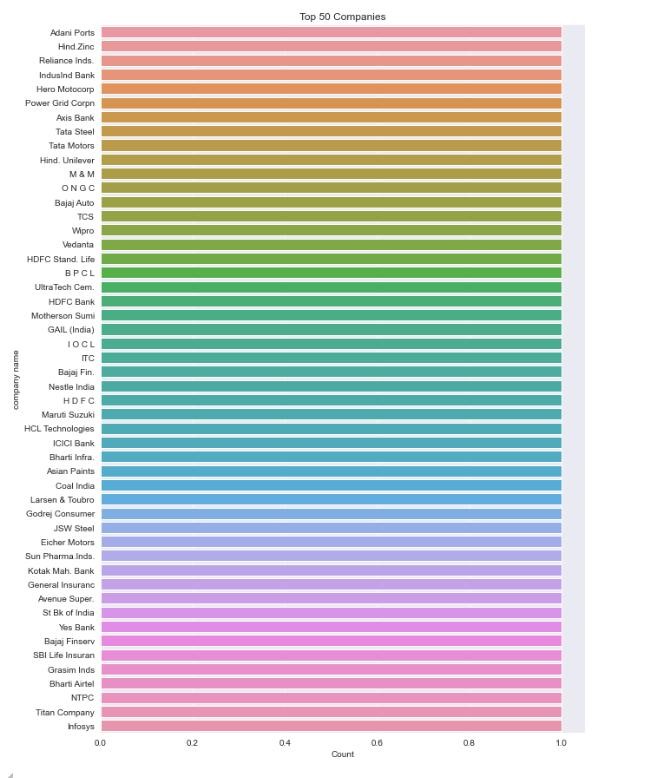
top\_50\_companies = data.head(50)

***Set the plot size for better visibility*** plt.figure(figsize=(10, 15))

## *Create a countplot for the 'Name' column of the top 50 companies*

sns.countplot(y=top\_50\_companies['Name'], order=top\_50\_companies['Name'].value\_counts().index)

plt.title('Top 50 Companies') plt.xlabel('Count') plt.ylabel('company name') plt.show()



***Calculate the difference between Mar Cap and Sales Qtr*** df['Difference'] = df['Mar Cap - Crore'] - df['Sales Qtr - Crore'] plt.figure(figsize=(18, 6))

***Bar plot for market capitalization, sales, and their difference*** width = 0.3

## *Bar plot for Market Cap*

plt.bar(df['Name'], df['Mar Cap - Crore'], width=width, label='Market Cap', align='center', color='r', edgecolor='black')

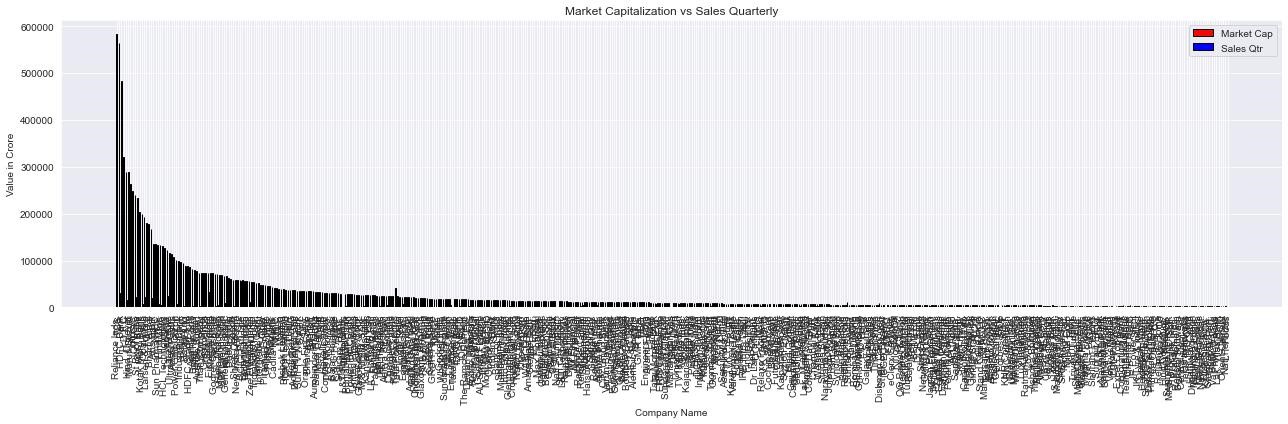
## *Bar plot for Sales Qtr*

plt.bar(df['Name'], df['Sales Qtr - Crore'], width=width, label='Sales Qtr', align='edge', color='b', edgecolor='black')

## *Rotate x-axis labels for better readability*

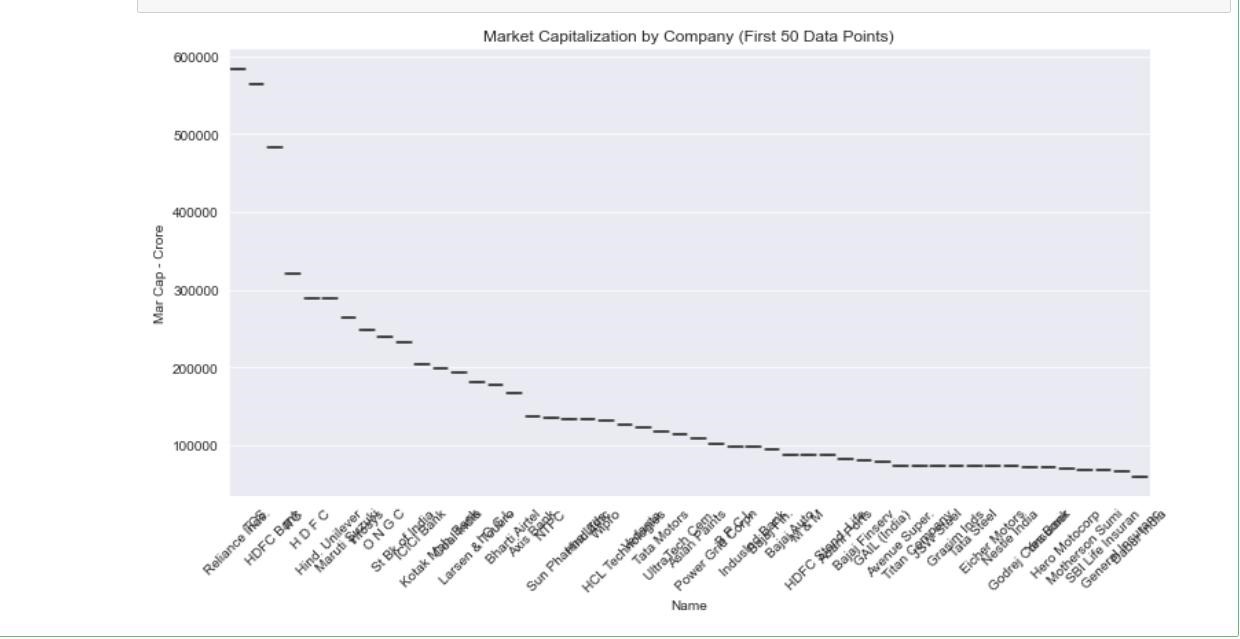
plt.xticks(rotation=90) plt.xlabel('Company Name') plt.ylabel('Value in Crore')

plt.title('Market Capitalization vs Sales Quarterly') plt.legend() plt.show()

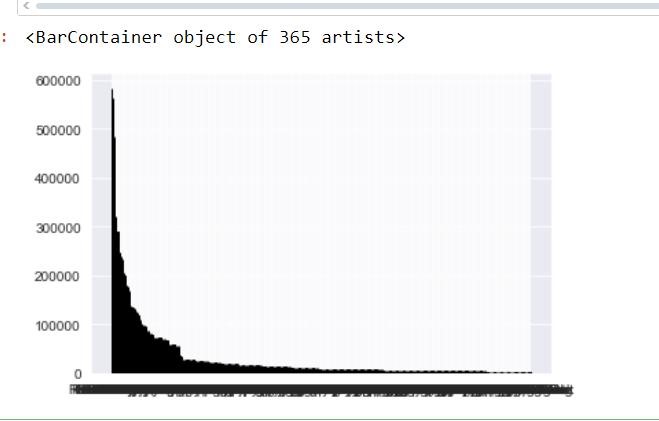


## *Create the box plot*

plt.figure(figsize=(12, 6)) sns.boxplot(x='Name', y='Mar Cap - Crore', data=df\_first\_50) plt.xticks(rotation=45) plt.title('Market Capitalization by Company (First 50 Data Points)') plt.show()



plt.bar(df['Name'], df['Mar Cap - Crore'], width=width, label='Market Cap', align='center', color='r', edgecolor='black')



***Sort the DataFrame by 'Mar Cap - Crore' in descending order and take the top 50 companies***

top\_50\_df = df.nlargest(50, 'Mar Cap - Crore')

plt.figure(figsize=(14, 6))

## *# Subplot 1: Bar Plot for Market Cap*

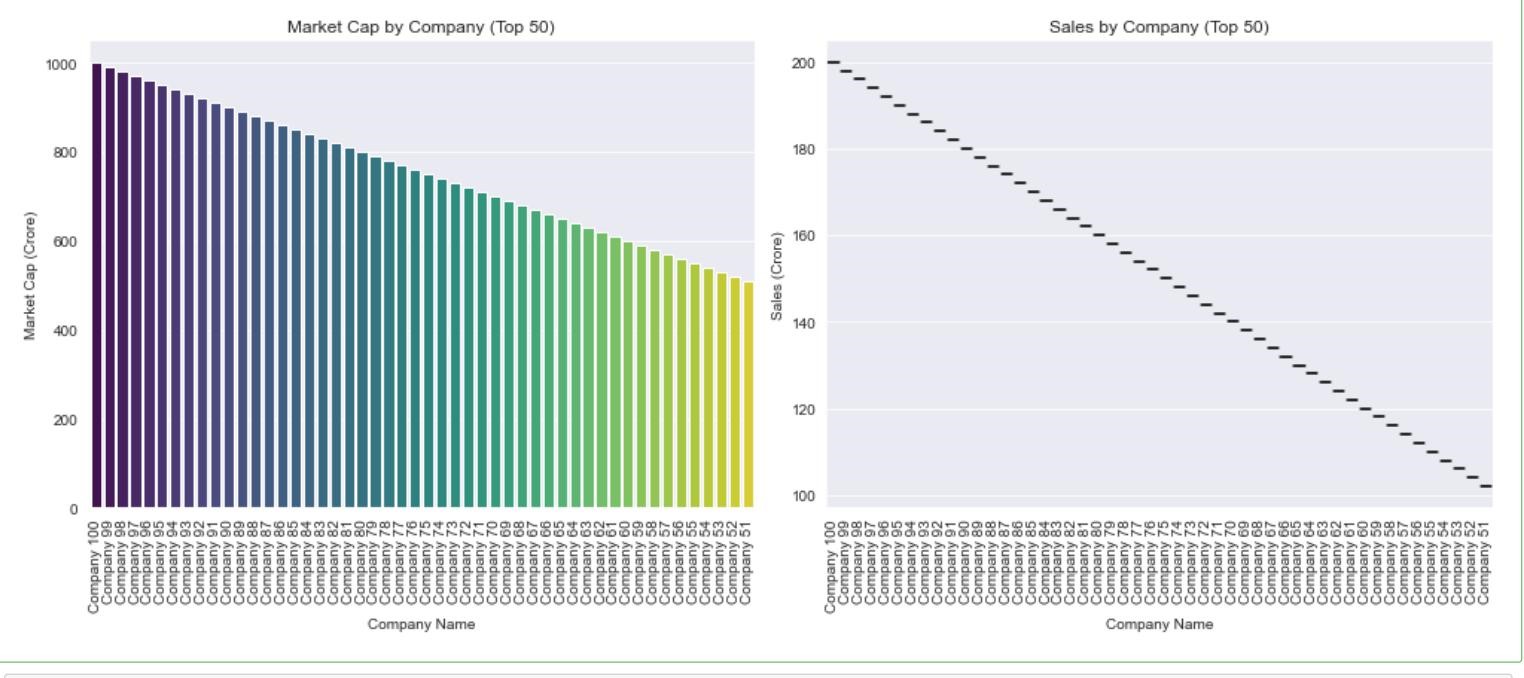
plt.subplot(1, 2, 1) sns.barplot(x=top\_50\_df['Name'], y=top\_50\_df['Mar Cap - Crore'], palette='viridis') plt.title('Market Cap by Company (Top 50)') plt.xlabel('Company Name') plt.ylabel('Market Cap (Crore)') plt.xticks(rotation=90) # Rotate x-axis labels for better readability

## *Subplot 2: Box Plot for Sales*

plt.subplot(1, 2, 2) sns.boxplot(x='Name', y='Sales Qtr - Crore', data=top\_50\_df, palette='plasma') plt.title('Sales by Company (Top 50)') plt.xlabel('Company Name')

plt.ylabel('Sales (Crore)') plt.xticks(rotation=90) # Rotate x-axis labels for better readability

plt.tight\_layout() plt.show()



## *Count of Market Capitalization Categories (assuming Mar Cap - Crore is binned)*

plt.subplot(1, 2, 2) sns.countplot(y=pd.cut(df['Mar Cap - Crore'], bins=10), palette='plasma') plt.title('Count of Market Capitalization Categories') plt.tight\_layout() plt.show()



## *Example data (replace this with your actual data)* data = {

'Name': [f'Company {i}' for i in range(1, 101)],

'Mar Cap - Crore': [i \* 10 for i in range(1, 101)],

'Sales Qtr - Crore': [i \* 2 for i in range(1, 101)]

}

df = pd.DataFrame(data)

# Sort the DataFrame by 'Mar Cap - Crore' in descending order and take the top 50 companies top\_50\_df = df.nlargest(50, 'Mar Cap - Crore')

plt.figure(figsize=(18, 6))

## *# Subplot 1: Average Market Capitalization by Company*

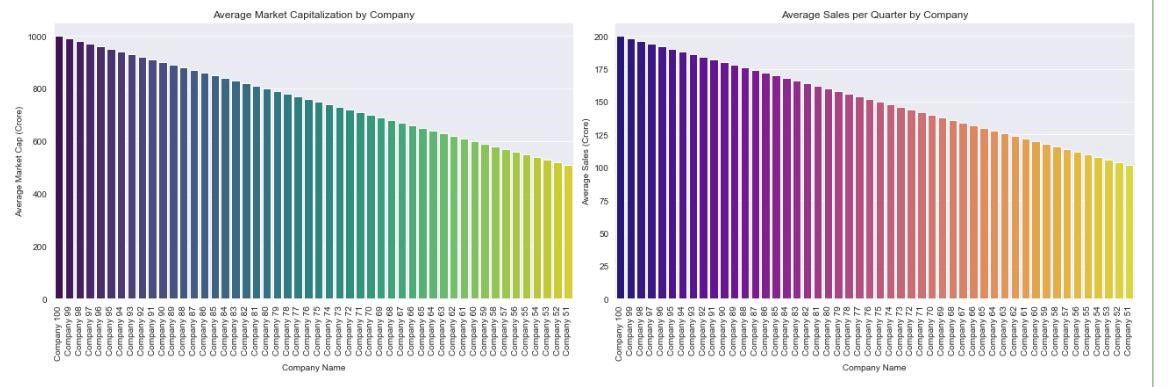
plt.subplot(1, 2, 1)

sns.barplot(x='Name', y='Mar Cap - Crore', data=top\_50\_df, estimator=np.mean, palette='viridis', ci=None) plt.xticks(rotation=90) # Rotate x-axis labels for better readability plt.title('Average Market Capitalization by Company') plt.xlabel('Company Name') plt.ylabel('Average Market Cap (Crore)')

## *# Subplot 2: Average Sales per Quarter by Company*

plt.subplot(1, 2, 2)

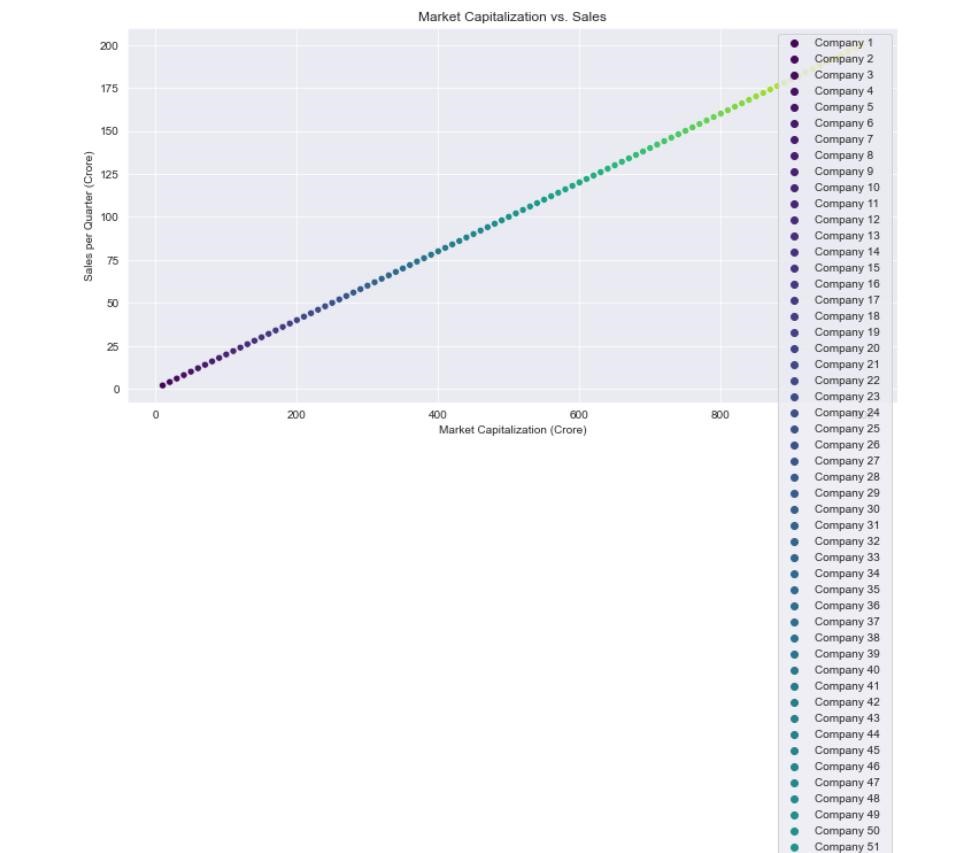
sns.barplot(x='Name', y='Sales Qtr - Crore', data=top\_50\_df, estimator=np.mean, palette='plasma', ci=None) plt.xticks(rotation=90) # Rotate x-axis labels for better readability plt.title('Average Sales per Quarter by Company') plt.xlabel('Company Name') plt.ylabel('Average Sales (Crore)') plt.tight\_layout() plt.show()



## *Scatter Plot*

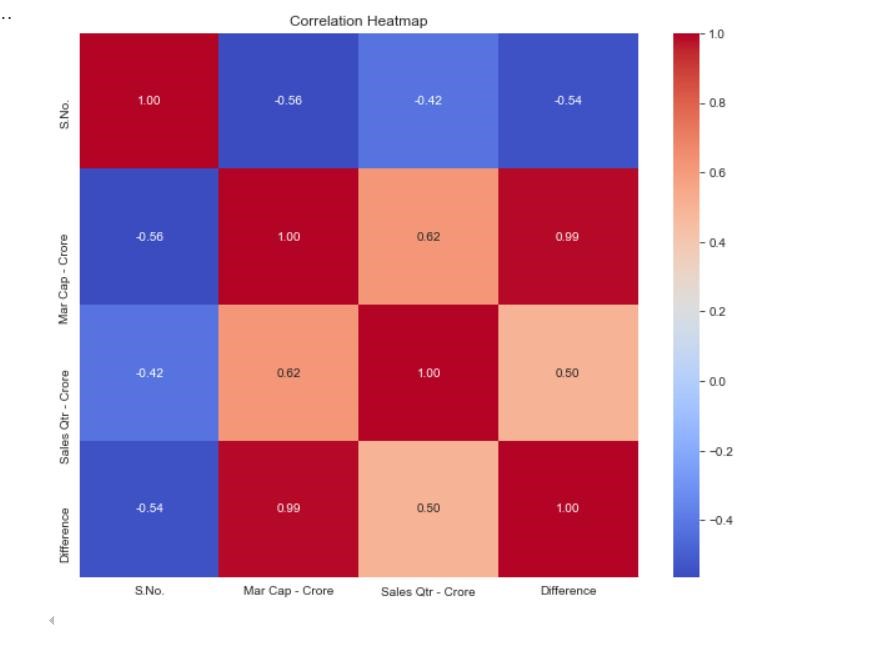
plt.figure(figsize=(12, 6))

sns.scatterplot(x=df['Mar Cap - Crore'], y=df['Sales Qtr - Crore'], hue=df['Name'], palette='viridis') plt.title('Market Capitalization vs. Sales') plt.xlabel('Market Capitalization (Crore)') plt.ylabel('Sales per Quarter (Crore)') plt.legend(loc='upper right') plt.show()



## Heatmap

plt.figure(figsize=(10, 8)) correlation\_matrix = df.corr() sns.heatmap(correlation\_matrix, annot=True, cmap='coolwarm', fmt='.2f') plt.title('Correlation Heatmap') plt.show()



# *Conclusion*

This analysis provides a comprehensive overview of the financial metrics of various companies. Key insights include the distribution and comparison of market capitalization and quarterly sales, as well as the correlation between these metrics. Visualizations like count plots, box plots, bar plots, scatter plots, and heatmaps help in understanding the financial standing and performance of the companies in the dataset.