Descriptive Statistics

Aim of the Experiment The main aim of this experiment is to explore the given dataset. A sample database is created and is available in the file Descriptive_statistics.csv.

```
import pandas as pd

df = pd.read_csv('Descriptive_statistics.csv')

display(df.head())
```

	id	first	last	gender	Marks	selected
0	1	Leone	Debrick	Female	50	True
1	2	Romola	Phinnessy	Female	60	False
2	3	Geri	Prium	Male	65	False
3	4	Sandy	Doveston	Female	95	False
4	5	Jacenta	Jansik	Female	31	True

description

```
print(df.describe())
```

```
             id      Marks
count  10.00000  10.000000
mean    5.50000  54.700000
std     3.02765  18.666964
min     1.00000  31.000000
25%     3.25000  45.000000
50%     5.50000  50.000000
75%     7.75000  63.750000
max    10.00000  95.000000
```

statistical operations

```
numeric_df = df.select_dtypes(include=['number'])

print ( numeric_df.mean())

print(numeric_df.sum())
print(numeric_df.median())
print(numeric_df.std())
print(numeric_df.mode())
print(numeric_df.min())
print(numeric_df.max())
print(numeric_df.count())
print(numeric_df.var())
print(numeric_df.quantile())
```

```
id        5.5
Marks    54.7
dtype: float64
id        5.5
Marks    50.0
dtype: float64
id        3.027650
Marks    18.666964
dtype: float64
   id  Marks
0   1   45.0
1   2   50.0
2   3    NaN
3   4    NaN
4   5    NaN
5   6    NaN
6   7    NaN
```

```
7   8    NaN
8   9    NaN
9  10    NaN
id        1
Marks    31
dtype: int64
id       10
Marks    95
dtype: int64
id       10
Marks    10
dtype: int64
id         9.166667
Marks    348.455556
dtype: float64
id        5.5
Marks    50.0
Name: 0.5, dtype: float64
```

Univariate Descriptive Statistics

Let's explore the descriptive statistics for a single variable, for example, the 'Marks' column.

```
print("Univariate Statistics for 'Marks':")
print(df['Marks'].describe())
```

```
Univariate Statistics for 'Marks':
count    10.000000
mean     54.700000
std      18.666964
min      31.000000
25%      45.000000
50%      50.000000
75%      63.750000
max      95.000000
Name: Marks, dtype: float64
```

Bivariate Descriptive Statistics

Now, let's explore the relationship between a numeric variable ('Marks') and a categorical variable ('gender'). We can use `groupby()` to achieve this.

```
print("\nBivariate Statistics for 'Marks' grouped by 'gender':")
print(df.groupby('gender')['Marks'].describe())
```

```

Bivariate Statistics for 'Marks' grouped by 'gender':
        count       mean        std   min    25%   50%    75%   max
gender
Female    6.0  55.166667  21.683327  31.0  46.25  50.0  57.50  95.0
Male      4.0  54.000000  16.145175  36.0  42.75  55.0  66.25  70.0
```