In [2]:
```python
# Import Library
import pandas as pd
import numpy as np
import seaborn as sns
import plotly.express as px
import matplotlib.pyplot as plt
```

In [54]:
```python
epa_data = pd.read_csv(r'C:\Users\HP\Desktop\Advance Data Analyst\4. The Power of Stats\1. Module 1\3. Calculate stats with python\Files\c4_epa_air_quality.csv')
epa_data.head(10)
```

Out[54]:

	Unnamed: 0	date_local	state_name	county_name	city_name	local_site_name	parameter_name	units_of_measure	arithmetic_mean	aqi
0	0	2018-01-01	Arizona	Maricopa	Buckeye	BUCKEYE	Carbon monoxide	Parts per million	0.473684	7
1	1	2018-01-01	Ohio	Belmont	Shadyside	Shadyside	Carbon monoxide	Parts per million	0.263158	5
2	2	2018-01-01	Wyoming	Teton	Not in a city	Yellowstone National Park - Old Faithful Snow ...	Carbon monoxide	Parts per million	0.111111	2
3	3	2018-01-01	Pennsylvania	Philadelphia	Philadelphia	North East Waste (NEW)	Carbon monoxide	Parts per million	0.300000	3
4	4	2018-01-01	Iowa	Polk	Des Moines	CARPENTER	Carbon monoxide	Parts per million	0.215789	3
5	5	2018-01-01	Hawaii	Honolulu	Not in a city	Kapolei	Carbon monoxide	Parts per million	0.994737	14
6	6	2018-01-01	Hawaii	Honolulu	Not in a city	Kapolei	Carbon monoxide	Parts per million	0.200000	2
7	7	2018-01-01	Pennsylvania	Erie	Erie	NaN	Carbon monoxide	Parts per million	0.200000	2
8	8	2018-01-01	Hawaii	Honolulu	Honolulu	Honolulu	Carbon monoxide	Parts per million	0.400000	5
9	9	2018-01-01	Colorado	Larimer	Fort Collins	Fort Collins - CSU - S. Mason	Carbon monoxide	Parts per million	0.300000	6

In [58]:
```python
# Get descriptive stats.

epa_data.describe()
```

Out[58]:

	Unnamed: 0	arithmetic_mean	aqi
count	260.000000	260.000000	260.000000
mean	129.500000	0.403169	6.757692
std	75.199734	0.317902	7.061707
min	0.000000	0.000000	0.000000
25%	64.750000	0.200000	2.000000
50%	129.500000	0.276315	5.000000
75%	194.250000	0.516009	9.000000
max	259.000000	1.921053	50.000000

In [60]:
```python
# Get descriptive stats about the states in the data.

epa_data["state_name"].describe()
```

Out[60]:
```
count            260
unique            52
top       California
freq              66
Name: state_name, dtype: object
```

In [62]:
```python
# Compute the mean value from the aqi column.

np.mean(epa_data["aqi"])
```

Out[62]: 6.757692307692308

In [64]:
```python
# Compute the median value from the aqi column.

np.median(epa_data["aqi"])
```

Out[64]: 5.0

In [66]:
```python
# Identify the minimum value from the aqi column.

np.min(epa_data["aqi"])
```

Out[66]: 0

In [68]:
```python
# Identify the maximum value from the aqi column.

np.max(epa_data["aqi"])
```

Out[68]: 50

In [70]:
```python
# Compute the standard deviation for the aqi column.

np.std(epa_data["aqi"], ddof=1)
```

Out[70]: 7.0617066788207215

In [ ]:
```python
# some key takeaways that you learned

# Functions in the pandas and numpy libraries can be used to find statistics that describe a dataset.
# The describe() function from pandas generates a table of descriptive statistics about numerical or categorical columns.
# The mean(), median(), min(), max(), and std() functions from numpy are useful for finding individual statistics about numerical data.
```

```
# What summary would you provide to stakeholders?

# 75% of the AQI values in the data are below 9, which is considered good air quality.
# Funding should be allocated for further investigation of the less healthy regions in order to learn how to improve the conditions.
```