## Розвідувальний аналіз даних (EDA). Складання аналітичного звіту

Виконав студент групи ПП-42:

Легейда А.О.

Викладач: Білий Р.О.

**Завдання**

### Виконайте дослідження domain experience стосовно американського ринку нерухомості. Ознайомтесь з декількома прикладами аналітичних продуктів від топових гравців на американському ринку, направлених на інвесторів. Питання, які потрібно опрацювати:

### Як топові компанії на ринку складають звіти по нерухомості?

### Які графіки використовуються для донесення інформації?

### Які співвідношення між якими даними по ринку є показовими для інвесторів / керівників агенцій нерухомості?

### Яка термінологія використовується для опису закономірностей на ринку нерухомості?

### Завантажити файли з даними у папку проекту з посилання:

### <https://www1.nyc.gov/site/finance/taxes/property-rolling-sales-data.page>

### Очистити дані.

### Виконайте розвідувальний аналіз, щоб дізнатися, де є викиди або відсутні значення, вирішіть, як ви їх обробляти, переконайтеся, що дати відформатовані правильно, значення, які ви вважаєте числовими, розглядаються як такі і т.д.

### Виконайте аналіз розвідувальних даних (отриманих результатів) для візуалізації та зіставлення за житловими масивами та за часом. Почніть шукати осмислені закономірності у цьому наборі.

### Зберіть висновки до невеликий звіт для генерального директора (графіки, висновки з текстом у окремому файлі), який потребує належного оформлення висновків, структури тощо.

**Хід роботи**

1. Добавлена колонка BOROUGH\_NAME для зручного групування в запитах

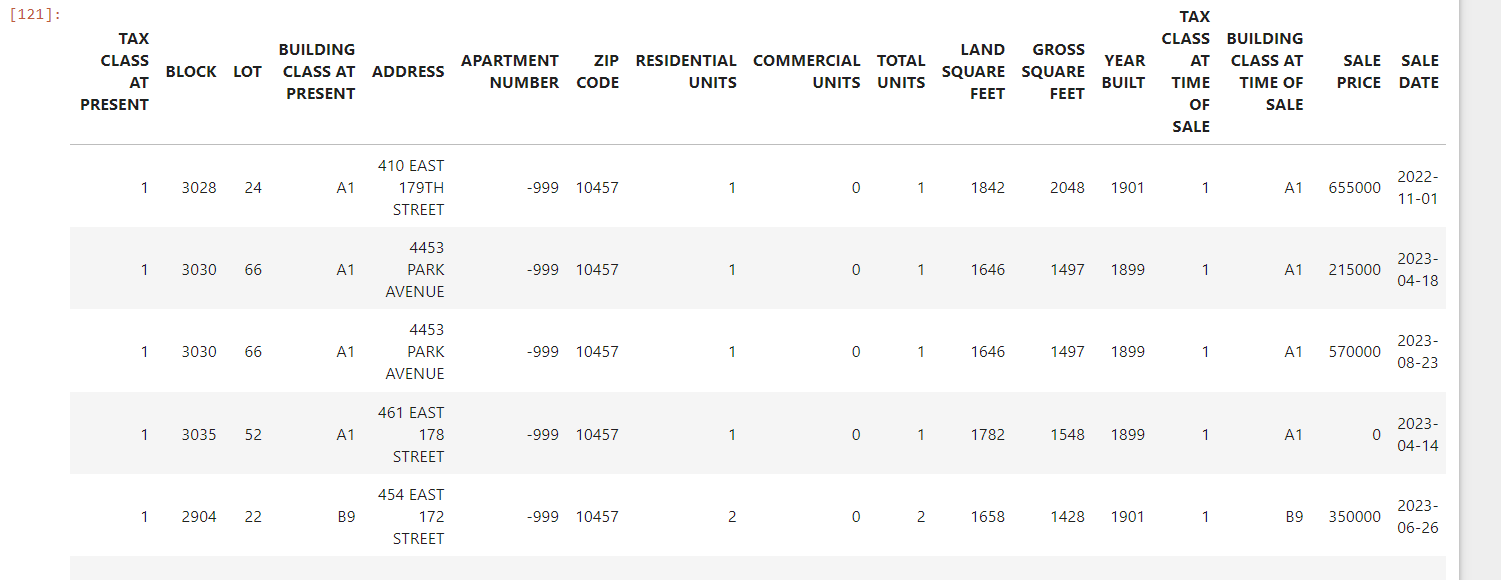
df.loc[df['BOROUGH'] == 1, 'BOROUGH\_NAME'] = 'manhattan'

df.loc[df['BOROUGH'] == 2, 'BOROUGH\_NAME'] = 'bronx'

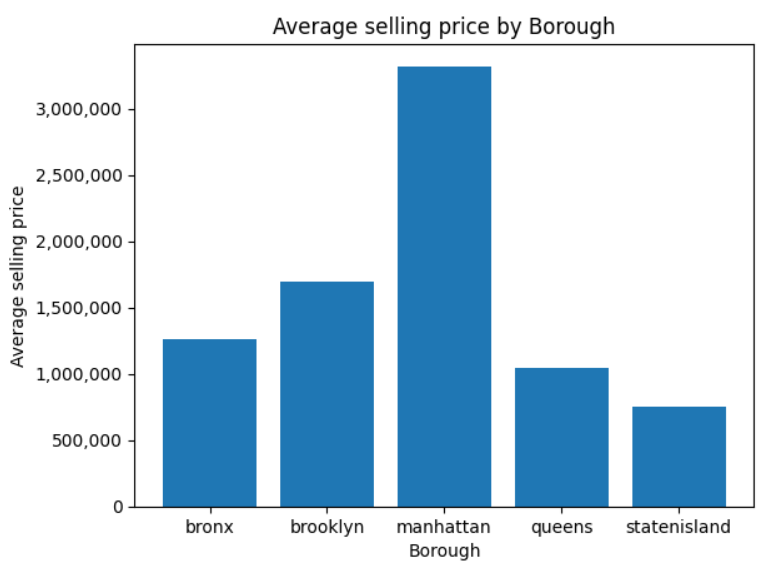
df.loc[df['BOROUGH'] == 3, 'BOROUGH\_NAME'] = 'brooklyn'

df.loc[df['BOROUGH'] == 4, 'BOROUGH\_NAME'] = 'queens'

df.loc[df['BOROUGH'] == 5, 'BOROUGH\_NAME'] = 'statenisland'



2. Середня ціна продажу по округу. Бачимо, що найвища ціна в окрузі Manhattan.



average\_price\_by\_borough = df[df['SALE PRICE'] != 0].groupby('BOROUGH\_NAME')['SALE PRICE'].mean().reset\_index()

plt.bar(average\_price\_by\_borough['BOROUGH\_NAME'], average\_price\_by\_borough['SALE PRICE'])

plt.gca().get\_yaxis().set\_major\_formatter(ticker.StrMethodFormatter('{x:,.0f}'))

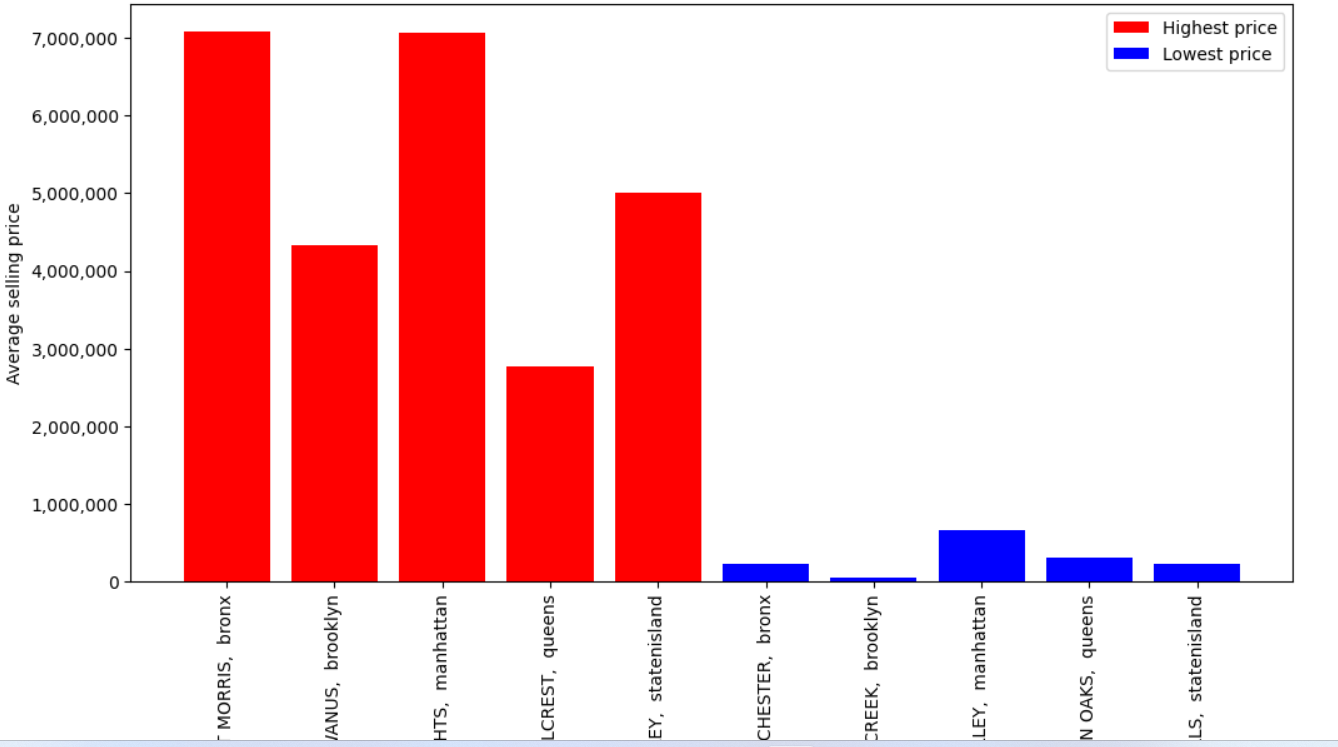
plt.xlabel('Borough')

plt.ylabel('Average selling price')

plt.title('Average selling price by Borough')

plt.show()

3. Район із найвищою та найнижчою середньою ціною. Бачимо, що найвища середня ціна в районах MOTT HEVEN/PORT MORRIS та GOWANUS, а найнища в SPRING GREEK



df\_pzh = df.groupby(['BOROUGH\_NAME', 'NEIGHBORHOOD'])['SALE PRICE'].mean().reset\_index()

df\_pzh2 = df\_pzh.groupby('BOROUGH\_NAME')['SALE PRICE'].idxmax()

max\_sales\_df2 = df\_pzh.loc[df\_pzh2]

df\_pzh3 = df\_pzh[df\_pzh['SALE PRICE'] != 0].groupby('BOROUGH\_NAME')['SALE PRICE'].idxmin()

min\_sales\_df2 = df\_pzh.loc[df\_pzh3]

plt.figure(figsize=(12, 6))

plt.gca().get\_yaxis().set\_major\_formatter(ticker.StrMethodFormatter('{x:,.0f}'))

plt.bar(max\_sales\_df2['NEIGHBORHOOD'] + ', ' + max\_sales\_df2['BOROUGH\_NAME'], max\_sales\_df2['SALE PRICE'], color='red', label='Highest price')

plt.bar(min\_sales\_df2['NEIGHBORHOOD'] + ', ' + min\_sales\_df2['BOROUGH\_NAME'], min\_sales\_df2['SALE PRICE'], color='blue', label='Lowest price')

plt.xticks(rotation=90)

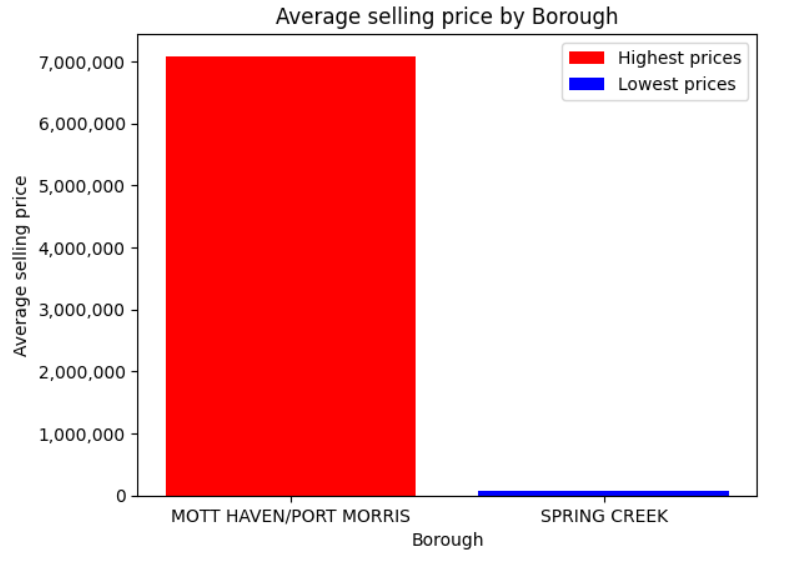
plt.legend()

plt.xlabel('Borough')

plt.ylabel('Average selling price')

plt.title('Neighborhood with the highest and lowest average price')

4. Найменша та найбільша середня ціна продажу по районам. Бачимо, що найменша ціна в районі SPRING GREEK, а найбільша в MOTT HEVEN/PORT MORRIS.



que = df.groupby('NEIGHBORHOOD')['SALE PRICE'].mean().reset\_index()

que2 = df[df['SALE PRICE'] != 0].groupby('NEIGHBORHOOD')['SALE PRICE'].mean().reset\_index()

max\_index = que['SALE PRICE'].idxmax()

max\_avg\_sale = que.loc[max\_index]

min\_index = que2['SALE PRICE'].idxmin()

min\_avg\_sale = que2.loc[min\_index]

plt.bar(max\_avg\_sale['NEIGHBORHOOD'], max\_avg\_sale['SALE PRICE'], color='red', label='Highest prices')

plt.bar(min\_avg\_sale['NEIGHBORHOOD'], min\_avg\_sale['SALE PRICE'], color='blue', label='Lowest prices')

plt.gca().get\_yaxis().set\_major\_formatter(ticker.StrMethodFormatter('{x:,.0f}'))

plt.legend()

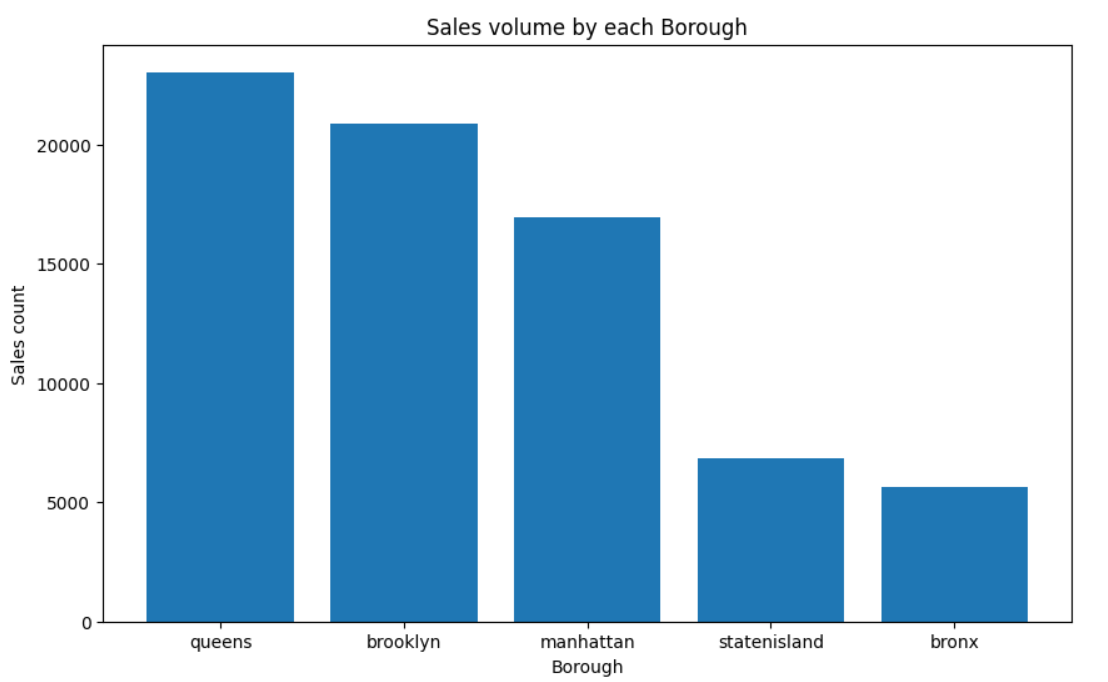
plt.xlabel('Borough')

plt.ylabel('Average selling price')

plt.title('Average selling price by Borough')

plt.show()

5. Обсяг продажів по кожному району. Бачимо, що найбільша ціна в районі queens, а найменша в bronx.



plt.figure(figsize=(10, 6))

res1 = df['BOROUGH\_NAME'].value\_counts().reset\_index()

plt.bar(res1['BOROUGH\_NAME'], res1['count'])

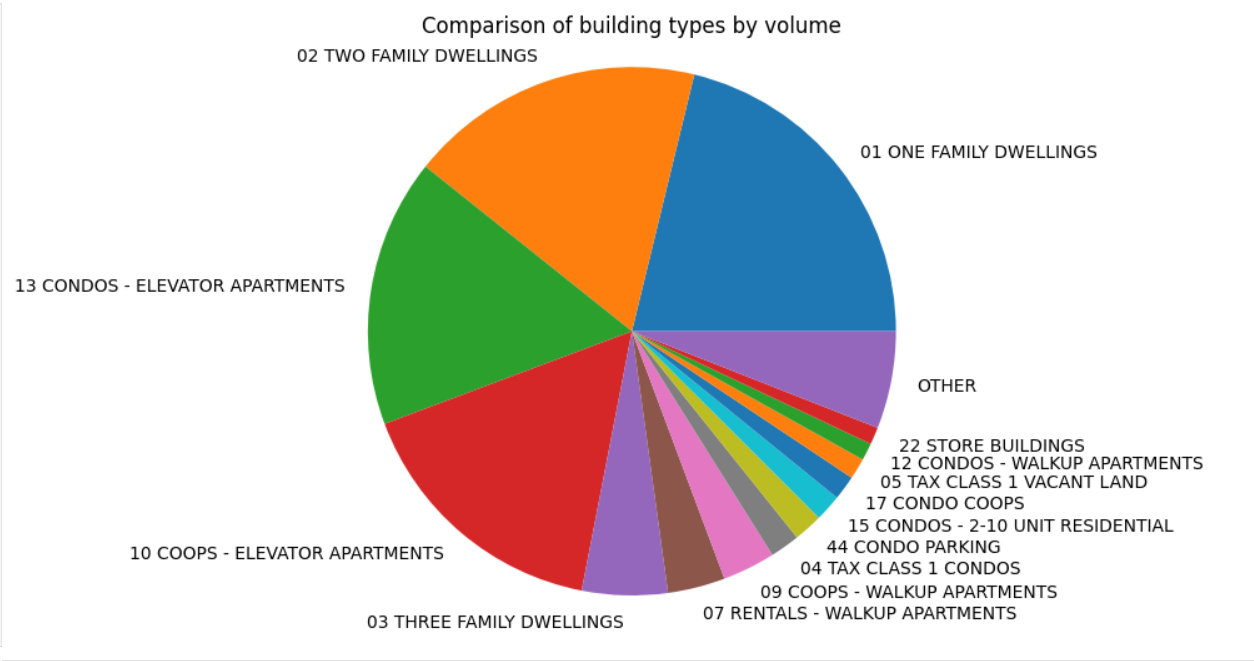
plt.xlabel('Borough')

plt.ylabel('Sales count')

plt.title('Sales volume by each Borough')

plt.show()

6. Порівняння типів будівель за об’ємом. Найбільші обсяги, як бачимо в 01 ONE FAMILY DWELLINGS.



building\_type\_counts = df['BUILDING CLASS CATEGORY'].value\_counts().reset\_index()

building\_type\_counts2 = building\_type\_counts[building\_type\_counts['count'] > 500].copy()

que20 = pd.Series({'BUILDING CLASS CATEGORY': 'OTHER', 'count': building\_type\_counts3['count'].sum()})

building\_type\_counts2.loc[len(building\_type\_counts2)] = que20

plt.figure(figsize=(6, 6))

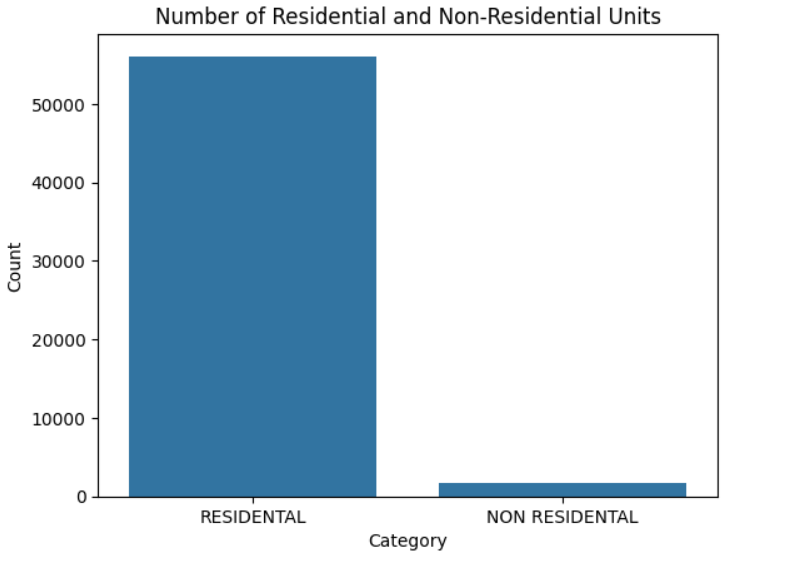
plt.pie(building\_type\_counts2['count'], labels=building\_type\_counts2['BUILDING CLASS CATEGORY'])

plt.axis('equal')

plt.title('Comparison of building types by volume')

plt.show()

7. Кількість житлових і нежитлових одиниць. Як бачимо кількість житлових одиниць перевщує 50000.



resid = pd.DataFrame()

resid = df[(df['TOTAL UNITS'] != -999) & (df['TOTAL UNITS'] >= 1)]

non\_resid = pd.DataFrame()

non\_resid = df[(df['TOTAL UNITS'] != -999) & (df['TOTAL UNITS'] == 0)]

non\_resid.shape[0]

res21 = pd.DataFrame({'RESIDENTAL': [resid.shape[0]], 'NON RESIDENTAL': [non\_resid.shape[0]]})

res21\_long = res21.melt(var\_name='Category', value\_name='Count')

sns.barplot(x='Category', y='Count', data=res21\_long)

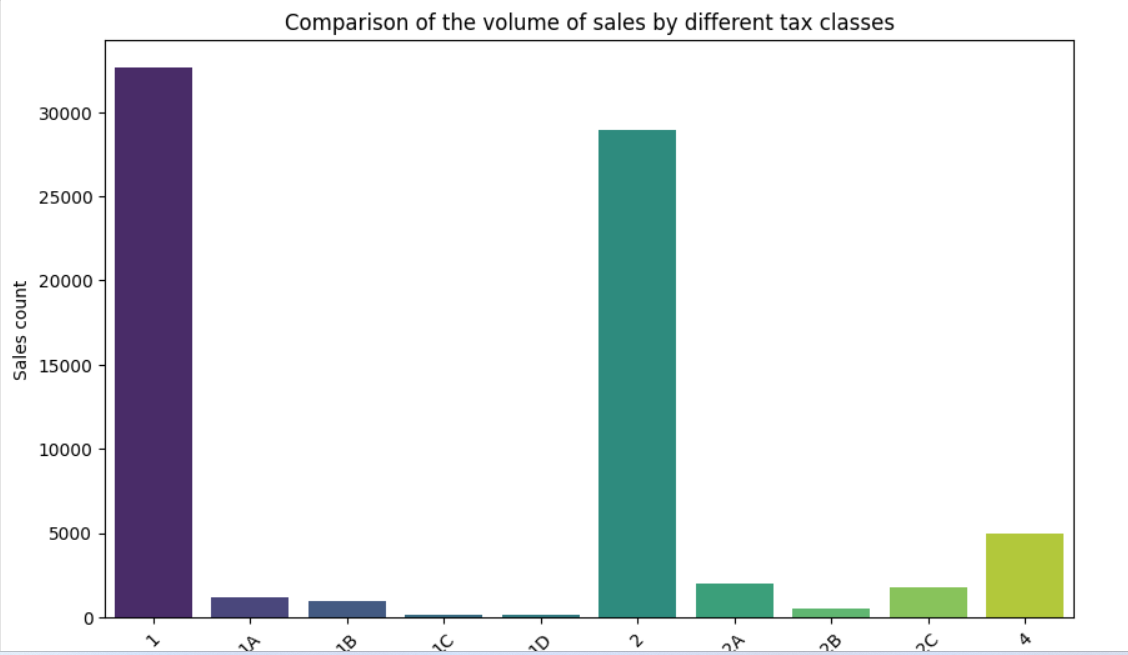
plt.xlabel('Category')

plt.ylabel('Count')

plt.title('Number of Residential and Non-Residential Units')

plt.show()

8. Порівняння обсягів за різними податковими класами. Як бачимо найбільші обсяги в податкових класах 1 та 2.



sales\_by\_tax\_class = df[df['TAX CLASS AT PRESENT'] != -999].groupby('TAX CLASS AT PRESENT')['SALE PRICE'].count().reset\_index()

plt.figure(figsize=(10, 6))

sns.barplot(x='TAX CLASS AT PRESENT', y='SALE PRICE', data=sales\_by\_tax\_class,hue ='TAX CLASS AT PRESENT',palette='viridis')

plt.xlabel('Tax Class')

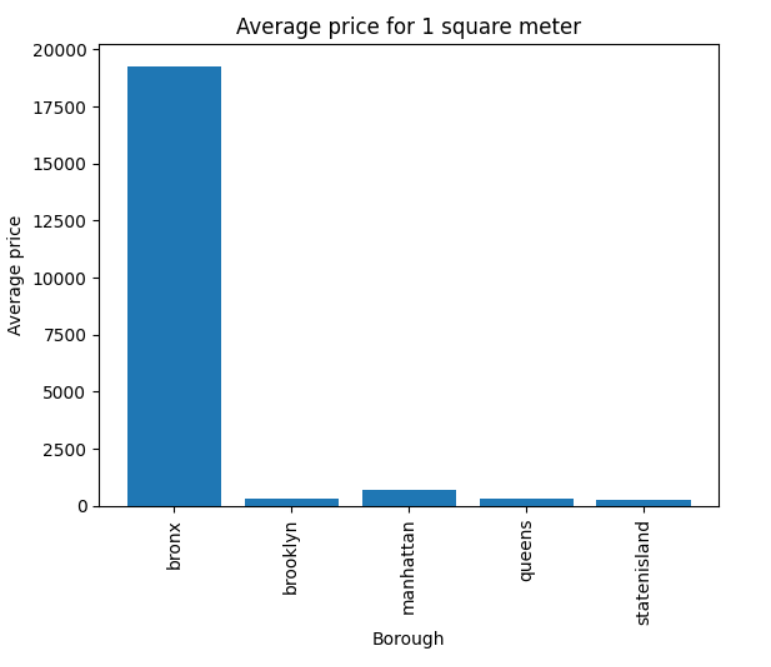
plt.ylabel('Sales count')

plt.title('Comparison of the volume of sales by different tax classes')

plt.xticks(rotation=45)

plt.show()

9. Середня ціна за 1 кв метр. Найбільша середня ціна як бачимо в районі Bronx.



filtered\_data = df[(df['GROSS SQUARE FEET'] > 0) & (df['GROSS SQUARE FEET'] != -999)]

filtered\_data2 = filtered\_data.copy()

filtered\_data2['PRICE PER SQFT'] = filtered\_data['SALE PRICE'] / filtered\_data['GROSS SQUARE FEET']

filtered\_data3 = filtered\_data2.groupby('BOROUGH\_NAME')['PRICE PER SQFT'].mean().reset\_index()

plt.bar(filtered\_data3['BOROUGH\_NAME'], filtered\_data3['PRICE PER SQFT'])

plt.xlabel('Borough')

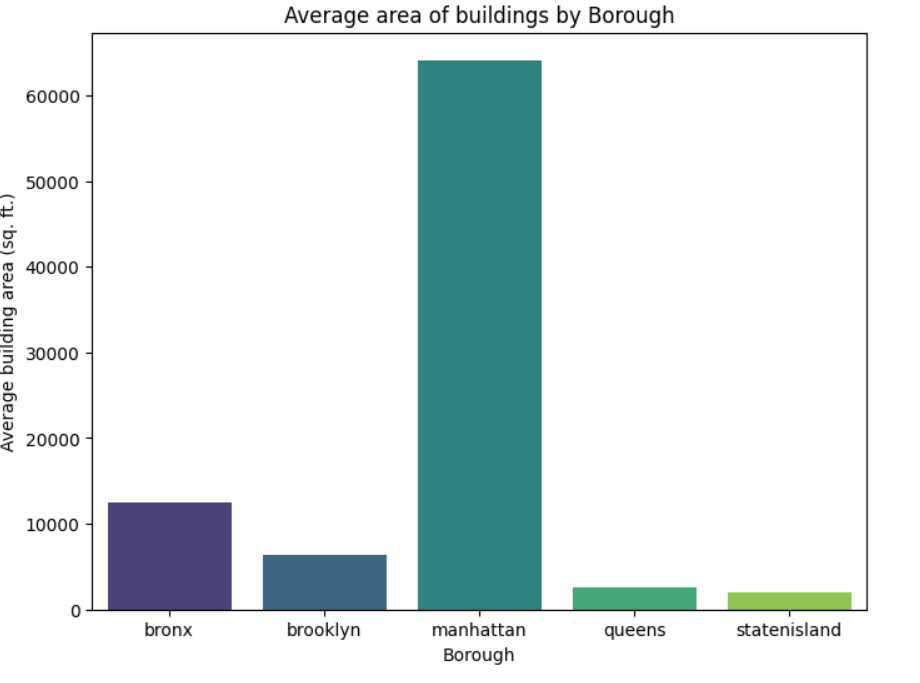
plt.ylabel('Average price')

plt.title('Average price for 1 square meter')

plt.xticks(rotation=90)

plt.show()

10. Середня площа забудови по районах. Найбільша середня площа в районі Manhattan.



filtered\_data = df[df['GROSS SQUARE FEET'] > 0]

average\_sqft\_by\_borough = filtered\_data.groupby('BOROUGH\_NAME')['GROSS SQUARE FEET'].mean().reset\_index()

plt.figure(figsize=(8, 6))

sns.barplot(data=average\_sqft\_by\_borough, x='BOROUGH\_NAME', y='GROSS SQUARE FEET', hue='BOROUGH\_NAME', palette='viridis')

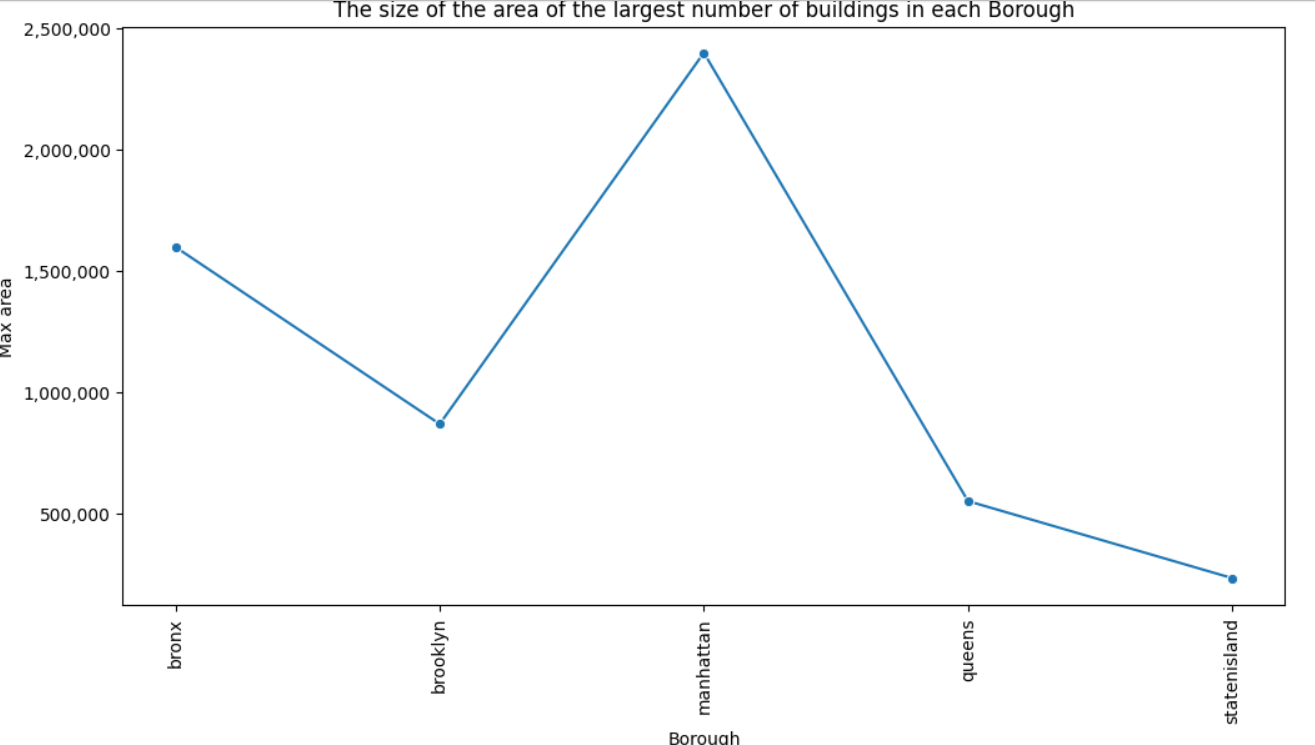
plt.xlabel('Borough')

plt.ylabel('Average building area (sq. ft.)')

plt.title('Average area of buildings by Borough')

plt.show()

11. Розмір площі найбільшої кількості будівель у кожному районі. Найбільша площа в районі Manhattan зі значенням близько 2.400.000.



max\_sqft\_by\_neighborhood = df.groupby('BOROUGH\_NAME')['GROSS SQUARE FEET'].idxmax()

max\_sqft\_by\_neighborhood\_df2 = df.loc[max\_sqft\_by\_neighborhood]

max\_sqft\_by\_neighborhood\_df2

plt.figure(figsize=(12, 6))

plt.gca().get\_yaxis().set\_major\_formatter(ticker.StrMethodFormatter('{x:,.0f}'))

sns.lineplot(data=max\_sqft\_by\_neighborhood\_df2, x='BOROUGH\_NAME', y='GROSS SQUARE FEET', marker='o')

plt.xlabel('Borough')

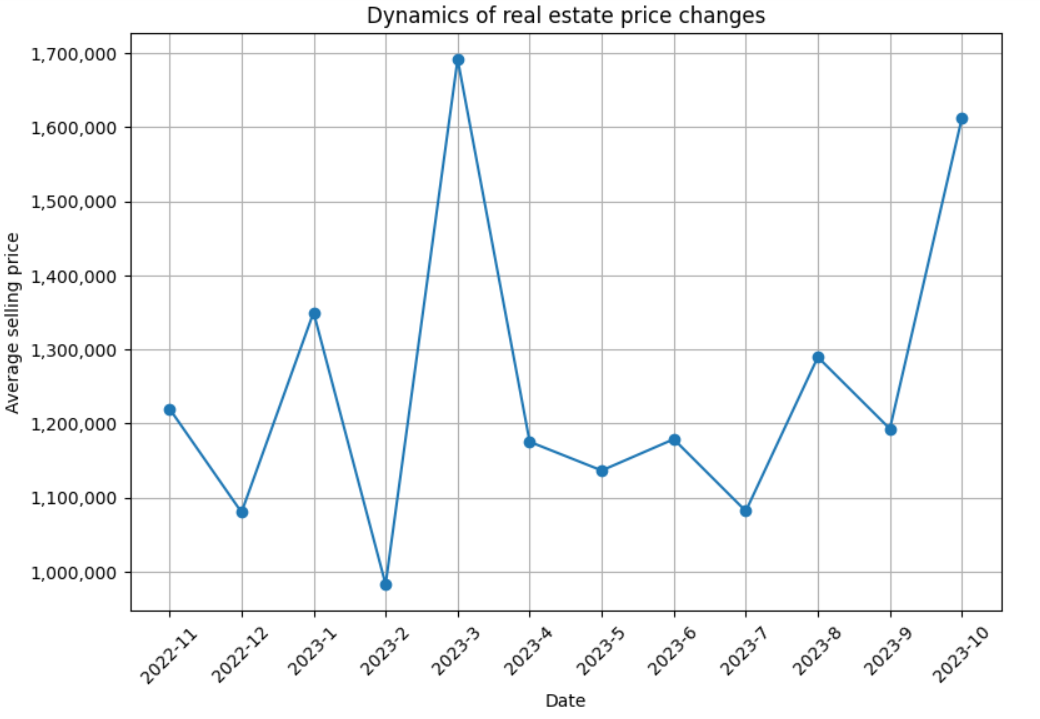
plt.ylabel('Max area')

plt.title('The size of the area of the largest number of buildings in each Borough')

plt.xticks(rotation=90)

plt.show()

12. Динаміка зміни цін на нерухомість. Як бачимо найбільша динаміка змін цін з 2023/02 по 2023/04.



df\_with\_ext\_date = df

df\_with\_ext\_date['Year'] = pd.to\_numeric(df\_with\_ext\_date['SALE DATE'].dt.year).astype(int)

df\_with\_ext\_date['Month'] = pd.to\_numeric(df\_with\_ext\_date['SALE DATE'].dt.month)

df\_with\_ext\_date['Day'] = pd.to\_numeric(df\_with\_ext\_date['SALE DATE'].dt.day)

average\_price\_by\_month = df\_with\_ext\_date.groupby(['Year', 'Month'])['SALE PRICE'].mean().reset\_index()

average\_price\_by\_month2 = average\_price\_by\_month.copy()

average\_price\_by\_month2['Date'] = average\_price\_by\_month['Year'].astype('str') + '-' + average\_price\_by\_month['Month'].astype('str')

plt.figure(figsize=(9, 6))

plt.gca().get\_yaxis().set\_major\_formatter(ticker.StrMethodFormatter('{x:,.0f}'))

plt.plot(average\_price\_by\_month2['Date'], average\_price\_by\_month2['SALE PRICE'], marker='o')

plt.xlabel('Date')

plt.ylabel('Average selling price')

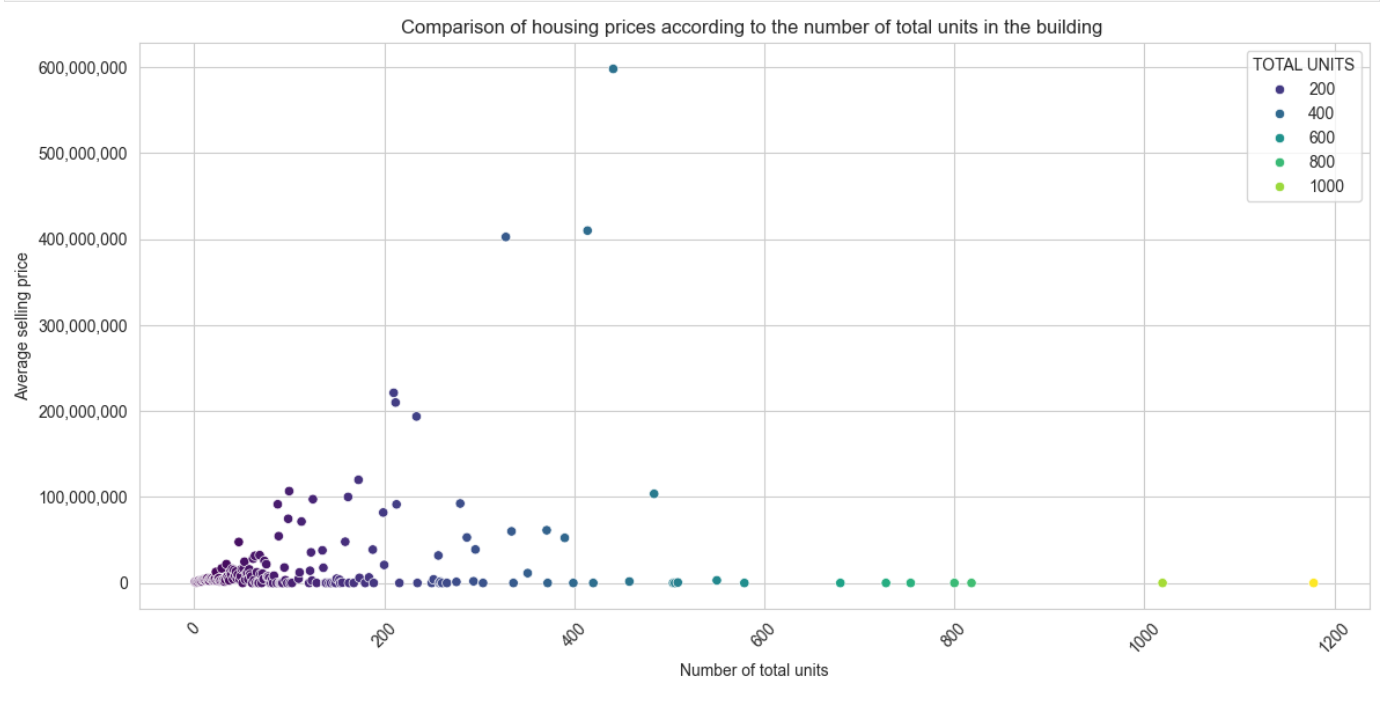
plt.title('Dynamics of real estate price changes')

plt.xticks(rotation=45)

plt.grid()

plt.show()

13. Порівняння цін на житло за загальною кількістю квартир у будинку. Як бачимо найбільше значення середньої ціни продажу в значенні 400 (тобто в тому районі де найбільший загальний середній об’єм SALE PRICE)



grouped\_data = df[(df['TOTAL UNITS'] > 0) & (df['SALE PRICE'] >= 0)].groupby('TOTAL UNITS')['SALE PRICE'].mean().reset\_index()

grouped\_data

plt.figure(figsize=(12, 6))

sns.set\_style("whitegrid")

sns.scatterplot(data=grouped\_data, x='TOTAL UNITS', y='SALE PRICE', hue='TOTAL UNITS', palette='viridis')

plt.title('Comparison of housing prices according to the number of total units in the building ')

plt.xlabel('Number of total units')

plt.ylabel('Average selling price')

plt.xticks(rotation=45)

plt.gca().get\_yaxis().set\_major\_formatter(ticker.StrMethodFormatter('{x:,.0f}'))

plt.tight\_layout()

plt.show()