Financial Analysis

```
#importing various packages
import pandas as pd
import numpy as np
import seaborn as sns
from sklearn.impute import KNNImputer


df=pd.read_csv('/content/Financial Analytics data.csv')



#displaying top 10 rows
df.head(10)
```

	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
5	6	Hind. Unilever	288265.26	8590.00	NaN
6	7	Maruti Suzuki	263493.81	19283.20	NaN
7	8	Infosys	248320.35	17794.00	NaN
8	9	O N G C	239981.50	22995.88	NaN
9	10	St Bk of India	232763.33	57014.08	NaN

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
#checking how many null values are there
df.isna().sum()
```

	0
S.No.	0
Name	0
Mar Cap - Crore	9
Sales Qtr - Crore	123
Unnamed: 4	394

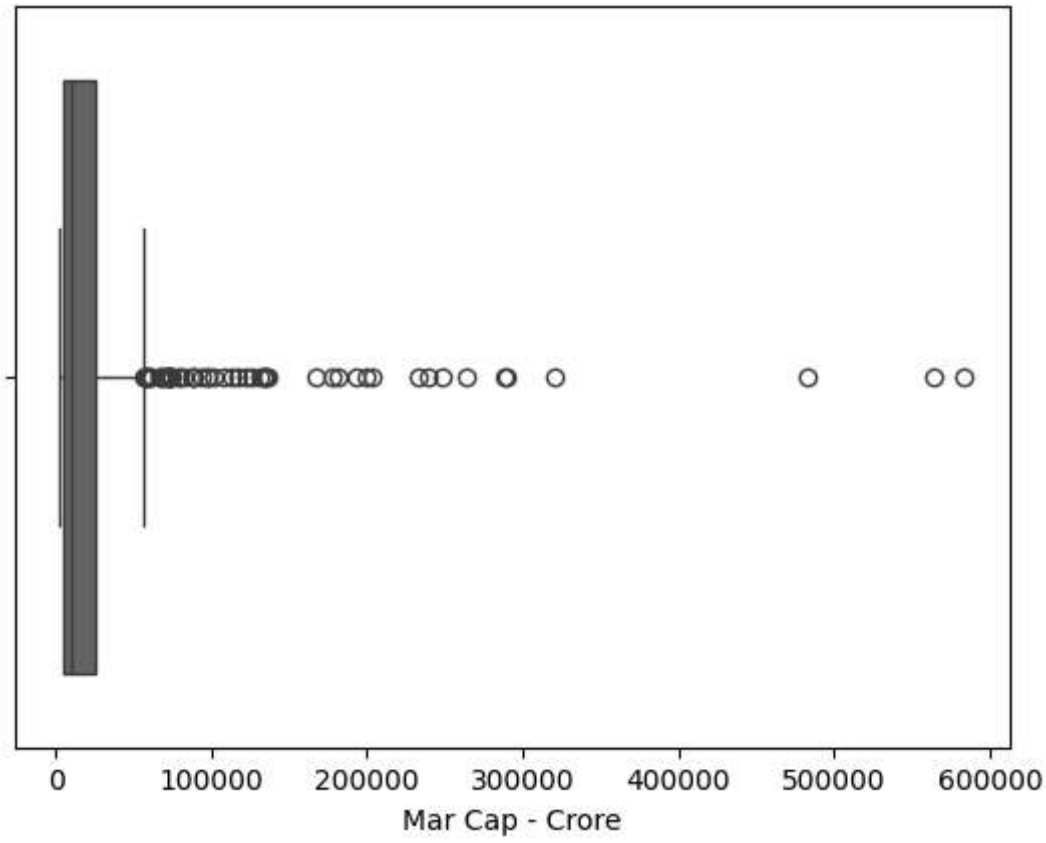
dtype: int64

```
#creating a copy of the original dataframe
df1=df


#storing the values of both these columns for KNN imputation
data = df1[['Mar Cap - Crore', 'Sales Qtr - Crore']]



#initialize the KNN imputer
imputer = KNNImputer(n_neighbors=5)



#impute the missing values
imputed_data = imputer.fit_transform(data)



#updating the dataset with imputed values
df1[['Mar Cap - Crore', 'Sales Qtr - Crore']] = imputed_data



#checking random rows of the data
df1.sample(10)
```

	S.No.	Name	Mar Cap - Crore	Sales Qtr - Crore	Unnamed: 4
104	105	United Breweries	27797.69	1197.100	NaN
25	26	Asian Paints	108044.04	4260.520	NaN
151	154	NBCC	17712.00	1321.500	NaN
88	89	Bank of Baroda	33364.23	11303.240	NaN
333	346	Timken India	5495.76	278.580	NaN
258	271	Mahindra CIE	8587.04	2321.488	NaN
303	316	Himadri Specialt	6646.41	500.080	NaN
203	210	Reliance Power	12033.99	2494.650	NaN
422	435	Multi Comm. Exc.	3847.19	456.808	60.97
302	315	Lak. Mach. Works	6654.81	542.420	NaN

```
#checking if the 'Mar Cap - Crore' column has any outliers
sns.boxplot(data=df1, x='Mar Cap - Crore')
```

```
<Axes: xlabel='Mar Cap - Crore'>
```

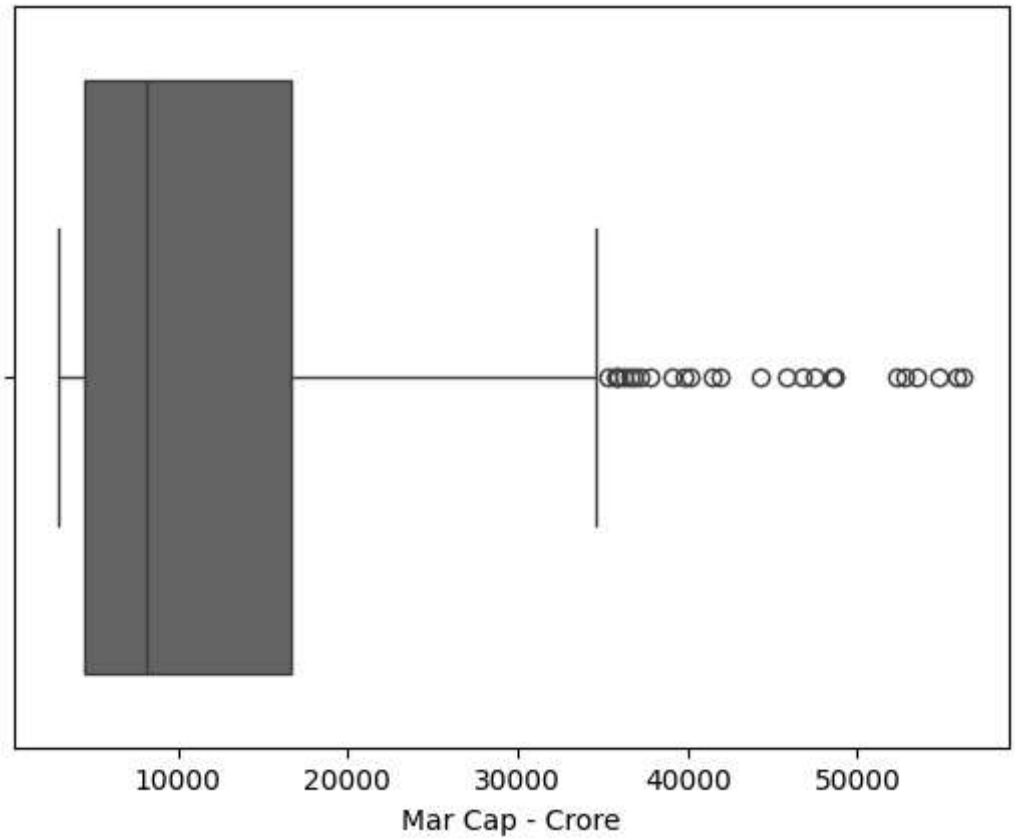


```
#I will use IQR method to eliminate the outliers
#calculating first and third quartiles
Q1 = df1['Mar Cap - Crore'].quantile(0.25)
Q3 = df1['Mar Cap - Crore'].quantile(0.75)

#Calculating IQR
IQR = Q3-Q1
```

```
#defining lower and upper limits
lower = Q1-1.5*IQR
upper = Q3+1.5*IQR

#filtering the dataset to remove the outliers
df1 = df1[(df1['Mar Cap - Crore'] >= lower) & (df1['Mar Cap - Crore'] <= upper)]
```

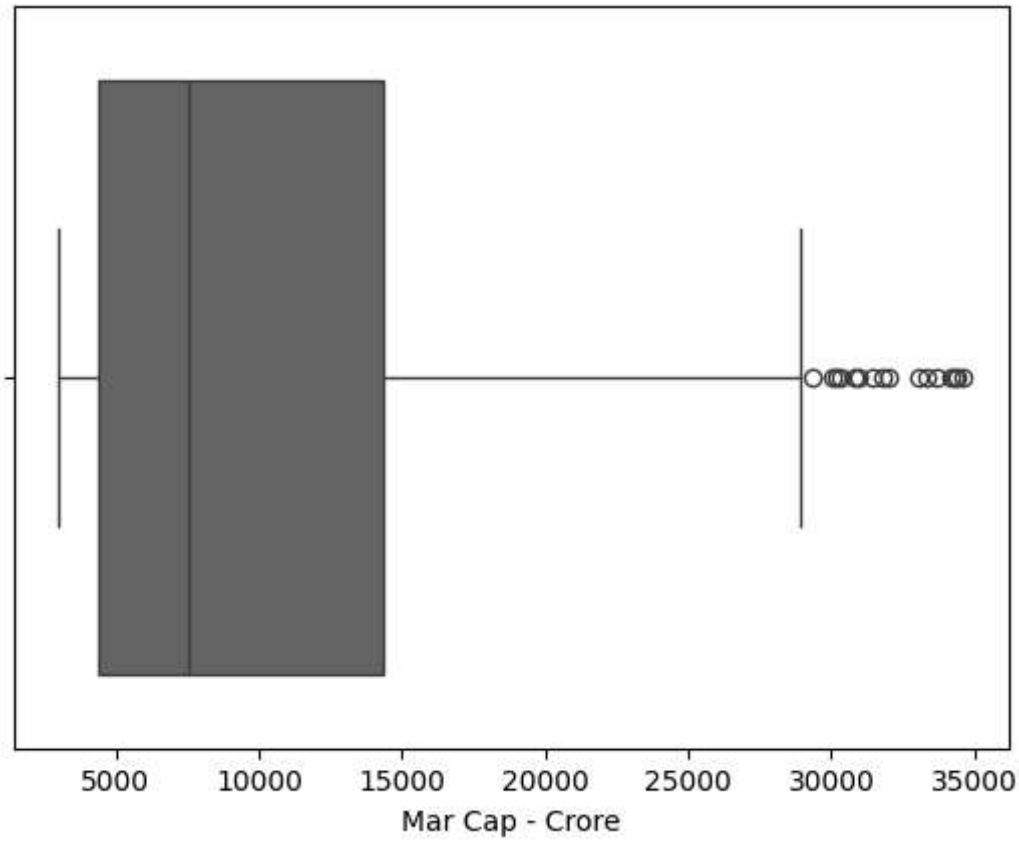
```
#checking if we still have the outliers using boxplot
sns.boxplot(data=df1, x='Mar Cap - Crore')
```

```
<Axes: xlabel='Mar Cap - Crore'>
```



```
#I will use IQR method to eliminate the outliers
#calculating first and third quartiles
Q1 = df1['Mar Cap - Crore'].quantile(0.25)
Q3 = df1['Mar Cap - Crore'].quantile(0.75)


#Calculating IQR
IQR = Q3-Q1

#defining lower and upper limits
lower = Q1-1.5*IQR
upper = Q3+1.5*IQR




#filtering the dataset to remove the outliers
df1 = df1[(df1['Mar Cap - Crore'] >= lower) & (df1['Mar Cap - Crore'] <= upper)]




#checking if we still have the outliers using boxplot
sns.boxplot(data=df1, x='Mar Cap - Crore')
```

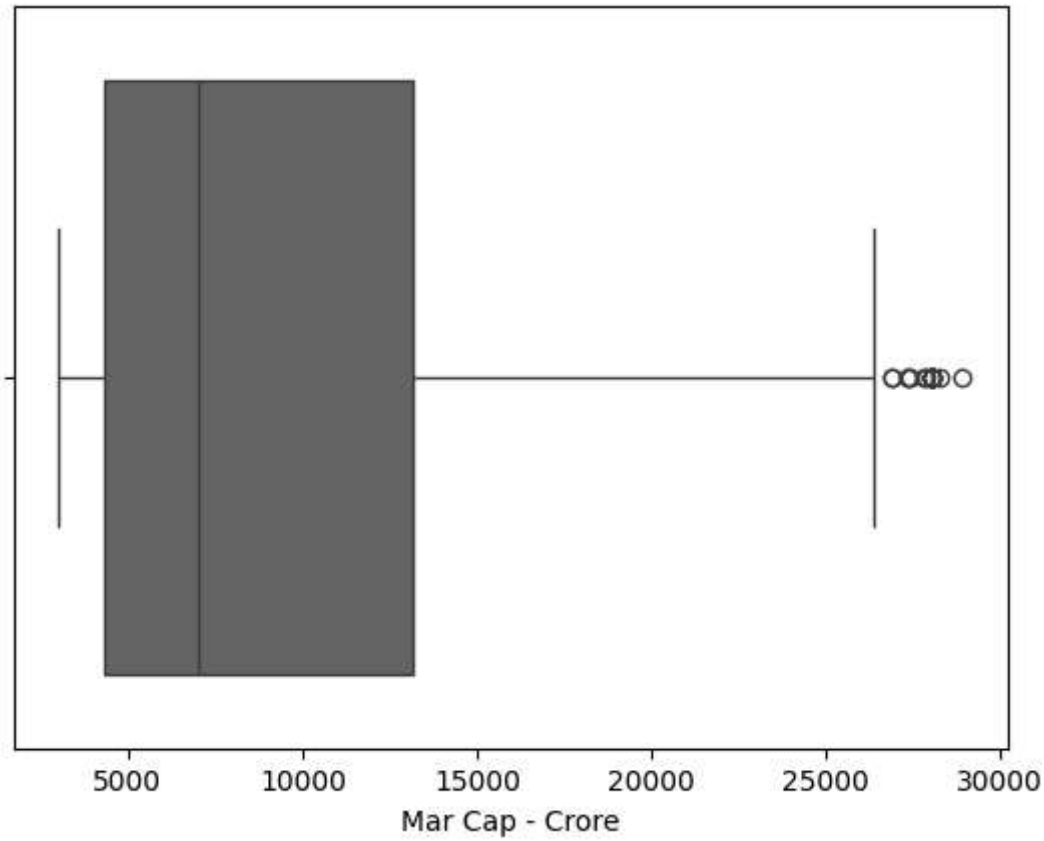
<Axes: xlabel='Mar Cap - Crore'>




```
#I will use IQR method to eliminate the outliers
#calculating first and third quartiles
Q1 = df1['Mar Cap - Crore'].quantile(0.25)
Q3 = df1['Mar Cap - Crore'].quantile(0.75)

#Calculating IQR
IQR = Q3-Q1

#defining lower and upper limits
lower = Q1-1.5*IQR
upper = Q3+1.5*IQR

#filtering the dataset to remove the outliers
df1 = df1[(df1['Mar Cap - Crore'] >= lower) & (df1['Mar Cap - Crore'] <= upper)]
```

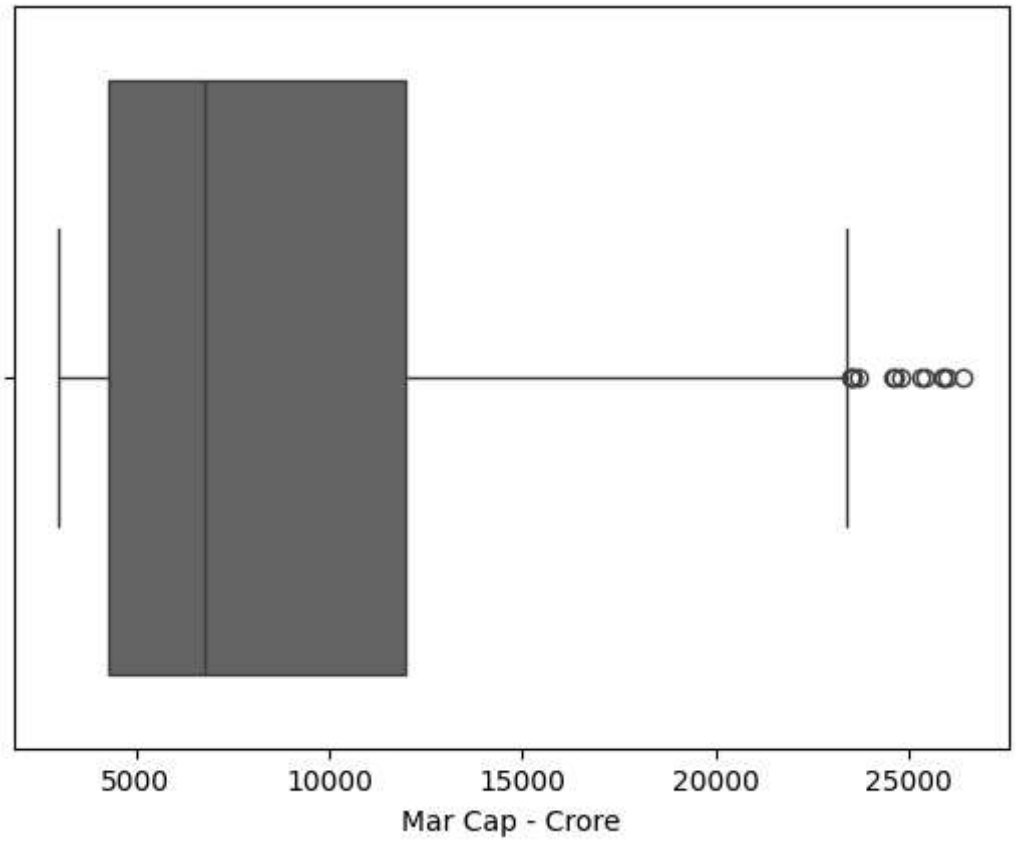
```
#checking if we still have the outliers using boxplot
sns.boxplot(data=df1, x='Mar Cap - Crore')
```

<Axes: xlabel='Mar Cap - Crore'>



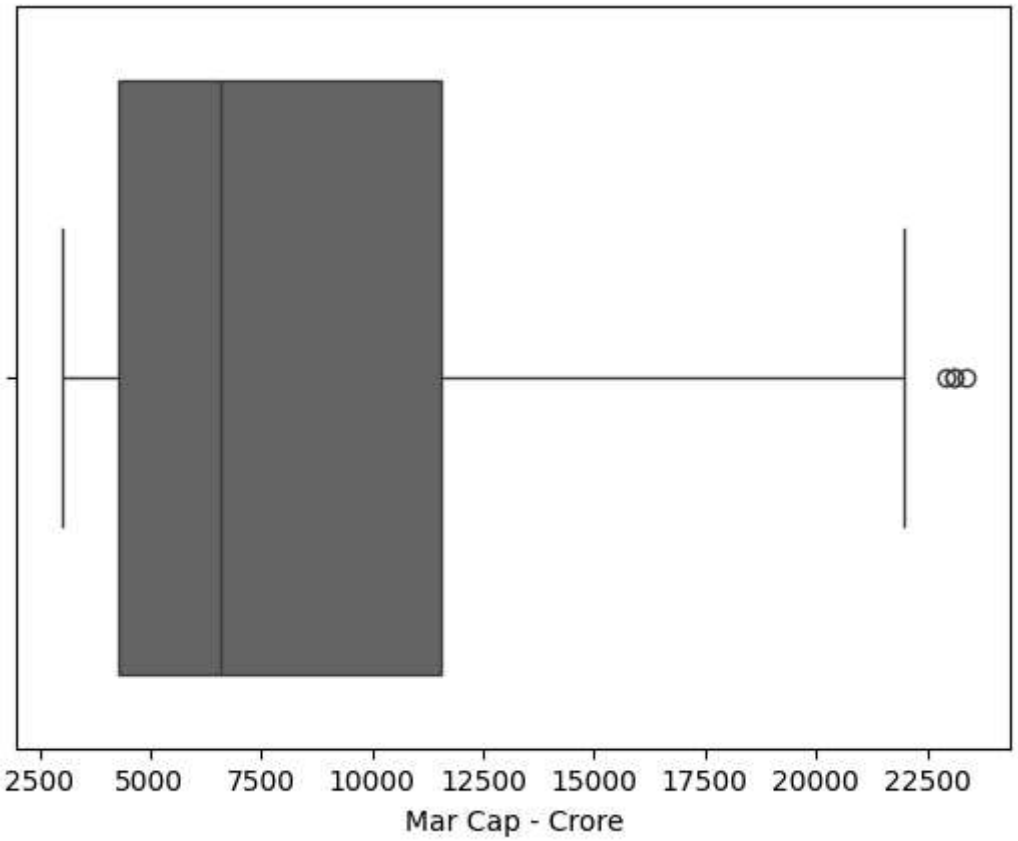

```
#I will use IQR method to eliminate the outliers
#calculating first and third quartiles
Q1 = df1['Mar Cap - Crore'].quantile(0.25)
Q3 = df1['Mar Cap - Crore'].quantile(0.75)

#Calculating IQR
IQR = Q3-Q1

#defining lower and upper limits
lower = Q1-1.5*IQR
upper = Q3+1.5*IQR

#filtering the dataset to remove the outliers
df1 = df1[(df1['Mar Cap - Crore'] >= lower) & (df1['Mar Cap - Crore'] <= upper)]



#checking if we still have the outliers using boxplot
sns.boxplot(data=df1, x='Mar Cap - Crore')
```

<Axes: xlabel='Mar Cap - Crore'>



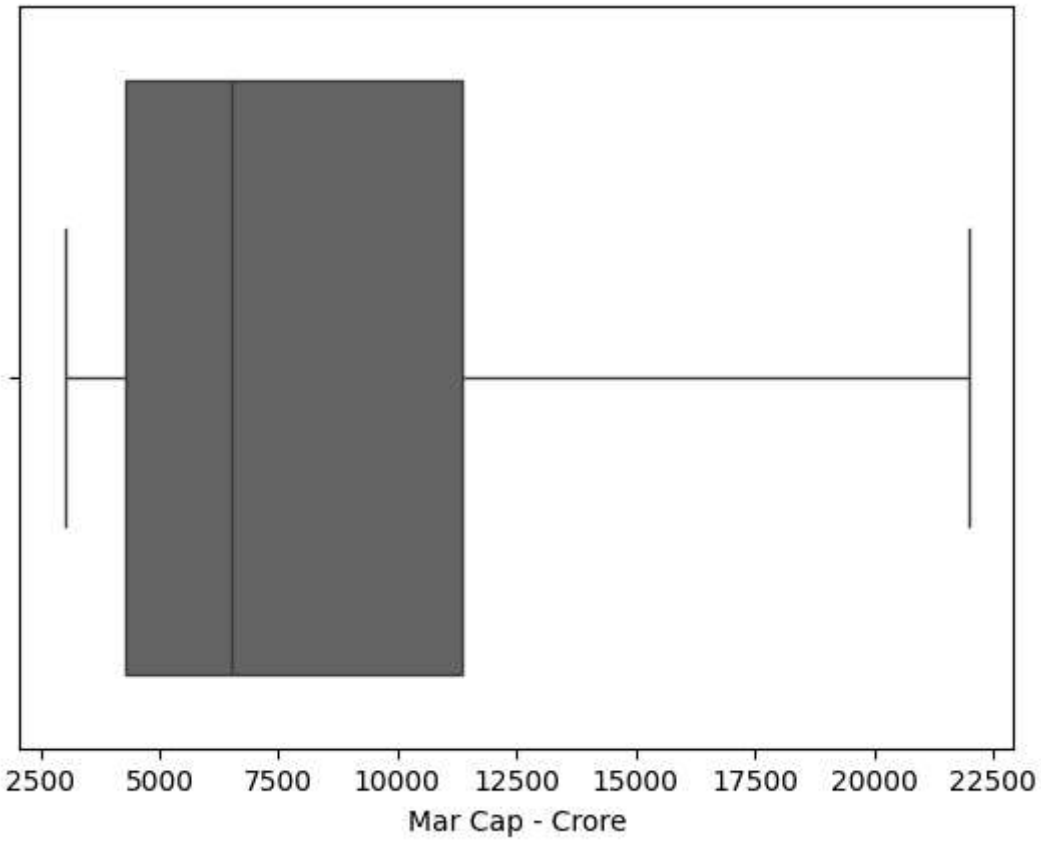

```
#I will use IQR method to eliminate the outliers
#calculating first and third quartiles
Q1 = df1['Mar Cap - Crore'].quantile(0.25)
Q3 = df1['Mar Cap - Crore'].quantile(0.75)

#Calculating IQR
IQR = Q3-Q1

#defining lower and upper limits
lower = Q1-1.5*IQR
upper = Q3+1.5*IQR

#filtering the dataset to remove the outliers
df1 = df1[(df1['Mar Cap - Crore'] >= lower) & (df1['Mar Cap - Crore'] <= upper)]



#checking if we still have the outliers using boxplot
sns.boxplot(data=df1, x='Mar Cap - Crore')
```

<Axes: xlabel='Mar Cap - Crore'>



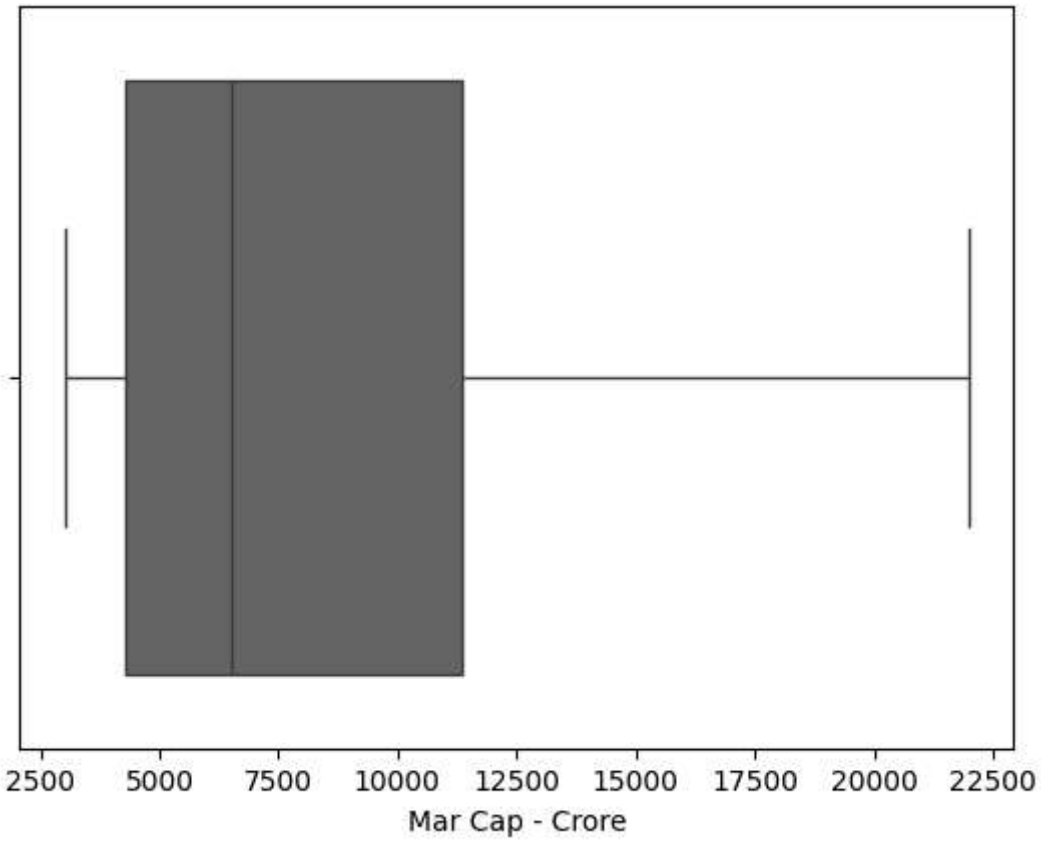

```
#I will use IQR method to eliminate the outliers
#calculating first and third quartiles
Q1 = df1['Mar Cap - Crore'].quantile(0.25)
Q3 = df1['Mar Cap - Crore'].quantile(0.75)

#Calculating IQR
IQR = Q3-Q1

#defining lower and upper limits
lower = Q1-1.5*IQR
upper = Q3+1.5*IQR

#filtering the dataset to remove the outliers
df1 = df1[(df1['Mar Cap - Crore'] >= lower) & (df1['Mar Cap - Crore'] <= upper)]



#checking if we still have the outliers using boxplot
sns.boxplot(data=df1, x='Mar Cap - Crore')
```

<Axes: xlabel='Mar Cap - Crore'>



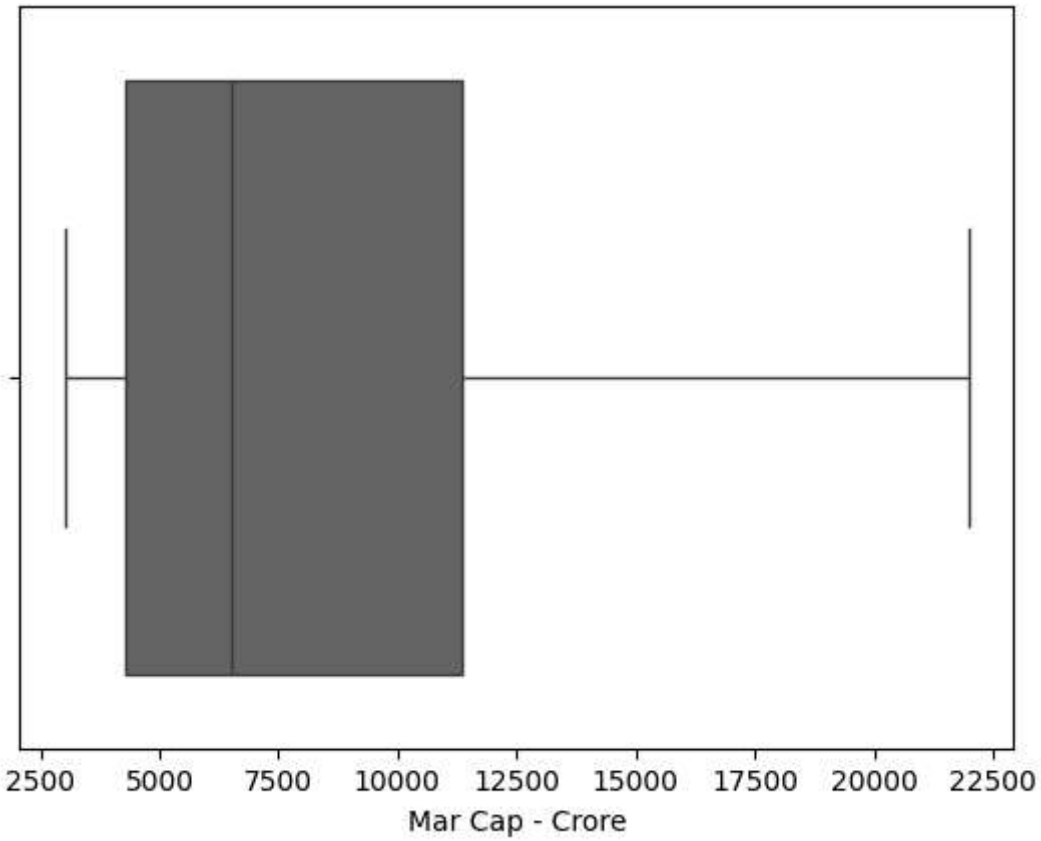

```
#I will use IQR method to eliminate the outliers
#calculating first and third quartiles
Q1 = df1['Mar Cap - Crore'].quantile(0.25)
Q3 = df1['Mar Cap - Crore'].quantile(0.75)

#Calculating IQR
IQR = Q3-Q1

#defining lower and upper limits
lower = Q1-1.5*IQR
upper = Q3+1.5*IQR

#filtering the dataset to remove the outliers
df1 = df1[(df1['Mar Cap - Crore'] >= lower) & (df1['Mar Cap - Crore'] <= upper)]



#checking if we still have the outliers using boxplot
sns.boxplot(data=df1, x='Mar Cap - Crore')
```

<Axes: xlabel='Mar Cap - Crore'>




```
#I will use IQR method to eliminate the outliers
#calculating first and third quartiles
Q1 = df1['Mar Cap - Crore'].quantile(0.25)
Q3 = df1['Mar Cap - Crore'].quantile(0.75)

#Calculating IQR
IQR = Q3-Q1

#defining lower and upper limits
lower = Q1-1.5*IQR
upper = Q3+1.5*IQR

#filtering the dataset to remove the outliers
df1 = df1[(df1['Mar Cap - Crore'] >= lower) & (df1['Mar Cap - Crore'] <= upper)]



#checking if we still have the outliers using boxplot
sns.boxplot(data=df1, x='Mar Cap - Crore')
```

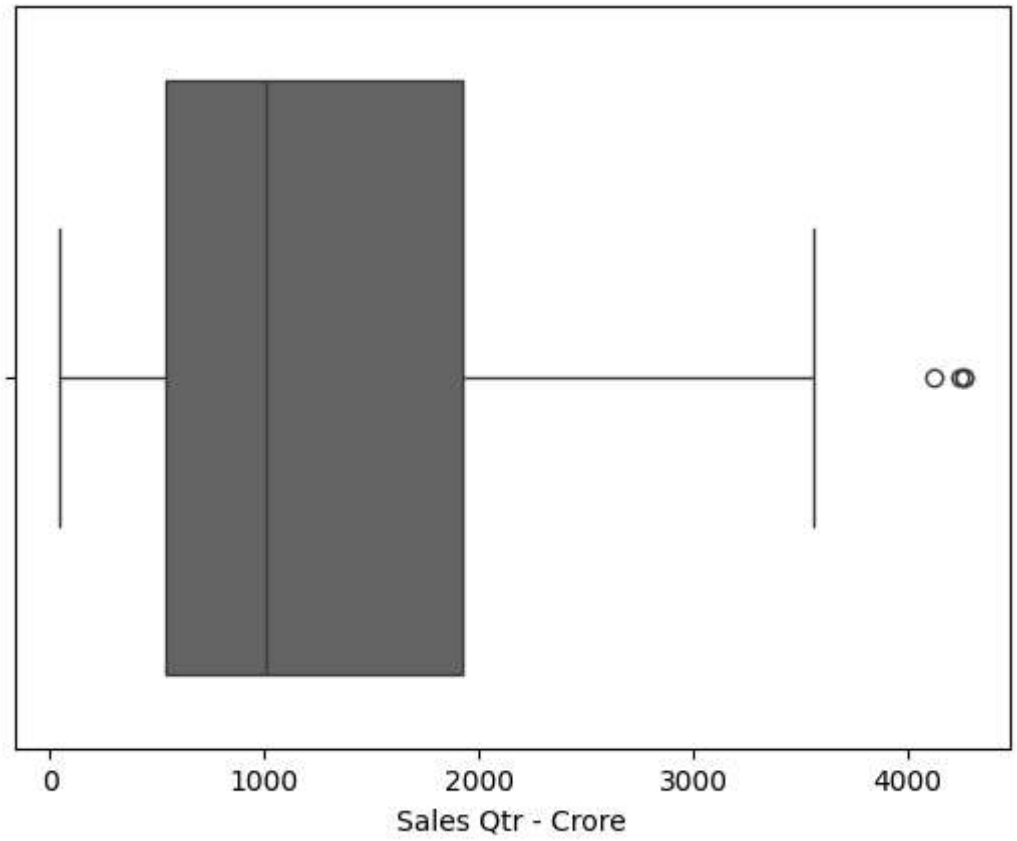
<Axes: xlabel='Mar Cap - Crore'>




```
#Now I will use IQR method to eliminate the outliers in 'Sales Qtr - Crore' column
#calculating first and third quartiles
Q1 = df1['Sales Qtr - Crore'].quantile(0.25)
Q3 = df1['Sales Qtr - Crore'].quantile(0.75)

#Calculating IQR
IQR = Q3-Q1

#defining lower and upper limits
lower = Q1-1.5*IQR
upper = Q3+1.5*IQR

#filtering the dataset to remove the outliers
df1 = df1[(df1['Sales Qtr - Crore'] >= lower) & (df1['Sales Qtr - Crore'] <= upper)]
```

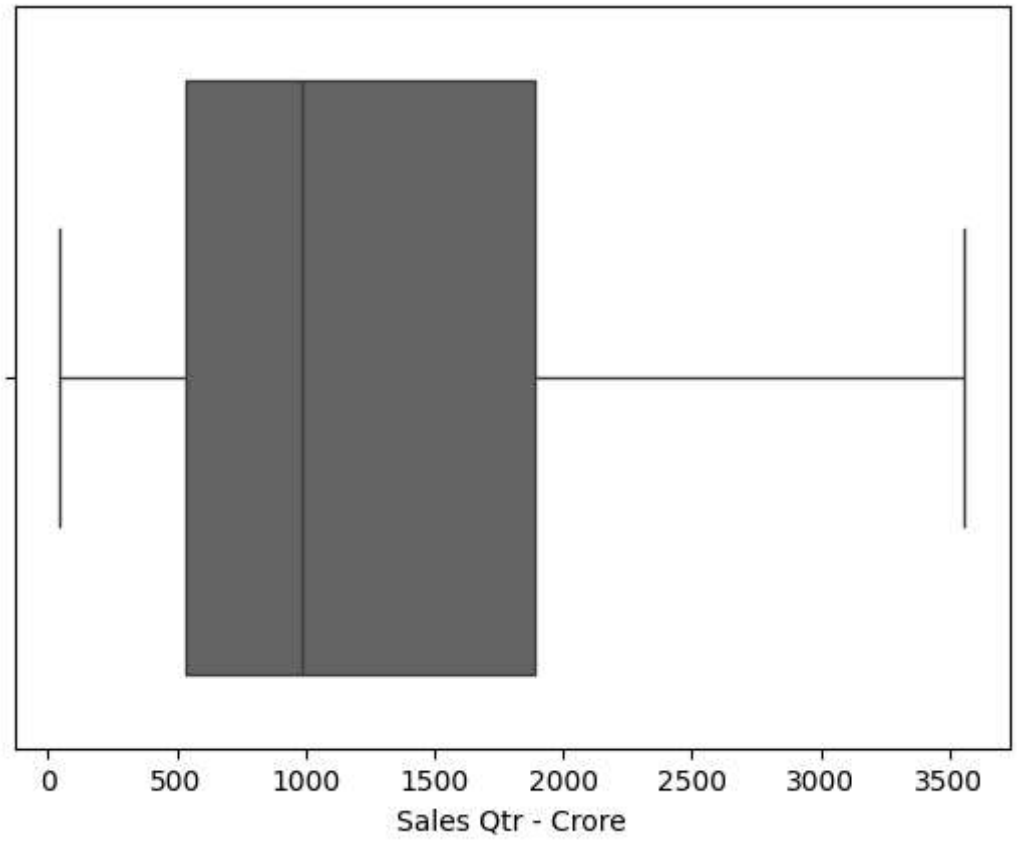
```
#checking if we still have the outliers using boxplot
sns.boxplot(data=df1, x='Sales Qtr - Crore')
```

<Axes: xlabel='Sales Qtr - Crore'>




```
#Now I will repeat IQR method to eliminate the outliers in 'Sales Qtr - Crore' column
#calculating first and third quartiles
Q1 = df1['Sales Qtr - Crore'].quantile(0.25)
Q3 = df1['Sales Qtr - Crore'].quantile(0.75)

#Calculating IQR
IQR = Q3-Q1

#defining lower and upper limits
lower = Q1-1.5*IQR
upper = Q3+1.5*IQR

#filtering the dataset to remove the outliers
df1 = df1[(df1['Sales Qtr - Crore'] >= lower) & (df1['Sales Qtr - Crore'] <= upper)]



#checking if we still have the outliers using boxplot
sns.boxplot(data=df1, x='Sales Qtr - Crore')
```

```
<Axes: xlabel='Sales Qtr - Crore'>
```




```
df1.describe()
```

	S.No.	Mar Cap - Crore	Sales Qtr - Crore	Unnamed: 4
count	331.000000	331.000000	331.000000	57.000000
mean	320.966767	8021.898550	1234.332157	454.456842
std	106.370004	4720.301442	840.960476	265.432660
min	130.000000	3017.070000	47.240000	0.000000
25%	229.500000	4196.320000	535.980000	185.650000
50%	323.000000	6469.510000	987.640000	473.420000
75%	411.500000	10909.750000	1892.885000	656.780000
max	499.000000	21976.740000	3557.940000	969.100000

```
#I can see that there is some zero values in 'Sales Qtr - Crore' column. Lets replace it wi
df1['Sales Qtr - Crore'].replace(0, np.nan, inplace=True)
```

```
#checking how many null values are there
df1['Sales Qtr - Crore'].isna().sum()
```

```
0
```

```
#applying KNN imputation method on Sales Qtr - Crore
data = df1[['Sales Qtr - Crore']]
imputer = KNNImputer(n_neighbors=5)
imputed_data = imputer.fit_transform(data)
df1['Sales Qtr - Crore'] = imputed_data
```

```
df1.isna().sum()
```

	0
S.No.	0
Name	0
Mar Cap - Crore	0
Sales Qtr - Crore	0
Unnamed: 4	274

dtype: int64

```
#lets check for the outliers again for the column 'Sales Qtr - Crore'
sns.boxplot(data=df1, x='Sales Qtr - Crore')
```

<Axes: xlabel='Sales Qtr - Crore'>

```
# we can see there is a outlier. So lets remove it using IQR method
#calculating first and third quartiles
Q1 = df1['Sales Qtr - Crore'].quantile(0.25)
Q3 = df1['Sales Qtr - Crore'].quantile(0.75)

#Calculating IQR
IQR = Q3-Q1

#defining lower and upper limits
lower = Q1-1.5*IQR
upper = Q3+1.5*IQR

#filtering the dataset to remove the outliers
df1 = df1[(df1['Sales Qtr - Crore'] >= lower) & (df1['Sales Qtr - Crore'] <= upper)]



sns.boxplot(data=df1, x='Sales Qtr - Crore')
```