

```
In [2]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
In [3]: df=pd.read_csv(r'C:\Users\Ishwar Chandra\Documents\CSV file\Airbnb NYC 2019.csv')
```

```
In [4]: df.head()
```

```
Out[4]:
```

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_nights
0	2539	Clean & quiet apt home by the park	2787	John	Brooklyn	Kensington	40.64749	-73.97237	Private room	149	1
1	2595	Skylit Midtown Castle	2845	Jennifer	Manhattan	Midtown	40.75362	-73.98377	Entire home/apt	225	1
2	3647	THE VILLAGE OF HARLEM....NEW YORK !	4632	Elisabeth	Manhattan	Harlem	40.80902	-73.94190	Private room	150	3
3	3831	Cozy Entire Floor of Brownstone	4869	LisaRoxanne	Brooklyn	Clinton Hill	40.68514	-73.95976	Entire home/apt	89	1
4	5022	Entire Apt: Spacious Studio/Loft by central park	7192	Laura	Manhattan	East Harlem	40.79851	-73.94399	Entire home/apt	80	10

```
In [5]: df.shape
```

```
Out[5]: (48895, 16)
```

```
In [6]: df.info()
```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48895 entries, 0 to 48894
Data columns (total 16 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   id                                     48895 non-null  int64
1   name                                  48879 non-null  object
2   host_id                               48895 non-null  int64
3   host_name                             48874 non-null  object
4   neighbourhood_group                   48895 non-null  object
5   neighbourhood                         48895 non-null  object
6   latitude                             48895 non-null  float64
7   longitude                             48895 non-null  float64
8   room_type                             48895 non-null  object
9   price                                 48895 non-null  int64
10  minimum_nights                       48895 non-null  int64
11  number_of_reviews                    48895 non-null  int64
12  last_review                          38843 non-null  object
13  reviews_per_month                    38843 non-null  float64
14  calculated_host_listings_count        48895 non-null  int64
15  availability_365                      48895 non-null  int64
dtypes: float64(3), int64(7), object(6)
memory usage: 6.0+ MB

```

```
In [7]: df.isnull().sum()
```

```

Out[7]: id                                     0
        name                                  16
        host_id                               0
        host_name                             21
        neighbourhood_group                   0
        neighbourhood                         0
        latitude                             0
        longitude                             0
        room_type                             0
        price                                 0
        minimum_nights                       0
        number_of_reviews                    0
        last_review                          10052
        reviews_per_month                    10052
        calculated_host_listings_count        0
        availability_365                      0
dtype: int64

```

```
In [8]: df.fillna({'reviews_per_month':0},inplace=True)
```

```
In [9]: df.isnull().sum()
```

```
Out[9]: id                0
        name              16
        host_id           0
        host_name         21
        neighbourhood_group 0
        neighbourhood      0
        latitude           0
        longitude          0
        room_type          0
        price              0
        minimum_nights     0
        number_of_reviews  0
        last_review       10052
        reviews_per_month  0
        calculated_host_listings_count 0
        availability_365    0
        dtype: int64
```

```
In [10]: df.columns
```

```
Out[10]: Index(['id', 'name', 'host_id', 'host_name', 'neighbourhood_group',
               'neighbourhood', 'latitude', 'longitude', 'room_type', 'price',
               'minimum_nights', 'number_of_reviews', 'last_review',
               'reviews_per_month', 'calculated_host_listings_count',
               'availability_365'],
              dtype='object')
```

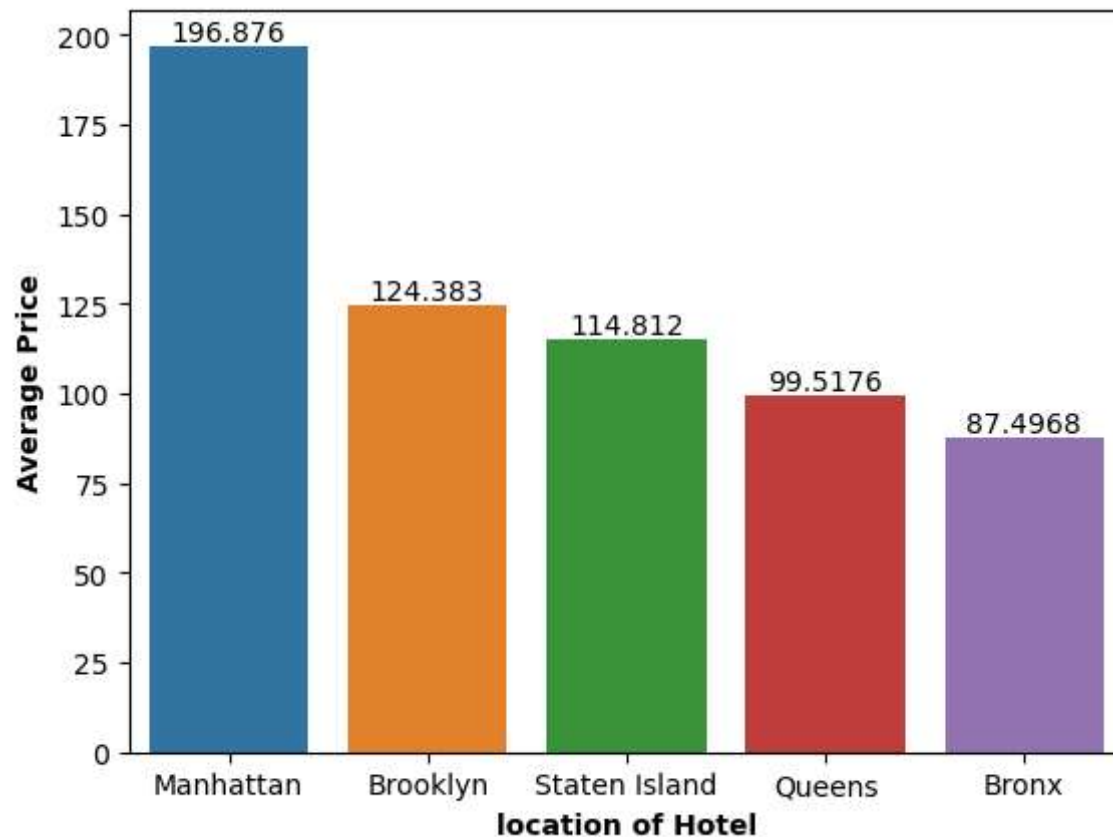
```
In [11]: graph_1=df.groupby(['neighbourhood_group'],as_index=False)['price'].mean().sort_values(['price'],ascending=False)
```

```
In [12]: graph_1
```

```
Out[12]:
```

	neighbourhood_group	price
2	Manhattan	196.875814
1	Brooklyn	124.383207
4	Staten Island	114.812332
3	Queens	99.517649
0	Bronx	87.496792

```
In [13]: ax=sns.barplot(x='neighbourhood_group',y='price',data=graph_1)
plt.xlabel('location of Hotel',fontweight='bold')
plt.ylabel('Average Price',fontweight='bold')
for bar in ax.containers:
    ax.bar_label(bar)
plt.show()
```



we can see above from graph that in manhattan , most of hotel room booking.Comprare to other location.and second brooklyn

```
In [14]: df['neighbourhood_group'].value_counts()
```

```
Out[14]: Manhattan      21661
          Brooklyn      20104
          Queens        5666
          Bronx         1091
          Staten Island   373
          Name: neighbourhood_group, dtype: int64
```

```
In [15]: filter_row=df[df['neighbourhood_group']=='Manhattan']
```

```
In [16]: filter_row.head()
```

```
Out[16]:
```

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_nights
1	2595	Skylit Midtown Castle	2845	Jennifer	Manhattan	Midtown	40.75362	-73.98377	Entire home/apt	225	1
2	3647	THE VILLAGE OF HARLEM....NEW YORK !	4632	Elisabeth	Manhattan	Harlem	40.80902	-73.94190	Private room	150	3
4	5022	Entire Apt: Spacious Studio/Loft by central park	7192	Laura	Manhattan	East Harlem	40.79851	-73.94399	Entire home/apt	80	10
5	5099	Large Cozy 1 BR Apartment In Midtown East	7322	Chris	Manhattan	Murray Hill	40.74767	-73.97500	Entire home/apt	200	3
7	5178	Large Furnished Room Near B'way	8967	Shunichi	Manhattan	Hell's Kitchen	40.76489	-73.98493	Private room	79	2

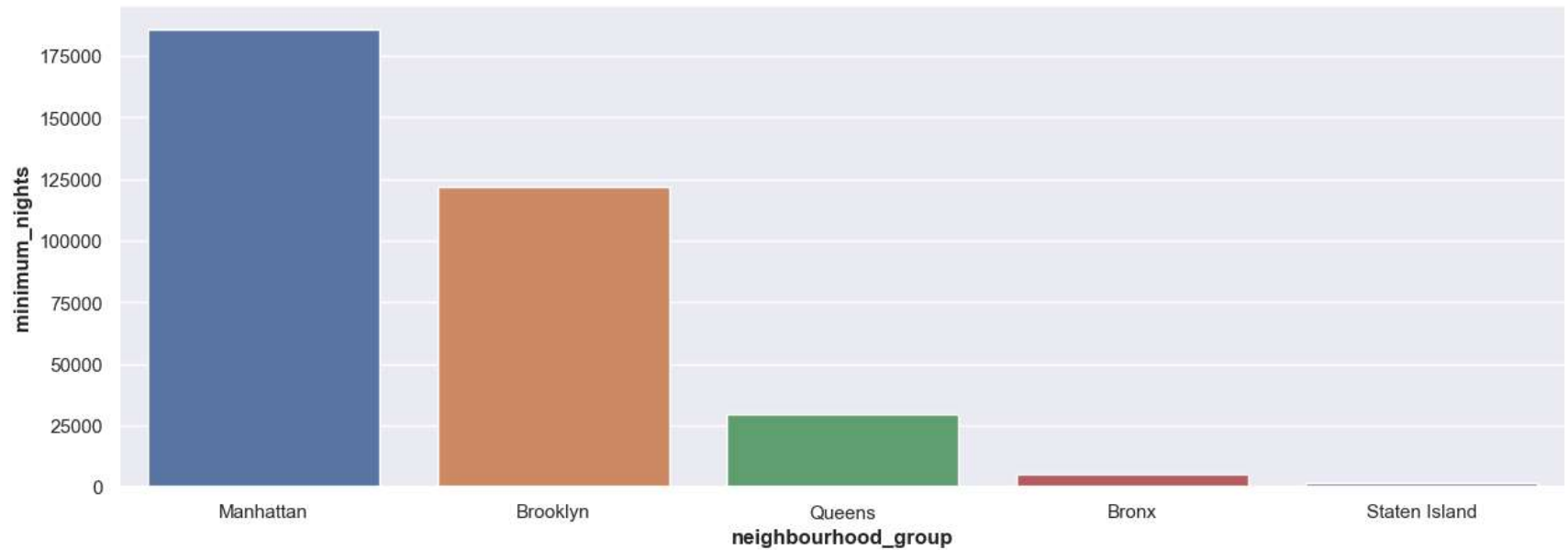
```
In [17]: minimum_nt=df.groupby(['neighbourhood_group'],as_index=False)['minimum_nights'].sum().sort_values(['minimum_nights'],as
```

```
In [18]: minimum_nt
```

```
Out[18]:
```

	neighbourhood_group	minimum_nights
2	Manhattan	185833
1	Brooklyn	121761
3	Queens	29358
0	Bronx	4976
4	Staten Island	1802

```
In [34]: sns.barplot(x='neighbourhood_group',y='minimum_nights',data=minimum_nt)
plt.xlabel('neighbourhood_group',fontweight='bold')
plt.ylabel('minimum_nights',fontweight='bold')
plt.show()
```



```
In [36]: #from above graph we can see that people maximum night spend or stay in manhattan.second position brooklyn
```

```
In [20]: pichart=df.groupby(['neighbourhood_group'],as_index=False)['number_of_reviews'].sum().sort_values(['number_of_reviews'])
```

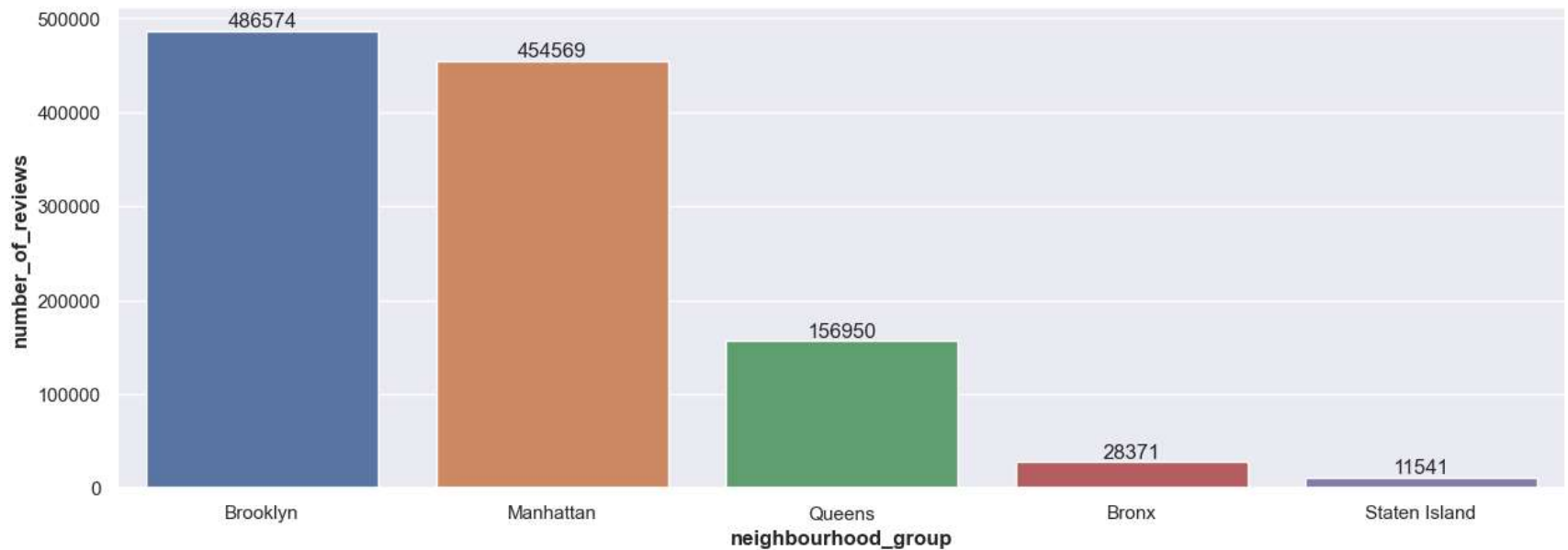
```
In [21]: pichart
```

Out[21]:

	neighbourhood_group	number_of_reviews
1	Brooklyn	486574
2	Manhattan	454569
3	Queens	156950
0	Bronx	28371
4	Staten Island	11541

1	Brooklyn	486574
2	Manhattan	454569
3	Queens	156950
0	Bronx	28371
4	Staten Island	11541

```
In [37]: ax=sns.barplot(x='neighbourhood_group',y='number_of_reviews',data=pichart)
for bar in ax.containers:
    ax.bar_label(bar)
plt.xlabel('neighbourhood_group',fontweight='bold')
plt.ylabel('number_of_reviews',fontweight='bold')
plt.show()
```



In [38]: *# from above graph we can see that most of number_review get from brooklyn.and second position in review manhattan.*

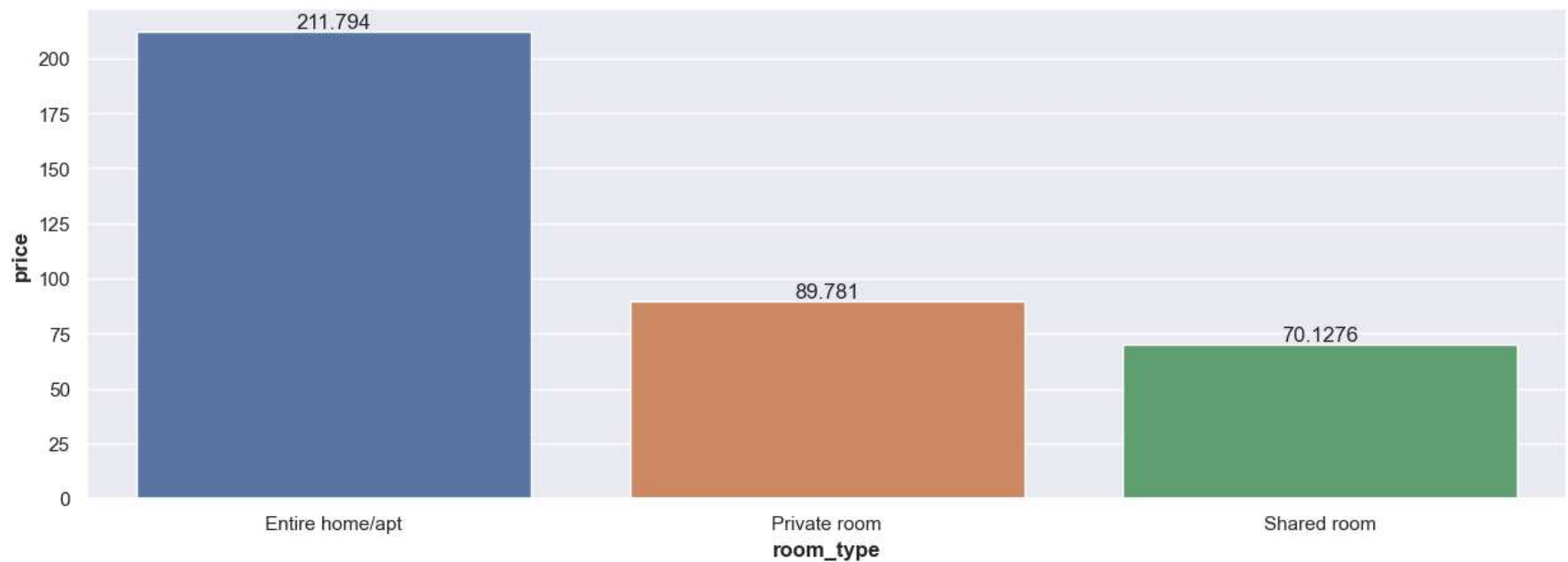
```
In [39]: room_ty=df.groupby(['room_type'],as_index=False)[['price']].mean()
```

```
In [40]: room_ty
```

```
Out[40]:
```

	room_type	price
0	Entire home/apt	211.794246
1	Private room	89.780973
2	Shared room	70.127586

```
In [44]: ax=sns.barplot(x='room_type',y='price',data=room_ty)
for bar in ax.containers:
    ax.bar_label(bar)
plt.xlabel('room_type',fontweight='bold')
plt.ylabel('price',fontweight='bold')
plt.show()
```



```
In [ ]: #from above graph we can see that most the room booking in Entire home/apt.
```

```
In [43]: df.groupby(['room_type','neighbourhood_group'],as_index=False)[['price']].mean()
```


Out[43]:

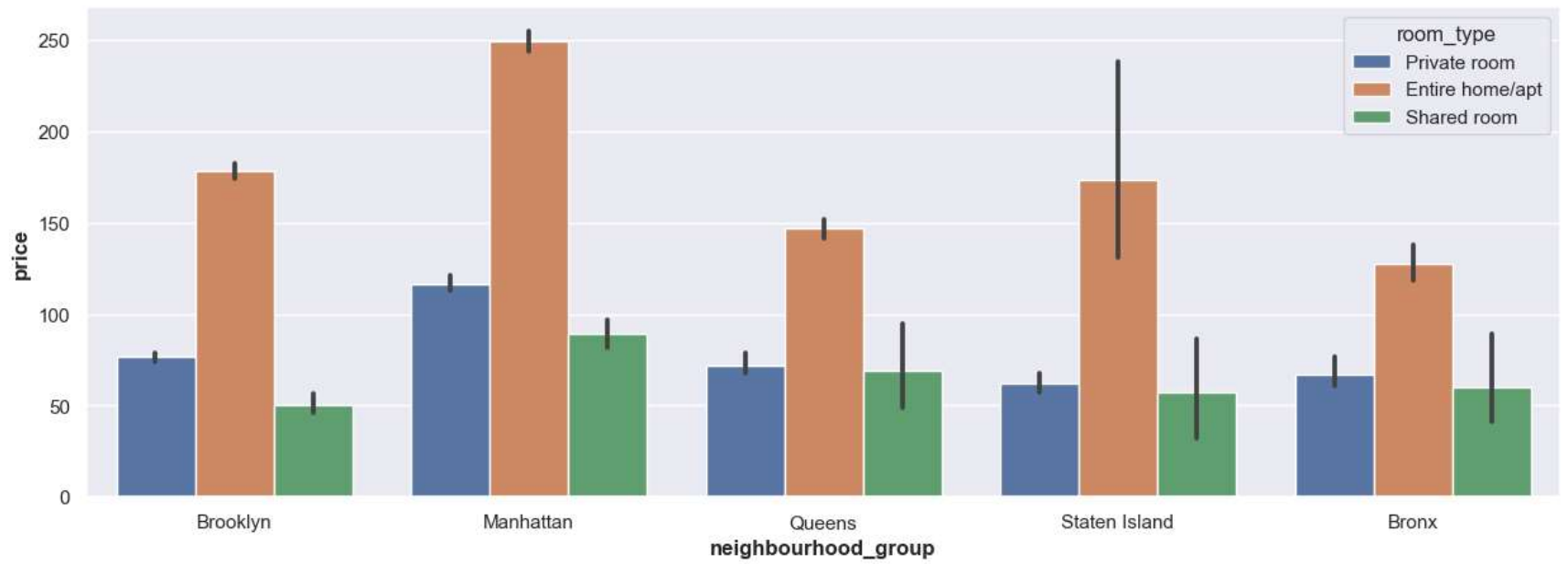
	room_type	neighbourhood_group	price
0	Entire home/apt	Bronx	127.506596
1	Entire home/apt	Brooklyn	178.327545
2	Entire home/apt	Manhattan	249.239109
3	Entire home/apt	Queens	147.050573
4	Entire home/apt	Staten Island	173.846591
5	Private room	Bronx	66.788344
6	Private room	Brooklyn	76.500099
7	Private room	Manhattan	116.776622
8	Private room	Queens	71.762456
9	Private room	Staten Island	62.292553
10	Shared room	Bronx	59.800000
11	Shared room	Brooklyn	50.527845
12	Shared room	Manhattan	88.977083
13	Shared room	Queens	69.020202
14	Shared room	Staten Island	57.444444

```
In [27]: df.groupby(['neighbourhood_group', 'room_type'], as_index=False)[['price']].mean()
```

Out[27]:

	neighbourhood_group	room_type	price
0	Bronx	Entire home/apt	127.506596
1	Bronx	Private room	66.788344
2	Bronx	Shared room	59.800000
3	Brooklyn	Entire home/apt	178.327545
4	Brooklyn	Private room	76.500099
5	Brooklyn	Shared room	50.527845
6	Manhattan	Entire home/apt	249.239109
7	Manhattan	Private room	116.776622
8	Manhattan	Shared room	88.977083
9	Queens	Entire home/apt	147.050573
10	Queens	Private room	71.762456
11	Queens	Shared room	69.020202
12	Staten Island	Entire home/apt	173.846591
13	Staten Island	Private room	62.292553
14	Staten Island	Shared room	57.444444

```
In [45]: sns.set(rc={'figure.figsize':(15,5)})
sns.barplot(x='neighbourhood_group',y='price',data=df,hue='room_type')
plt.xlabel('neighbourhood_group',fontweight='bold')
plt.ylabel('price',fontweight='bold')
plt.show()
```



In []: *#we can see from above graph,most of room booking in manhattan but Entire home/apt is greater booking from Private room*

In [29]: `df.head()`

Out[29]:

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_nights
0	2539	Clean & quiet apt home by the park	2787	John	Brooklyn	Kensington	40.64749	-73.97237	Private room	149	1
1	2595	Skylit Midtown Castle	2845	Jennifer	Manhattan	Midtown	40.75362	-73.98377	Entire home/apt	225	1
2	3647	THE VILLAGE OF HARLEM.....NEW YORK !	4632	Elisabeth	Manhattan	Harlem	40.80902	-73.94190	Private room	150	3
3	3831	Cozy Entire Floor of Brownstone	4869	LisaRoxanne	Brooklyn	Clinton Hill	40.68514	-73.95976	Entire home/apt	89	1
4	5022	Entire Apt: Spacious Studio/Loft by central park	7192	Laura	Manhattan	East Harlem	40.79851	-73.94399	Entire home/apt	80	10



In [30]:

```
df.info()
```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48895 entries, 0 to 48894
Data columns (total 16 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   id                                     48895 non-null  int64
1   name                                  48879 non-null  object
2   host_id                               48895 non-null  int64
3   host_name                             48874 non-null  object
4   neighbourhood_group                   48895 non-null  object
5   neighbourhood                         48895 non-null  object
6   latitude                             48895 non-null  float64
7   longitude                             48895 non-null  float64
8   room_type                            48895 non-null  object
9   price                                48895 non-null  int64
10  minimum_nights                       48895 non-null  int64
11  number_of_reviews                    48895 non-null  int64
12  last_review                          38843 non-null  object
13  reviews_per_month                    48895 non-null  float64
14  calculated_host_listings_count       48895 non-null  int64
15  availability_365                     48895 non-null  int64
dtypes: float64(3), int64(7), object(6)
memory usage: 6.0+ MB

```

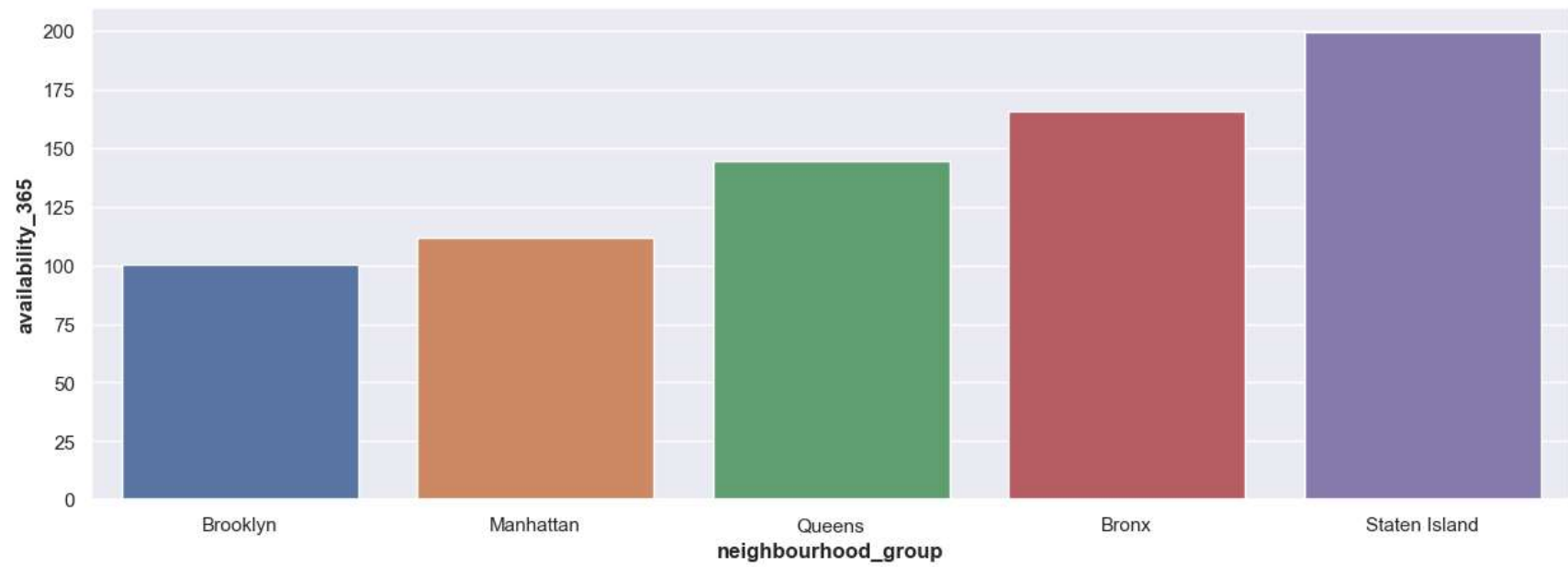
```
In [31]: available_room=df.groupby(['neighbourhood_group'],as_index=False)['availability_365'].mean().sort_values(['availability_365'])
```

```
In [32]: available_room
```

```
Out[32]:
```

	neighbourhood_group	availability_365
1	Brooklyn	100.232292
2	Manhattan	111.979410
3	Queens	144.451818
0	Bronx	165.758937
4	Staten Island	199.678284

```
In [33]: sns.barplot(x='neighbourhood_group',y='availability_365',data=available_room)
plt.xlabel('neighbourhood_group',fontweight='bold')
plt.ylabel('availability_365',fontweight='bold')
plt.show()
```



In []: *#less room availability in brooklyn and greater room availability in Staten island compare to other.less room availabil*