Berlin AirBnB Dataset

BY: Tanishka Shah

Data Analysis

Imports

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import OneHotEncoder
pd.options.display.max_columns = None
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
```

Reading The Data and basic display

```
data = pd.read_csv("/kaggle/input/berlin-airbnb-data/listings.csv")
```

```
data
```

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_n
0	2015	Berlin-Mitte Value! Quiet courtyard/very central	2217	Ian	Mitte	Brunnenstr. Süd	52.534537	13.402557	Entire home/apt	60	
1	2695	Prenzlauer Berg close to Mauerpark	2986	Michael	Pankow	Prenzlauer Berg Nordwest	52.548513	13.404553	Private room	17	
2	3176	Fabulous Flat in great Location	3718	Britta	Pankow	Prenzlauer Berg Südwest	52.534996	13.417579	Entire home/apt	90	
3	3309	BerlinSpot Schöneberg near KaDeWe	4108	Jana	Tempelhof - Schöneberg	Schöneberg-Nord	52.498855	13.349065	Private room	26	
4	7071	BrightRoom with sunny greenview!	17391	Bright	Pankow	Helmholtzplatz	52.543157	13.415091	Private room	42	
...	...	...	...	...	...	...	...	...	...	...	
22547	29856708	Cozy Apartment right in the center of Berlin	87555909	Ulisses	Mitte	Brunnenstr. Süd	52.533865	13.400731	Entire home/apt	60	
22548	29857108	Altbau/ Schöneberger Kiez / Schlafsofa	67537363	Jörg	Tempelhof - Schöneberg	Schöneberg-Nord	52.496211	13.341738	Shared room	20	
22549	29864272	Artists loft with garden in the center of Berlin	3146923	Martin	Pankow	Prenzlauer Berg Südwest	52.531800	13.411999	Entire home/apt	85	
22550	29866805	Room for two with private shower / WC	36961901	Arte Luise	Mitte	Alexanderplatz	52.520802	13.378688	Private room	99	
22551	29867352	Sunny, modern and cozy flat in Berlin Neukölln :)	177464875	Sebastian	Neukölln	Schillerpromenade	52.473762	13.424447	Private room	45	

22552 rows × 16 columns

```
data.describe()
```

	id	host_id	latitude	longitude	price	minimum_nights	number_of_reviews	reviews_per_month	calculated_
count	2.255200e+04	2.255200e+04	22552.000000	22552.000000	22552.000000	22552.000000	22552.000000	18638.000000	
mean	1.571560e+07	5.403355e+07	52.509824	13.406107	67.143668	7.157059	17.840679	1.135525	
std	8.552069e+06	5.816290e+07	0.030825	0.057964	220.266210	40.665073	36.769624	1.507082	
min	2.015000e+03	2.217000e+03	52.345803	13.103557	0.000000	1.000000	0.000000	0.010000	
25%	8.065954e+06	9.240002e+06	52.489065	13.375411	30.000000	2.000000	1.000000	0.180000	
50%	1.686638e+07	3.126711e+07	52.509079	13.416779	45.000000	2.000000	5.000000	0.540000	
75%	2.258393e+07	8.067518e+07	52.532669	13.439259	70.000000	4.000000	16.000000	1.500000	
max	2.986735e+07	2.245081e+08	52.651670	13.757642	9000.000000	5000.000000	498.000000	36.670000	

We can see that there are some irregularity in the data. Some points to be noted are the following -

1. Most features have same number of rows except for reviews per month, it has some missing columns that need to be adjusted.
2. For the pirce feature we can see that the mean is 67 that is acceptable but we can also see extremes on both ends, such as the max price is 9000 and minimum is 0. This caused the variance and std to increase for the price data, it requires further observation.
3. Similar extreme unacceptable values can be seen with minimum_nights as well as the max is 5000 nights which is practically impossible.

Let's explore the data a bit more

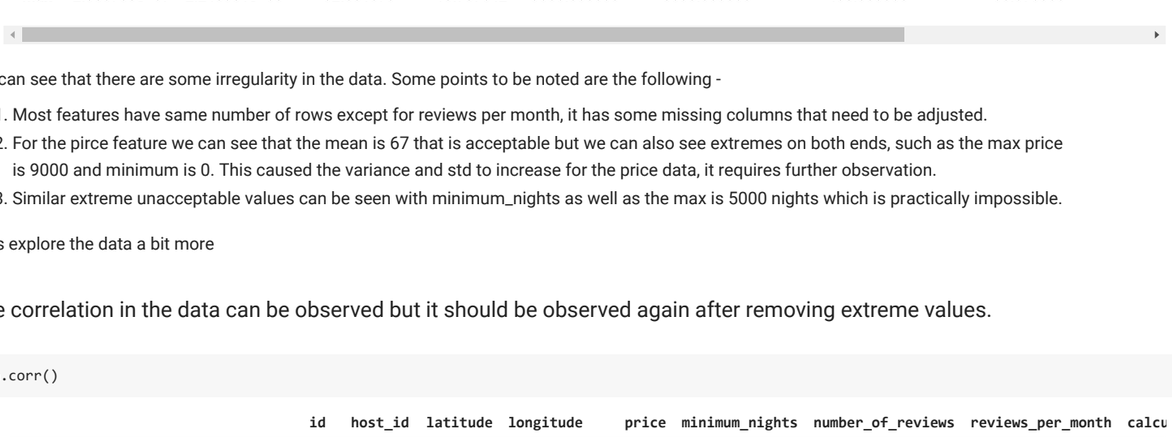
The correlation in the data can be observed but it should be observed again after removing extreme values.

```
data.corr()
```

	id	host_id	latitude	longitude	price	minimum_nights	number_of_reviews	reviews_per_month	calcu
id	1.000000	0.527680	0.006060	-0.018406	0.030992	-0.041777	-0.308877	0.250475	
host_id	0.527680	1.000000	0.007230	-0.047042	0.037808	-0.028396	-0.148577	0.203368	
latitude	0.006060	0.007230	1.000000	-0.107228	0.002181	0.011863	0.037973	0.042594	
longitude	-0.018406	-0.047042	-0.107228	1.000000	-0.042662	-0.026450	-0.020905	-0.041718	
price	0.030992	0.037808	0.002181	-0.042662	1.000000	0.003626	-0.001235	0.010060	
minimum_nights	-0.041777	-0.028396	0.011863	-0.026450	0.003626	1.000000	-0.021685	-0.047410	
number_of_reviews	-0.308877	-0.148577	0.037973	-0.020905	-0.001235	-0.021685	1.000000	0.556738	
reviews_per_month	0.250475	0.203368	0.042594	-0.041718	0.010060	-0.047410	0.556738	1.000000	
calculated_host_listings_count	0.038248	-0.048570	0.039785	-0.044122	0.073244	0.027285	0.085898	0.121327	
availability_365	-0.059485	-0.009365	0.012116	-0.086532	0.105959	0.099350	0.277621	0.232556	

```
plt.figure(figsize=(30,10))
sns.heatmap(data=data.corr(), annot=True)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd8e6cba090>
```



No strong correlation can be observed between the data. However we will look into each of the features and look for patterns amongst the data. After which we will process the data a bit and then check for correlations again.

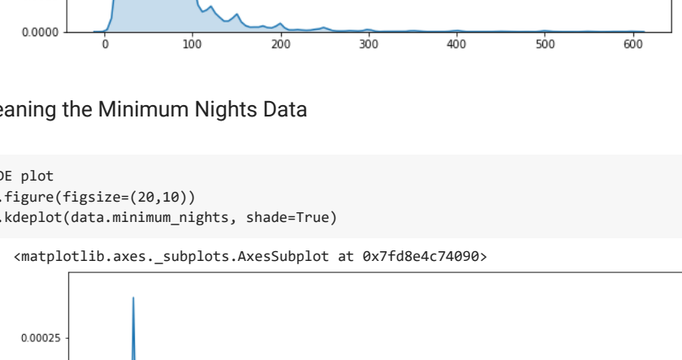
Cleaning The Data

1. We will clean the data for extreme values of prices and minimum_nights. We can see that the mean for the price is 67 and std is 200 so we can assume prices that are over 1000 are extremes and we will remove them. also the prices that are 0 will be removed as we will not consider any listing to be free.

Cleaning the Price Data

```
# KDE plot
plt.figure(figsize=(10,5))
sns.kdeplot(data=data.price, shade=True)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd8e46ce050>
```



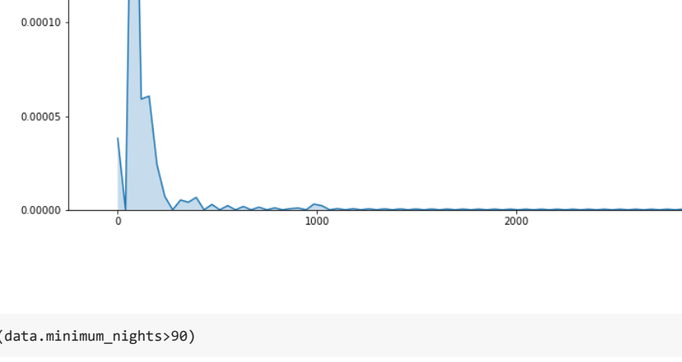

```
sum(data.price == 0) + sum(data.price >600)
```

```
64
```

```
data = data[data.price != 0 ]
data = data[data.price <=600 ]
```

```
# KDE plot
plt.figure(figsize=(10,5))
sns.kdeplot(data=data.price, shade=True)
```

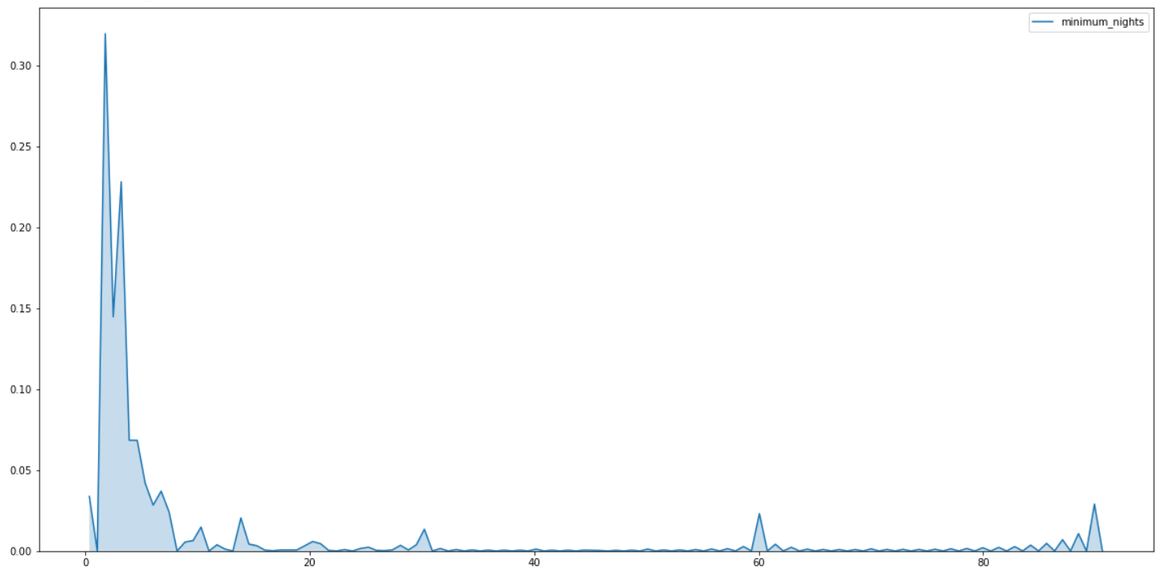
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd8e4ce0b50>
```



Cleaning the Minimum Nights Data

```
# KDE plot
plt.figure(figsize=(20,10))
sns.kdeplot(data.minimum_nights, shade=True)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd8e4c74090>
```




```
sum(data.minimum_nights>90)
```

```
148
```

```
data = data[data.minimum_nights <=90 ]
```

```
# KDE plot
plt.figure(figsize=(20,10))
sns.kdeplot(data.minimum_nights, shade=True)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd8e4c5e0d0>
```


Analysis of ID, Name, Host ID and Hostname

Our main focus is on id and host id as we assume that the names refer to one of the IDs or host ids. We can see that there are fewer number of host id, that means soem hosts have multiple listings(rooms/apartments).

```
sum(data.host_id.isnull()==True)
```

```
0
```

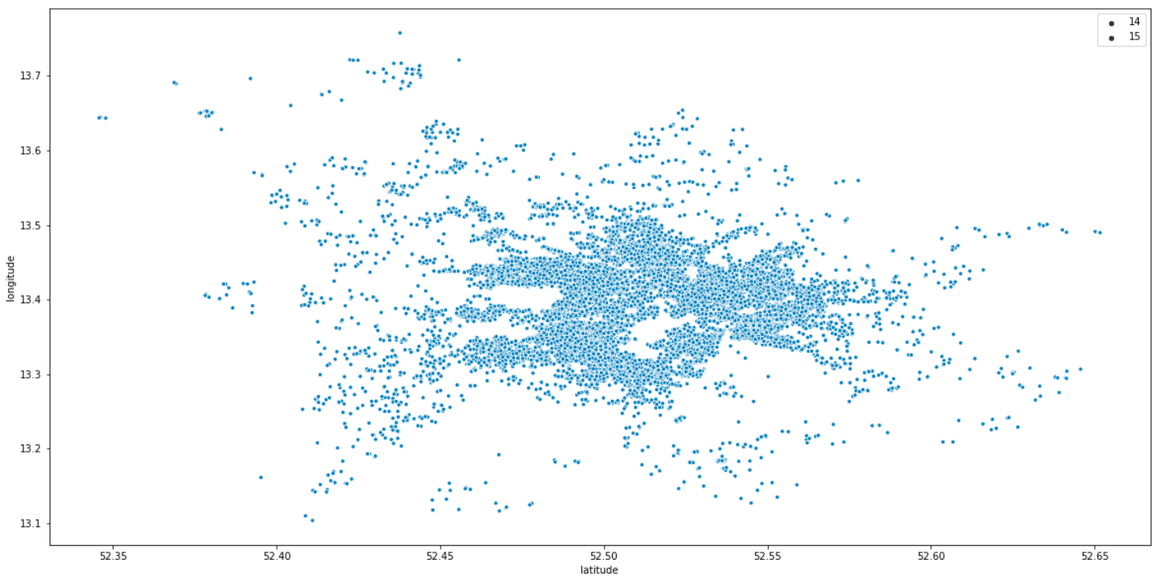
```
print("Unique ID : ", len(data.id.unique()))
print("Unique host ID : ", len(data.host_id.unique()))
```

```
Unique ID :  22340
Unique host ID :  19058
```

Geo-Spatial Visualization of the Data

We will use the longitude and lattitude to plot the geospatial data. For some of the upcoming features we will also plot them geospatially to see if patterns can be identified.

```
plt.figure(figsize=(20,10))

sns.scatterplot(x=data['latitude'], y=data['longitude'],size=15, color=sns.color_palette('winter', n_colors=1))
plt.show()
```



Analysis of Neighbourhood Groups

```
data.neighbourhood_group.isnull().any()
```

```
False
```

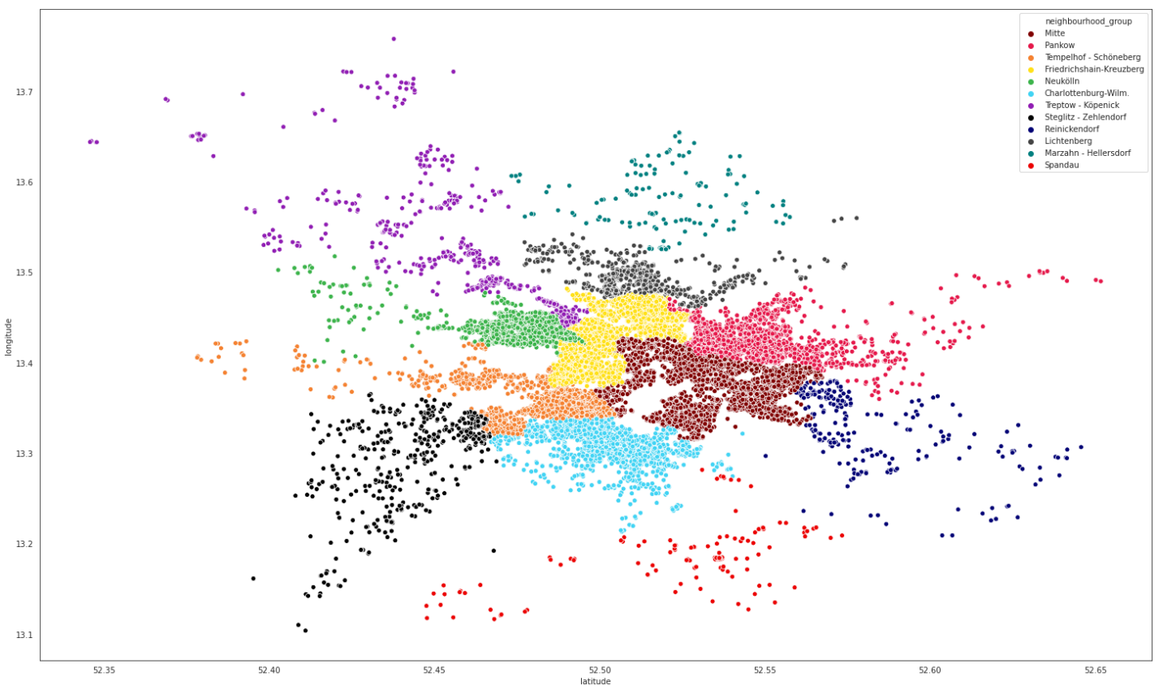
Geographical View of the neighbourhoods

Fantastic!!! This is brilliant. We have a good representation of the neighbourhood data, we can see the different neighbourhoods locations, the density of listings in the neighbourhoods. The points that we can draw are-

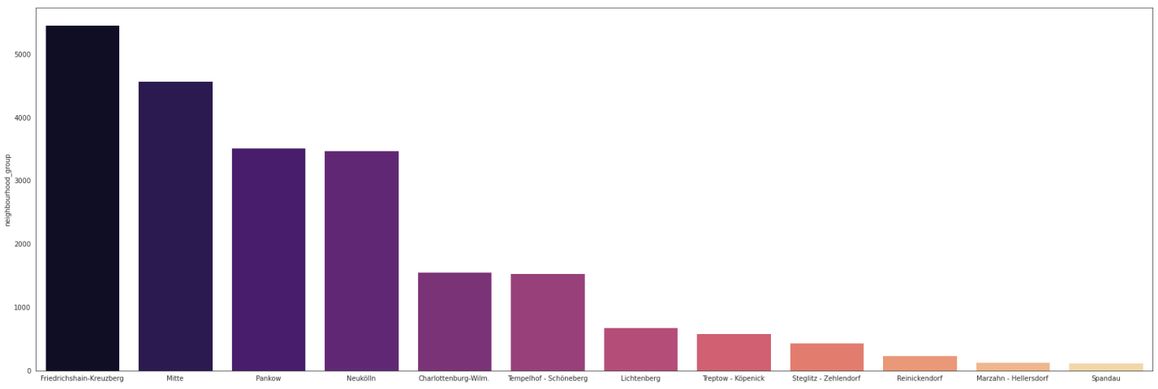
1. There are more listings in the central neighbourhoods(We can also see from the value counts), which are mainly - Friedrichshain-Kreuzberg, Mitte, Pankow, Neukölln. These places are closer to the city center, therefore has more listings.
2. We can see that after the central neighbourhoods, The SOUTHERN NEIGHBOURHOODS have more density in listings which are Schoneberg and Charlottenberg. That means that residential facilities(housing,communication, markets) are more available on the SOUTHERN PART of the city.

We also have the smaller neighbourhood feature. We will look into it later, for now we want to analyse the larger neighbourhood groups more, in terms of relationships with other features

```
plt.figure(figsize=(25,15))
sns.set_style('white')
customPalette = ['#800000', '#e6194B', '#f58231', '#ffe119', '#3cb44b', '#42d4f4', '#911eb4', '#000000', '#000075', '#444444', '#008080', '#ec0101']
sns.scatterplot(x=data['latitude'], y=data['longitude'],hue=data["neighbourhood_group"], palette=sns.set_palette(customPalette))
plt.show()
```




```
plt.figure(figsize=(30,10))
sns.barplot(x=data.neighbourhood_group.value_counts().index, y=data.neighbourhood_group.value_counts(),palette=sns.color_palette('magma', n_colors=12))
plt.show()
```

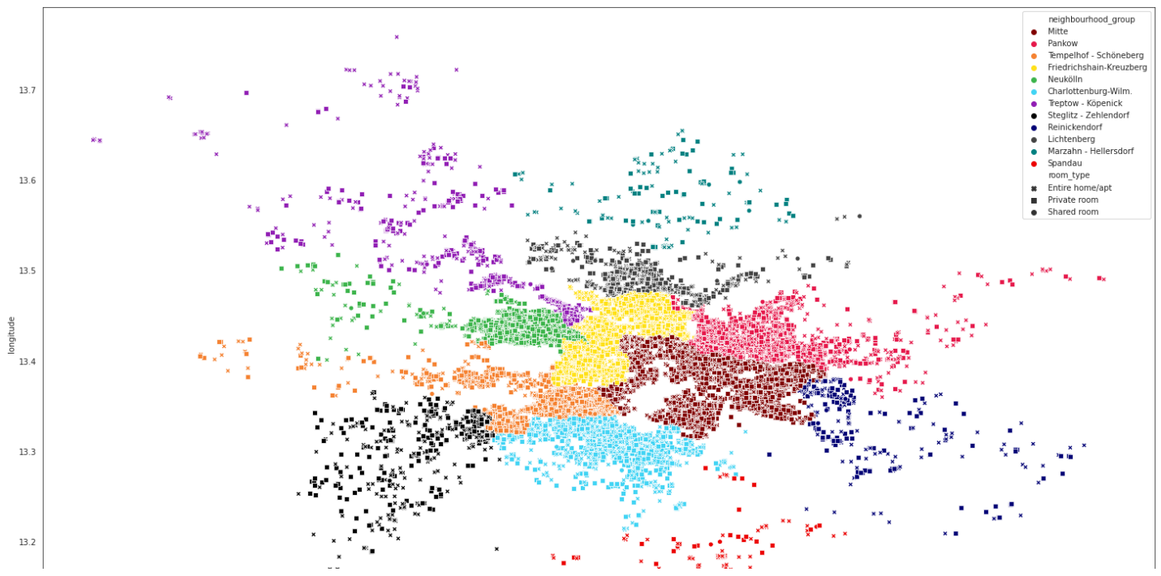



```
data.room_type.value_counts()
```

```
Private room       11473
Entire home/apt    10573
Shared room          294
Name: room_type, dtype: int64
```

It is difficult observe the relation between types of rooms and neighbourhood as it is very dense but it seems like Private rooms and Entire home/apt are evenly distributed as their numbers are roughly the same

```
plt.figure(figsize=(25,15))
markers = {"Private room": "s", "Entire home/apt": "X", "Shared room":"o"}
customPalette = ['#800000', '#e6194B', '#f58231', '#ffe119', '#3cb44b', '#42d4f4', '#911eb4', '#000000', '#000075', '#444444', '#008080', '#ec0101']
sns.scatterplot(x=data['latitude'], y=data['longitude'],hue=data["neighbourhood_group"], palette=sns.set_palette(customPalette),style=data['room_type'], markers=markers)
plt.show()
```



Now we want to see the relationship between price and the neighbourhood

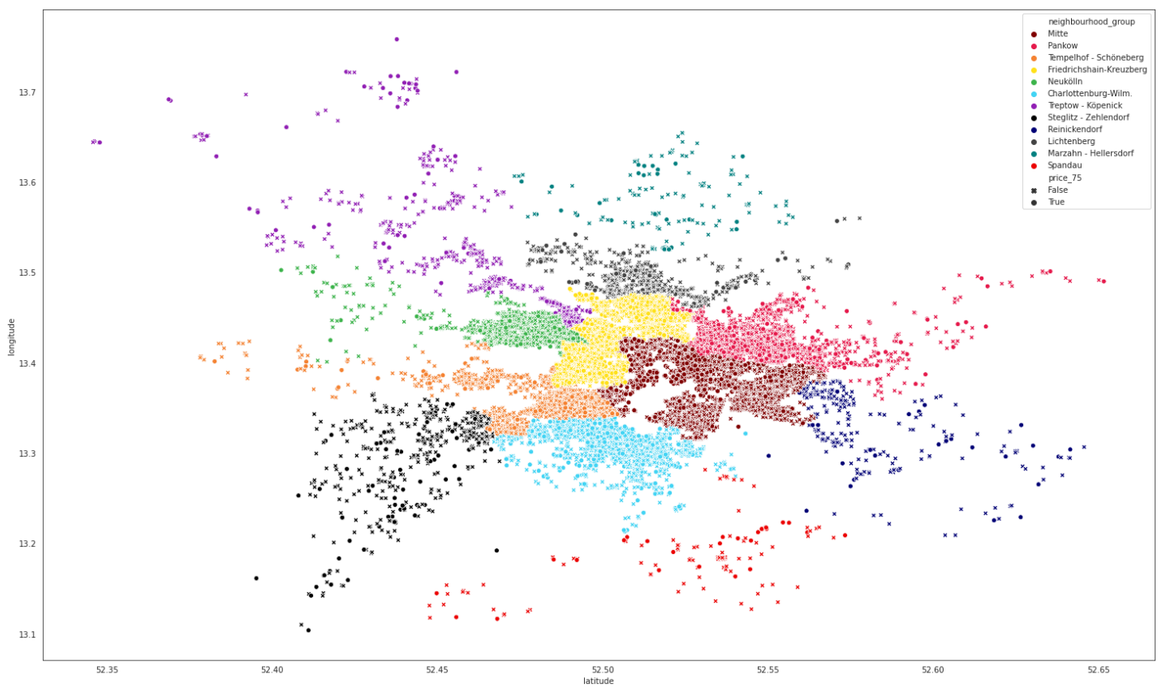
```
data.price.min()
```

```
1
```

```
temp = data
temp["price_75"] = data.price>data.price.quantile(0.75)
```

As it is hard to understand the pattern between neighbourhood and price from geaospatial view, we will try a different approach

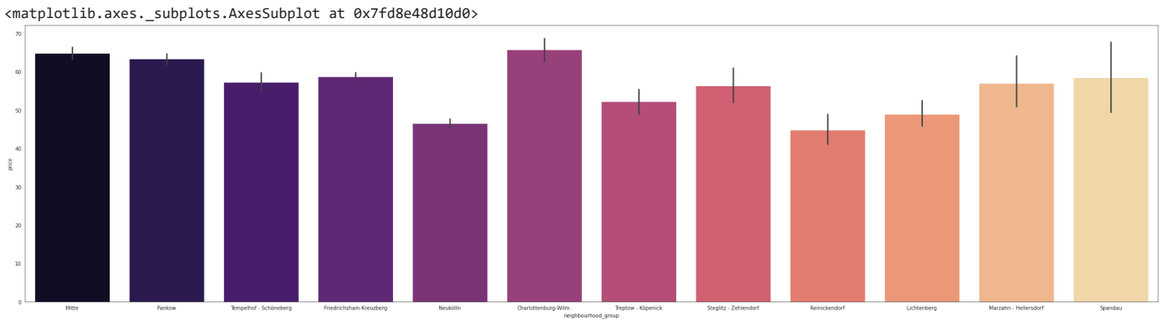
```
plt.figure(figsize=(25,15))
markers = {True: "o", False: "X"}

sns.scatterplot(x=temp['latitude'], y=temp['longitude'],hue=temp["neighbourhood_group"], palette=sns.set_palette(customPalette),style=temp['price_75'], markers=markers)
plt.show()
```



Overall comparison of price and neighbourhood

```
plt.figure(figsize=(40,10))
sns.barplot(x=data['neighbourhood_group'], y=data['price'], palette=sns.color_palette('magma', n_colors=12))
```

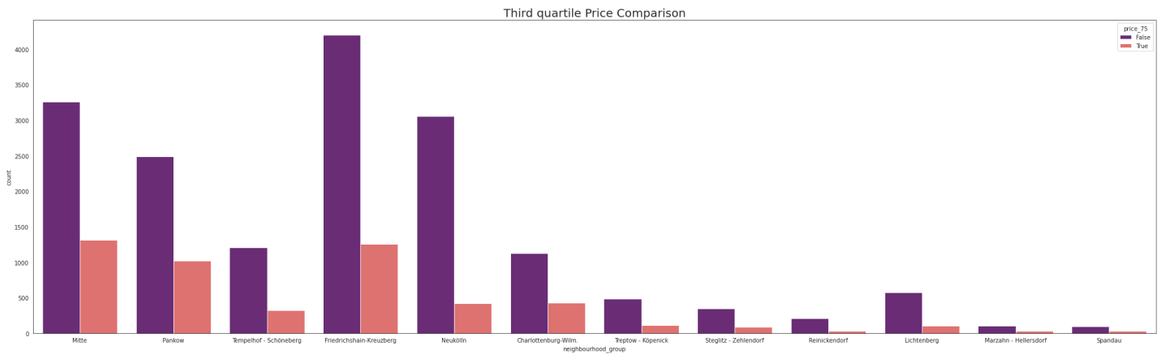
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd8e48d10d0>
```



From the overall view, much cannot be observed, seems like the price in all neighbourhoods are almost the same. Two neighbourhoods, charlotteberg and Schoneberg seems to have relatively higher price. We may see better patterns if we look at count of listings comparing the min, mix, third quantile. However, we cannot compare with min and max, as they are extreme values, but hopefully the top third quartile value will give a good insight.

Comparing counts of third quantile of price and Neighbourhood

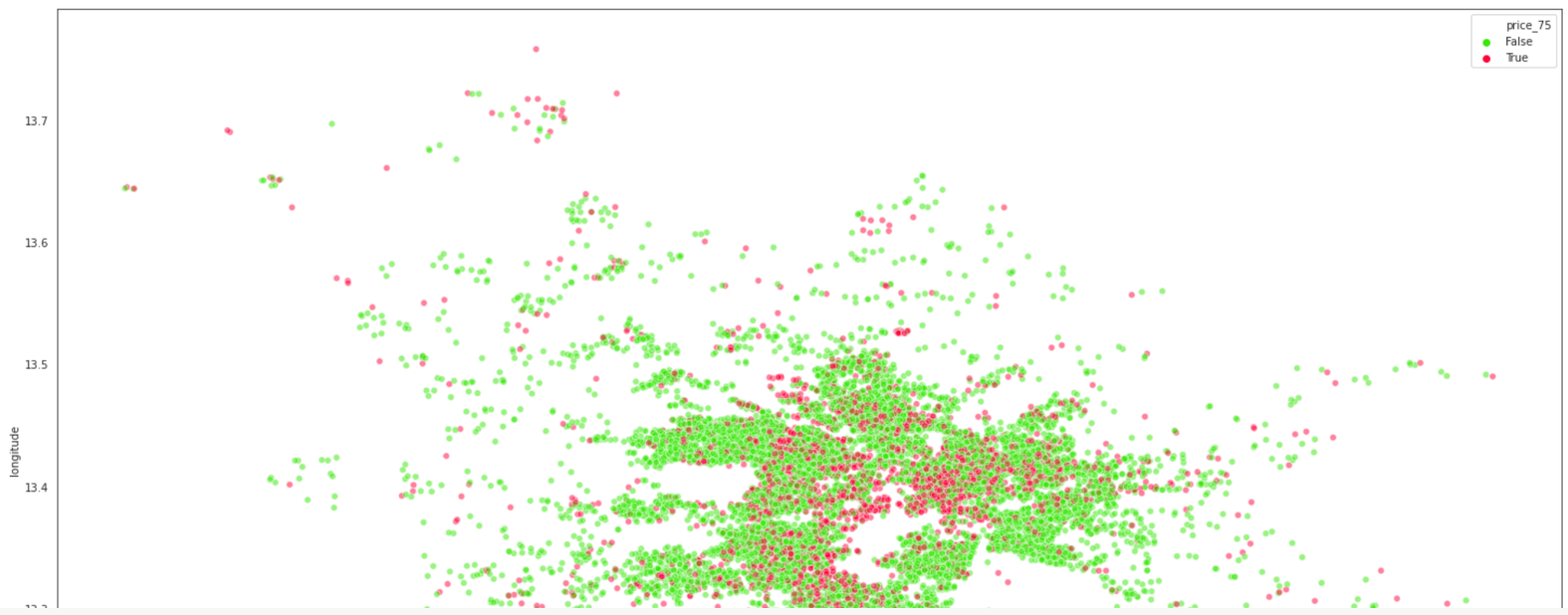
```
plt.figure(figsize=(35,10))
plt.title("Third quartile Price Comparison",fontsize=20)
sns.countplot(temp.neighbourhood_group,hue=temp["price_75"], palette=sns.color_palette('magma', n_colors=2))
plt.show()
```



Great !!! from the 3rd quantile we can get quite a few ideas and it supports the hypothesis of price being higher in city center more.

We can see that the listing which are in the neighbourhoods of the city center have a higher count of the listings costing more than the third quantile. It may imply that the price of the listings near the city center have a higher price. We can plot this geospatially and see.

```
plt.figure(figsize=(25,15))

sns.scatterplot(x=data['latitude'], y=data['longitude'],hue=temp.price_75, palette=sns.color_palette('prism', n_colors=2), alpha=0.5)
plt.show()
```




```
plt.figure(figsize=(35,10))
sns.swarmplot(x=data['neighbourhood_group'],
              y=data['price'])
```

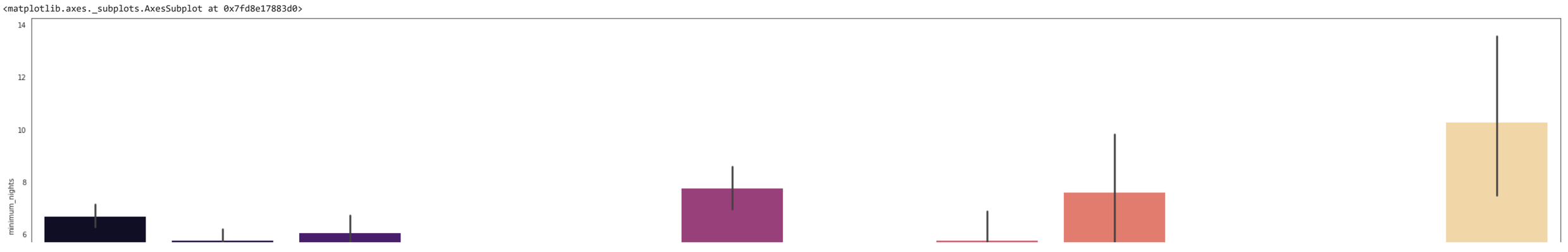
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd8e188e6d0>
```

Comparison of Minimum Nights and Neighbourhoods

As this will be hard to see from geospatial plotting we will use bar charts to display the data.

```
plt.figure(figsize=(40,10))
sns.barplot(x=data['neighbourhood_group'], y=data['minimum_nights'], palette=sns.color_palette('magma', n_colors=12))
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd8e17883d0>
```



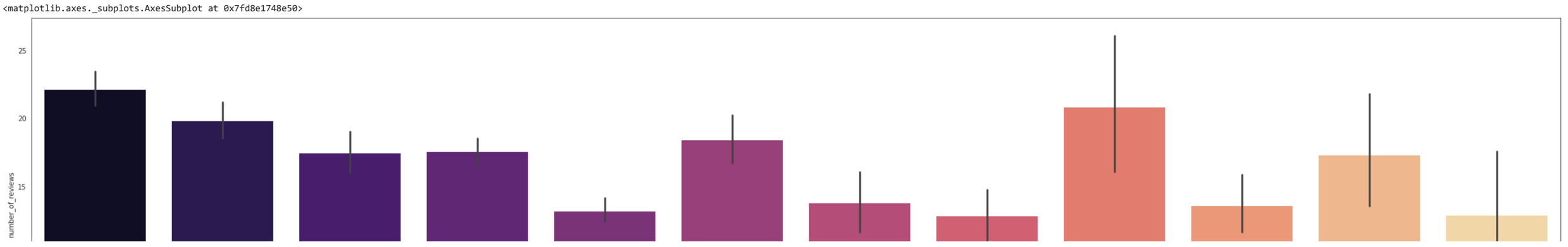
We can see that the neighbourhood Spandou has on average higher number of minimum nights of stay, a reason maybe due to it being far away from the city center. Also it has less dense listing, so we can see a alarge standard error in the data.

That's pretty much all that be observed from the data for comparison between the neighbourhood groups and minimum nights.

Observation between Reviews, Reviews per Month and Neighbourhood

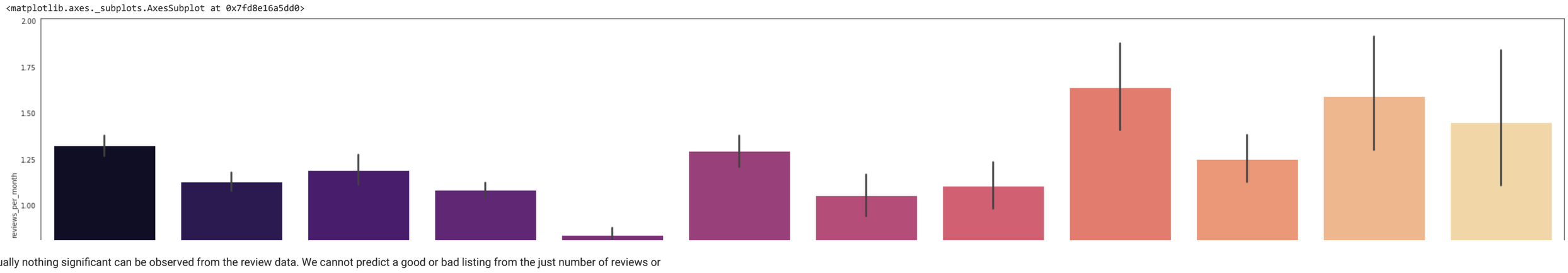
```
plt.figure(figsize=(40,10))
sns.barplot(x=data['neighbourhood_group'], y=data['number_of_reviews'], palette=sns.color_palette('magma', n_colors=12))
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd8e1748e50>
```




```
plt.figure(figsize=(40,10))
sns.barplot(x=data['neighbourhood_group'], y=data['reviews_per_month'], palette=sns.color_palette('magma', n_colors=12))
```

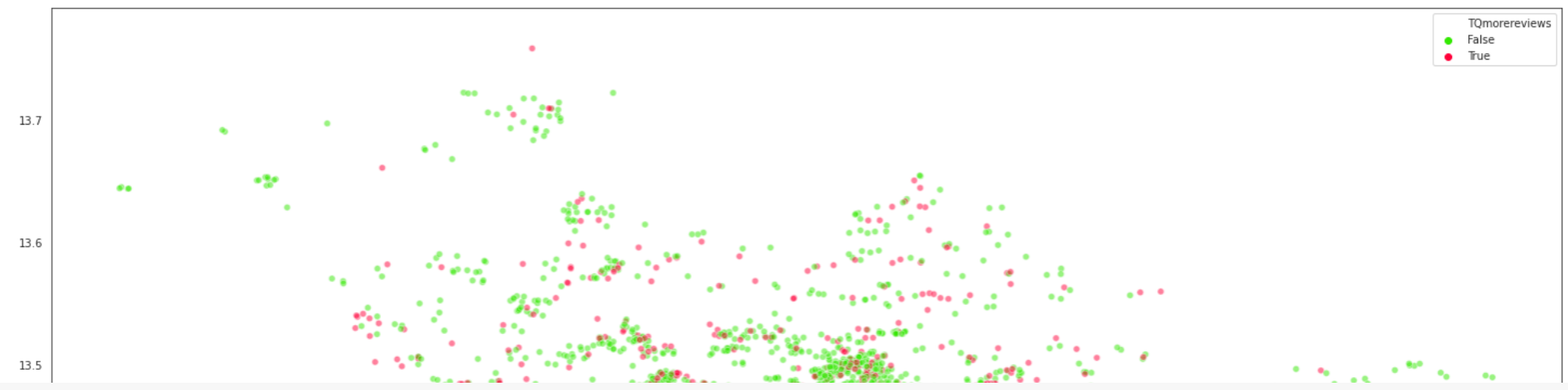
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd8e16a5dd0>
```



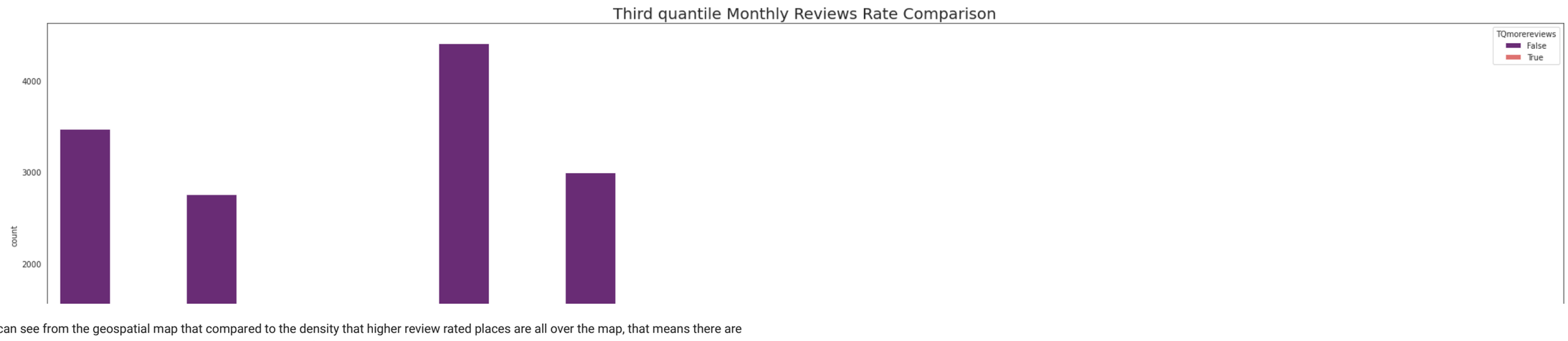
Actually nothing significant can be observed from the review data. We cannot predict a good or bad listing from the just number of reviews or the reviews rate. However this data may suggest that which neighbourhoods listings get more visitors as more reviews may mean more visitors. We can have a look at the third quantile range crossing review rate to see if we can observe any pattern in which neighbourhood listings are more busy.

```
data["TQmorereviews"] = data["reviews_per_month"]>data["reviews_per_month"].quantile(0.75)
```

```
plt.figure(figsize=(25,15))

sns.scatterplot(x=data['latitude'], y=data['longitude'],hue=data.TQmorereviews, palette=sns.color_palette('prism', n_colors=2), alpha=0.5)
plt.show()
```




```
plt.figure(figsize=(35,10))
plt.title("Third quantile Monthly Reviews Rate Comparison",fontsize=20)
sns.countplot(data.neighbourhood_group,hue=data["TQmorereviews"], palette=sns.color_palette('magma', n_colors=2))
plt.show()
```



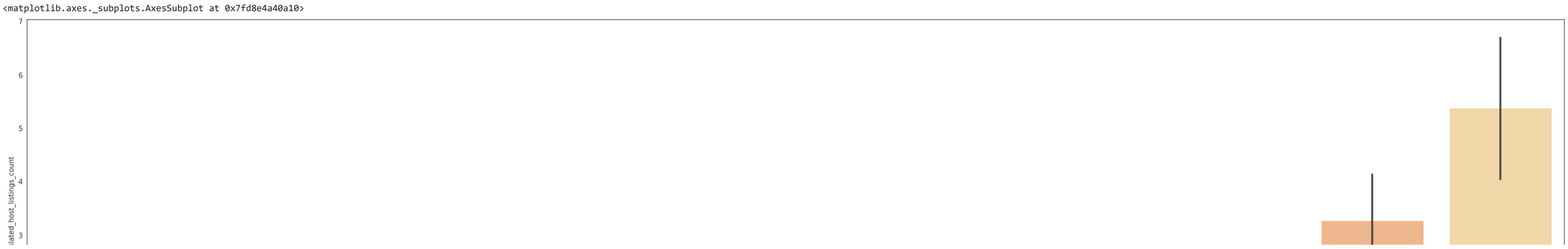
We can see from the geospatial map that compared to the density that higher review rated places are all over the map, that means there are places in every neighbourhood that receives higher number of reviews and it is not centralized. The Bar comparison shows that some neighbourhoods have higher counts of greater number of reviews but it is due to the fact that the density of listings in those regions are lower. If we look at the neighbourhoods far from the city center we see the ratio of false to true get lower.

Occupied Time of Listings Depending on the Neighbourhood - Relation of host listings count, availability and neighbourhood

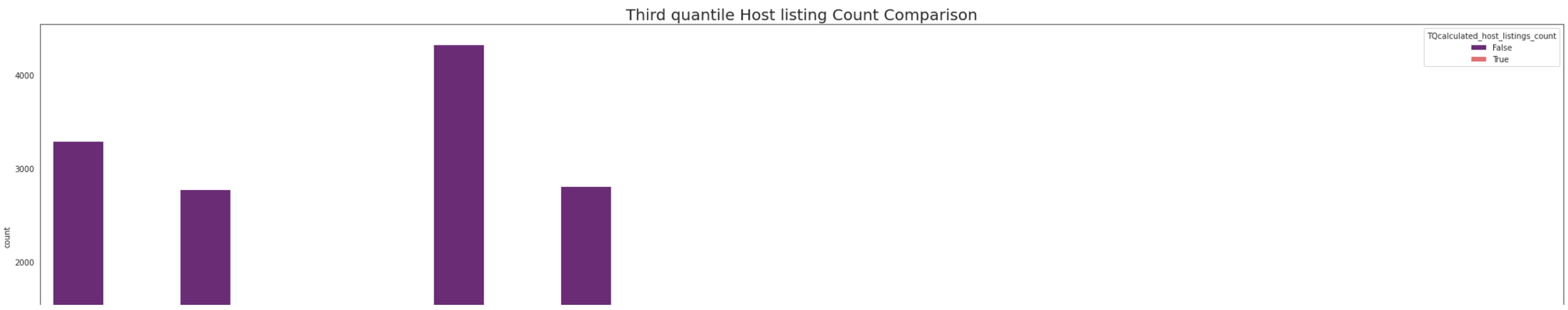
```
temp = data
temp["TQcalculated_host_listings_count"] = temp["calculated_host_listings_count"]>temp["calculated_host_listings_count"].quantile(0.75)
```

```
plt.figure(figsize=(40,10))
sns.barplot(x=data['neighbourhood_group'], y=data['calculated_host_listings_count'], palette=sns.color_palette('magma', n_colors=12))
```

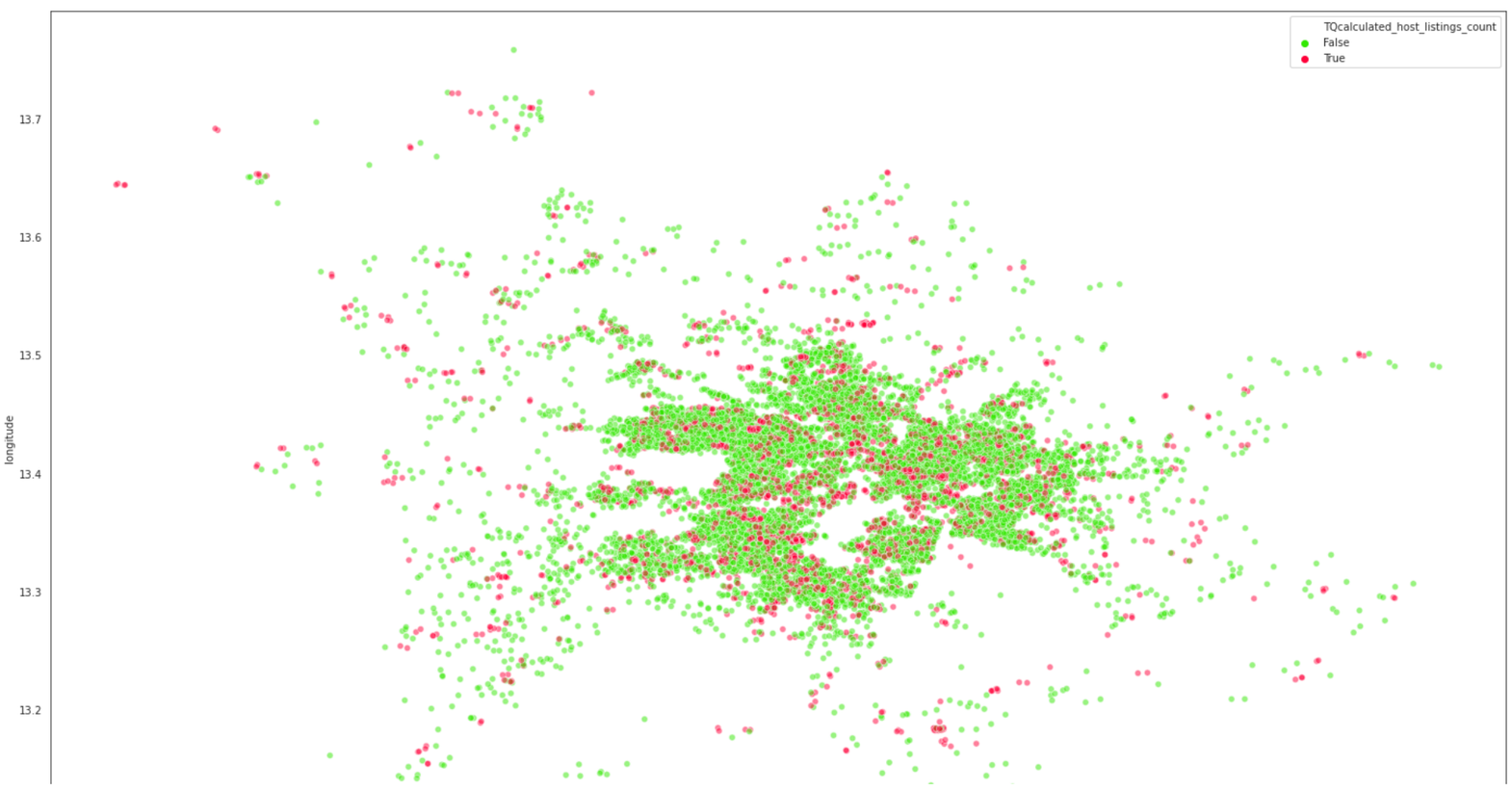
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd8e4a40a10>
```




```
plt.figure(figsize=(35,10))
plt.title("Third quantile Host listing Count Comparison",fontsize=20)
sns.countplot(temp.neighbourhood_group,hue=temp["TQcalculated_host_listings_count"], palette=sns.color_palette('magma', n_colors=2))
plt.show()
```



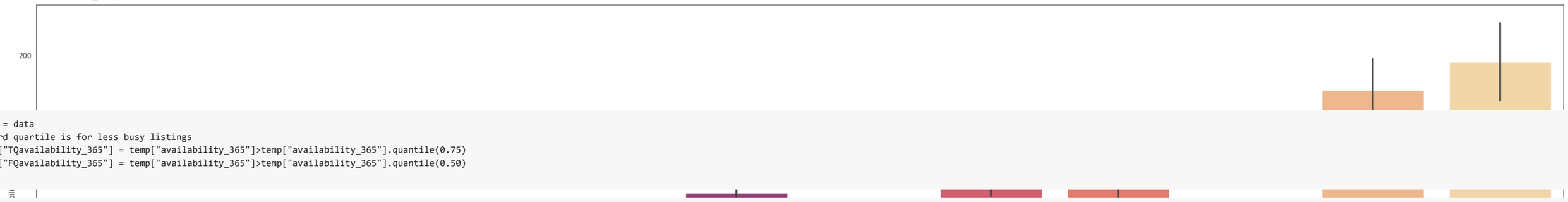

```
plt.figure(figsize=(25,15))

sns.scatterplot(x=temp['latitude'], y=temp['longitude'],hue=temp.TQcalculated_host_listings_count, palette=sns.color_palette('prism', n_colors=2), alpha=0.5)
plt.show()
```



We can see that the ratio of busy to not busy listings ratio is more on the neighbourhoods far from the center. It may be due to the fact that there are less options as you move far from the city center so you have to pick the local best option. This makes the loacally best listings to get more visitors.

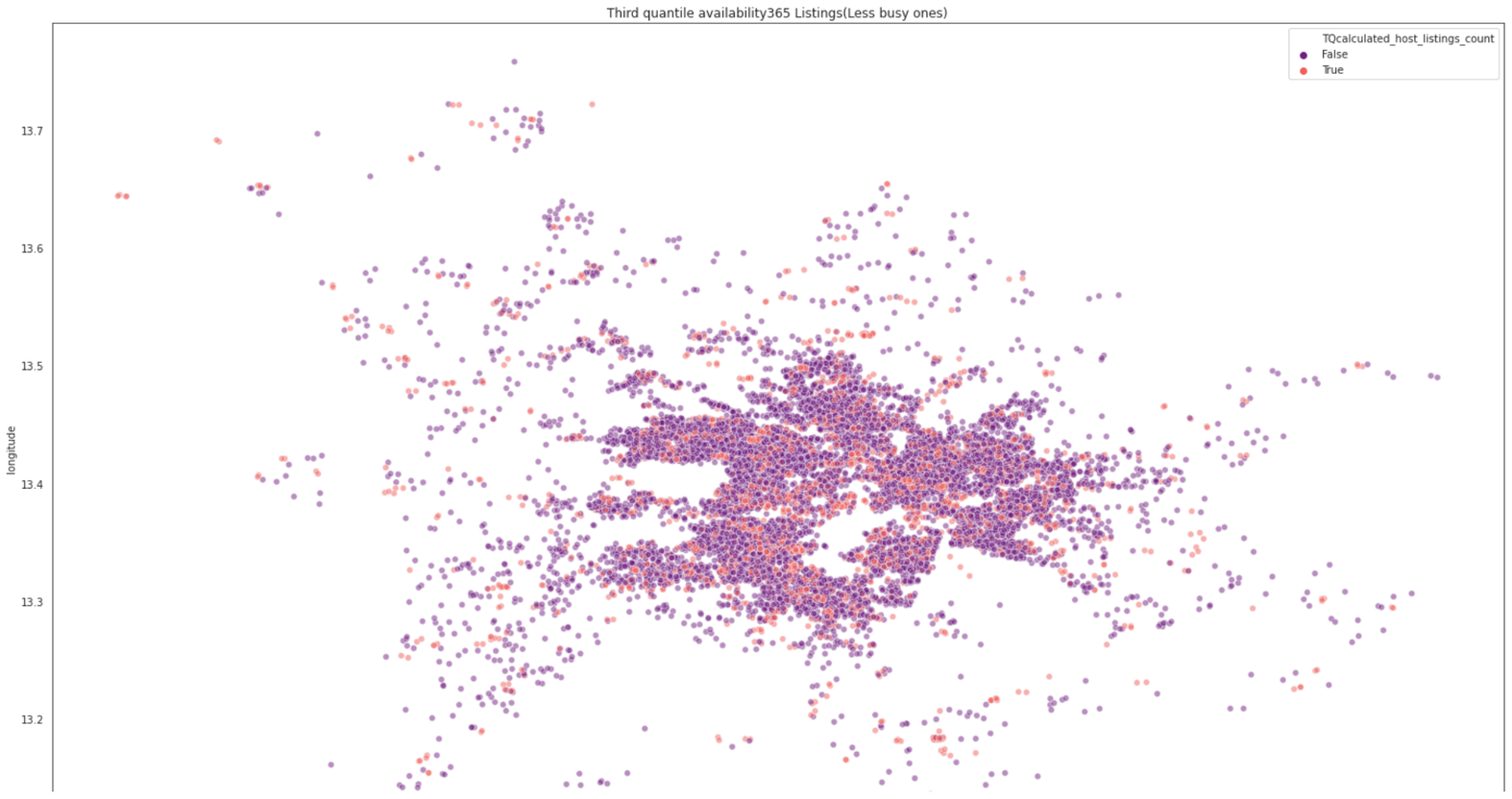
```
plt.figure(figsize=(40,10))
sns.barplot(x=data['neighbourhood_group'], y=data['availability_365'], palette=sns.color_palette('magma', n_colors=12))
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd8e4b3f510>
```

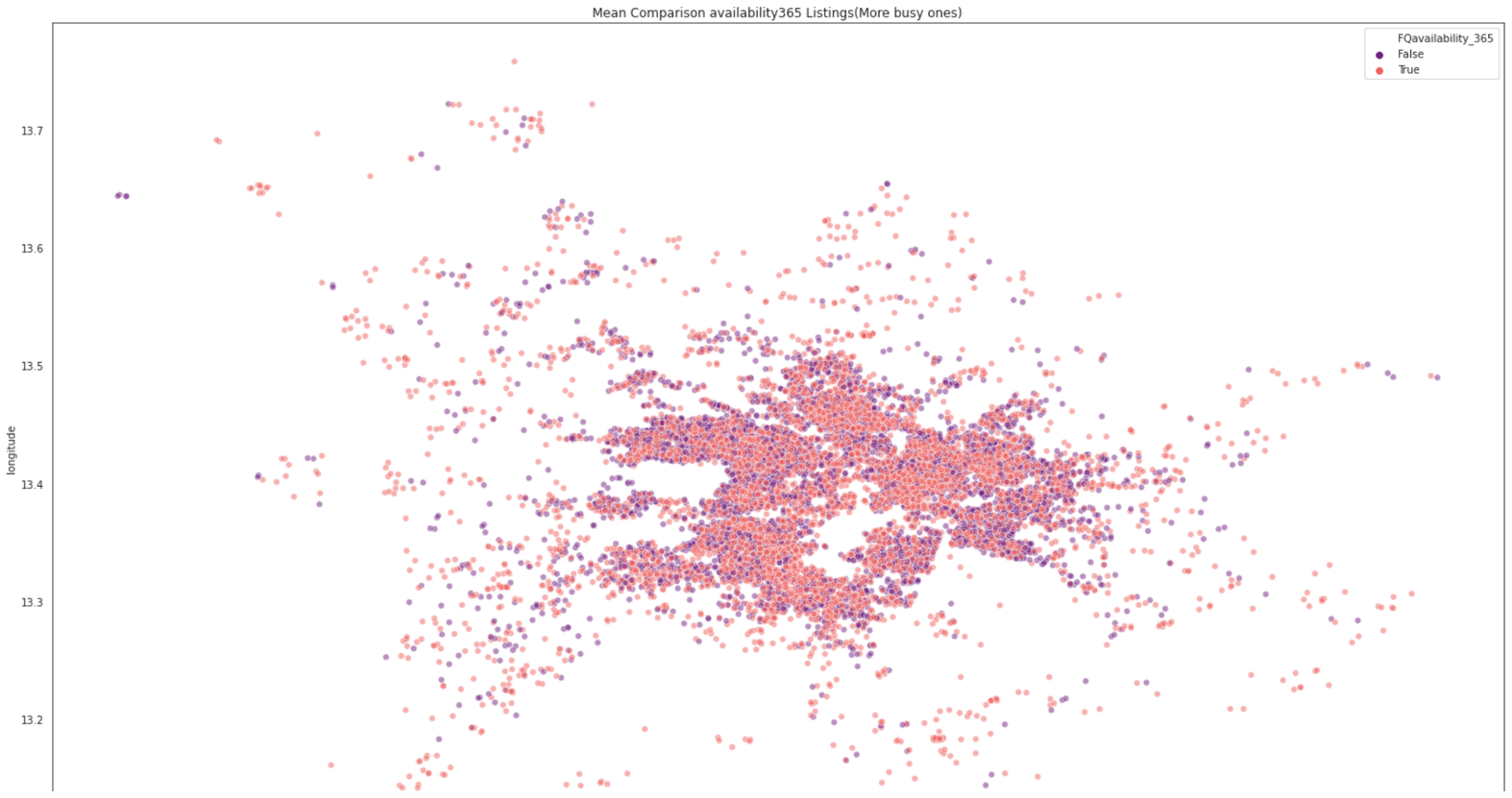



```
temp = data
#Third quartile is for less busy listings
temp["TQavailability_365"] = temp["availability_365"]>temp["availability_365"].quantile(0.75)
temp["FQavailability_365"] = temp["availability_365"]>temp["availability_365"].quantile(0.50)
```

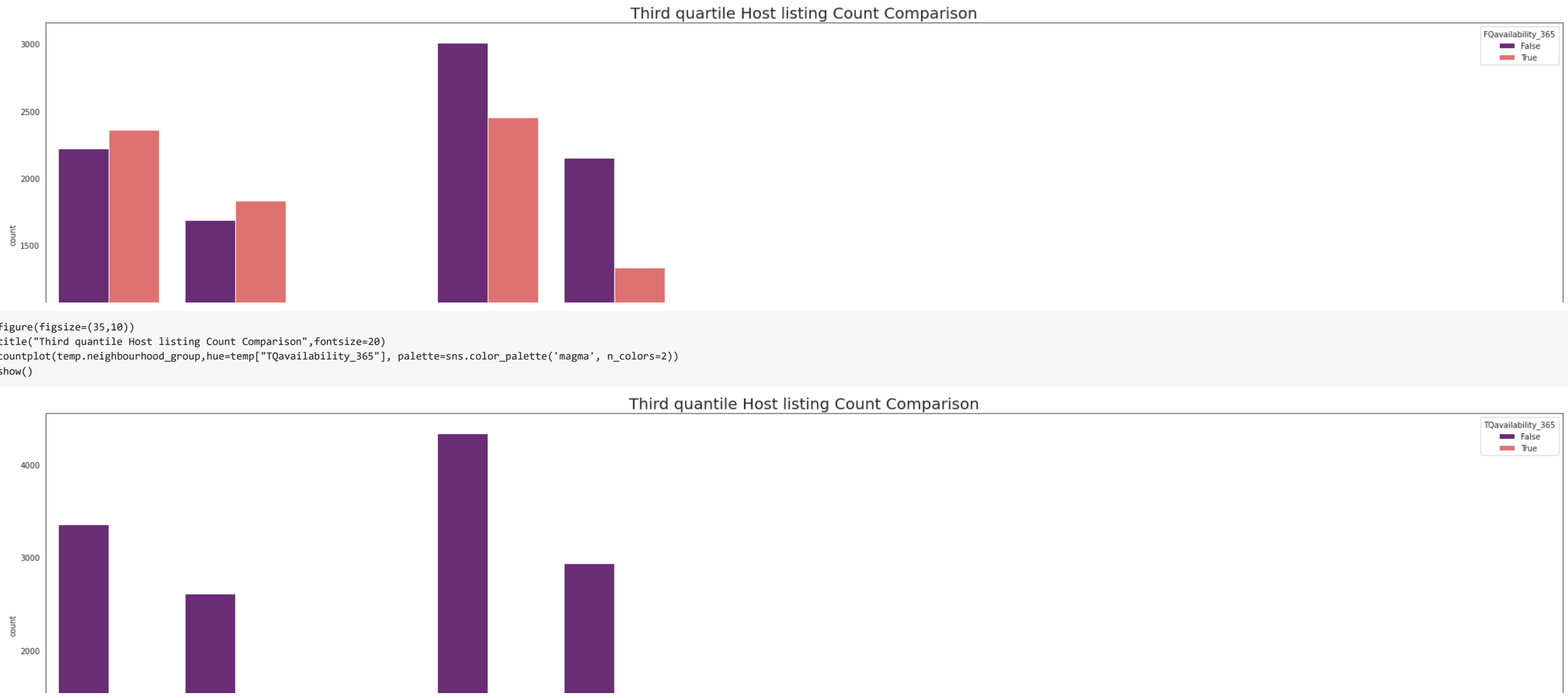
```
plt.figure(figsize=(25,15))
plt.title("Third quantile availability365 Listings(Less busy ones)")
sns.scatterplot(x=temp['latitude'], y=temp['longitude'],hue=temp.TQcalculated_host_listings_count, palette=sns.color_palette('magma', n_colors=2), alpha=0.5)
plt.show()
```




```
plt.figure(figsize=(25,15))
plt.title("Mean Comparison availability365 Listings(More busy ones)")
sns.scatterplot(x=temp['latitude'], y=temp['longitude'],hue=temp.FQavailability_365, palette=sns.color_palette('magma', n_colors=2), alpha=0.5)
plt.show()
```




```
plt.figure(figsize=(35,10))
plt.title("Third quantile Host listing Count Comparison",fontsize=20)
sns.countplot(temp.neighbourhood_group,hue=temp["FQavailability_365"], palette=sns.color_palette('magma', n_colors=2))
plt.show()
```




```
plt.figure(figsize=(35,10))
plt.title("Third quantile Host listing Count Comparison",fontsize=20)
sns.countplot(temp.neighbourhood_group,hue=temp["TQavailability_365"], palette=sns.color_palette('magma', n_colors=2))
plt.show()
```


It seems like that the listings that are near the city center are more available that means that they less number of hosts throughout the year. We can see the centra; most neighbourhood kreuzberg has more listings on average free than the busy ones. This maybe due to the fact that there is more competition in the city center and also there are hotels near the city center therefore they get less hosts and are more free.

It is hard to see from the third quartile mapping but if we see from the mean mapping, we see that on the outskirts of the city very few listings have availability higher than the mean which means they are more occupied with hosts.

```
data.head()
```

	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	price_75	TQmorereviews	TQcalculated_host_listings_count	TQavailab
0	2015	Berlin-Mitte Value! Quiet courtyard/very central	2217	Ian	Mitte	Brunnenstr. Süd	52.534537	13.402557	Entire home/apt	60	4	118	2018-10-28	3.76	4	141	False	True	True	
1	2695	Prenzlauer Berg close to Mauerpark	2986	Michael	Pankow	Prenzlauer Berg Nordwest	52.548513	13.404553	Private room	17	2	6	2018-10-01	1.42	1	0	False	False	False	
2	3176	Fabulous Flat in great Location	3718	Britta	Pankow	Prenzlauer Berg Südwest	52.534996	13.417579	Entire home/apt	90	62	143	2017-03-20	1.25	1	220	True	False	False	
3	3309	BerlinSpot Schöneberg near KaDeWe	4108	Jana	Tempelhof - Schöneberg	Schöneberg-Nord	52.498855	13.349065	Private room	26	5	25	2018-08-16	0.39	1	297	False	False	False	
4	7071	BrightRoom with sunny greenview!	17391	Bright	Pankow	Helmholtzplatz	52.543157	13.415091	Private room	42	2	197	2018-11-04	1.75	1	26	False	True	False	

```
data = data.drop(["price_75","TQmorereviews","TQcalculated_host_listings_count","TQavailability_365","FQavailability_365"], axis=1)
```

We have analysed the data for the larger negihbourhood groups. Now we can have a look at the smaller groups and see if there are any patterns with the prices.

We will take 4 Larger neighbourhoods and analyse the price distribution in those geographical data

```
data.neighbourhood_group.value_counts()
```

```
Friedrichshain-Kreuzberg    5461
Mitte                       4576
Pankow                      3516
Neukölln                    3482
Charlottenburg-Wilm.        1556
Tempelhof - Schöneberg      1536
Lichtenberg                  680
Treptow - Köpenick           591
Steglitz - Zehlendorf        436
Reinickendorf                244
Marzahn - Hellersdorf        139
Spandau                      123
Name: neighbourhood_group, dtype: int64
```

Friedrichshain-Kreuzberg Data

```
data.shape
```

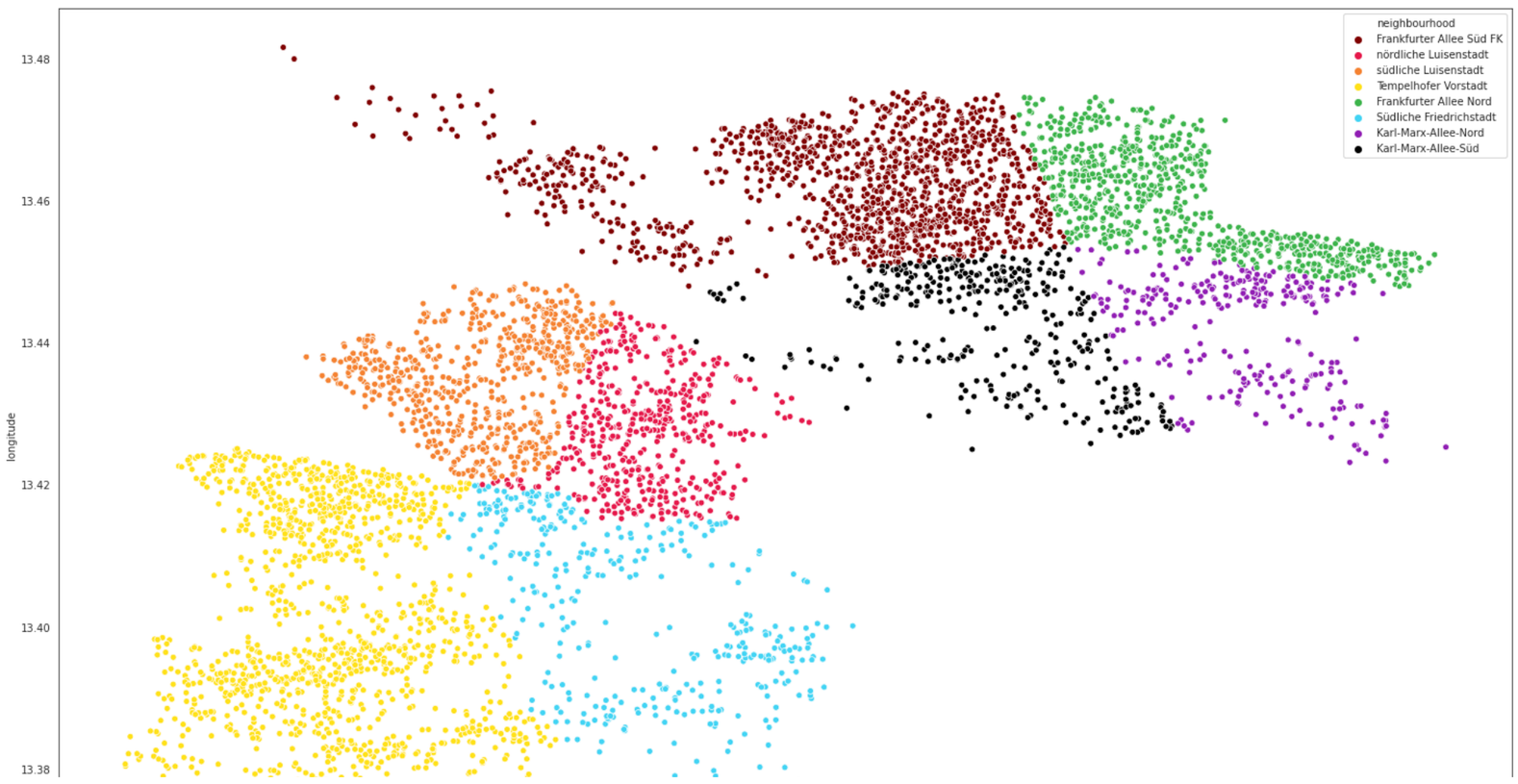
```
(22340, 16)
```

```
Kreu_data = data[data.neighbourhood_group == "Friedrichshain-Kreuzberg"]
```

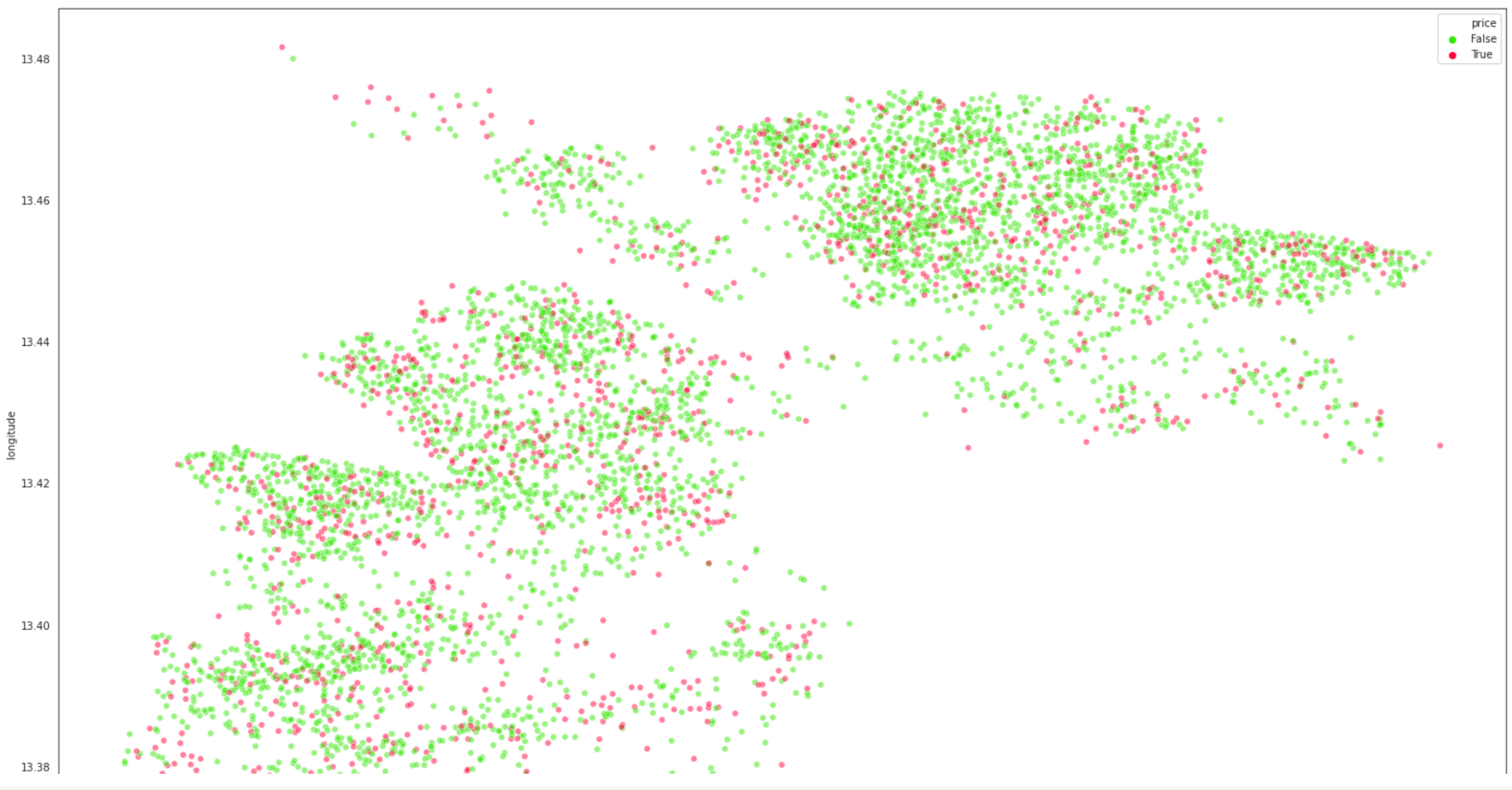
```
Kreu_data.shape
```

```
(5461, 16)
```

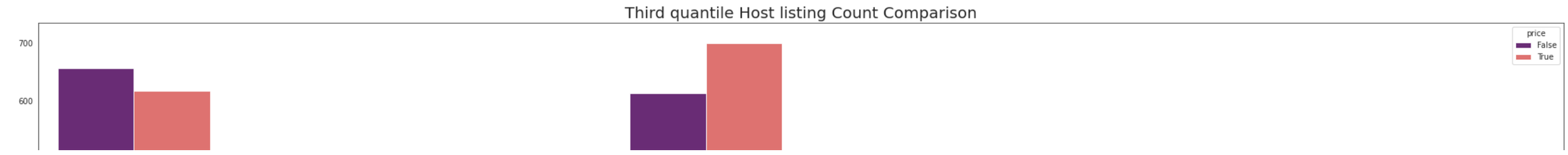
```
plt.figure(figsize=(25,15))
sns.set_style('white')
customPalette = ['#800000', '#e6194B', '#f58231', '#ffe119', '#3cb44b', '#42d4f4', '#911eb4', '#000000', '#000075', '#444444', '#008080', '#ec0101']
sns.scatterplot(x=Kreu_data['latitude'], y=Kreu_data['longitude'],hue=Kreu_data["neighbourhood"], palette=sns.set_palette(customPalette))
plt.show()
```




```
plt.figure(figsize=(25,15))

sns.scatterplot(x=Kreu_data['latitude'], y=Kreu_data['longitude'],hue=Kreu_data.price>Kreu_data.price.quantile(0.75), palette=sns.color_palette('prism', n_colors=2), alpha=0.5)
plt.show()
```




```
plt.figure(figsize=(35,10))
plt.title("Third quantile Host listing Count Comparison",fontsize=20)
sns.countplot(Kreu_data.neighbourhood,hue=Kreu_data.price>Kreu_data.price.quantile(0.5), palette=sns.color_palette('magma', n_colors=2))
plt.show()
```



We can see that for Kreuzberg, the listings price are well distributed, no region specifically has more expensive listings.

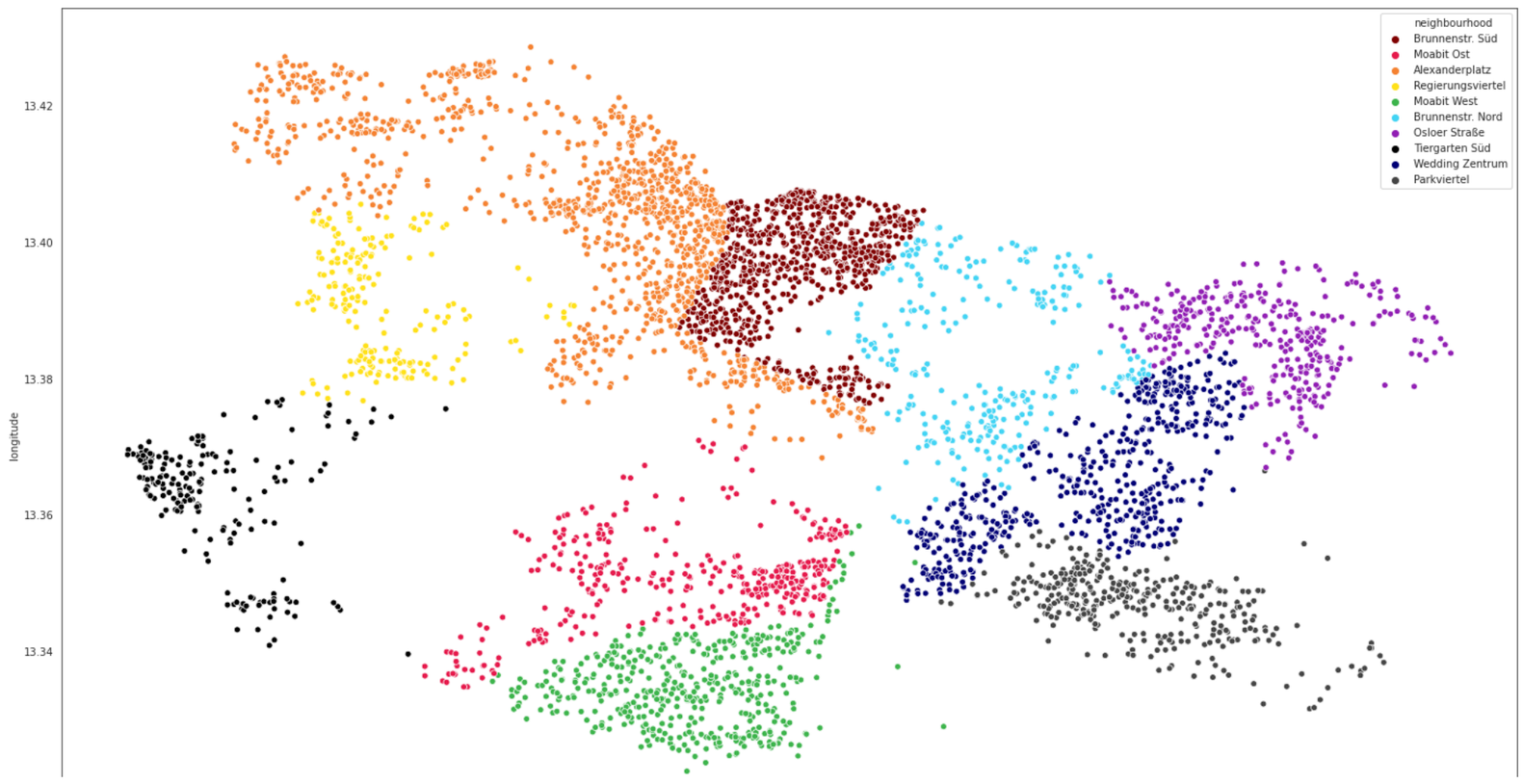
Mitte Data

```
mitte_data = data[data.neighbourhood_group == "Mitte"]
```

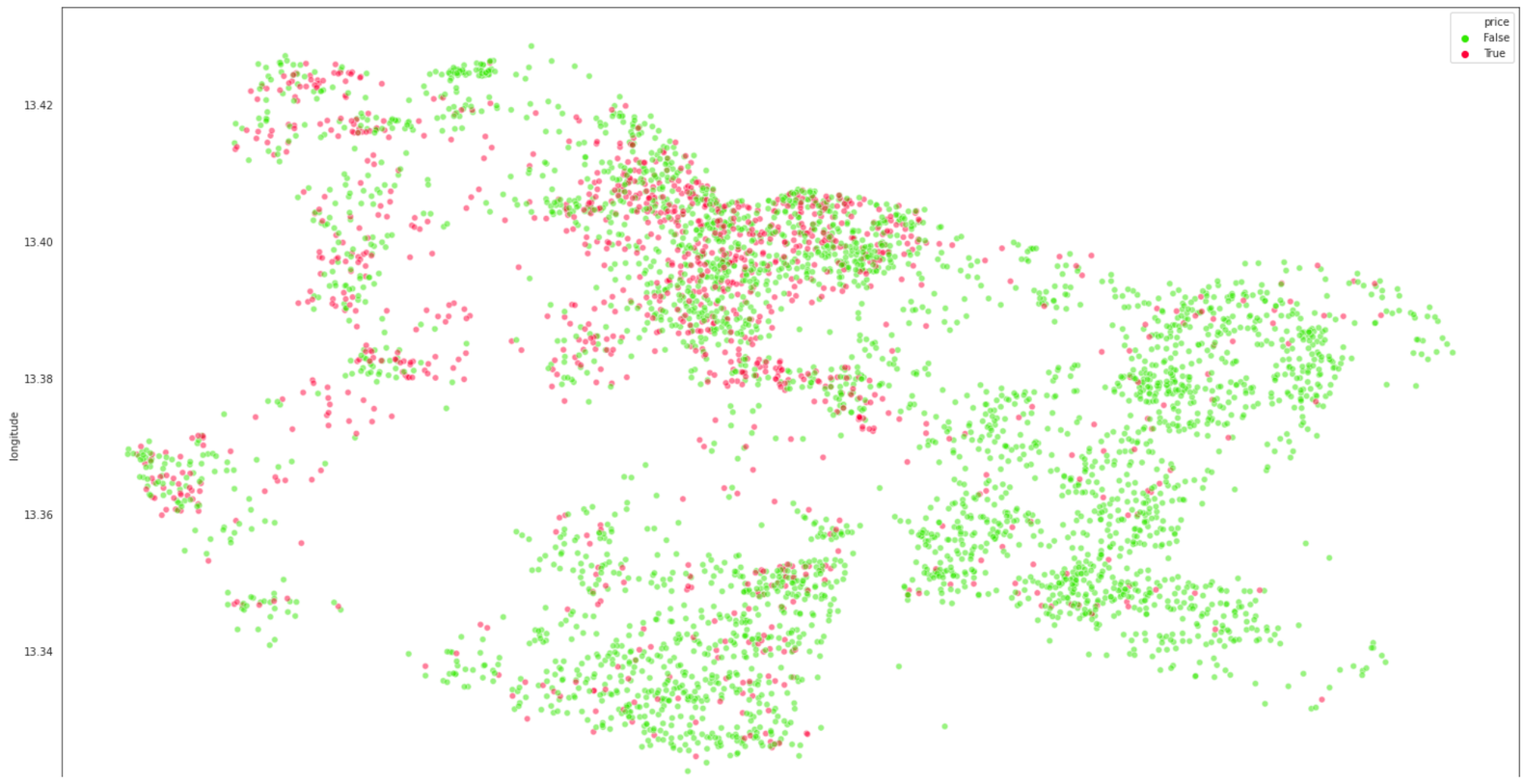
```
mitte_data.shape
```

```
(4576, 16)
```

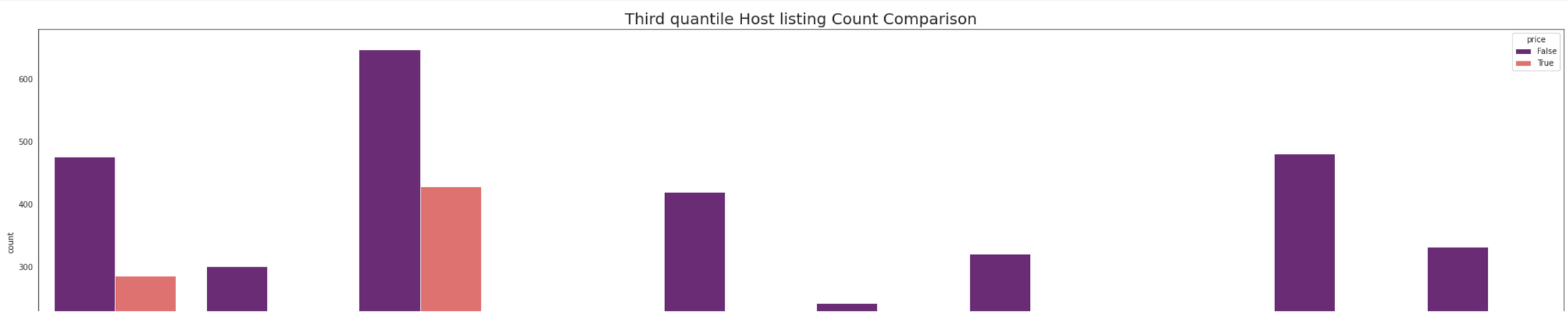
```
plt.figure(figsize=(25,15))
sns.set_style('white')
customPalette = ['#800000', '#e6194B', '#f58231', '#ffe119', '#3cb44b', '#42d4f4', '#911eb4', '#000000', '#000075', '#444444', '#008080', '#ec0101']
sns.scatterplot(x=mitte_data['latitude'], y=mitte_data['longitude'],hue=mitte_data["neighbourhood"], palette=sns.set_palette(customPalette))
plt.show()
```




```
plt.figure(figsize=(25,15))

sns.scatterplot(x=mitte_data['latitude'], y=mitte_data['longitude'],hue=mitte_data.price>mitte_data.price.quantile(0.75), palette=sns.color_palette('prism', n_colors=2), alpha=0.5)
plt.show()
```




```
plt.figure(figsize=(35,10))
plt.title("Third quantile Host listing Count Comparison",fontsize=20)
sns.countplot(mitte_data.neighbourhood,hue=mitte_data.price>mitte_data.price.quantile(0.75), palette=sns.color_palette('magma', n_colors=2))
plt.show()
```



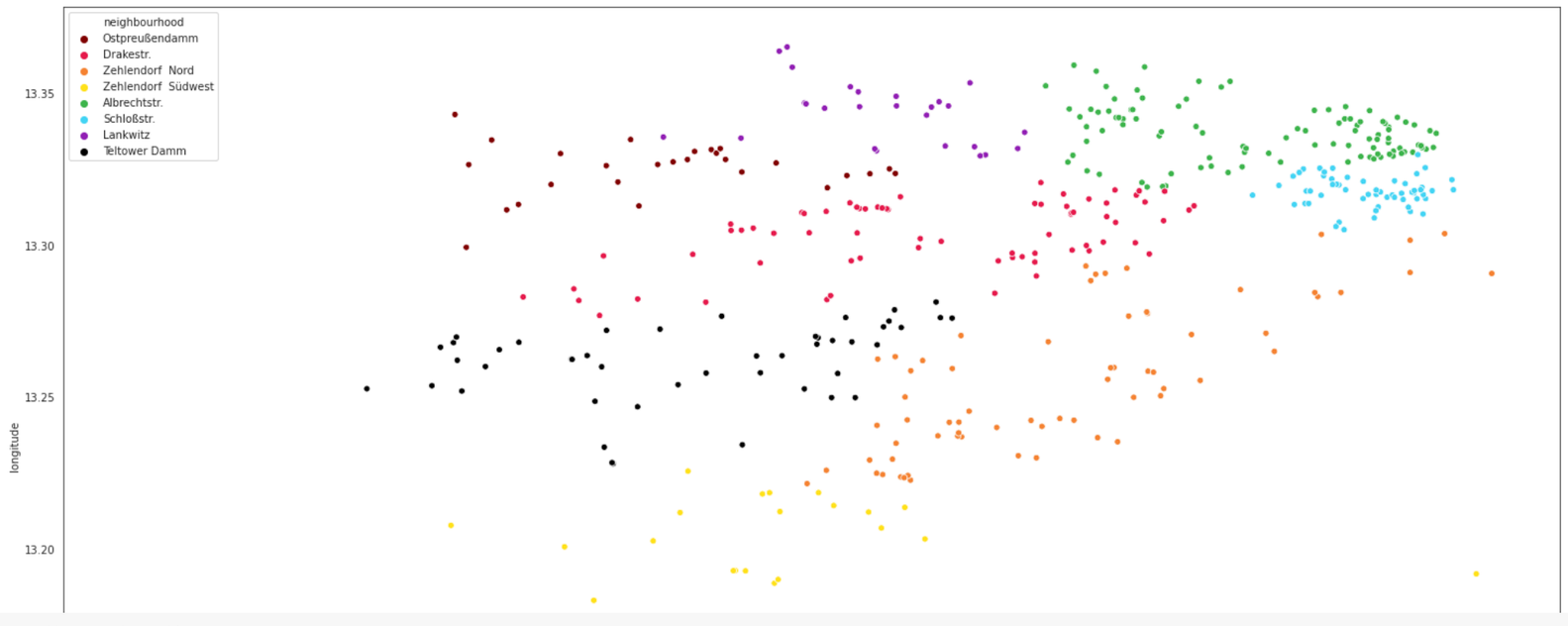

```
# plt.figure(figsize=(35,10))
# sns.swarmplot(x=mitte_data['neighbourhood'],
#               y=mitte_data['price'])
```

We can see something interesting for the Mitte neighbourhood data. We can see that the expensive places are more on the left side, meaning closer to the city center. So for this neighbourhood we can see that some smaller neighbourhoods have effect on the price.

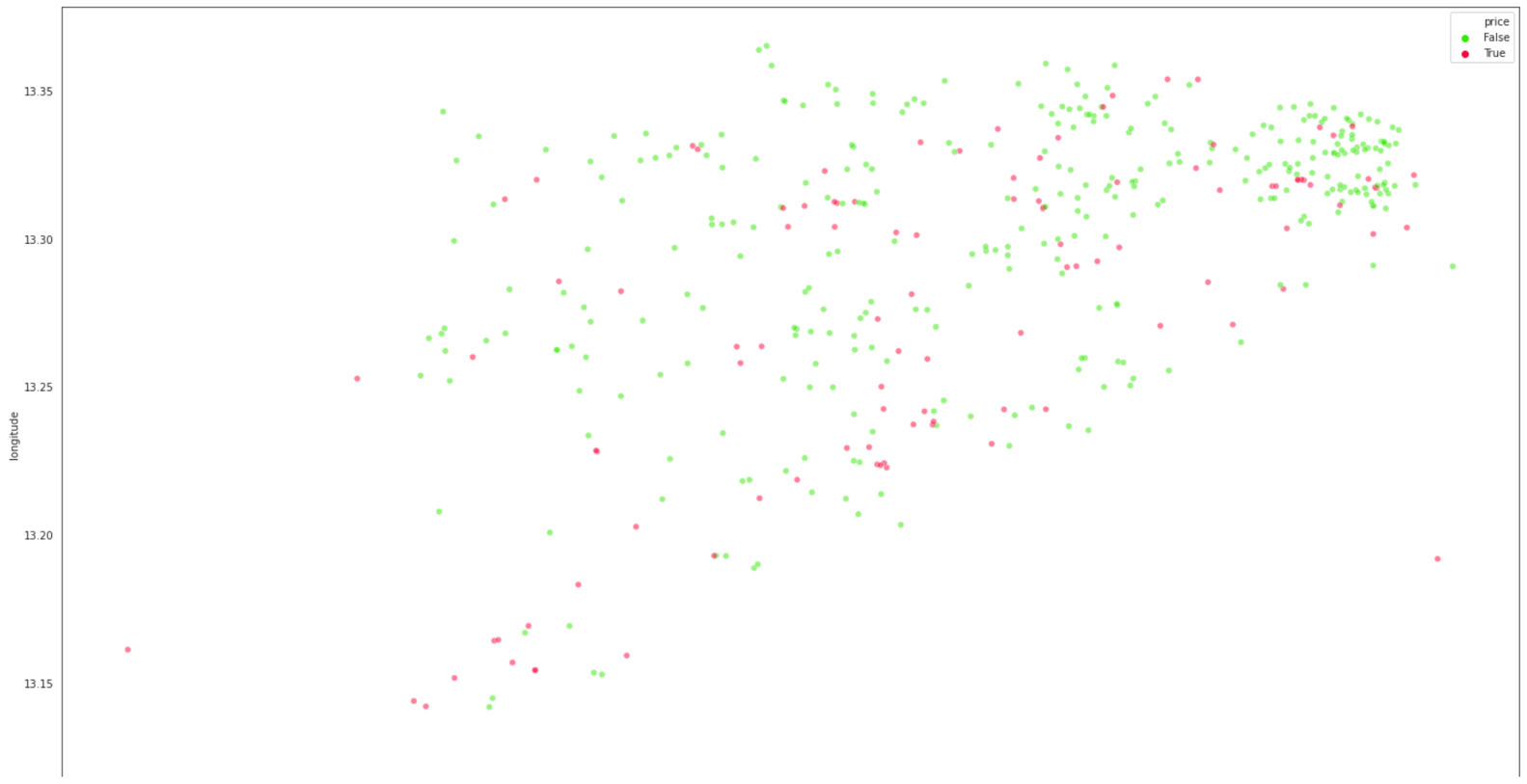
Steglitz - Zehlendorf Data

```
#Steglitz - Zehlendorf
Zehl_data = data[data.neighbourhood_group == "Steglitz - Zehlendorf"]
```

```
plt.figure(figsize=(25,15))
sns.set_style('white')
customPalette = ['#800000', '#e6194B', '#f58231', '#ffe119', '#3cb44b', '#42d4f4', '#911eb4', '#000000', '#000075', '#444444', '#008080', '#ec0101']
sns.scatterplot(x=Zehl_data['latitude'], y=Zehl_data['longitude'],hue=Zehl_data["neighbourhood"], palette=sns.set_palette(customPalette))
plt.show()
```




```
plt.figure(figsize=(25,15))

sns.scatterplot(x=Zehl_data['latitude'], y=Zehl_data['longitude'],hue=Zehl_data.price>Zehl_data.price.quantile(0.75), palette=sns.color_palette('prism', n_colors=2), alpha=0.5)
plt.show()
```




```
plt.figure(figsize=(35,10))
plt.title("Third quantile Host listing Count Comparison",fontsize=20)
sns.countplot(Zehl_data.neighbourhood,hue=Zehl_data.price>Zehl_data.price.quantile(0.75), palette=sns.color_palette('magma', n_colors=2))
plt.show()
```

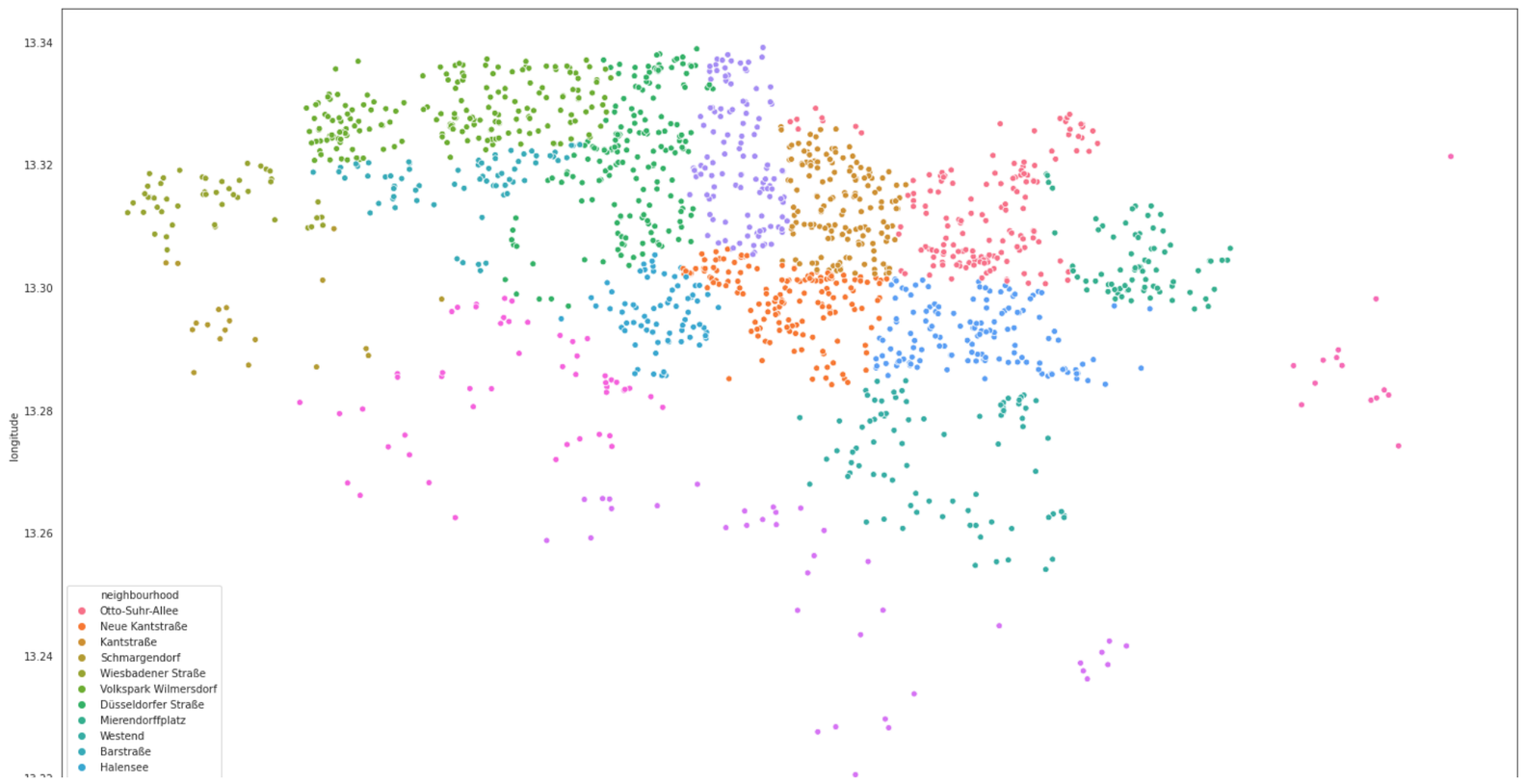
Third quantile Host listing Count Comparison

This neighbourhood doesnt have much density in listings so it is hard to see any patterns of listing prices distribution

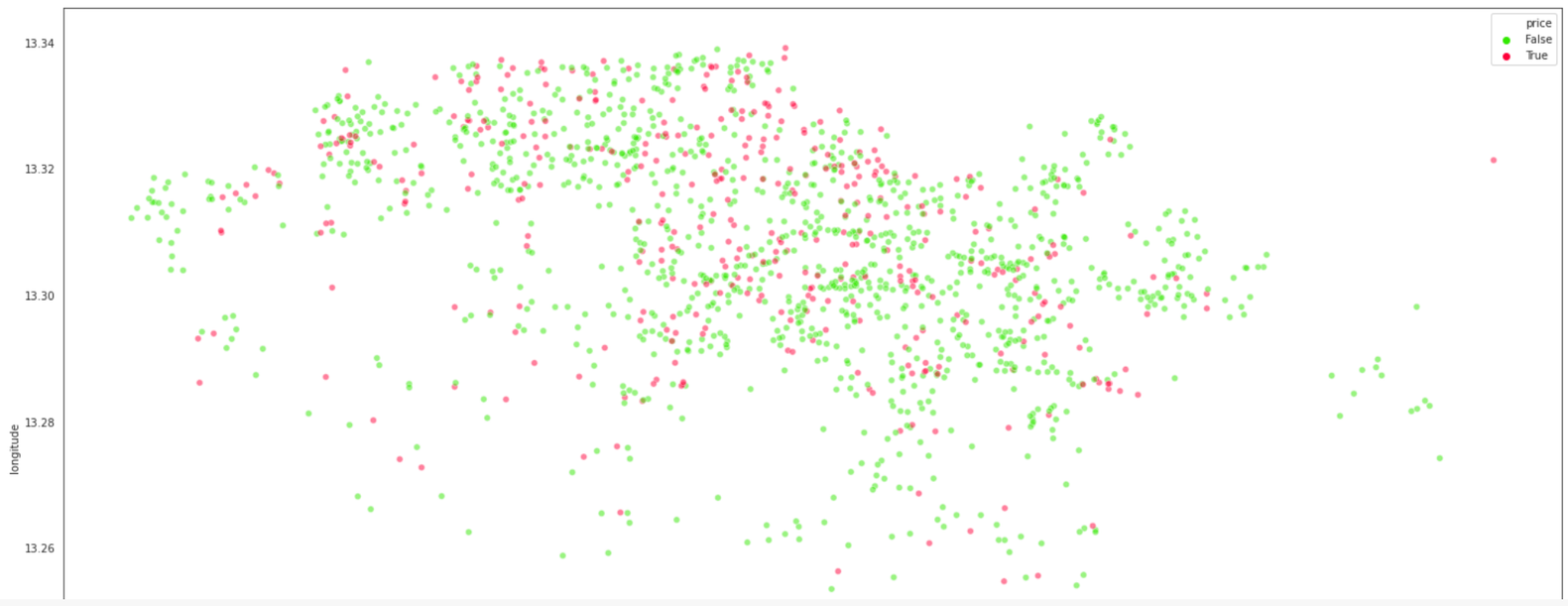
Charlottenburg-Wilm. Data

```
#Charlottenburg-Wilm.
Char_data = data[data.neighbourhood_group == "Charlottenburg-Wilm."]
```

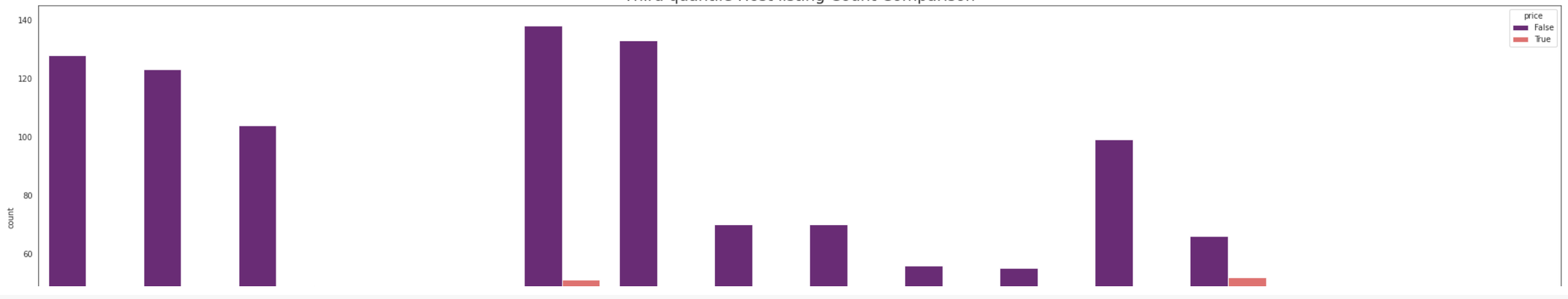
```
plt.figure(figsize=(25,15))
sns.set_style('white')
customPalette = ['#800000', '#e6194B', '#f58231', '#ffe119', '#3cb44b', '#42d4f4', '#911eb4', '#000000', '#000075', '#444444', '#008080', '#ec0101']
sns.scatterplot(x=Char_data['latitude'], y=Char_data['longitude'],hue=Char_data["neighbourhood"], palette=sns.set_palette(customPalette))
plt.show()
```




```
plt.figure(figsize=(25,15))

sns.scatterplot(x=Char_data['latitude'], y=Char_data['longitude'],hue=Char_data.price>Char_data.price.quantile(0.75), palette=sns.color_palette('prism', n_colors=2), alpha=0.5)
plt.show()
```

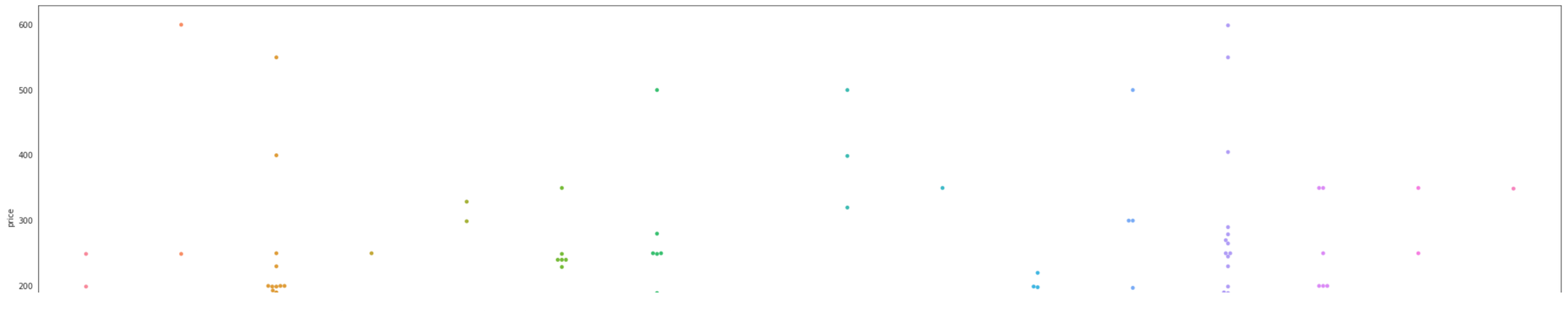



```
plt.figure(figsize=(35,10))
plt.title("Third quantile Host listing Count Comparison",fontsize=20)
sns.countplot(Char_data.neighbourhood,hue=Char_data.price>Char_data.price.quantile(0.75), palette=sns.color_palette('magma', n_colors=2))
plt.show()
```




```
plt.figure(figsize=(35,10))
sns.swarmplot(x=Char_data['neighbourhood'],
              y=Char_data['price'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd8e1548110>
```



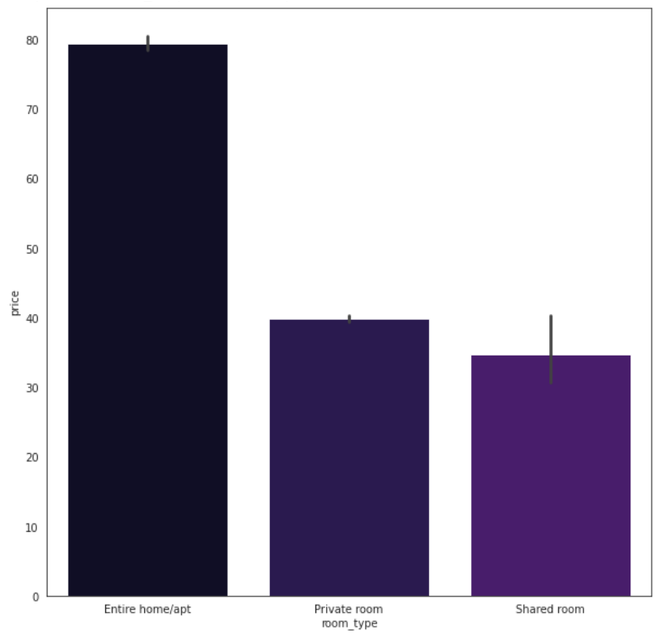
We can see that some neighbourhoods do have a higher number of listings that have a price higher than third quantile. We can conclude that the smaller neighbourhood areas also have an effect on the price. Therefore we will consider it for training the model.

We have analysed that the geolocation has certain effect on the price. Now we will examine room types and prices

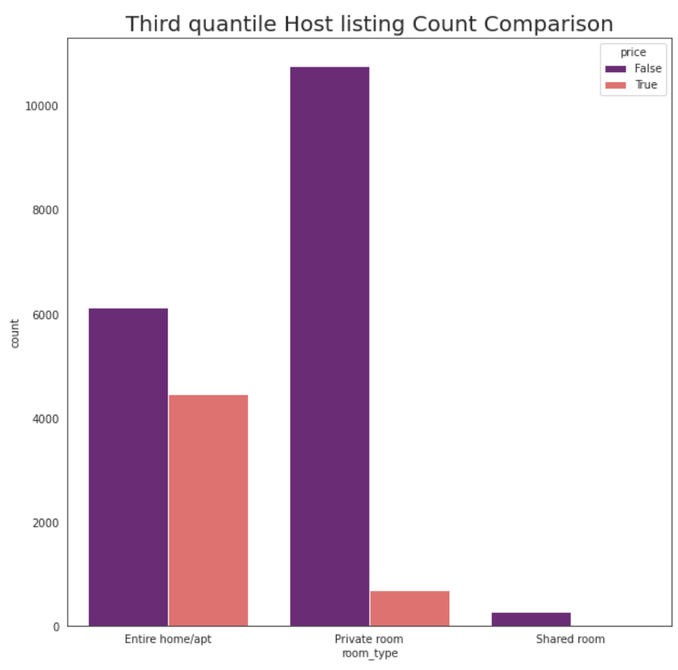
Room Type and Price

```
plt.figure(figsize=(10,10))
sns.barplot(x=data['room_type'], y=data['price'], palette=sns.color_palette('magma', n_colors=12))
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd8e1509810>
```



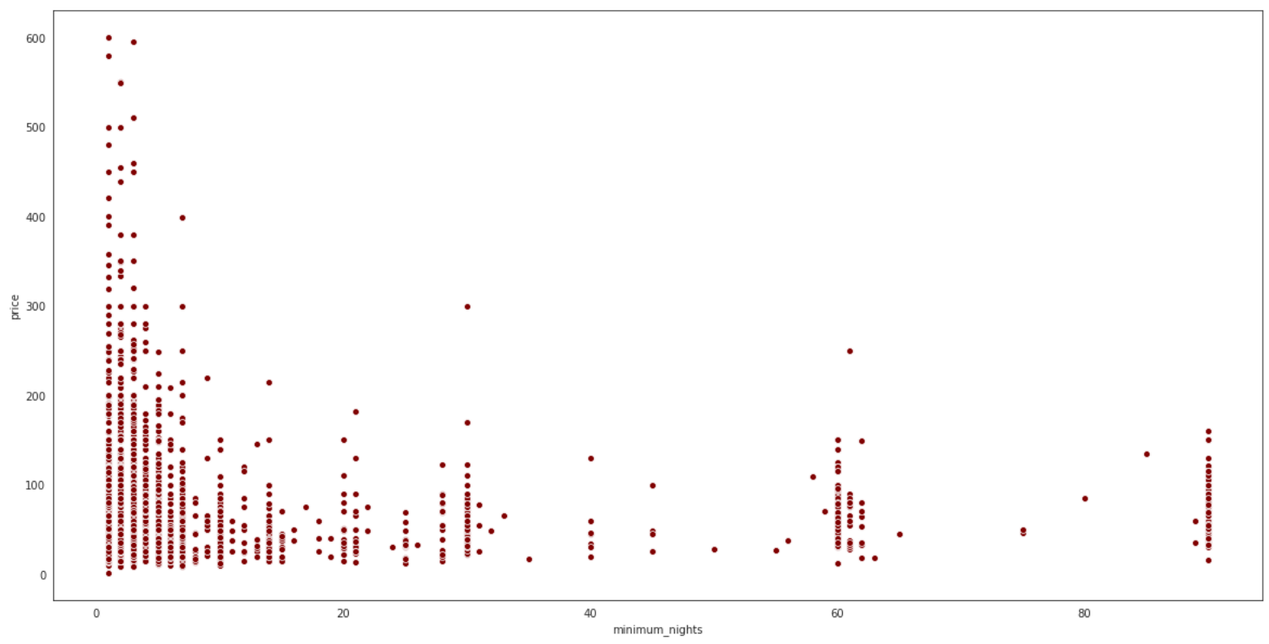

```
plt.figure(figsize=(10,10))
plt.title("Third quantile Host listing Count Comparison",fontsize=20)
sns.countplot(data.room_type,hue=data.price>data.price.quantile(0.75), palette=sns.color_palette('magma', n_colors=2))
plt.show()
```



As expected we can see that the price of entire home/apt is higher than a private room. Therefore this will be a crucial feature for determining the price.

Observation of Minimum Nights and Price

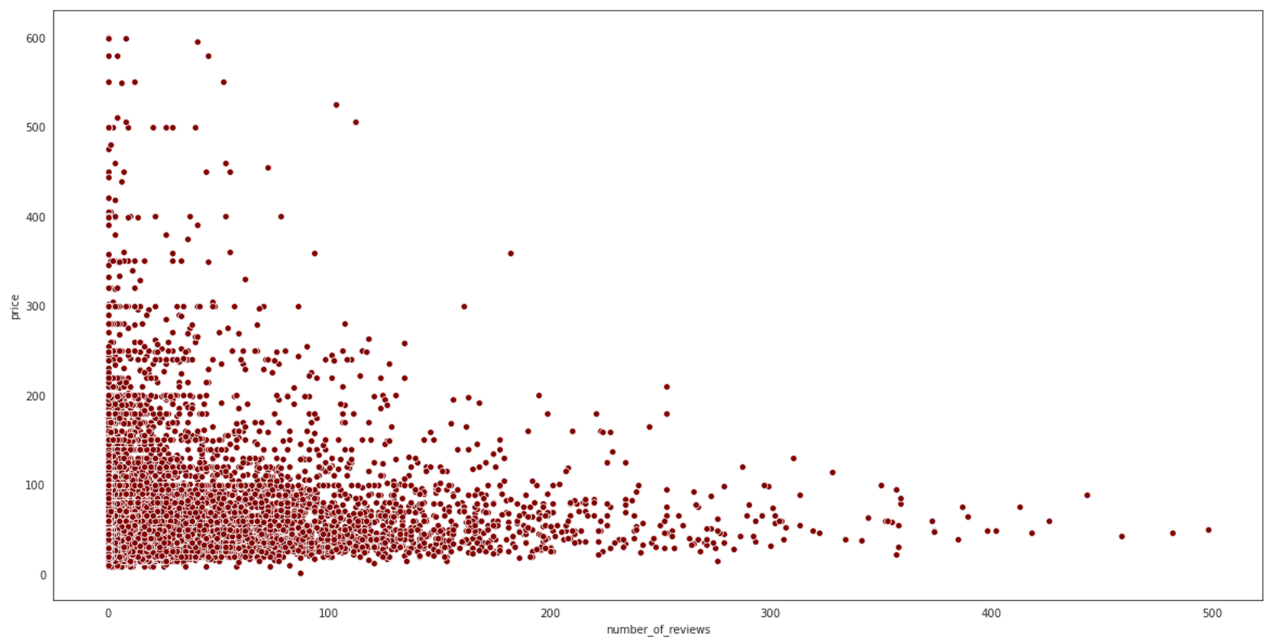
```
plt.figure(figsize=(20,10))
sns.set_style('white')

sns.scatterplot(x=mitte_data['minimum_nights'], y=mitte_data['price'], palette=sns.color_palette('Blues', n_colors=2))
plt.show()
```



It seems like the listings that have a lower minimum nights have higher value but this is not the case. Most rooms have lower minimum nights thats why the expensive rooms seem to have lower minimum nights.

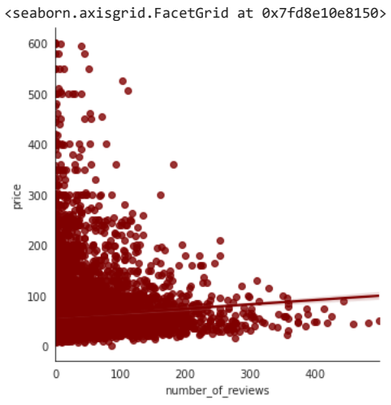
Reviews and Prices

```
plt.figure(figsize=(20,10))
sns.set_style('white')

sns.scatterplot(x=data['number_of_reviews'], y=data['price'], palette=sns.color_palette('plasma', n_colors=2))
plt.show()
```

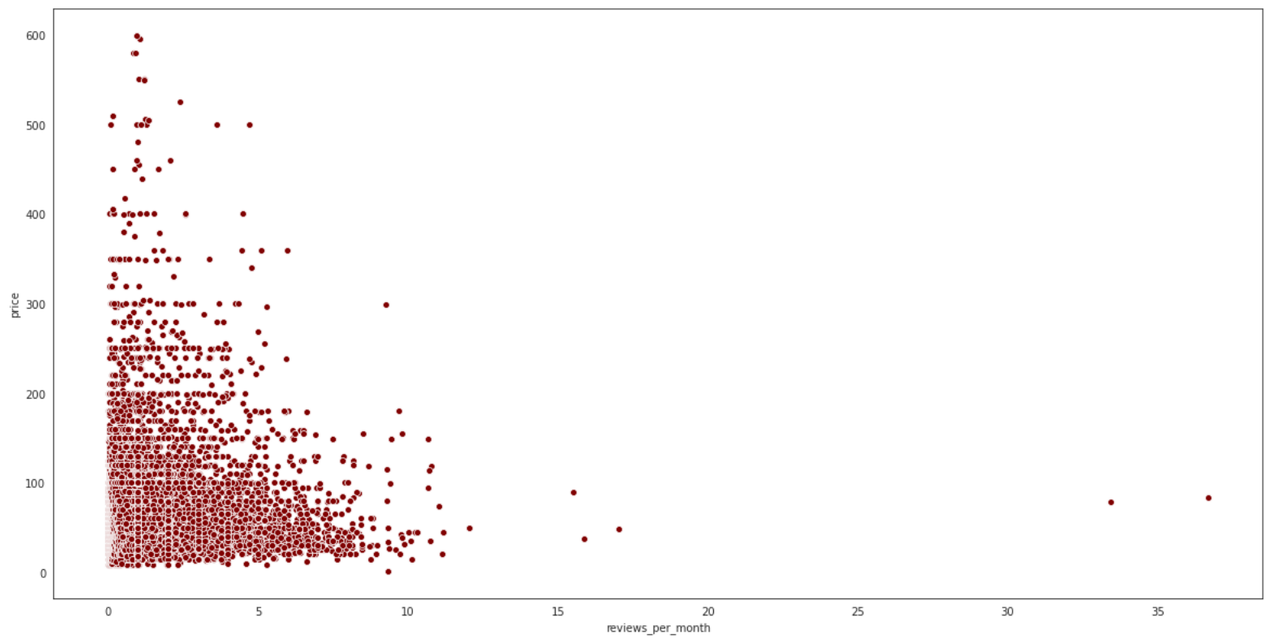



```
sns.lmplot(x="number_of_reviews", y="price",  data=data)
```

```
<seaborn.axisgrid.FacetGrid at 0x7fd8e10e8150>
```




```
plt.figure(figsize=(20,10))
sns.set_style('white')

sns.scatterplot(x=data['reviews_per_month'], y=data['price'], palette=sns.color_palette('plasma', n_colors=2))
plt.show()
```



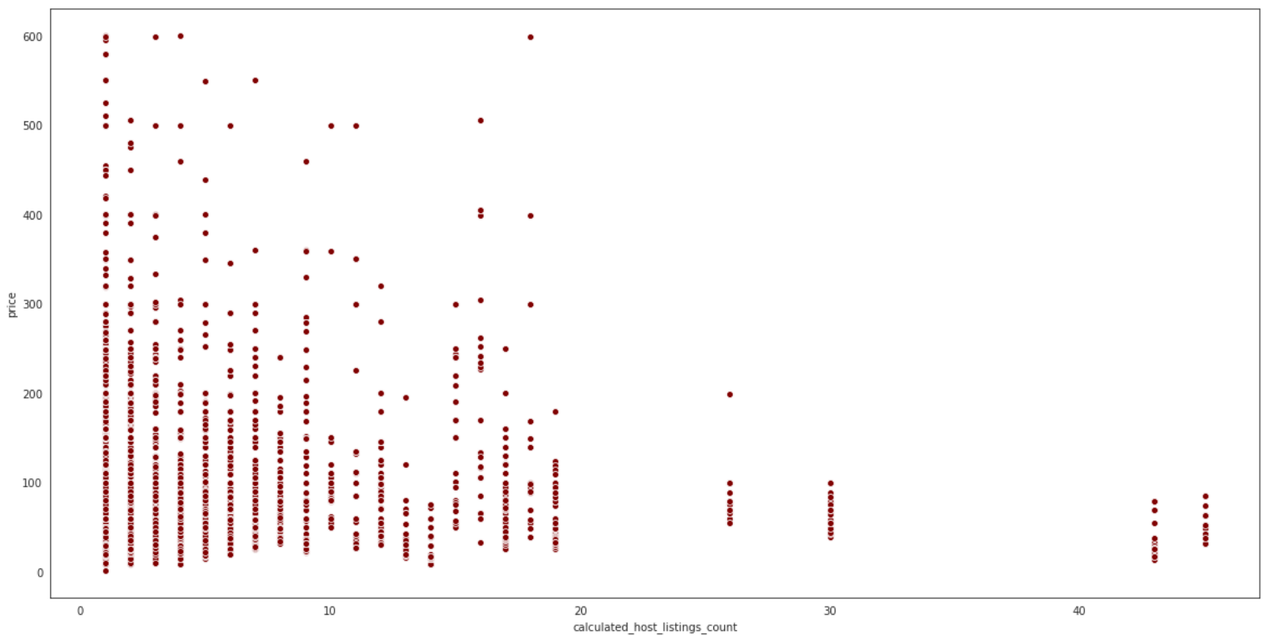

```
sns.lmplot(x="reviews_per_month", y="price",  data=data)
```

```
<seaborn.axisgrid.FacetGrid at 0x7fd8e1020790>
```

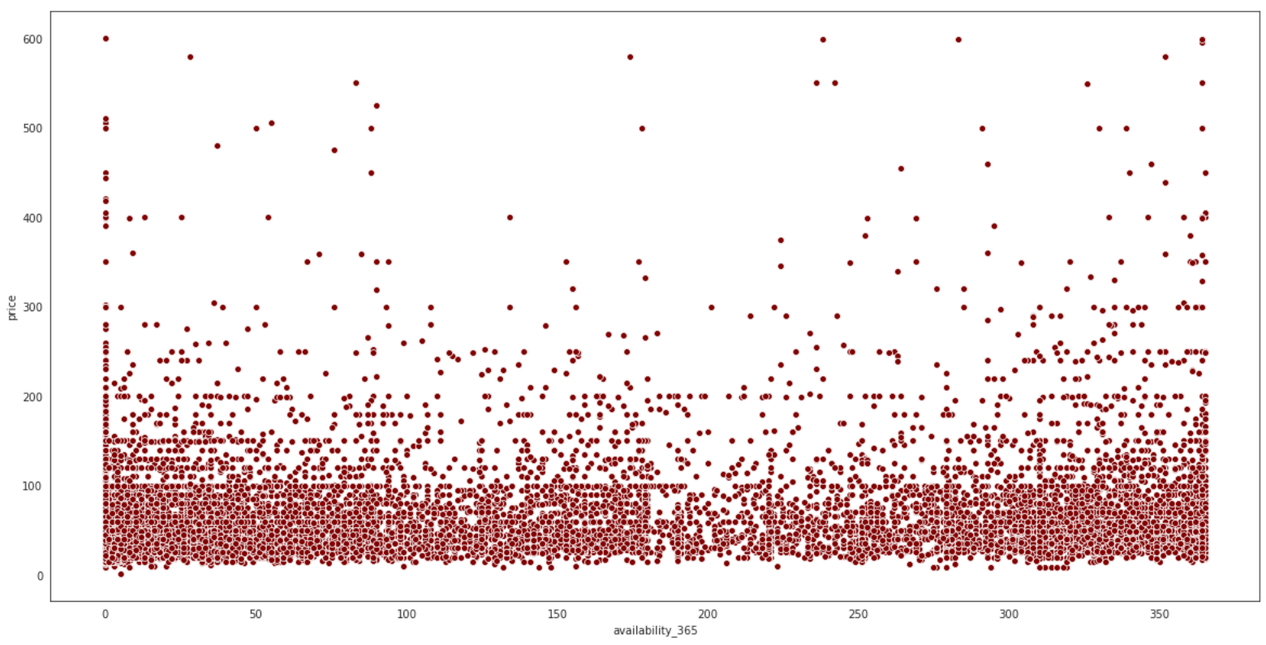
It doesn't seem like there is any strong relationship between reviews and prices.

Listings Occupied Time and Price


```
plt.figure(figsize=(20,10))
sns.set_style('white')

sns.scatterplot(x=data['calculated_host_listings_count'], y=data['price'], palette=sns.color_palette('plasma', n_colors=2))
plt.show()
```




```
plt.figure(figsize=(20,10))
sns.set_style('white')

sns.scatterplot(x=data['availability_365'], y=data['price'], palette=sns.color_palette('plasma', n_colors=2))
plt.show()
```




```
data.head()
```

	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
0	2015	Berlin-Mitte Value! Quiet courtyard/very central	2217	Ian	Mitte	Brunnenstr. Süd	52.534537	13.402557	Entire home/apt	60	4	118	2018-10-28	3.76	4	141
1	2695	Prenzlauer Berg close to Mauerpark	2986	Michael	Pankow	Prenzlauer Berg Nordwest	52.548513	13.404553	Private room	17	2	6	2018-10-01	1.42	1	0
2	3176	Fabulous Flat in great Location	3718	Britta	Pankow	Prenzlauer Berg Südwest	52.534996	13.417579	Entire home/apt	90	62	143	2017-03-20	1.25	1	220
3	3309	BerlinSpot Schöneberg near KaDeWe	4108	Jana	Tempelhof - Schöneberg	Schöneberg-Nord	52.498855	13.349065	Private room	26	5	25	2018-08-16	0.39	1	297
4	7071	BrightRoom with sunny greenview!	17391	Bright	Pankow	Helmholtzplatz	52.543157	13.415091	Private room	42	2	197	2018-11-04	1.75	1	26