

```
In [2]: import pandas as pd
import numpy as np
import statistics as st
```

```
In [4]: # Load the data
data = pd.read_csv("phase2.csv")
df = pd.DataFrame(data)
```

```
In [5]: # Print data
print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100 entries, 0 to 99
Data columns (total 19 columns):
Q1      100 non-null int64
Q2      100 non-null int64
Q3      100 non-null int64
Q4      99 non-null float64
Q5      100 non-null int64
Q6      100 non-null int64
Q7      100 non-null int64
Q8      100 non-null int64
Q9      100 non-null int64
Q10     100 non-null int64
Q11     100 non-null int64
Q12     100 non-null int64
Q13     100 non-null object
Q14     100 non-null object
Q15     100 non-null int64
Q16     100 non-null int64
Q17     100 non-null int64
Q18     100 non-null int64
Q19     100 non-null int64
dtypes: float64(1), int64(16), object(2)
memory usage: 15.0+ KB
None
```

```
In [6]: # Get mean
print("\nThe mean of the following: ")
print(df.mean())
```

The mean of the following:

Q1	2.390000
Q2	1.560000
Q3	3.180000
Q4	2.616162
Q5	1.800000
Q6	1.050000
Q7	1.100000
Q8	1.630000
Q9	2.050000
Q10	1.140000
Q11	1.140000
Q12	1.710000
Q15	1.040000
Q16	1.750000
Q17	1.170000
Q18	1.110000
Q19	2.500000
dtype:	float64

```
In [7]: # Get median
print("\nThe median of the following: ")
print(df.median())
```

The median of the following:

Q1	2.0
Q2	2.0
Q3	3.0
Q4	2.0
Q5	2.0
Q6	1.0
Q7	1.0
Q8	2.0
Q9	2.0
Q10	1.0
Q11	1.0
Q12	2.0
Q15	1.0
Q16	2.0
Q17	1.0
Q18	1.0
Q19	3.0
dtype:	float64

```
In [8]: # Get standard deviation
print("\nThe standard deviation of the following: ")
print(df.std())
```

The standard deviation of the following:

Q1	1.483206
Q2	0.498888
Q3	1.166710
Q4	1.201250
Q5	0.402015
Q6	0.219043
Q7	0.301511
Q8	0.613896
Q9	0.729951
Q10	0.348735
Q11	0.348735
Q12	0.456048
Q15	0.196946
Q16	0.435194
Q17	0.377525
Q18	0.314466
Q19	0.643538
dtype:	float64

```
In [9]: # Get variance
print("\nThe variance of the following: ")
print(df.var())
```

The variance of the following:

Q1	2.199899
Q2	0.248889
Q3	1.361212
Q4	1.443001
Q5	0.161616
Q6	0.047980
Q7	0.090909
Q8	0.376869
Q9	0.532828
Q10	0.121616
Q11	0.121616
Q12	0.207980
Q15	0.038788
Q16	0.189394
Q17	0.142525
Q18	0.098889
Q19	0.414141
dtype:	float64

```
In [10]: # Get skewness
print("\nThe skewness of the following: ")
print(df.skew())
```

The skewness of the following:

Q1	0.379869
Q2	-0.245444
Q3	-0.164547
Q4	1.468615
Q5	-1.522940
Q6	4.192637
Q7	2.707449
Q8	0.422313
Q9	0.081490
Q10	2.106739
Q11	2.106739
Q12	-0.939751
Q15	4.766655
Q16	-1.172360
Q17	1.783906
Q18	2.531015
Q19	-0.928169
dtype:	float64

```
In [11]: #Get correlation
print("\nThe correlation of the following: ")
print(df.corr())
```

The correlation of the following:

	Q1	Q2	Q3	Q4	Q5	Q6
\						
Q1	1.000000	0.193296	-0.017628	0.035554	-4.099552e-01	-9.171835e-02
Q2	0.193296	1.000000	-0.070804	-0.213220	-3.424748e-01	1.109211e-01
Q3	-0.017628	-0.070804	1.000000	0.122503	1.292144e-02	3.952507e-03
Q4	0.035554	-0.213220	0.122503	1.000000	1.962498e-01	-1.189014e-01
Q5	-0.409955	-0.342475	0.012921	0.196250	1.000000e+00	-2.507229e-17
Q6	-0.091718	0.110921	0.003953	-0.118901	-2.507229e-17	1.000000e+00
Q7	-0.178438	0.094013	0.005743	-0.133631	8.333333e-02	5.353034e-01
Q8	-0.083978	0.221634	-0.103515	-0.094375	1.882718e-01	4.394368e-01
Q9	0.140413	-0.188615	0.155374	-0.017106	3.442142e-02	-2.053172e-01
Q10	0.127716	0.067348	0.136046	-0.039445	5.763904e-02	5.686020e-01
Q11	-0.067568	0.299582	-0.087387	-0.039445	1.296878e-01	5.686020e-01
Q12	0.318226	0.143846	0.080113	-0.039580	-1.542659e-01	-3.589660e-01
Q15	0.222690	-0.024673	-0.031651	-0.105787	1.020621e-01	-4.682929e-02
Q16	0.105629	0.046524	-0.129310	0.046766	-1.154701e-01	1.324532e-01
Q17	-0.155678	0.025743	-0.001376	0.034181	1.597305e-01	3.847691e-01
Q18	0.015376	0.182855	-0.026981	-0.047809	9.588042e-02	5.059189e-01
Q19	-0.005291	-0.157311	0.067266	0.055223	-3.904344e-02	1.074862e-01

	Q7	Q8	Q9	Q10	Q11	Q12	Q
15 \							
Q1	-0.178438	-0.083978	0.140413	0.127716	-0.067568	0.318226	0.222690
Q2	0.094013	0.221634	-0.188615	0.067348	0.299582	0.143846	-0.024673
Q3	0.005743	-0.103515	0.155374	0.136046	-0.087387	0.080113	-0.031651
Q4	-0.133631	-0.094375	-0.017106	-0.039445	-0.039445	-0.039580	-0.105787
Q5	0.083333	0.188272	0.034421	0.057639	0.129688	-0.154266	0.102062
Q6	0.535303	0.439437	-0.205317	0.568602	0.568602	-0.358966	-0.046829
Q7	1.000000	0.147343	0.068843	0.249769	0.345834	-0.227726	-0.068041
Q8	0.147343	1.000000	-0.341498	0.291583	0.385947	-0.206735	0.123647
Q9	0.068843	-0.341498	1.000000	0.170626	0.130945	0.195713	-0.154577
Q10	0.249769	0.291583	0.170626	1.000000	0.335548	-0.250239	-0.082359
Q11	0.345834	0.385947	0.130945	0.335548	1.000000	-0.059702	-0.082359
Q12	-0.227726	-0.206735	0.195713	-0.250239	-0.059702	1.000000	0.130456
Q15	-0.068041	0.123647	-0.154577	-0.082359	-0.082359	0.130456	1.000000
Q16	0.038490	0.255206	-0.246428	-0.099834	0.166390	0.394433	0.117851
Q17	0.292839	0.404892	-0.104465	0.277736	0.124291	-0.532127	-0.092380
Q18	0.308948	0.526897	-0.288230	0.226585	0.226585	-0.197919	-0.071762
Q19	-0.052058	0.063920	-0.032254	0.045009	-0.045009	-0.017209	-0.159394

	Q16	Q17	Q18	Q19
Q1	0.105629	-0.155678	0.015376	-0.005291
Q2	0.046524	0.025743	0.182855	-0.157311
Q3	-0.129310	-0.001376	-0.026981	0.067266
Q4	0.046766	0.034181	-0.047809	0.055223
Q5	-0.115470	0.159730	0.095880	-0.039043
Q6	0.132453	0.384769	0.505919	0.107486
Q7	0.038490	0.292839	0.308948	-0.052058
Q8	0.255206	0.404892	0.526897	0.063920
Q9	-0.246428	-0.104465	-0.288230	-0.032254
Q10	-0.099834	0.277736	0.226585	0.045009
Q11	0.166390	0.124291	0.226585	-0.045009
Q12	0.394433	-0.532127	-0.197919	-0.017209
Q15	0.117851	-0.092380	-0.071762	-0.159394
Q16	1.000000	-0.230551	0.055357	0.162301
Q17	-0.230551	1.000000	0.521562	0.228669
Q18	0.055357	0.521562	1.000000	0.074870
Q19	0.162301	0.228669	0.074870	1.000000

```
In [12]: #Get range
print("\nThe range of the following: ")
print(df.max() - df.min())
```

The range of the following:

```
Q1      4.0
Q2      1.0
Q3      5.0
Q4      7.0
Q5      1.0
Q6      1.0
Q7      1.0
Q8      2.0
Q9      3.0
Q10     1.0
Q11     1.0
Q12     1.0
Q13     1.0
Q14     4.0
Q15     1.0
Q16     1.0
Q17     1.0
Q18     1.0
Q19     2.0
dtype: float64
```

```
In [16]: #Get quartiles and outliers for Age
sorted(df['Q1']) #Sort data
q1, q3= np.percentile(df['Q1'],[25,75]) #Find 1st quartile and 3rd quartile
iqr = q3 - q1 #Find interquartile range
lower_bound = q1 -(1.5 * iqr) #Calculate lower bound
upper_bound = q3 +(1.5 * iqr) #Calculate upper bound
print("\n25th quartile for Q1 is", q1)
print("75th quartile for Q1 is", q3)
print("Outlier range for Q1 is", lower_bound,"and", upper_bound)
```

```
25th quartile for Q1 is 1.0
75th quartile for Q1 is 4.0
Outlier range for Q1 is -3.5 and 8.5
```

In []: