In [22]:
```
# This Python 3 environment comes with many helpful analytics libraries instal
led
# It is defined by the kaggle/python Docker image: https://github.com/kaggle/d
ocker-python
# For example, here's several helpful packages to load

import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)

# Input data files are available in the read-only "../input/" directory
# For example, running this (by clicking run or pressing Shift+Enter) will lis
t all files under the input directory

import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))

# You can write up to 20GB to the current directory (/kaggle/working/) that ge
ts preserved as output when you create a version using "Save & Run All"
# You can also write temporary files to /kaggle/temp/, but they won't be saved
outside of the current session
```

```
/kaggle/input/weight-and-heightcsv/weight-height.csv
```

In [23]:
```
import pandas as pd
import numpy as np
from scipy import stats
```

In [24]:
```
data = pd.read_csv('/kaggle/input/weight-and-heightcsv/weight-height.csv')
data.head()
```

Out[24]:

	Gender	Height	Weight
0	Male	73.847017	241.893563
1	Male	68.781904	162.310473
2	Male	74.110105	212.740856
3	Male	71.730978	220.042470
4	Male	69.881796	206.349801

In [25]:
```
data.describe()
```

Out[25]:

	Height	Weight
count	10000.000000	10000.000000
mean	66.367560	161.440357
std	3.847528	32.108439
min	54.263133	64.700127
25%	63.505620	135.818051
50%	66.318070	161.212928
75%	69.174262	187.169525
max	78.998742	269.989699

In [26]:
```
data
```

Out[26]:

	Gender	Height	Weight
0	Male	73.847017	241.893563
1	Male	68.781904	162.310473
2	Male	74.110105	212.740856
3	Male	71.730978	220.042470
4	Male	69.881796	206.349801
...	...	...	...
9995	Female	66.172652	136.777454
9996	Female	67.067155	170.867906
9997	Female	63.867992	128.475319
9998	Female	69.034243	163.852461
9999	Female	61.944246	113.649103

10000 rows × 3 columns

Mean

In [27]:
```
height_mean = np.mean(data['Height'])
weight_mean = np.mean(data['Weight'])
print("Mean of Height: ",height_mean)
print("Mean of weight: ",weight_mean)
```

```
Mean of Height:  66.36755975482124
Mean of weight:  161.44035683283076
```

Median

In [28]:
```
height_med = np.median(data['Height'])
weight_med = np.median(data['Weight'])
print("Median of Height: ",height_med)
print("Median of weight: ",weight_med)
```

```
Median of Height:  66.31807008178464
Median of weight:  161.21292769948298
```

Standard Deviation

In [29]:
```
height_std = np.std(data['Height'])
weight_std = np.std(data['Weight'])
print("Standard Deviation of Height: ",height_std)
print("Standard Deviation of weight: ",weight_std)
```

```
Standard Deviation of Height:  3.84733573955754
Standard Deviation of weight:  32.106833544431716
```

skewness and kurtosis

In [30]:
```
h = data['Height']
w = data['Weight']
```

In [18]:
```
print('Height skewness', stats.skew(h))
```

```
Height skewness 0.04936168370490536
```

In [19]:
```
print('Height kurtosis', stats.kurtosis(h))
```

```
Height kurtosis -0.4744964699248557
```

In [20]:
```
print('Weight skewness', stats.skew(w))
```

```
Weight skewness 0.032949561064277405
```

In [21]:
```
print('Weight kurtosis', stats.kurtosis(w))
```

```
Weight kurtosis -0.794820272056854
```

In [ ]:
```

```