Financial Analytics Project 1

Data Loading

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
df = pd.read_csv(r"/content/Financial Analytics data.csv")
```

Data Exploration and Understanding

```
df.head()
```

	S.No.	Name	Mar Cap - Crore	Sales Qtr - Crore	Unnamed: 4
0	1	Reliance Inds.	583436.72	99810.00	NaN
1	2	TCS	563709.84	30904.00	NaN
2	3	HDFC Bank	482953.59	20581.27	NaN
3	4	ITC	320985.27	9772.02	NaN
4	5	H D F C	289497.37	16840.51	NaN

Next steps: Generate code with df View recommended plots

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 488 entries, 0 to 487
Data columns (total 5 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   S.No.              488 non-null    int64
 1   Name               488 non-null    object
 2   Mar Cap - Crore    479 non-null    float64
 3   Sales Qtr - Crore  365 non-null    float64
 4   Unnamed: 4         94 non-null     float64
dtypes: float64(3), int64(1), object(1)
memory usage: 19.2+ KB
```

```
df.describe()
```

	S.No.	Mar Cap - Crore	Sales Qtr - Crore	Unnamed: 4
count	488.000000	479.000000	365.000000	94.000000
mean	251.508197	28043.857119	4395.976849	1523.870106
std	145.884078	59464.615831	11092.206185	1800.008836
min	1.000000	3017.070000	47.240000	0.000000
25%	122.750000	4843.575000	593.740000	407.167500
50%	252.500000	9885.050000	1278.300000	702.325000
75%	378.250000	23549.900000	2840.750000	2234.815000
max	500.000000	583436.720000	110666.930000	7757.060000

Data Cleaning and Preprocessing

```
nan_sales = df['Sales Qtr - Crore'].isna()
nan_sales
```

```
0      False
1      False
2      False
3      False
```

```
4      False
       ...
483    False
484    False
485    False
486    False
487     True
Name: Sales Qtr - Crore, Length: 488, dtype: bool
```

```
df.fillna(0, inplace = True)
```

```
df['Sales Qtr (in Crore)'] = df['Sales Qtr - Crore'] + df['Unnamed: 4']
df.head()
```

	S.No.	Name	Mar Cap - Crore	Sales Qtr - Crore	Unnamed: 4	Sales Qtr (in Crore)
0	1	Reliance Inds.	583436.72	99810.00	0.0	99810.00
1	2	TCS	563709.84	30904.00	0.0	30904.00
2	3	HDFC Bank	482953.59	20581.27	0.0	20581.27
3	4	ITC	320985.27	9772.02	0.0	9772.02

Next steps: Generate code with df View recommended plots

Dropping the redundant values

```
df.drop('Sales Qtr - Crore', axis = 1, inplace = True)
```

```
df.drop('Unnamed: 4', axis = 1, inplace = True)
```

```
df.shape
```

```
(488, 4)
```

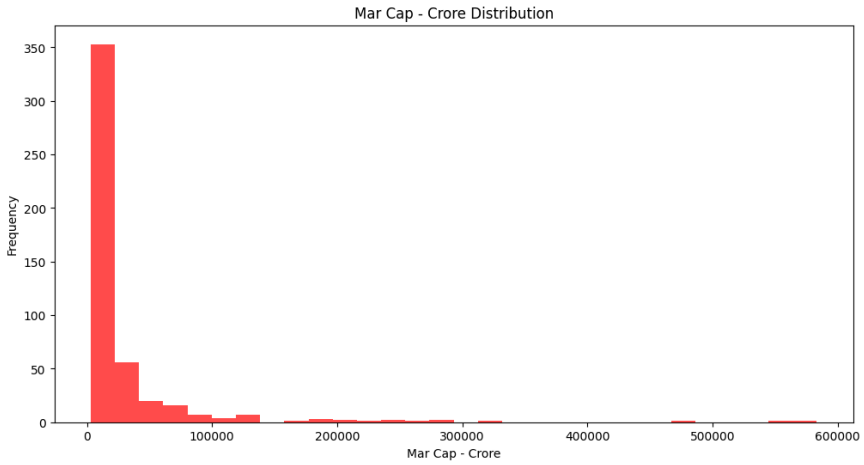
```
df = df.drop(df[(df['Mar Cap - Crore'] == 0) & (df['Sales Qtr (in Crore)'] == 0)].index)
df
```

	S.No.	Name	Mar Cap - Crore	Sales Qtr (in Crore)
0	1	Reliance Inds.	583436.72	99810.00
1	2	TCS	563709.84	30904.00
2	3	HDFC Bank	482953.59	20581.27
3	4	ITC	320985.27	9772.02
4	5	H D F C	289497.37	16840.51
...	...	...	...	...
482	495	Prime Focus	3031.50	609.61
483	496	Lak. Vilas Bank	3029.57	790.17
484	497	NOCIL	3026.26	249.27
485	498	Orient Cement	3024.32	511.53
486	499	Natl.Fertilizer	3017.07	2840.75

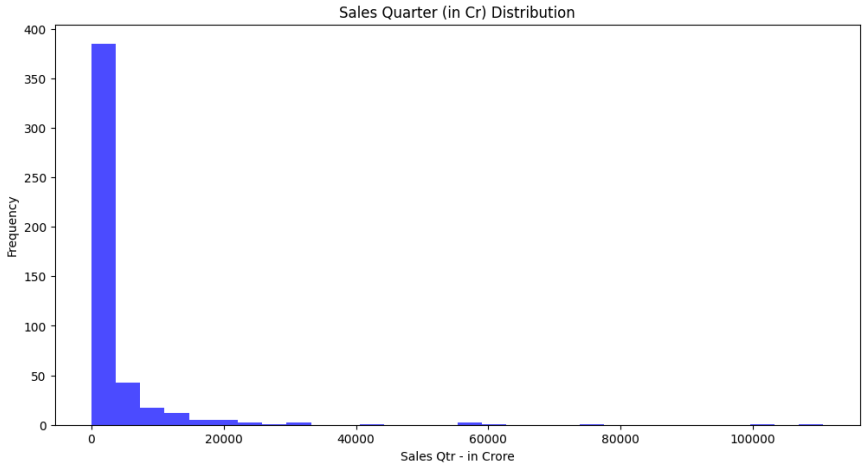
479 rows × 4 columns

Next steps: Generate code with df View recommended plots

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

plt.hist(df['Mar Cap - Crore'].dropna(), bins = 30, color = 'red', alpha = 0.7)
plt.title("Mar Cap - Crore Distribution")
plt.xlabel('Mar Cap - Crore')
plt.ylabel('Frequency')
plt.show()
```




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

plt.hist(df['Sales Qtr (in Crore)'].dropna(), bins = 30, color = 'Blue', alpha = 0.7)
plt.title("Sales Quarter (in Cr) Distribution")
plt.xlabel('Sales Qtr - in Crore')
plt.ylabel('Frequency')
plt.show()
```

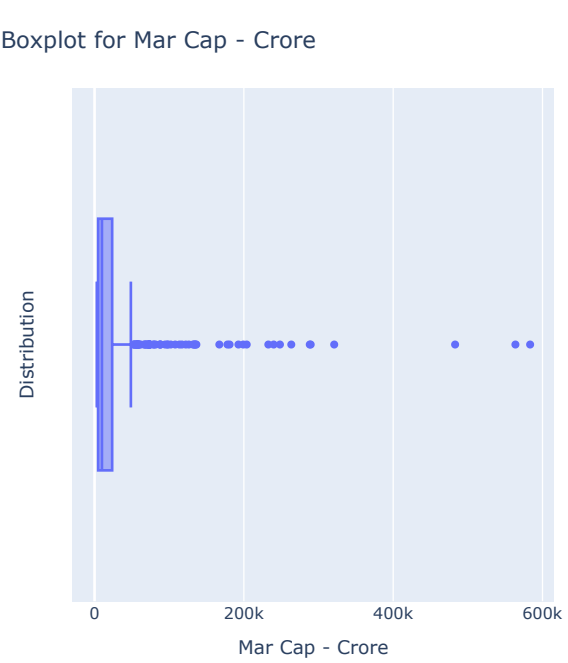



```
zeros = (df==0).sum()
zeros
```

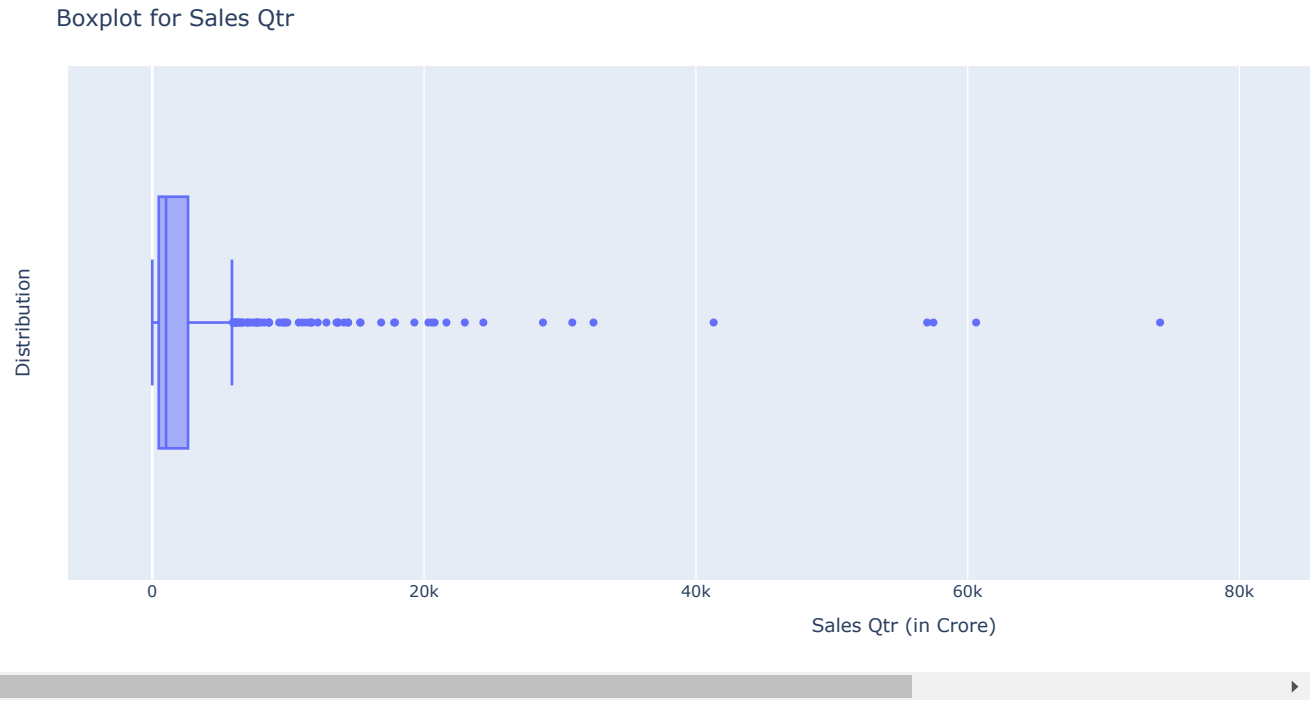
```
S.No.                   0
Name                    0
Mar Cap - Crore         0
Sales Qtr (in Crore)    21
dtype: int64
```

```
import plotly.express as px


fig = px.box(df, x='Mar Cap - Crore', orientation='h')
```

```
fig.update_layout(title=f'Boxplot for Mar Cap - Crore', xaxis_title='Mar Cap - Crore', yaxis_title='Distribution')
fig.show()
```




```
fig1 = px.box(df, x = 'Sales Qtr (in Crore)', orientation = 'h')
fig1.update_layout(title = 'Boxplot for Sales Qtr', xaxis_title = 'Sales Qtr (in Crore)', yaxis_title = 'Distribution')
fig1.show()
```




```
df_sale_0 = df[(df['Mar Cap - Crore'] != 0) & (df['Sales Qtr (in Crore)'] == 0)]
df_sale_0
```

	S.No.	Name	Mar Cap - Crore	Sales Qtr (in Crore)
49	50	Bharti Infra.	61776.92	0.0
171	176	Info Edg.(India)	14845.05	0.0
185	192	Max Financial	13401.76	0.0
224	231	Bombay Burmah	10864.53	0.0
241	248	Sundaram Clayton	10074.36	0.0
258	271	Mahindra CIE	8587.04	0.0
314	327	Prism Cement	6176.23	0.0
332	345	GE Power	5497.40	0.0
338	351	MMTC	5300.00	0.0
370	383	Swan Energy	4721.49	0.0
374	387	Shoppers St.	4558.06	0.0
379	392	Stand.Chart.PLC	4487.31	0.0
393	406	Ujjivan Fin.Ser.	4293.42	0.0
396	409	Jindal Saw	4278.31	0.0
398	411	Linde India	4198.33	0.0
409	422	JP Associates	4074.37	0.0
418	431	HMT	3973.50	0.0
424	437	Gayatri Projects	3835.73	0.0
446	459	JP Power Ven.	3597.60	0.0
451	464	Amber Enterp.	3529.87	0.0
459	472	Hind.Construct.	3452.57	0.0

Next steps: Generate code with df_sale_0 View recommended plots

Feature Engineering

```
def market_cap_to_sales_ratio(row):
    if row['Sales Qtr (in Crore)'] != 0:
        return row['Mar Cap - Crore'] / row['Sales Qtr (in Crore)']
    else:
        return 0

df['Market Cap-to-Sales Ratio'] = df.apply(market_cap_to_sales_ratio, axis=1)
df
```

	S.No.	Name	Mar Cap - Crore	Sales Qtr (in Crore)	Market Cap-to-Sales Ratio
0	1	Reliance Inds.	583436.72	99810.00	5.845474
1	2	TCS	563709.84	30904.00	18.240676
2	3	HDFC Bank	482953.59	20581.27	23.465685
3	4	ITC	320985.27	9772.02	32.847382
4	5	H D F C	289497.37	16840.51	17.190535
...	...	...	...	...	...
482	495	Prime Focus	3031.50	609.61	4.972851
483	496	Lak. Vilas Bank	3029.57	790.17	3.834074
484	497	NOCIL	3026.26	249.27	12.140490
485	498	Orient Cement	3024.32	511.53	5.912302
486	499	Natl.Fertilizer	3017.07	2840.75	1.062068

479 rows × 5 columns

Next steps: Generate code with df View recommended plots

```
df_d = df[(df['Sales Qtr (in Crore)'] == 0)]
df_d
```

	S.No.	Name	Mar Cap - Crore	Sales Qtr (in Crore)	Market Cap-to-Sales Ratio
49	50	Bharti Infra.	61776.92	0.0	0.0
171	176	Info Edg.(India)	14845.05	0.0	0.0
185	192	Max Financial	13401.76	0.0	0.0
224	231	Bombay Burmah	10864.53	0.0	0.0
241	248	Sundaram Clayton	10074.36	0.0	0.0
258	271	Mahindra CIE	8587.04	0.0	0.0
314	327	Prism Cement	6176.23	0.0	0.0
332	345	GE Power	5497.40	0.0	0.0
338	351	MMTC	5300.00	0.0	0.0
370	383	Swan Energy	4721.49	0.0	0.0
374	387	Shoppers St.	4558.06	0.0	0.0
379	392	Stand.Chart.PLC	4487.31	0.0	0.0
393	406	Ujjivan Fin.Ser.	4293.42	0.0	0.0
396	409	Jindal Saw	4278.31	0.0	0.0
398	411	Linde India	4198.33	0.0	0.0
409	422	JP Associates	4074.37	0.0	0.0
418	431	HMT	3973.50	0.0	0.0
424	437	Gayatri Projects	3835.73	0.0	0.0
446	459	JP Power Ven.	3597.60	0.0	0.0
451	464	Amber Enterp.	3529.87	0.0	0.0
459	472	Hind.Construct.	3452.57	0.0	0.0

Next steps: Generate code with df_d View recommended plots

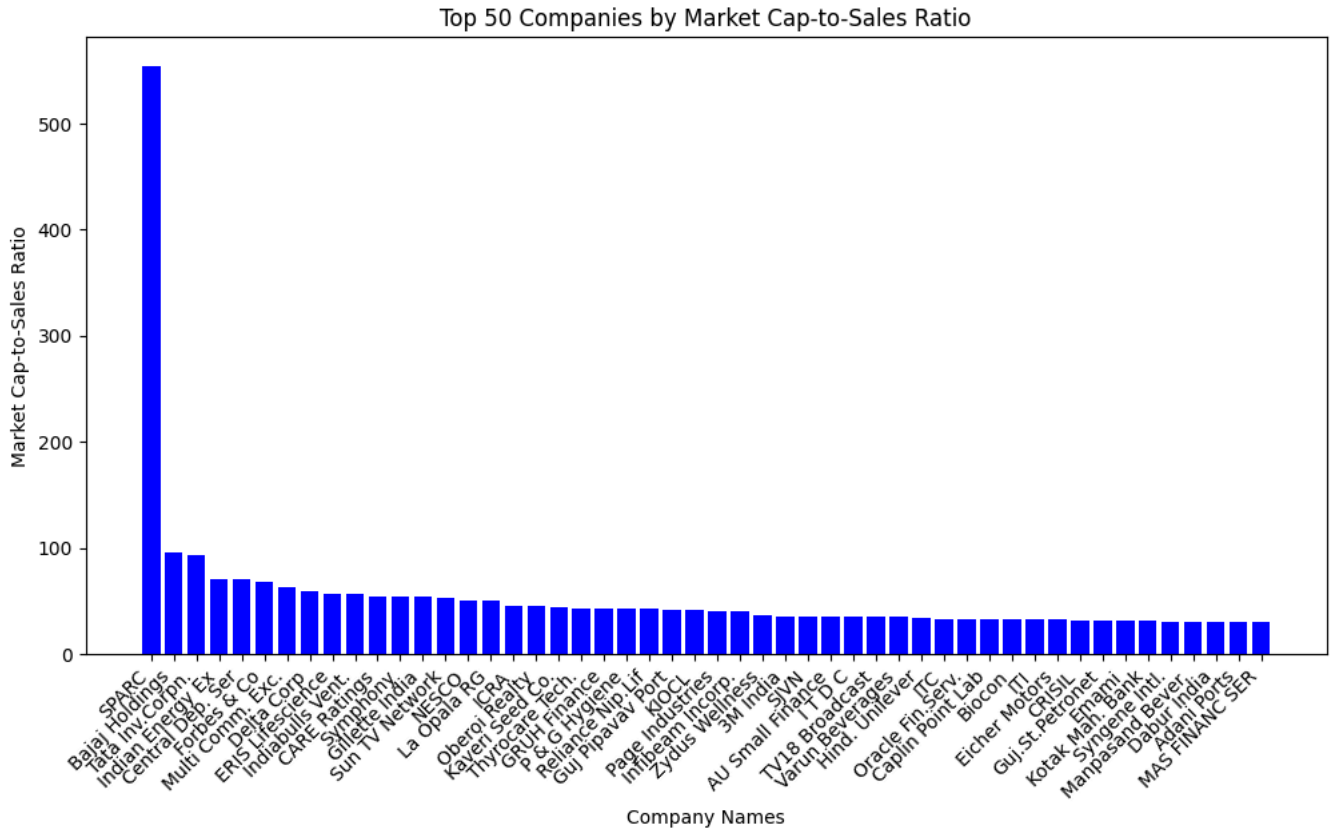
```
df.sort_values('Market Cap-to-Sales Ratio', ascending = False)
df
```

	S.No.	Name	Mar Cap - Crore	Sales Qtr (in Crore)	Market Cap-to-Sales Ratio
0	1	Reliance Inds.	583436.72	99810.00	5.845474
1	2	TCS	563709.84	30904.00	18.240676
2	3	HDFC Bank	482953.59	20581.27	23.465685
3	4	ITC	320985.27	9772.02	32.847382
4	5	H D F C	289497.37	16840.51	17.190535
...	...	...	...	...	...
482	495	Prime Focus	3031.50	609.61	4.972851
483	496	Lak. Vilas Bank	3029.57	790.17	3.834074
484	497	NOCIL	3026.26	249.27	12.140490
485	498	Orient Cement	3024.32	511.53	5.912302
486	499	Natl.Fertilizer	3017.07	2840.75	1.062068

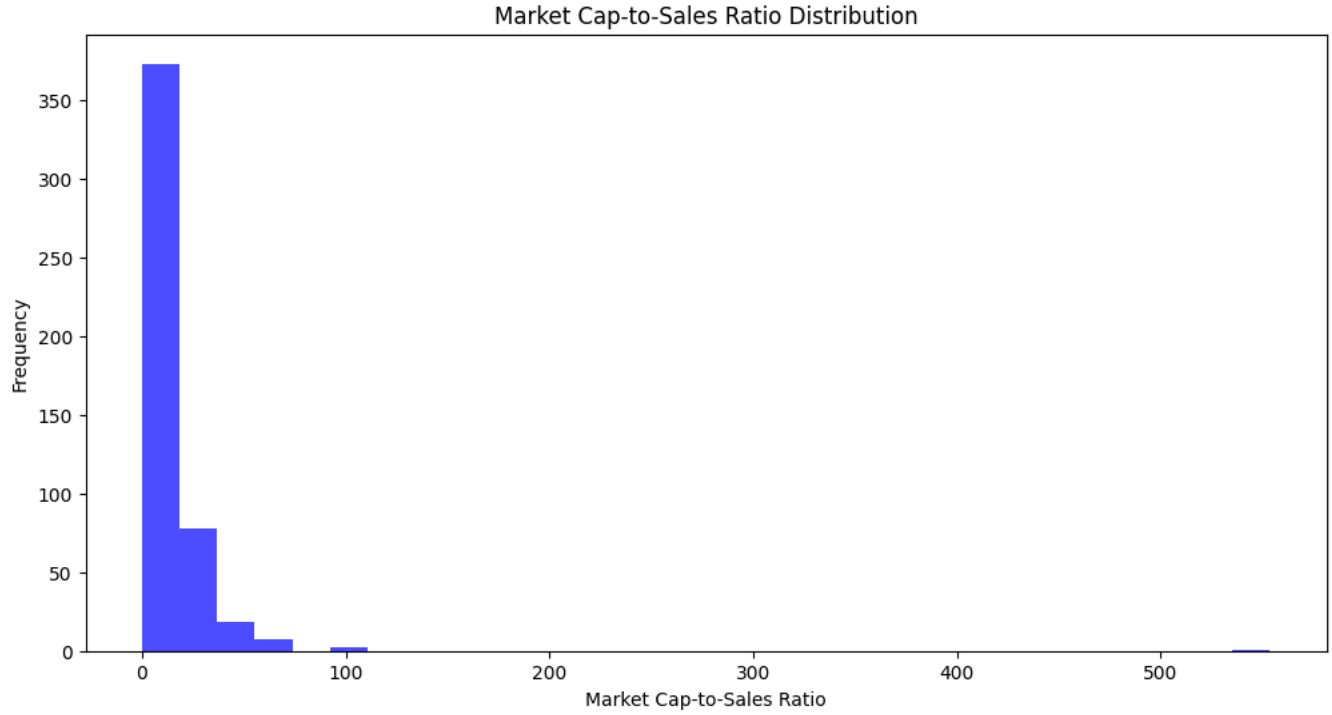
479 rows × 5 columns

Next steps: Generate code with df View recommended plots

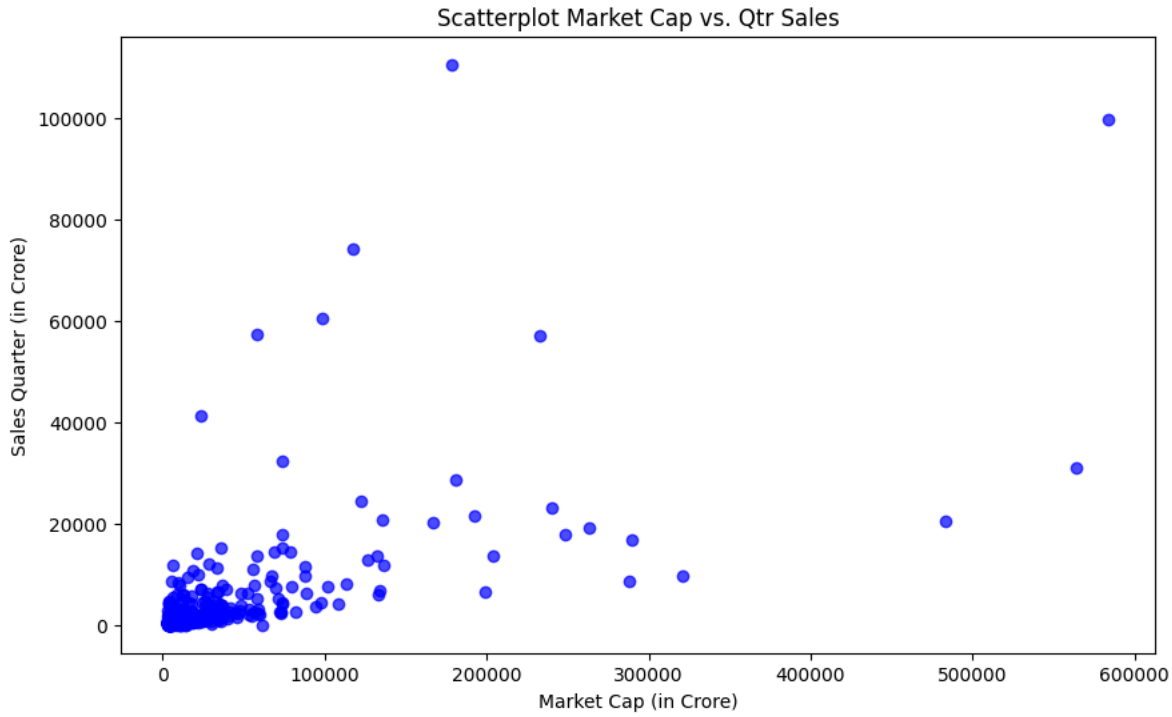
```
top_companies = df.nlargest(50, 'Market Cap-to-Sales Ratio')

plt.figure(figsize=(12,6))
plt.bar(top_companies['Name'], top_companies['Market Cap-to-Sales Ratio'], color='blue')
plt.title('Top 50 Companies by Market Cap-to-Sales Ratio')
plt.xlabel('Company Names')
plt.ylabel('Market Cap-to-Sales Ratio')
plt.xticks(rotation=45, ha='right')
plt.show()
```



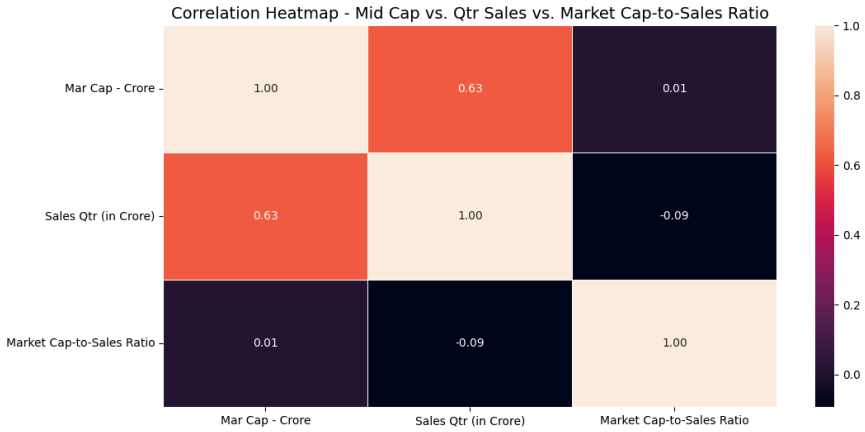

```
plt.figure(figsize=(12,6))

plt.hist(df['Market Cap-to-Sales Ratio'].dropna(), bins = 30, color = 'Blue', alpha = 0.7)
plt.title("Market Cap-to-Sales Ratio Distribution")
plt.xlabel('Market Cap-to-Sales Ratio')
plt.ylabel('Frequency')
plt.show()
```



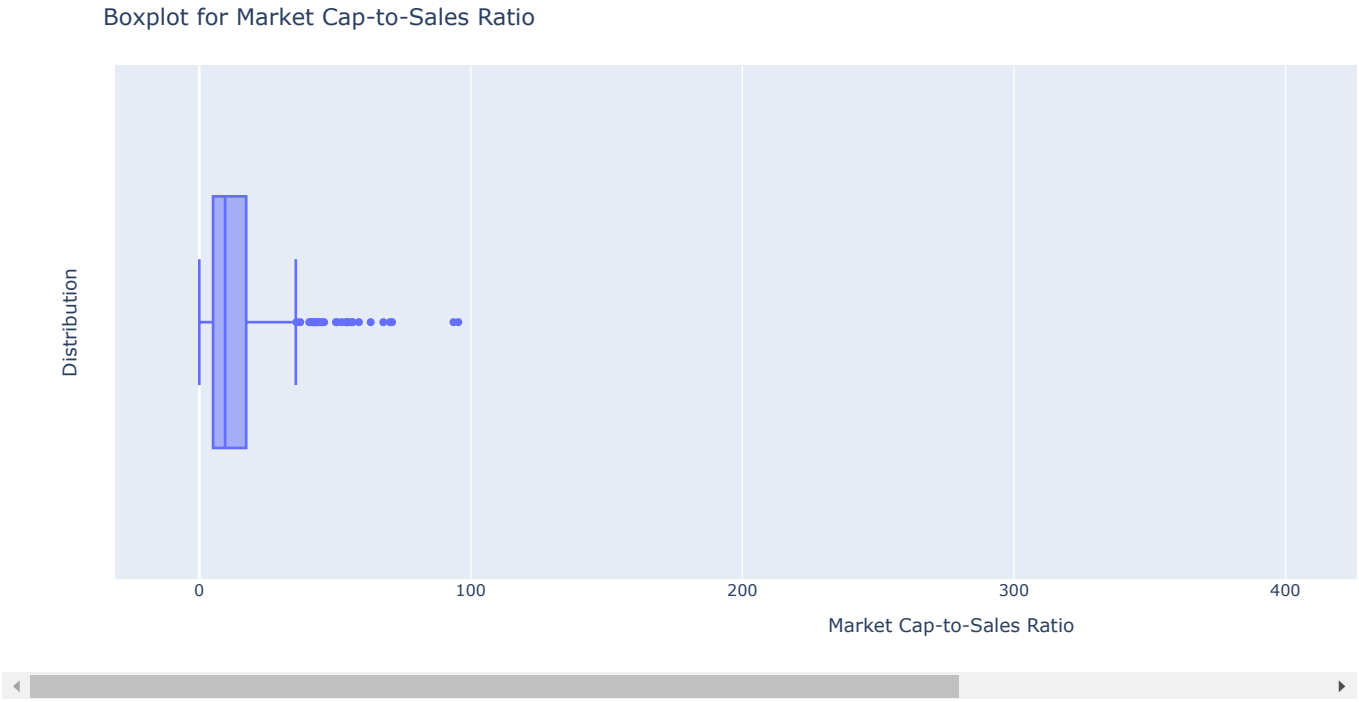

```
plt.figure(figsize=(10, 6))
plt.scatter(df['Mar Cap - Crore'], df['Sales Qtr (in Crore)'], color='blue', alpha=0.7)

# Set plot title and labels
plt.title('Scatterplot Market Cap vs. Qtr Sales')
plt.xlabel('Market Cap (in Crore)')
plt.ylabel('Sales Quarter (in Crore)')
plt.show()
```




```
correlation_data = df[['Mar Cap - Crore', 'Sales Qtr (in Crore)', 'Market Cap-to-Sales Ratio']]

correlation_matrix = correlation_data.corr()

plt.figure(figsize=(12,6))
sns.heatmap(correlation_matrix, annot=True, fmt='.2f', linewidths=.5)

plt.title('Correlation Heatmap - Mid Cap vs. Qtr Sales vs. Market Cap-to-Sales Ratio',fontsize = 14)
plt.xticks(rotation=0)
plt.yticks(rotation=0)

plt.show()
```




```
fig2 = px.box(df, x = 'Market Cap-to-Sales Ratio', orientation = 'h')
fig2.update_layout(title = 'Boxplot for Market Cap-to-Sales Ratio', xaxis_title = 'Market Cap-to-Sales Ratio', yaxis_title = 'Distribut
fig2.show()
```



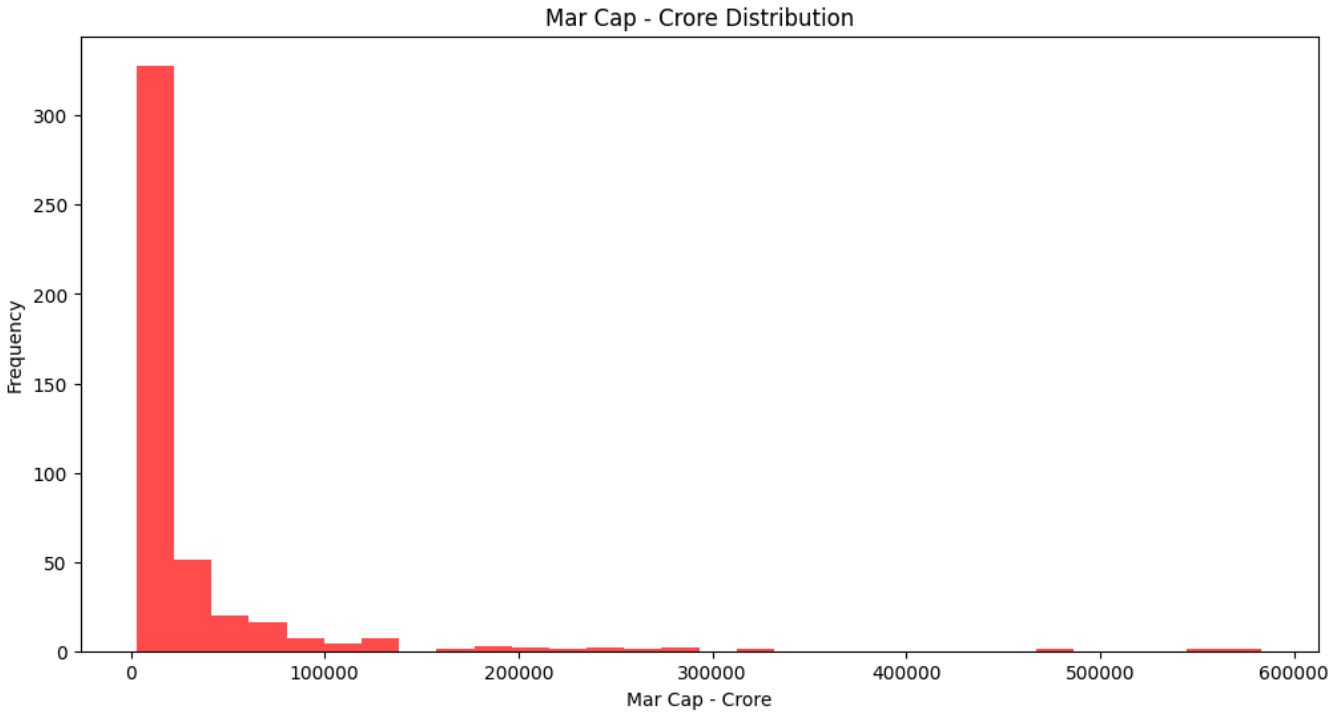

```
Q1 = df['Market Cap-to-Sales Ratio'].quantile(0.25)
Q3 = df['Market Cap-to-Sales Ratio'].quantile(0.75)

IQR = Q3 - Q1

lowerbound = Q1 - 1.5 * IQR
higherbound = Q3 + 1.5 * IQR

df_cleaned = df[(df['Market Cap-to-Sales Ratio'] >= lowerbound) & (df['Market Cap-to-Sales Ratio'] <= higherbound)]
df_cleaned
```

	S.No.	Name	Mar Cap - Crore	Sales Qtr (in Crore)	Market Cap-to-Sales Ratio
0	1	Reliance Inds.	583436.72	99810.00	5.845474
1	2	TCS	563709.84	30904.00	18.240676
2	3	HDFC Bank	482953.59	20581.27	23.465685
3	4	ITC	320985.27	9772.02	32.847382
4	5	H D F C	289497.37	16840.51	17.190535
...	...	...	...	...	...
482	495	Prime Focus	3031.50	609.61	4.972851
483	496	Lak. Vilas Bank	3029.57	790.17	3.834074
484	497	NOCIL	3026.26	249.27	12.140490
485	498	Orient Cement	3024.32	511.53	5.912302
486	499	Natl.Fertilizer	3017.07	2840.75	1.062068

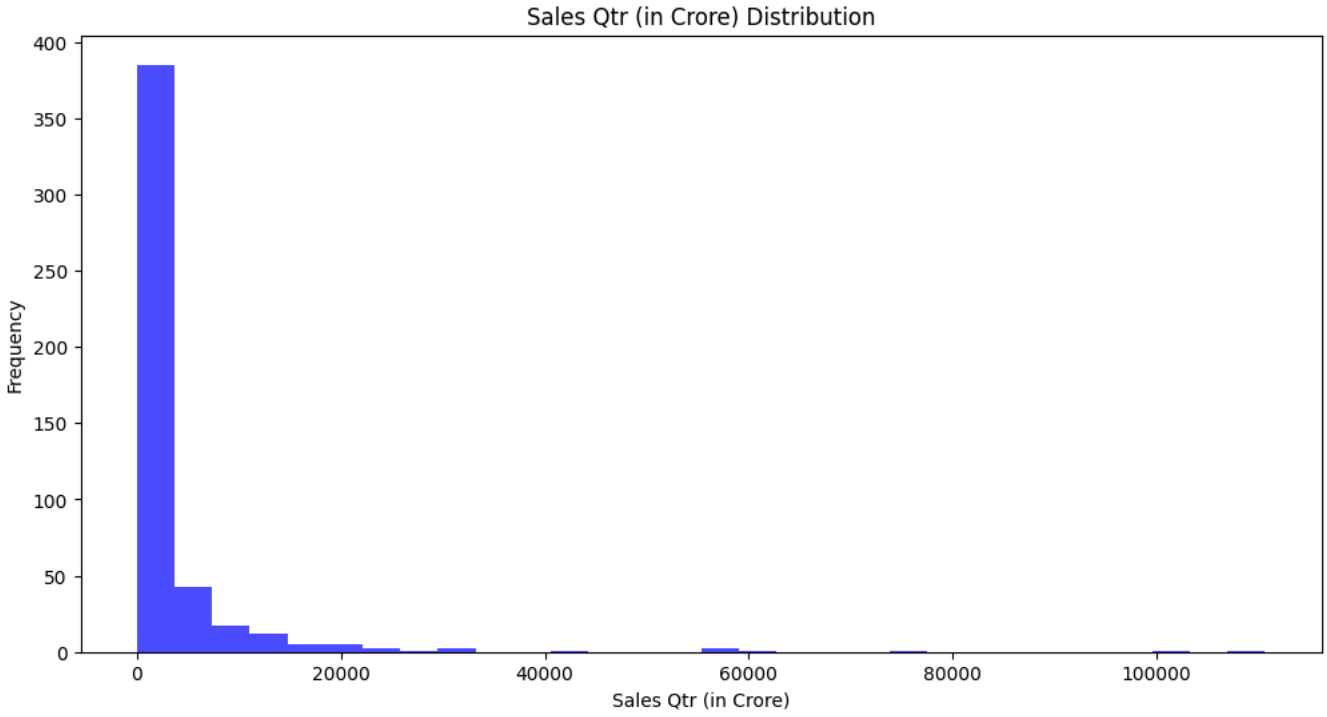
449 rows × 5 columns

Next steps: Generate code with df_cleaned View recommended plots

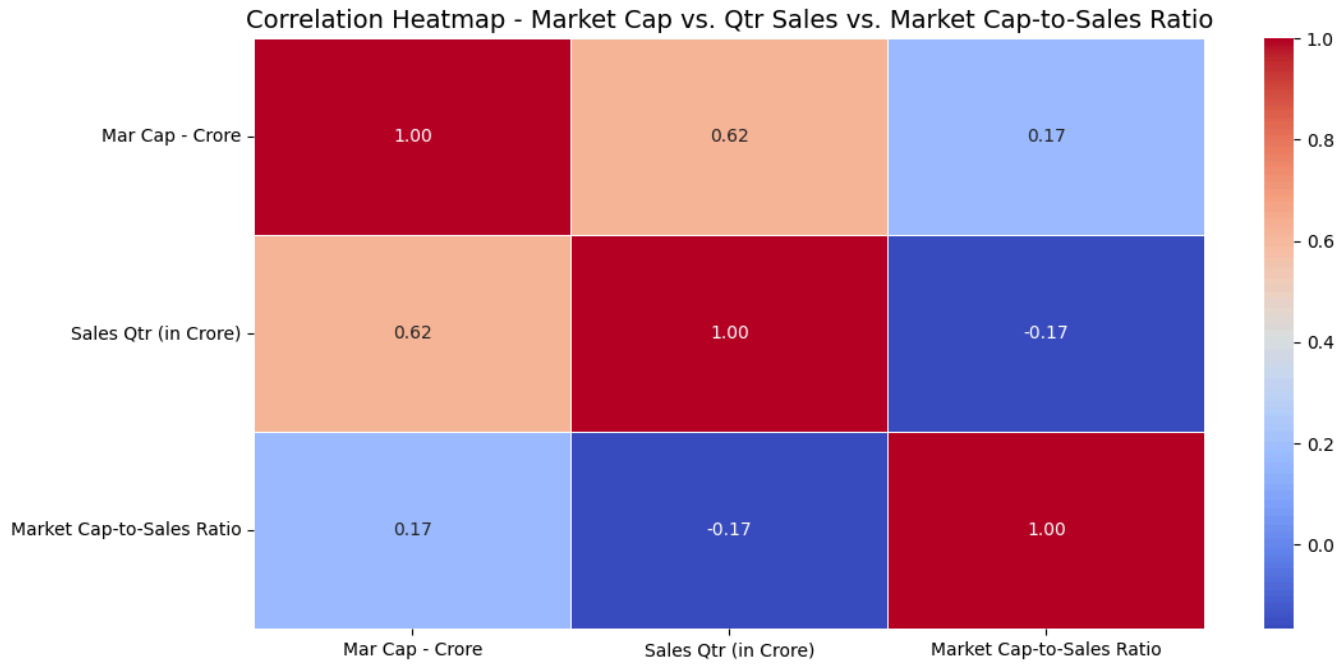
```
plt.figure(figsize=(12,6))

plt.hist(df_cleaned['Mar Cap - Crore'].dropna(), bins = 30, color = 'red', alpha = 0.7)
plt.title("Mar Cap - Crore Distribution")
plt.xlabel('Mar Cap - Crore')
plt.ylabel('Frequency')
plt.show()
```



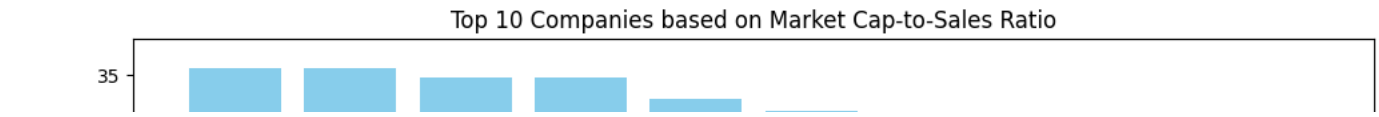

```
plt.figure(figsize=(12,6))

plt.hist(df['Sales Qtr (in Crore)'].dropna(), bins = 30, color = 'blue', alpha = 0.7)
plt.title("Sales Qtr (in Crore) Distribution")
plt.xlabel('Sales Qtr (in Crore)')
plt.ylabel('Frequency')
plt.show()
```



```
correlation_data = df_cleaned[['Mar Cap - Crore', 'Sales Qtr (in Crore)','Market Cap-to-Sales Ratio']]

correlation_matrix = correlation_data.corr()

plt.figure(figsize=(12, 6))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f", linewidths=.5)

plt.title('Correlation Heatmap - Market Cap vs. Qtr Sales vs. Market Cap-to-Sales Ratio',fontsize = 14)
plt.xticks(rotation=0)
plt.yticks(rotation=0)

plt.show()
```




```
top_companies = df_cleaned.nlargest(10, 'Market Cap-to-Sales Ratio')

plt.figure(figsize=(12,6))
plt.bar(top_companies['Name'], top_companies['Market Cap-to-Sales Ratio'], color='skyblue')
plt.title('Top 10 Companies based on Market Cap-to-Sales Ratio')
plt.xlabel('Name of Company')
plt.ylabel('Market Cap-to-Sales Ratio')
plt.xticks(rotation=45, ha='right')
plt.show()
```




```
top_companies = df_cleaned.nsmallest(10, 'Market Cap-to-Sales Ratio')

top_companies
```

	S.No.	Name	Mar Cap - Crore	Sales Qtr (in Crore)	Market Cap-to-Sales Ratio
49	50	Bharti Infra.	61776.92	0.0	0.0
171	176	Info Edg.(India)	14845.05	0.0	0.0
185	192	Max Financial	13401.76	0.0	0.0
224	231	Bombay Burmah	10864.53	0.0	0.0
241	248	Sundaram Clayton	10074.36	0.0	0.0
258	271	Mahindra CIE	8587.04	0.0	0.0
314	327	Prism Cement	6176.23	0.0	0.0
332	345	GE Power	5497.40	0.0	0.0
338	351	MMTC	5300.00	0.0	0.0
370	383	Swan Energy	4721.49	0.0	0.0

Next steps: Generate code with top_companies View recommended plots

```
plt.figure(figsize=(12, 6))
plt.boxplot(df_cleaned['Market Cap-to-Sales Ratio'], vert=False, sym='b.')
plt.title(f'Boxplot for Market Cap-to-Sales Ratio')
plt.xlabel('Market Cap-to-Sales Ratio')
plt.ylabel('Distribution')
plt.show()
```