

Untitled

August 22, 2024

```
[32]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings("ignore")
%matplotlib inline
import seaborn as sn
```

```
[2]: df = pd.read_csv("Financial Analytics data.csv")
df.head()
```

```
[2]:
```

	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

0.1 Exploratory Data Analysis

```
[3]: df.columns
```

```
[3]: Index(['S.No.', 'Name', 'Mar Cap - Crore', 'Sales Qtr - Crore', 'Unnamed: 4'],
dtype='object')
```

```
[4]: df['Qtr Sales - Crore'] = df['Sales Qtr - Crore'].combine_first(df['Unnamed: 4']
↪4'])
```

```
[5]: df.tail()
```

```
[5]:
```

	S.No.	Name	Mar Cap - Crore	Sales Qtr - Crore	Unnamed: 4	\
483	496	Lak. Vilas Bank	3029.57	790.17	NaN	
484	497	NOCIL	3026.26	249.27	NaN	
485	498	Orient Cement	3024.32	511.53	NaN	
486	499	Natl.Fertilizer	3017.07	2840.75	NaN	
487	500	L T Foods	NaN	NaN	NaN	

	Qtr Sales - Crore
483	790.17
484	249.27
485	511.53
486	2840.75
487	NaN

```
[6]: df.drop(['Sales Qtr - Crore', 'Unnamed: 4'], inplace = True, axis = 1)
```

```
[7]: df.head()
```

```
[7]:
```

	S.No.	Name	Mar Cap - Crore	Qtr Sales - 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

```
[29]: df.columns
```

```
[29]: Index(['S.No.', 'Name', 'Mar Cap - Crore', 'Qtr Sales - Crore'], dtype='object')
```

```
[26]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 488 entries, 0 to 487
Data columns (total 4 columns):
#   Column                Non-Null Count  Dtype
---  -
0   S.No.                 488 non-null   int64
1   Name                  488 non-null   object
2   Mar Cap - Crore       488 non-null   float64
3   Qtr Sales - Crore     488 non-null   float64
dtypes: float64(2), int64(1), object(1)
memory usage: 15.4+ KB
```

```
[27]: df.describe()
```

```
[27]:
```

	S.No.	Mar Cap - Crore	Qtr Sales - Crore
count	488.000000	488.000000	488.000000
mean	251.508197	27596.880460	3807.789412
std	145.884078	59002.947102	9687.458059
min	1.000000	3017.070000	0.000000
25%	122.750000	4643.832500	570.035000
50%	252.500000	9460.155000	1276.255000
75%	378.250000	23400.815000	3753.280000
max	500.000000	583436.720000	110666.930000

```
[8]: df.isna().sum()
```

```
[8]: S.No.          0
     Name          0
     Mar Cap - Crore    9
     Qtr Sales - Crore 29
     dtype: int64
```

```
[9]: df.fillna((df['Qtr Sales - Crore']).mean(), inplace = True, axis = 0)
```

```
[10]: df['Name'].nunique()
```

```
[10]: 488
```

```
[11]: df['Name'].dtype
```

```
[11]: dtype('O')
```

```
[12]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 488 entries, 0 to 487
Data columns (total 4 columns):
 #   Column                Non-Null Count  Dtype
---  -
 0   S.No.                 488 non-null   int64
 1   Name                  488 non-null   object
 2   Mar Cap - Crore       488 non-null   float64
 3   Qtr Sales - Crore     488 non-null   float64
dtypes: float64(2), int64(1), object(1)
memory usage: 15.4+ KB
```

```
[13]: df.describe()
```

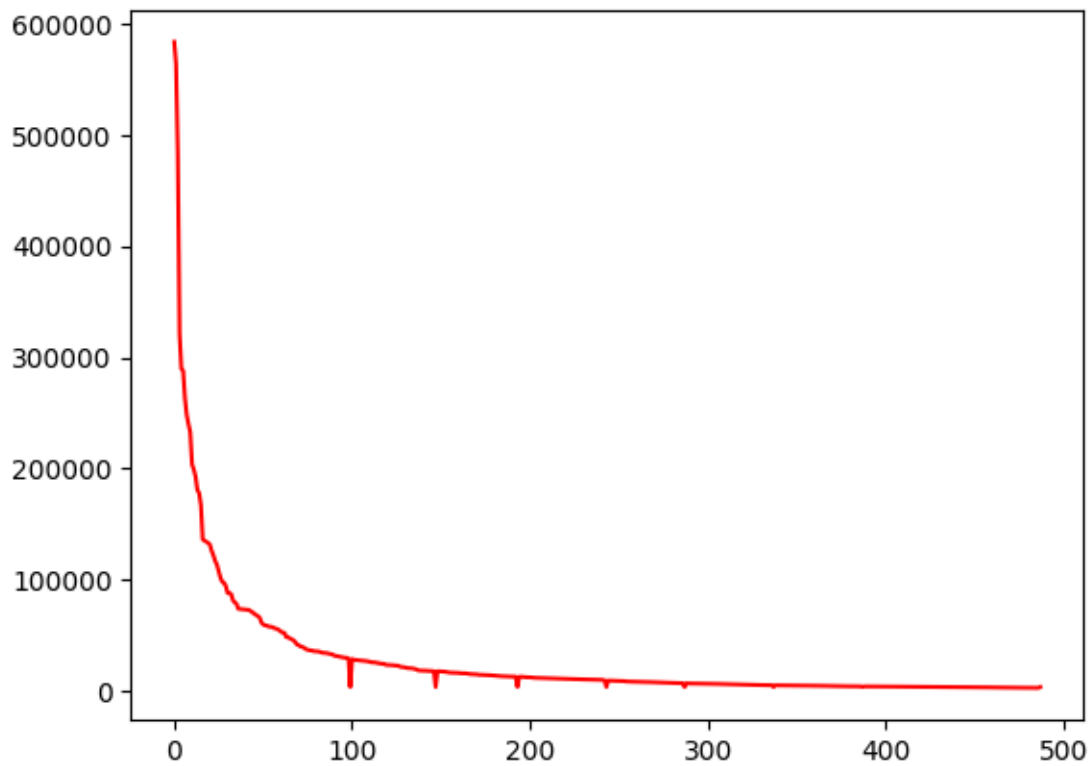
```
[13]:
```

	S.No.	Mar Cap - Crore	Qtr Sales - Crore
count	488.000000	488.000000	488.000000
mean	251.508197	27596.880460	3807.789412
std	145.884078	59002.947102	9687.458059
min	1.000000	3017.070000	0.000000
25%	122.750000	4643.832500	570.035000
50%	252.500000	9460.155000	1276.255000
75%	378.250000	23400.815000	3753.280000
max	500.000000	583436.720000	110666.930000

```
[14]: df.sort_index(inplace = True)
```

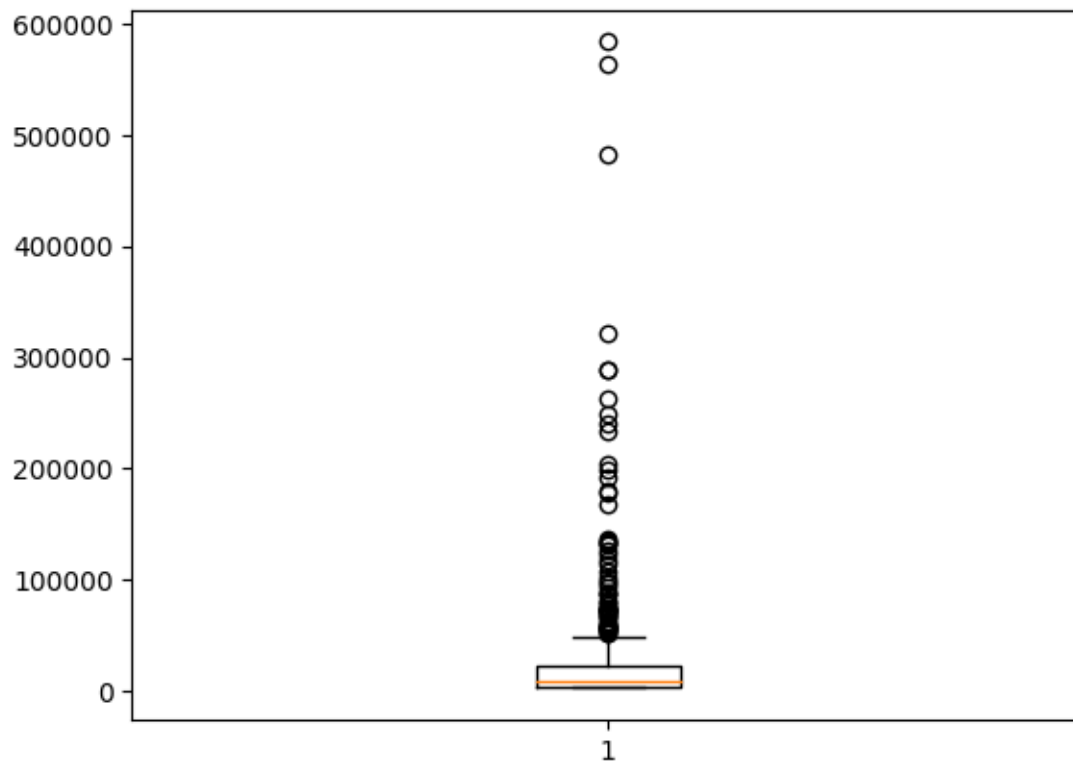
```
[15]: plt.plot(df['Mar Cap - Crore'],color = 'red')
```

```
[15]: [<matplotlib.lines.Line2D at 0x1d77fc85570>]
```



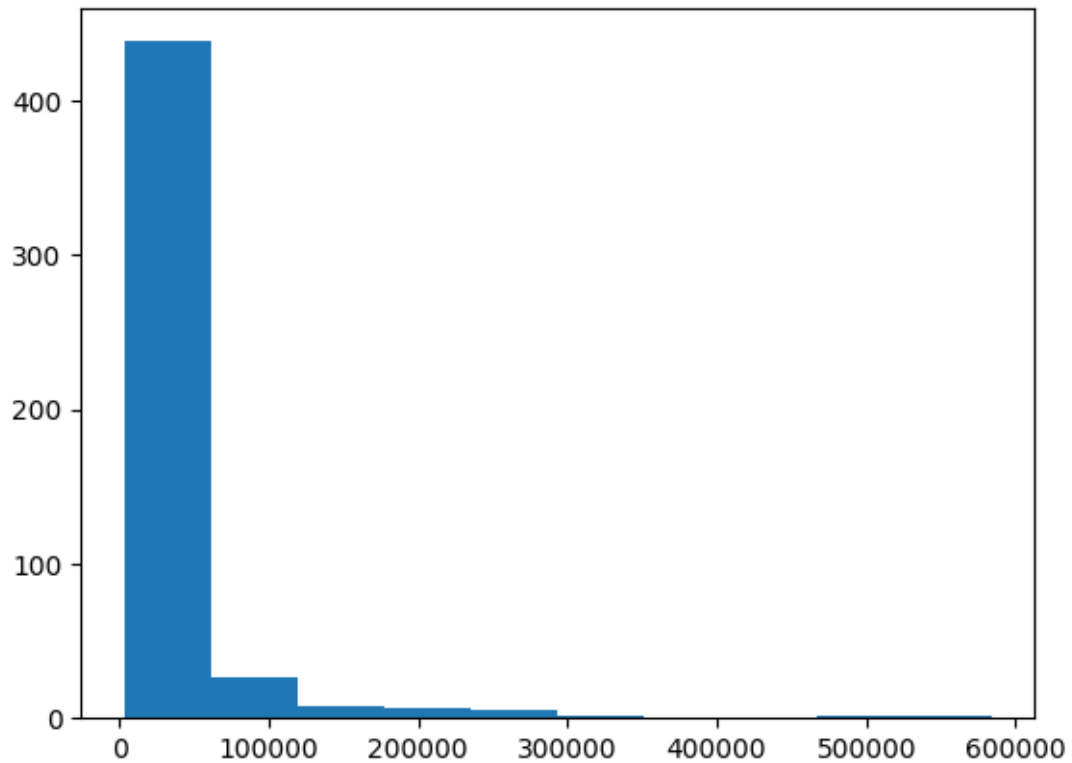
```
[16]: plt.boxplot(df['Mar Cap - Crore'])
```

```
[16]: {'whiskers': [<matplotlib.lines.Line2D at 0x1d77fec6080>,
<matplotlib.lines.Line2D at 0x1d77fec6320>],
'caps': [<matplotlib.lines.Line2D at 0x1d77fec65c0>,
<matplotlib.lines.Line2D at 0x1d77fec6860>],
'boxes': [<matplotlib.lines.Line2D at 0x1d77fec5de0>],
'medians': [<matplotlib.lines.Line2D at 0x1d77fec6b00>],
'fliers': [<matplotlib.lines.Line2D at 0x1d77fec6da0>],
'means': []}
```



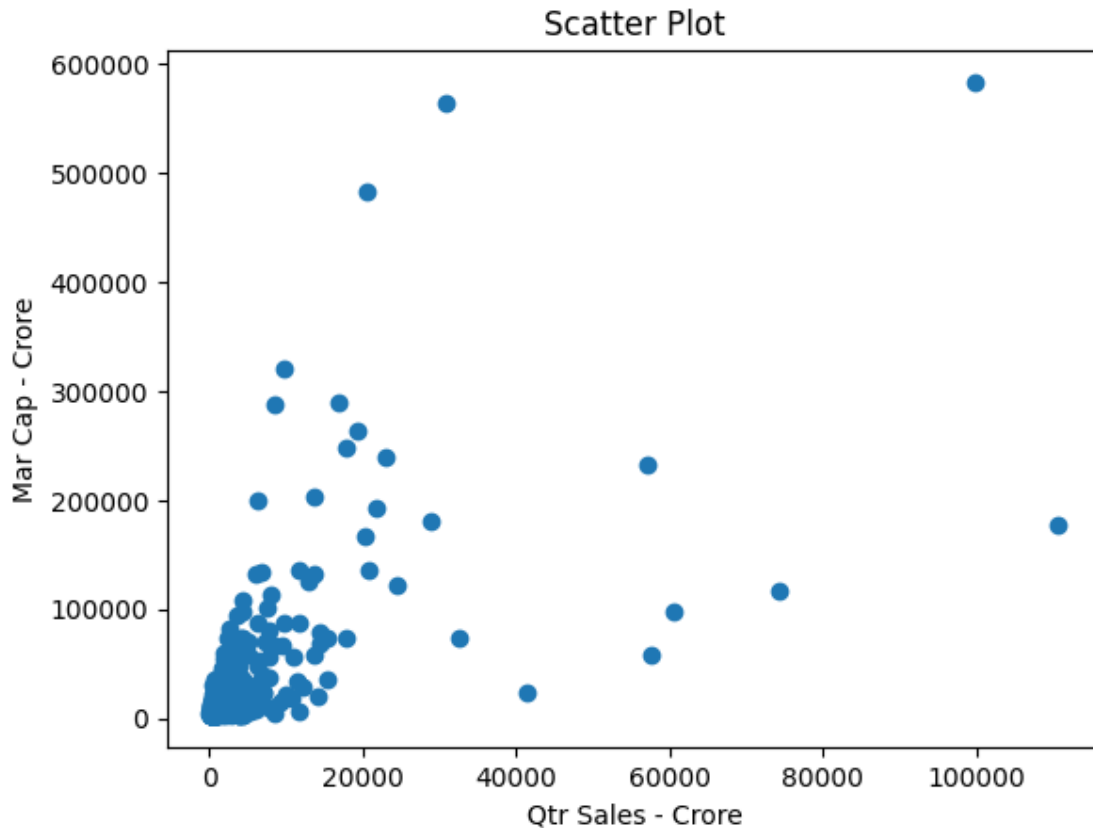
```
[17]: plt.hist(df['Mar Cap - Crore'])
```

```
[17]: (array([438., 27., 8., 6., 5., 1., 0., 0., 1., 2.]),
array([ 3017.07 , 61059.035, 119101.   , 177142.965, 235184.93 ,
        293226.895, 351268.86 , 409310.825, 467352.79 , 525394.755,
        583436.72 ]),
<BarContainer object of 10 artists>)
```



```
[18]: plt.xlabel('Qtr Sales - Crore')
plt.ylabel('Mar Cap - Crore')
plt.title('Scatter Plot')
plt.scatter(x = df['Qtr Sales - Crore'], y=df['Mar Cap - Crore'])
```

```
[18]: <matplotlib.collections.PathCollection at 0x1d71403f310>
```



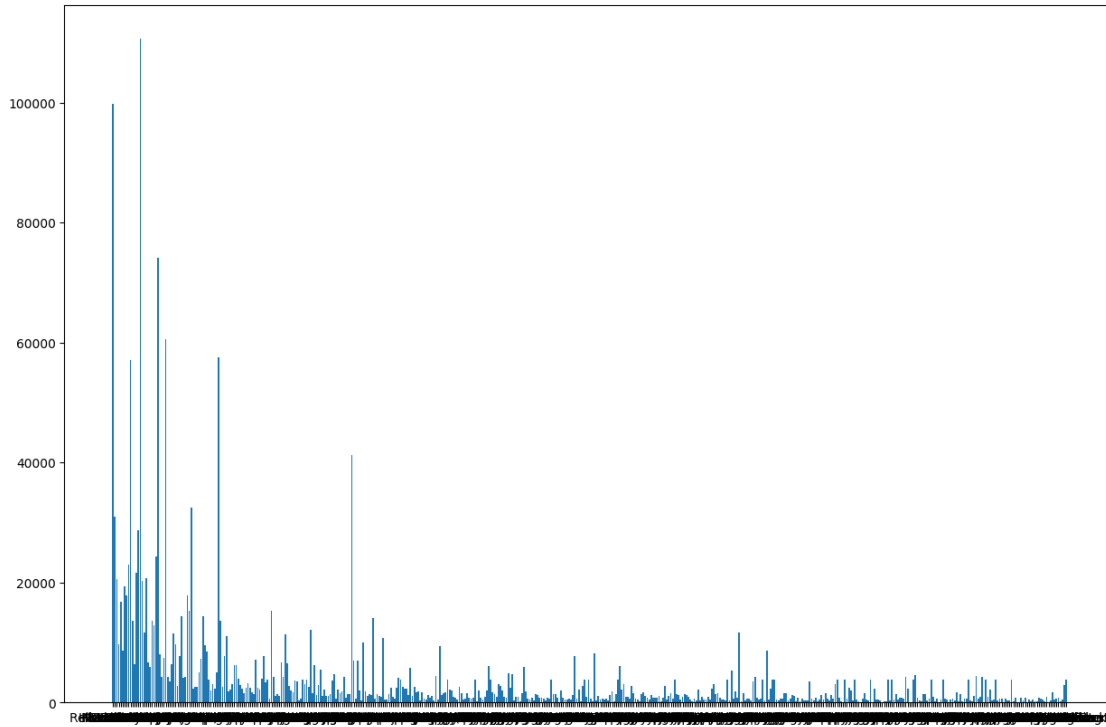
```
[19]: df.head()
```

```
[19]:
```

	S.No.	Name	Mar Cap - Crore	Qtr Sales - 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

```
[20]: plt.figure(figsize = (15,10), dpi = 100)
plt.bar(x = df['Name'], height = df['Qtr Sales - Crore'])
```

```
[20]: <BarContainer object of 488 artists>
```



```
[21]: df.head(10)
```

```
[21]:
```

	S.No.	Name	Mar Cap - Crore	Qtr Sales - 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
5	6	Hind. Unilever	288265.26	8590.00
6	7	Maruti Suzuki	263493.81	19283.20
7	8	Infosys	248320.35	17794.00
8	9	O N G C	239981.50	22995.88
9	10	St Bk of India	232763.33	57014.08

```
[22]: df.tail(10)
```

```
[22]:
```

	S.No.	Name	Mar Cap - Crore	Qtr Sales - Crore
478	491	Kaveri Seed Co.	3125.830000	70.640000
479	492	Star Ferro Cem.	3115.980000	393.490000
480	493	Deepak Fert.	3079.060000	1644.920000
481	494	Va Tech Wabag	3041.930000	460.890000
482	495	Prime Focus	3031.500000	609.610000
483	496	Lak. Vilas Bank	3029.570000	790.170000
484	497	NOCIL	3026.260000	249.270000

485	498	Orient Cement	3024.320000	511.530000
486	499	Natl.Fertilizer	3017.070000	2840.750000
487	500	L T Foods	3807.789412	3807.789412

```
[23]: df_sorted = df.sort_values(by = 'Qtr Sales - Crore', ascending = False)
```

```
[61]: #Top 10 Companies with quarterly sales
df_sorted.head(10)
```

```
[61]:
```

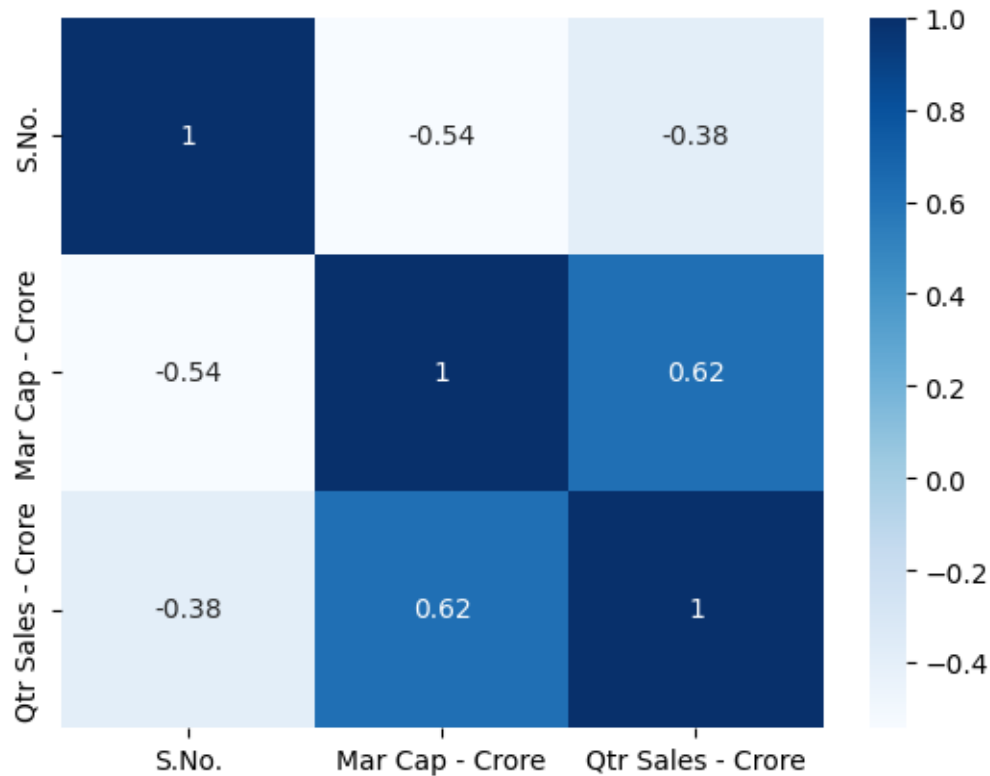
	S.No.	Name	Mar Cap - Crore	Qtr Sales - Crore	
	14	15	I O C L	178017.48	110666.93
	0	1	Reliance Inds.	583436.72	99810.00
	23	24	Tata Motors	117071.87	74156.07
	27	28	B P C L	98278.00	60616.36
	54	55	H P C L	58034.78	57474.25
	9	10	St Bk of India	232763.33	57014.08
	122	123	Rajesh Exports	23495.54	41304.84
	40	41	Tata Steel	73376.14	32464.14
	1	2	TCS	563709.84	30904.00
	13	14	Larsen & Toubro	180860.74	28747.45

```
[25]: #Lowest 10 Companies with quarterly sales
df_sorted.tail(10)
```

```
[25]:
```

	S.No.	Name	Mar Cap - Crore	Qtr Sales - Crore	
	408	421	CARE Ratings	4090.69	74.82
	478	491	Kaveri Seed Co.	3125.83	70.64
	455	468	La Opala RG	3510.93	69.77
	373	386	Indian Energy Ex	4595.70	64.75
	388	401	Forbes & Co	4331.82	63.93
	422	435	Multi Comm. Exc.	3847.19	60.97
	467	480	Central Dep. Ser	3316.31	47.24
	382	395	Tata Inv.Corpnn.	4401.66	47.02
	228	235	SPARC	10755.13	19.42
	393	406	Ujjivan Fin.Ser.	4293.42	0.00

```
[33]: #Finding the correlation between variables
sns.heatmap(df.corr(), annot = True, cmap = 'Blues')
plt.show()
```



0.2 Fitting a Linear Regression model

```
[39]: from sklearn.model_selection import train_test_split
      from sklearn.linear_model import LinearRegression
      from sklearn.metrics import mean_squared_error, r2_score
```

```
[40]: X = df['Mar Cap - Crore']
      y = df['Qtr Sales - Crore']
```

```
[46]: X = X.array.reshape(-1, 1)
```

```
[47]: X_train, X_test, y_train, y_test = train_test_split(X,y, test_size = 20,
      ↪random_state = 42)
```

```
[48]: model = LinearRegression()
      model.fit(X_train, y_train)
```

```
[48]: LinearRegression()
```

```
[52]: y_pred = model.predict(X_test)
```

```
[53]: mse = mean_squared_error(y_test, y_pred)
      r2 = r2_score(y_test, y_pred)
```

```
[54]: print(f"Mean Squared Error is: {mse}")
      print(f"R2 score is: {r2}")
```

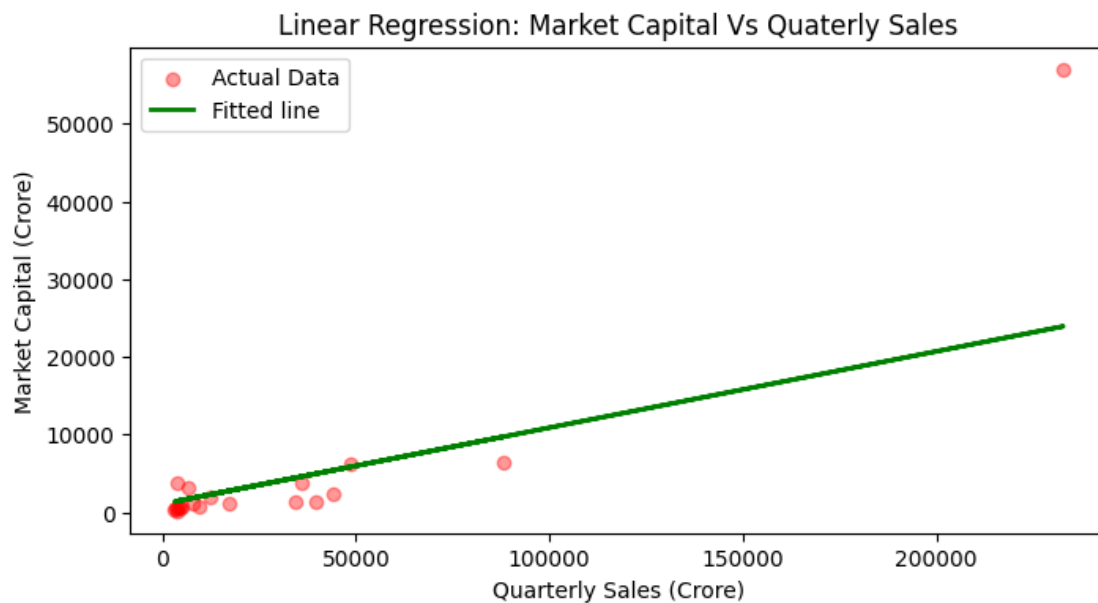
Mean Squared Error is: 57904968.63235223
R2 score is: 0.6067523795831016

0.2.1 Goodness of Fit:

An R^2 score of around 0.60 suggests that model has a decent fit but leaves room for improvement. It indicates that your model is capturing a significant portion of the variance but not all of it.

0.2.2 Visualizing the model

```
[70]: plt.figure(figsize = (8, 4))
      plt.scatter(X_test, y_test, alpha = 0.4,color = 'red', label = 'Actual Data')
      plt.plot(X_test, y_pred, color = 'green', linewidth = 2, label = 'Fitted line')
      plt.title('Linear Regression: Market Capital Vs Quaterly Sales')
      plt.xlabel('Quarterly Sales (Crore)')
      plt.ylabel('Market Capital (Crore)')
      plt.legend()
      plt.show()
```



```
[ ]:
```