In [1]:
```
import pandas as pd
import numpy as np
import statistics as st
import warnings
warnings.filterwarnings("ignore")
# Load the data
df = pd.read_csv("iris.csv")
print(df.shape)
print(df.info())
```

```
(150, 6)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 6 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Id             150 non-null    int64
 1   SepalLengthCm  150 non-null    float64
 2   SepalWidthCm   150 non-null    float64
 3   PetalLengthCm  150 non-null    float64
 4   PetalWidthCm   150 non-null    float64
 5   Species        150 non-null    object
dtypes: float64(4), int64(1), object(1)
memory usage: 7.2+ KB
None
```

In [2]:
```
df.mean()
```

Out[2]:
```
Id               75.500000
SepalLengthCm     5.843333
SepalWidthCm      3.054000
PetalLengthCm     3.758667
PetalWidthCm      1.198667
dtype: float64
```

In [3]:
```
print(df.loc[:,'SepalLengthCm'].mean())
print(df.loc[:,'PetalLengthCm'].mean())
```

```
5.843333333333335
3.7586666666666693
```

In [4]:
```
df.mean(axis = 1)[0:10]
```

Out[4]:
```
0    2.24
1    2.30
2    2.48
3    2.68
4    3.04
5    3.48
6    3.34
7    3.62
8    3.58
9    3.92
dtype: float64
```

In [5]:
```
df.median(axis=1)
```

Out[5]:
```
0      1.4
1      2.0
```

```
2      3.0
3      3.1
4      3.6
      ...
145    5.2
146    5.0
147    5.2
148    5.4
149    5.1
Length: 150, dtype: float64
```

In [6]:
```
df.median()
```

Out[6]:
```
Id               75.50
SepalLengthCm     5.80
SepalWidthCm      3.00
PetalLengthCm     4.35
PetalWidthCm      1.30
dtype: float64
```

In [7]:
```
#to compute a median of a some column
print(df.loc[:,'SepalLengthCm'].median())
print(df.loc[:,'PetalLengthCm'].median())

df.median(axis = 1)[0:10]
```

Out[7]:
```
5.8
4.35
0    1.4
1    2.0
2    3.0
3    3.1
4    3.6
5    3.9
6    3.4
7    3.4
8    2.9
9    3.1
dtype: float64
```

In [8]:
```
df.mode()
```

Out[8]:

| | Id | SepalLengthCm | SepalWidthCm | PetalLengthCm | PetalWidthCm | Species |
|---|---|---|---|---|---|---|
| **0** | 1 | 5.0 | 3.0 | 1.5 | 0.2 | Iris-setosa |
| **1** | 2 | NaN | NaN | NaN | NaN | Iris-versicolor |
| **2** | 3 | NaN | NaN | NaN | NaN | Iris-virginica |
| **3** | 4 | NaN | NaN | NaN | NaN | NaN |
| **4** | 5 | NaN | NaN | NaN | NaN | NaN |
| **...** | ... | ... | ... | ... | ... | ... |
| **145** | 146 | NaN | NaN | NaN | NaN | NaN |
| **146** | 147 | NaN | NaN | NaN | NaN | NaN |
| **147** | 148 | NaN | NaN | NaN | NaN | NaN |
| **148** | 149 | NaN | NaN | NaN | NaN | NaN |
| **149** | 150 | NaN | NaN | NaN | NaN | NaN |

| | Id | SepalLengthCm | SepalWidthCm | PetalLengthCm | PetalWidthCm | Species |
|---|---|---|---|---|---|---|

150 rows × 6 columns

In [9]:
```
#Measure the Standard deviation
df.std()
```

Out[9]:
```
Id               43.445368
SepalLengthCm     0.828066
SepalWidthCm      0.433594
PetalLengthCm     1.764420
PetalWidthCm      0.763161
dtype: float64
```

In [10]:
```
print(df.loc[:,'SepalLengthCm'].std())
print(df.loc[:,'PetalLengthCm'].std())

#calculate the standard deviation of the first five rows
df.std(axis = 1)[0:10]
```

Out[10]:
```
0.8280661279778629
1.7644204199522617
0    2.010721
1    1.772005
2    1.754138
3    1.813009
4    2.165179
5    2.391025
6    2.645373
7    3.054832
8    3.416431
9    3.842135
dtype: float64
```

In [11]:
```
#easure the Variance
df.var()
```

Out[11]:
```
Id               1887.500000
SepalLengthCm       0.685694
SepalWidthCm        0.188004
PetalLengthCm       3.113179
PetalWidthCm        0.582414
dtype: float64
```

In [12]:
```
#Measures the Interquartile Range (IQR)
from scipy.stats import iqr
iqr(df['SepalLengthCm'])
```

Out[12]:
```
1.3000000000000007
```

In [13]:
```
print(df.skew())
```

```
Id               0.000000
SepalLengthCm    0.314911
SepalWidthCm     0.334053
PetalLengthCm   -0.274464
PetalWidthCm    -0.104997
dtype: float64
```

In [14]:
```
df.describe()
```

Out[14]:

| | Id | SepalLengthCm | SepalWidthCm | PetalLengthCm | PetalWidthCm |
|---|---|---|---|---|---|
| **count** | 150.000000 | 150.000000 | 150.000000 | 150.000000 | 150.000000 |
| **mean** | 75.500000 | 5.843333 | 3.054000 | 3.758667 | 1.198667 |
| **std** | 43.445368 | 0.828066 | 0.433594 | 1.764420 | 0.763161 |
| **min** | 1.000000 | 4.300000 | 2.000000 | 1.000000 | 0.100000 |
| **25%** | 38.250000 | 5.100000 | 2.800000 | 1.600000 | 0.300000 |
| **50%** | 75.500000 | 5.800000 | 3.000000 | 4.350000 | 1.300000 |
| **75%** | 112.750000 | 6.400000 | 3.300000 | 5.100000 | 1.800000 |
| **max** | 150.000000 | 7.900000 | 4.400000 | 6.900000 | 2.500000 |

In [15]:
```
df.describe(include='all')
```

Out[15]:

| | Id | SepalLengthCm | SepalWidthCm | PetalLengthCm | PetalWidthCm | Species |
|---|---|---|---|---|---|---|
| **count** | 150.000000 | 150.000000 | 150.000000 | 150.000000 | 150.000000 | 150 |
| **unique** | NaN | NaN | NaN | NaN | NaN | 3 |
| **top** | NaN | NaN | NaN | NaN | NaN | Iris-setosa |
| **freq** | NaN | NaN | NaN | NaN | NaN | 50 |
| **mean** | 75.500000 | 5.843333 | 3.054000 | 3.758667 | 1.198667 | NaN |
| **std** | 43.445368 | 0.828066 | 0.433594 | 1.764420 | 0.763161 | NaN |
| **min** | 1.000000 | 4.300000 | 2.000000 | 1.000000 | 0.100000 | NaN |
| **25%** | 38.250000 | 5.100000 | 2.800000 | 1.600000 | 0.300000 | NaN |
| **50%** | 75.500000 | 5.800000 | 3.000000 | 4.350000 | 1.300000 | NaN |
| **75%** | 112.750000 | 6.400000 | 3.300000 | 5.100000 | 1.800000 | NaN |
| **max** | 150.000000 | 7.900000 | 4.400000 | 6.900000 | 2.500000 | NaN |

In [16]:
```
df.groupby('SepalLengthCm')['PetalLengthCm'].sum()
```

Out[16]:
```
SepalLengthCm
4.3     1.1
4.4     4.0
4.5     1.3
4.6     5.3
4.7     2.9
4.8     7.9
4.9    13.7
5.0    18.4
5.1    15.5
5.2     8.3
5.3     1.5
5.4    12.2
5.5    22.6
5.6    25.2
```

```
5.7    28.7
5.8    28.5
5.9    14.1
6.0    27.9
6.1    28.5
6.2    19.0
6.3    46.2
6.4    36.1
6.5    26.2
6.6     9.0
6.7    42.1
6.8    16.2
6.9    21.1
7.0     4.7
7.1     5.9
7.2    17.9
7.3     6.3
7.4     6.1
7.6     6.6
7.7    26.4
7.9     6.4
Name: PetalLengthCm, dtype: float64
```

In [17]:

```
print('Iris-setosa')
```

```
Iris-setosa
```

In [19]:

```
setosa = df['Species'] == 'Iris-setosa'
print(df[setosa].describe())
```

```
             Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm
count  50.00000       50.00000     50.000000      50.000000      50.00000
mean   25.50000        5.00600      3.418000       1.464000       0.24400
std    14.57738        0.35249      0.381024       0.173511       0.10721
min     1.00000        4.30000      2.300000       1.000000       0.10000
25%    13.25000        4.80000      3.125000       1.400000       0.20000
50%    25.50000        5.00000      3.400000       1.500000       0.20000
75%    37.75000        5.20000      3.675000       1.575000       0.30000
max    50.00000        5.80000      4.400000       1.900000       0.60000
```

In [20]:

```
setosa = df['Species'] == 'Iris-versicolor'
print(df[setosa].describe())
print('\nIris-virginica')
setosa = df['Species'] == 'Iris-virginica'
print(df[setosa].describe())
```

```
             Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm
count   50.00000      50.000000     50.000000      50.000000     50.000000
mean    75.50000       5.936000      2.770000       4.260000      1.326000
std     14.57738       0.516171      0.313798       0.469911      0.197753
min     51.00000       4.900000      2.000000       3.000000      1.000000
25%     63.25000       5.600000      2.525000       4.000000      1.200000
50%     75.50000       5.900000      2.800000       4.350000      1.300000
75%     87.75000       6.300000      3.000000       4.600000      1.500000
max    100.00000       7.000000      3.400000       5.100000      1.800000

Iris-virginica
             Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm
count   50.00000       50.00000     50.000000      50.000000      50.00000
mean   125.50000        6.58800      2.974000       5.552000       2.02600
std     14.57738        0.63588      0.322497       0.551895       0.27465
min    101.00000        4.90000      2.200000       4.500000       1.40000
```

| | | | | | |
|---|---|---|---|---|---|
| 25% | 113.25000 | 6.22500 | 2.800000 | 5.100000 | 1.80000 |
| 50% | 125.50000 | 6.50000 | 3.000000 | 5.550000 | 2.00000 |
| 75% | 137.75000 | 6.90000 | 3.175000 | 5.875000 | 2.30000 |
| max | 150.00000 | 7.90000 | 3.800000 | 6.900000 | 2.50000 |

In [ ]: