

In [9]: *# Importing the Necessary Libraries*

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
```

In [10]: *# 1.Load a dataset in your IDE*

```
dataset = pd.read_csv("/content/googleplaystore_v2.csv")
```

In [11]: *# 2.Observe the statistics of all the features*

```
dataset.describe()
```

Out[11]:

	Rating	Size
count	9367.000000	10841.000000
mean	4.193338	21516.529524
std	0.537431	20746.537567
min	1.000000	8.500000
25%	4.000000	5900.000000
50%	4.300000	18000.000000
75%	4.500000	26000.000000
max	19.000000	100000.000000

In [12]: *# 3.Obtain the shape of the dataset*

```
dataset.shape
```

Out[12]: (10841, 13)

In [13]: *# 4.Separate all the features*

```
dataset.info()
```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10841 entries, 0 to 10840
Data columns (total 13 columns):
#   Column                Non-Null Count  Dtype
---  -
0   App                    10841 non-null  object
1   Category               10841 non-null  object
2   Rating                 9367 non-null   float64
3   Reviews                10841 non-null  object
4   Size                   10841 non-null  float64
5   Installs               10841 non-null  object
6   Type                   10840 non-null  object
7   Price                  10841 non-null  object
8   Content Rating         10840 non-null  object
9   Genres                 10841 non-null  object
10  Last Updated           10841 non-null  object
11  Current Ver            10833 non-null  object
12  Android Ver            10838 non-null  object
dtypes: float64(2), object(11)
memory usage: 1.1+ MB

```

In [14]: *# 5.Fill the missing values, if any, using the statistically relevant value*

Remove the Observation Having the NULL Values

```
dataset = dataset.drop(dataset[dataset['Rating'].isnull()].index)
```

Replacing the NULL Values with DUMMY Values

```

dataset['Android Ver'].fillna(value = '4.1 and up' , inplace = True)
dataset['Current Ver'].fillna(value = 'Varies with device' , inplace = True)
dataset['Content Rating'].fillna(value = 'Everyone' , inplace = True)
dataset['Type'].fillna(value = 'Free' , inplace = True)

```

In [17]: *# 6.Observe the Box-Plot of each feature*

```

sns.set_style('darkgrid')
fig, axes = plt.subplots(1, 2, figsize=(12, 6))
colors = ['#FFA07A', '#6495ED']

# Box plot for Rating
axes[0].boxplot(dataset['Rating'], patch_artist=True, boxprops=dict(facecolor=colors[0], medianprops=dict(color='black', linewidth=2), whiskerprops=dict(
axes[0].set_title('Rating', fontsize=14, fontweight='bold', color=colors[0])
axes[0].set_ylabel('Values', fontsize=12)

# Box plot for Size
axes[1].boxplot(dataset['Size'], patch_artist=True, boxprops=dict(facecolor=colors[1], medianprops=dict(color='black', linewidth=2), whiskerprops=dict(
axes[1].set_title('Size', fontsize=14, fontweight='bold', color=colors[1])
axes[1].set_ylabel('Values', fontsize=12)

plt.tight_layout()
fig.suptitle('Box Plots for Rating and Size', fontsize=16, fontweight='bold', color=colors[0])
plt.show()

```

