In [4]:

```
# Import Libraries
import pandas as pd
import numpy as np
```

In [14]:

```
# Read the Data and Showed 10 rows
df = pd.read_csv(r'C:\Users\HP\Desktop\Advance Data Analyst\2. Getting started with Python\4. Module 4\2. Dictionaries and sets\Files\c2_epa_air_quality.csv')
df.head(10)
```

Out[14]:

	state_code	state_name	county_code	county_name	aqi	state_code_int	county_code_int
0	4	Arizona	13	Maricopa	18.0	4	13
1	4	Arizona	13	Maricopa	9.0	4	13
2	4	Arizona	19	Pima	20.0	4	19
3	6	California	1	Alameda	11.0	6	1
4	6	California	7	Butte	6.0	6	7
5	6	California	19	Fresno	11.0	6	19
6	6	California	29	Kern	7.0	6	29
7	6	California	29	Kern	3.0	6	29
8	6	California	29	Kern	7.0	6	29
9	6	California	37	Los Angeles	13.0	6	37

In [24]:

```
# Assign to new variables
state_list = df['state_name']
county_list = df['county_name']
aqi_list = df['aqi']
```

In [26]:

```
# Create a list of tuples
epa_tuples = list(zip(state_list, county_list, aqi_list))
```

In [30]:

```
# Create a dictionary
aqi_dict = {}
for state, county, aqi in epa_tuples:
    if state in aqi_dict:
        aqi_dict[state].append((county, aqi))
    else:
        aqi_dict[state] = [(county, aqi)]

aqi_dict['Vermont']
```

Out[30]:

```
[('Chittenden', 18.0),
 ('Chittenden', 20.0),
 ('Chittenden', 3.0),
 ('Chittenden', 49.0),
 ('Rutland', 15.0),
 ('Chittenden', 3.0),
 ('Chittenden', 6.0),
 ('Rutland', 3.0),
 ('Rutland', 6.0),
 ('Chittenden', 5.0),
 ('Chittenden', 2.0)]
```

In [32]:

```
# Calculate how many readings were recorded in the state of Arizona
len(aqi_dict['Arizona'])
```

Out[32]:

```
72
```

In [36]:

```
# Calculate the mean AQI from the state of California
ca_aqi_list = [aqi for county, aqi in aqi_dict['California']]
ca_aqi_mean = sum(ca_aqi_list) / len(ca_aqi_list)
ca_aqi_mean
```

Out[36]:

```
9.412280701754385
```

In [38]:

```
# Define a county_counter() function
def county_counter(state):
    county_dict = {}
    for county, aqi in aqi_dict[state]:
        if county in county_dict:
            county_dict[county] +=1
        else:
            county_dict[county] = 1
    return county_dict
```

In [40]:

```
# Use the function to check Washington County, PA
pa_dict = county_counter('Pennsylvania')
pa_dict['Washington']
```

Out[40]:

```
7
```

In [42]:

```
# Use the function to check the different counties in Indiana
county_counter('Indiana').keys()
```

Out[42]:

```
dict_keys(['Marion', 'St. Joseph', 'Vanderburgh', 'Allen', 'Vigo', 'Hendricks', 'Lake'])
```

In [44]:

```
# Use sets to determine how many counties share names
all_counties = []
for state in aqi_dict.keys():
    counties = list(county_counter(state).keys())
    all_counties += counties

len(all_counties)
```

Out[44]:

```
277
```

In [50]:

```
# Calculate how many counties share names
shared_count = 0

for county in set(all_counties):
    count = all_counties.count(county)
    if count > 1:
        shared_count += count

shared_count
```

Out[50]:

```
41
```

In [ ]:

```
# Conclusion
# Python has many built-in functions that are useful for building dictionaries and sets.
# Dictionaries in Python are useful for representing data in terms of keys mapped to values.
# A set will not allow duplicate values.
# The values a set contains are unchangable and unordered.
# Functions and loop iteration can be used to perform calculations on dictionary values.
# Once the values have been calculated, they can be saved to other data types, such as tuples, lists, and sets.
# There are many ways to access data stored inside a dictionary.
```