In [1]:
```
#dependencies and setup
import pandas as pd
pd.options.display.float_format = '{:,.2f}'.format
import os
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
import plotly.express as px
from openpyxl import Workbook
import numpy as np
# SQLite dependencies
import sqlite3
from sqlalchemy import create_engine, text
from sqlalchemy import Column, Integer, String, Float
from pandas_profiling import ProfileReport
```

In [2]:
```
# SQLite dependencies
import sqlite3
from sqlalchemy import create_engine, text
from sqlalchemy import Column, Integer, String, Float
from pandas_profiling import ProfileReport
#  SQLite DB creation and establishing connection
database_path = "NJ_County_DB.sqlite"
engine = create_engine(f"sqlite:///{database_path}", echo=True)
sqlite_connection = engine.connect()
```

In [3]:
```
sql_query = """
SELECT * FROM nj_zillow_house_value_index
 AS T1
INNER JOIN nj_population AS T2 ON T1.county_name = T2.county_name AND T1.year=T2.year
INNER JOIN nj_poverty_median_income AS T3 ON T1.county_name = T3.county_name AND T1.year=T3.year
INNER JOIN
(SELECT county_name , year, AVG(tax_rate) AS tax_rate
FROM nj_property_tax GROUP BY 1,2)AS T4 ON T1.county_name = T4.county_name AND T1.year=T4.year
INNER JOIN nj_mortgage_rates AS T5 ON T1.year=T5.year
;
"""
all_df = pd.read_sql(sql_query,sqlite_connection)
all_df.info()
```

```
2023-03-25 12:15:59,256 INFO sqlalchemy.engine.Engine PRAGMA main.table_info("
SELECT * FROM nj_zillow_house_value_index
 AS T1
INNER JOIN nj_population AS T2 ON T1.county_name = T2.county_name AND T1.year=T2.year
INNER JOIN nj_poverty_median_income AS T3 ON T1.county_name = T3.county_name AND T1.year=T3.year
INNER JOIN
(SELECT county_name , year, AVG(tax_rate) AS tax_rate
FROM nj_property_tax GROUP BY 1,2)AS T4 ON T1.county_name = T4.county_name AND T1.year=T4.year
INNER JOIN nj_mortgage_rates AS T5 ON T1.year=T5.year
;
")
2023-03-25 12:15:59,260 INFO sqlalchemy.engine.Engine [raw sql] ()
2023-03-25 12:15:59,262 INFO sqlalchemy.engine.Engine PRAGMA temp.table_info("
SELECT * FROM nj_zillow_house_value_index
 AS T1
INNER JOIN nj_population AS T2 ON T1.county_name = T2.county_name AND T1.year=T2.year
INNER JOIN nj_poverty_median_income AS T3 ON T1.county_name = T3.county_name AND T1.year=T3.year
INNER JOIN
(SELECT county_name , year, AVG(tax_rate) AS tax_rate
FROM nj_property_tax GROUP BY 1,2)AS T4 ON T1.county_name = T4.county_name AND T1.year=T4.year
INNER JOIN nj_mortgage_rates AS T5 ON T1.year=T5.year
;
")
2023-03-25 12:15:59,263 INFO sqlalchemy.engine.Engine [raw sql] ()
2023-03-25 12:15:59,265 INFO sqlalchemy.engine.Engine
SELECT * FROM nj_zillow_house_value_index
 AS T1
INNER JOIN nj_population AS T2 ON T1.county_name = T2.county_name AND T1.year=T2.year
INNER JOIN nj_poverty_median_income AS T3 ON T1.county_name = T3.county_name AND T1.year=T3.year
INNER JOIN
(SELECT county_name , year, AVG(tax_rate) AS tax_rate
FROM nj_property_tax GROUP BY 1,2)AS T4 ON T1.county_name = T4.county_name AND T1.year=T4.year
INNER JOIN nj_mortgage_rates AS T5 ON T1.year=T5.year
;

2023-03-25 12:15:59,266 INFO sqlalchemy.engine.Engine [raw sql] ()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1260 entries, 0 to 1259
Data columns (total 23 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   county_name        1260 non-null   object 
 1   year               1260 non-null   int64  
 2   num_of_bedrooms    1260 non-null   int64  
 3   house_value_index  1260 non-null   float64
 4   county_name        1260 non-null   object 
 5   est_pop            1260 non-null   int64  
 6   year               1260 non-null   int64  
 7   county_name        1260 non-null   object 
 8   median_hh_income   1260 non-null   int64  
 9   poverty_count      1260 non-null   int64  
 10  poverty_rate       1260 non-null   float64
 11  st_abb             1260 non-null   object 
 12  year               1260 non-null   int64  
 13  state_code         1260 non-null   object 
 14  county_code        1260 non-null   object 
 15  county_name        1260 non-null   object 
 16  year               1260 non-null   int64  
 17  tax_rate           1260 non-null   float64
 18  year               1260 non-null   int64  
 19  apr_30             1260 non-null   float64
 20  points_30          1260 non-null   float64
 21  apr_15             1260 non-null   float64
 22  points_15          1260 non-null   float64
dtypes: float64(7), int64(9), object(7)
memory usage: 226.5+ KB
```

In [4]:
```
all_df=all_df.loc[:,~all_df.columns.duplicated()].copy()
all_df
```

Out[4]:

	county_name	year	num_of_bedrooms	house_value_index	est_pop	median_hh_income	poverty_count	poverty_rate	st_abb	state_code
0	ATLANTIC	2010	1	120,414.14	274654	51457	36693	13.60	NJ	034
1	ATLANTIC	2011	1	106,680.39	274635	49983	35108	13.10	NJ	034
2	ATLANTIC	2012	1	100,139.16	274657	50881	38245	14.20	NJ	034
3	ATLANTIC	2013	1	94,991.76	274360	51668	46281	17.10	NJ	034
4	ATLANTIC	2014	1	92,839.52	272634	54208	40761	15.10	NJ	034
...	...	...	...	...	...	...	...	...	...	...
1255	WARREN	2017	5	337,688.95	105761	79633	7770	7.40	NJ	034
1256	WARREN	2018	5	348,528.58	105709	77571	7006	6.70	NJ	034
1257	WARREN	2019	5	352,652.50	105455	83998	7313	7.10	NJ	034
1258	WARREN	2020	5	365,908.92	105624	80412	7539	7.30	NJ	034
1259	WARREN	2021	5	433,923.52	110731	82900	10140	9.30	NJ	034

1260 rows × 16 columns

In [5]:
```
all_df.to_csv('../Resources/final_data2.csv',index=False)
```

In [6]:
```
# close connection
sqlite_connection.close()
```