[1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

[2]:
```
rollno = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]
name = ["Piyush", "Shishir", "Pratik", "Gunjan", "Kaushal", "Aditya", "Dhruv", "Ajay", "Sanyam", "Dnyaneshwar", np.nan, np.nan, "Sumedh", "Prasad", "Shantanu"]
marks = [40, 23, 50, 78, 48, 89, 90, 67, 84, 96, 76, np.nan, 97, np.nan, 65]
grade = ["F", "F", "P", "P", "P", "P", "P", "P", "P", "P", "P", "F", "P", np.nan, np.nan]
```

[5]:
```
df=pd.DataFrame({"Rollno":rollno ,"Name":name,"Marks":marks,"Grade":grade})
```

[6]:
```
df
```

[6]:

	Rollno	Name	Marks	Grade
0	1	Piyush	40.0	F
1	2	Shishir	23.0	F
2	3	Pratik	50.0	P
3	4	Gunjan	78.0	P
4	5	Kaushal	48.0	P
5	6	Aditya	89.0	P
6	7	Dhruv	90.0	P
7	8	Ajay	67.0	P
8	9	Sanyam	84.0	P
9	10	Dnyaneshwar	96.0	P
10	11	NaN	76.0	P
11	12	NaN	NaN	F

10	11	NaN	76.0	P
11	12	NaN	NaN	F
12	13	Sumedh	97.0	P
13	14	Prasad	NaN	NaN
14	15	Shantanu	65.0	NaN

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15 entries, 0 to 14
Data columns (total 4 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Rollno  15 non-null     int64
 1   Name    13 non-null     object
 2   Marks   13 non-null     float64
 3   Grade   13 non-null     object
dtypes: float64(1), int64(1), object(2)
memory usage: 608.0+ bytes
```

[8]:
```
df.describe()
```

[8]:

	Rollno	Marks
count	15.000000	13.000000
mean	8.000000	69.461538
std	4.472136	23.247277
min	1.000000	23.000000
25%	4.500000	50.000000
50%	8.000000	76.000000
75%	11.500000	89.000000
max	15.000000	97.000000

+ Code + Markdown

[9]:
```
df.dtypes
```

[9]:
```
Rollno      int64
Name       object
Marks     float64
Grade      object
dtype: object
```

[10]:
```
df.columns
```

[10…
```
Index(['Rollno', 'Name', 'Marks', 'Grade'], dtype='object')
```

[11]:
```
df.isna().sum()
```

[11…
```
Rollno    0
Name      2
Marks     2
Grade     2
dtype: int64
```

[12]:
```
df.to_csv("Academic_Performance.csv")
```

[13]:
```
df["Marks"] = df["Marks"].fillna(df["Marks"].mean())
```

[14]:
```
df
```

[12]:
```
df.to_csv("Academic_Performance.csv")
```

[13]:
```
df["Marks"] = df["Marks"].fillna(df["Marks"].mean())
```

[14]:
```
df
```

[14...

	RollNo	Name	Marks	Grade
0	1	Piyush	40.000000	F
1	2	Shishir	23.000000	F
2	3	Pratik	50.000000	P
3	4	Gunjan	78.000000	P
4	5	Kaushal	48.000000	P
5	6	Aditya	89.000000	P
6	7	Dhruv	90.000000	P
7	8	Ajay	67.000000	P
8	9	Sanyam	84.000000	P
9	10	Dnyaneshwar	96.000000	P
10	11	NaN	76.000000	P
11	12	NaN	69.461538	F
12	13	Sumedh	97.000000	P
13	14	Prasad	69.461538	NaN
14	15	Shantanu	65.000000	NaN

[16]:
```
df = df[df['Name'].notna()]
```

[17]:
```
df
```

[17...

	RollNo	Name	Marks	Grade
0	1	Piyush	40.000000	F
1	2	Shishir	23.000000	F
2	3	Pratik	50.000000	P
3	4	Gunjan	78.000000	P
4	5	Kaushal	48.000000	P
5	6	Aditya	89.000000	P
6	7	Dhruv	90.000000	P
7	8	Ajay	67.000000	P
8	9	Sanyam	84.000000	P
9	10	Dnyaneshwar	96.000000	P
12	13	Sumedh	97.000000	P
13	14	Prasad	69.461538	NaN
14	15	Shantanu	65.000000	NaN

+ Code + Markdown

[19]:
```
for index, row in df.iterrows():
    if (row['Marks'] > 40):
        df.loc[index, 'Grade'] = 'P'
    else:
        df.loc[index, 'Grade'] = 'F'
```

[20]:
```
df
```

[20...

	RollNo	Name	Marks	Grade
0	1	Piyush	40.000000	F
1	2	Shishir	23.000000	F
2	3	Pratik	50.000000	P
3	4	Gunjan	78.000000	P
4	5	Kaushal	48.000000	P
5	6	Aditya	89.000000	P
6	7	Dhruv	90.000000	P
7	8	Ajay	67.000000	P
8	9	Sanyam	84.000000	P
9	10	Dnyaneshwar	96.000000	P
12	13	Sumedh	97.000000	P
13	14	Prasad	69.461538	P
14	15	Shantanu	65.000000	P

[22]:
```
first_outlier = [16, 'Ganesh', 200, 'P']
second_outlier = [17, 'Dinesh', -100, 'F']
```

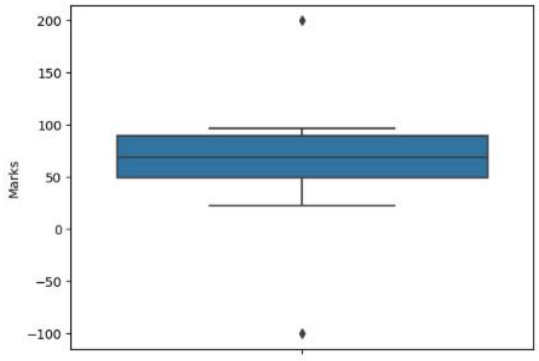
[23]:
```
df.loc[15] = first_outlier
df.loc[16] = second_outlier
```

[24]:
```
df
```

[24...

	Rollno	Name	Marks	Grade
0	1	Piyush	40.000000	F
1	2	Shishir	23.000000	F
2	3	Pratik	50.000000	P
3	4	Gunjan	78.000000	P
4	5	Kaushal	48.000000	P
5	6	Aditya	89.000000	P
6	7	Dhruv	90.000000	P
7	8	Ajay	67.000000	P
8	9	Sanyam	84.000000	P
9	10	Dnyaneshwar	96.000000	P
12	13	Sumedh	97.000000	P
13	14	Prasad	69.461538	P
14	15	Shantanu	65.000000	P
15	16	Ganesh	200.000000	P
16	17	Dinesh	-100.000000	F

```
sns.boxplot(data=df, y='Marks');
```



+ Code + Markdown

[27]:
```
df = df.drop([15,16], axis=0)
```

[28]:
```
df
```

[28...

	RollNo	Name	Marks	Grade
0	1	Piyush	40.000000	F
1	2	Shishir	23.000000	F
2	3	Pratik	50.000000	P
3	4	Gunjan	78.000000	P
4	5	Kaushal	48.000000	P
5	6	Aditya	89.000000	P
6	7	Dhruv	90.000000	P
7	8	Ajay	67.000000	P
8	9	Sanyam	84.000000	P
9	10	Dnyaneshwar	96.000000	P
12	13	Sumedh	97.000000	P
13	14	Prasad	69.461538	P
14	15	Shantanu	65.000000	P

+ Code + Markdown

[29]:
```
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()
```

[30]:
```
df[['Marks']] = scaler.fit_transform(df[['Marks']])
```

[30]:
```
df[['Marks']] = scaler.fit_transform(df[['Marks']])
```

[31]:
```
df
```

[31…

	RollNo	Name	Marks	Grade
0	1	Piyush	0.229730	F
1	2	Shishir	0.000000	F
2	3	Pratik	0.364865	P
3	4	Gunjan	0.743243	P
4	5	Kaushal	0.337838	P
5	6	Aditya	0.891892	P
6	7	Dhruv	0.905405	P
7	8	Ajay	0.594595	P
8	9	Sanyam	0.824324	P
9	10	Dnyaneshwar	0.986486	P
12	13	Sumedh	1.000000	P
13	14	Prasad	0.627859	P
14	15	Shantanu	0.567568	P

[ ]:
```

```

+ Code + Markdown