```
# Measures of Dispersion or Measures of Spread (2nd Business Moment)
```

```
# Range, IQR, Variance, Std.deviation (all Applied on )
```

```
import numpy as np
import pandas as pd
```

```
df = pd.DataFrame({"x":[1,2,3,4,5]})
df
```

	x
0	1
1	2
2	3
3	4

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
# Minimum
```

```
df["x"].min()
```

```
1
```

```
# Maximum
```

```
df["x"].max()
```

```
5
```

```
# Range = Maximum value - Minimum value
```

```
df["x"].max() - df["x"].min()
```

```
4
```

```
# Deviation (x-μ)
# Deviation = Data deviated from the mean = how dispersed the data is from the central value.
```

```
df["x-μ"] = df["x"] - df["x"].mean()
df
```

	x	x-μ
0	1	-2.0
1	2	-1.0
2	3	0.0
3	4	1.0

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
# Mean deviation
```

```
df["x-μ"].mean()
```

```
np.float64(0.0)
```

```
# for any data set mean deviation is always zero.
```

```
# Population varience = how far the data is from the mean
```

```
df["x"].var(ddof=0)
```

```
2.0
```

```
# Population Standered deviation
df["x"].std(ddof=0)
```

```
1.4142135623730951
```

```
# Sample variance
```

```
df["x"].var(ddof=1)
```

```
2.5
```

```
# Sample Std. deviation
```

```
df["x"].std(ddof=1)
```

```
1.5811388300841898
```

```
# Coefficient of variation
```

```
df["x"].std(ddof=0)/df["x"].mean()
```

```
np.float64(0.47140452079103173)
```

```
# Percentile = describe the percentage of data value that fall at or below the value.
```

```
df=pd.DataFrame({"x":[1,2,3,4,5,6,7,8,9,10]})
df
```

	x
0	1
1	2
2	3
3	4
4	5
5	6
6	7
7	8
8	9

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
# 0 percentile or minimum
df["x"].quantile(0)
```

```
np.float64(1.0)
```

```
#25 percentile (Q1)
q1 = df["x"].quantile(0.25)
q1
```

```
np.float64(3.25)
```

```
# 50 percentile
q2 = df["x"].quantile(0.5)
q2
```

```
np.float64(5.5)
```

```
# 75 percentile
q3 = df["x"].quantile(0.75)
q3
```

```
np.float64(7.75)
```

```
# Inter Quartile Range (IQR) = q3-q1
```

```
IQR = q3-q1
IQR
```

```
np.float64(4.5)
```

```
# Lower limit
ll = q1-1.5*IQR
ll
```

```
np.float64(-3.5)
```

```
# upper limit q3 + (1.5* IQR)
```

```
ul = q3 + 1.5*IQR
ul
```

```
np.float64(14.5)
```