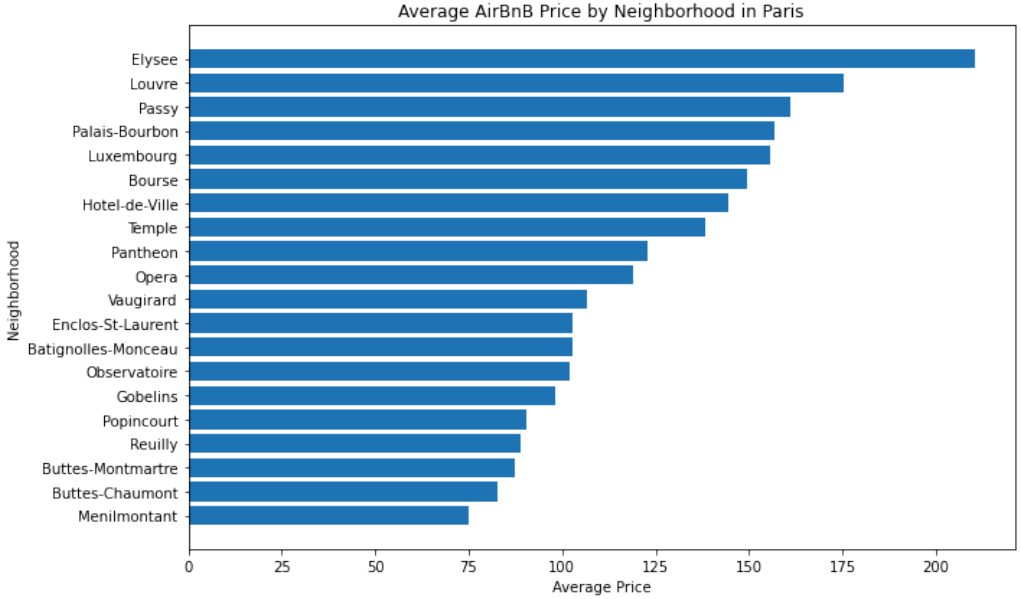
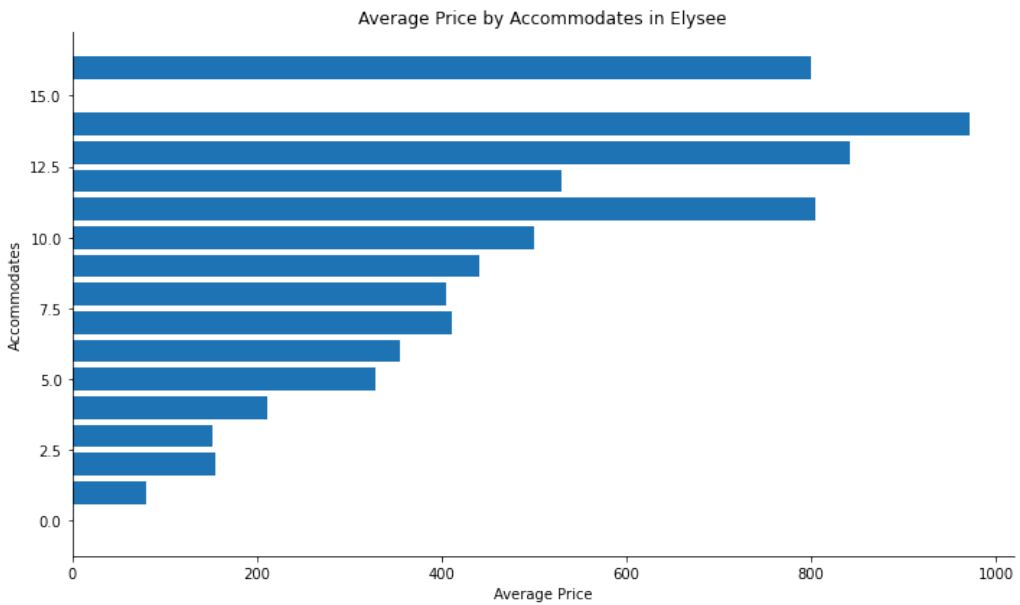
In [1]:

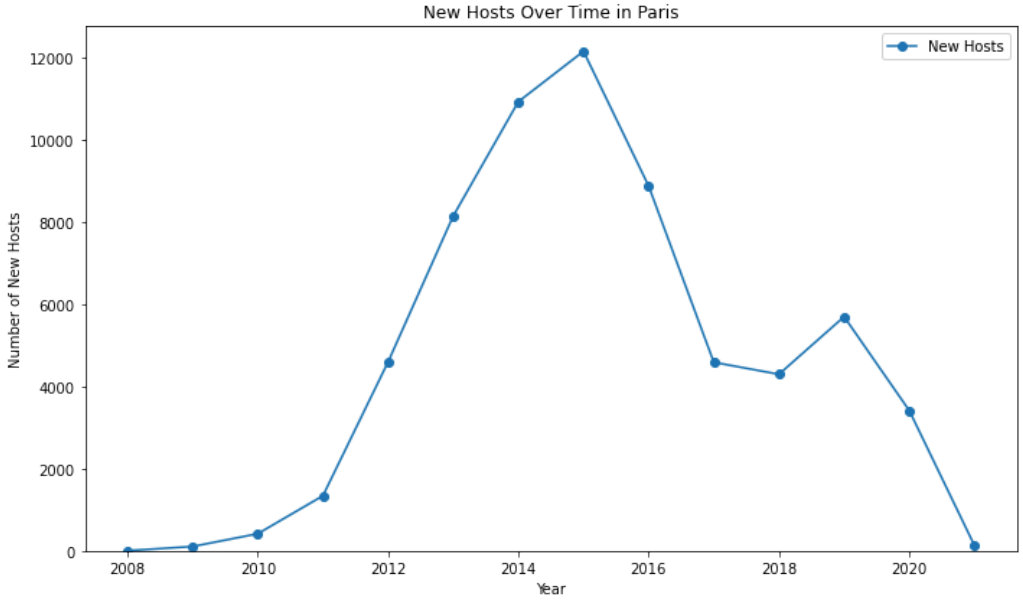
```
# AirBnB Listing Analysis for Paris
# Objective: Analyze the impact of recent regulations on AirBnB listings in Paris

# Import necessary libraries
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Ensure plots are displayed inline in Jupyter Notebooks
%matplotlib inline

# Objective 1: Profile & QA the data

# Step 1: Import the Listings.csv file with appropriate settings
listings = pd.read_csv('Listings.csv', low_memory=False, encoding="ISO-8859-1", parse_dates=['host_since'])

# Step 2: Filter data for Paris listings and select relevant columns
paris_listings = listings.query("city == 'Paris'").loc[:, ['host_since', 'neighbourhood', 'city', 'accommodates', 'price']]

# Step 3: Data Quality Assessment
print("Data Info:\n", paris_listings.info())
print("\nData Description:\n", paris_listings.describe())
print("\nMissing Values:\n", paris_listings.isnull().sum())

# Objective 2: Prepare the data for visualization

# Step 4: Group by 'neighbourhood' and calculate the mean price
paris_listings['price'] = paris_listings['price'].replace('[\$,]', '', regex=True).astype(float)
paris_listings_neighbourhood = paris_listings.groupby('neighbourhood')['price'].mean().sort_values().reset_index()

# Step 5: Identify the most expensive neighborhood and analyze 'accommodates'
most_expensive_neighbourhood = paris_listings_neighbourhood.iloc[-1]['neighbourhood']
most_expensive_data = paris_listings.query("neighbourhood == @most_expensive_neighbourhood")
paris_listings_accommodations = most_expensive_data.groupby('accommodates')['price'].mean().sort_values().reset_index()

# Step 6: Group by 'host_since' year and calculate average price and count of new hosts
paris_listings['host_since_year'] = paris_listings['host_since'].dt.year
paris_listings_over_time = paris_listings.groupby('host_since_year').agg({'price': 'mean', 'host_since': 'count'}).rename(columns={'host_since': 'new_hosts'}).reset_index()

# Objective 3: Visualize the data and summarize findings

# Step 7: Horizontal bar chart for average price by neighborhood
plt.figure(figsize=(10, 6))
plt.barh(paris_listings_neighbourhood['neighbourhood'], paris_listings_neighbourhood['price'])
plt.xlabel('Average Price')
plt.ylabel('Neighborhood')
plt.title('Average AirBnB Price by Neighborhood in Paris')
plt.tight_layout()
plt.show()

# Step 8: Horizontal bar chart for average price by accommodates in the most expensive neighborhood
plt.figure(figsize=(10, 6))
plt.barh(paris_listings_accommodations['accommodates'], paris_listings_accommodations['price'])
plt.xlabel('Average Price')
plt.ylabel('Accommodates')
plt.title(f'Average Price by Accommodates in {most_expensive_neighbourhood}')
sns.despine()
plt.tight_layout()
plt.show()

# Step 9: Line charts for new hosts and average price over time
plt.figure(figsize=(10, 6))
plt.plot(paris_listings_over_time['host_since_year'], paris_listings_over_time['new_hosts'], marker='o', label='New Hosts')
plt.xlabel('Year')
plt.ylabel('Number of New Hosts')
plt.title('New Hosts Over Time in Paris')
```

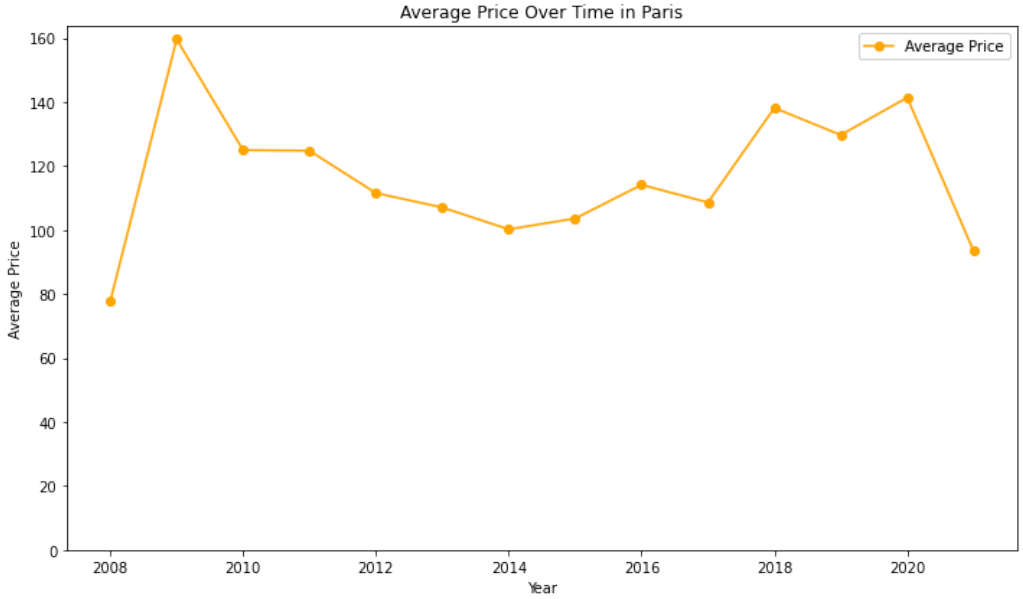
```
plt.ylim(0)
plt.legend()
plt.tight_layout()
plt.show()

plt.figure(figsize=(10, 6))
plt.plot(paris_listings_over_time['host_since_year'], paris_listings_over_time['price'], marker='o', label='Average Price', color='orange')
plt.xlabel('Year')
plt.ylabel('Average Price')
plt.title('Average Price Over Time in Paris')
plt.ylim(0)
plt.legend()
plt.tight_layout()
plt.show()

# Step 10: Dual axis line chart for new hosts and average price over time
fig, ax1 = plt.subplots(figsize=(10, 6))

color = 'tab:blue'
ax1.set_xlabel('Year')
ax1.set_ylabel('New Hosts', color=color)
ax1.plot(paris_listings_over_time['host_since_year'], paris_listings_over_time['new_hosts'], color=color, marker='o')
ax1.tick_params(axis='y', labelcolor=color)

ax2 = ax1.twinx()
color = 'tab:orange'
ax2.set_ylabel('Average Price', color=color)
ax2.plot(paris_listings_over_time['host_since_year'], paris_listings_over_time['price'], color=color, marker='o')
ax2.tick_params(axis='y', labelcolor=color)

plt.title('New Hosts and Average Price Over Time in Paris')
fig.tight_layout()
plt.show()

# Step 11: Identify the neighborhood with the highest average price
highest_price_neighbourhood = paris_listings_neighbourhood.iloc[-1]['neighbourhood']
print(f"The neighborhood in Paris with the highest average AirBnB listing price is: {highest_price_neighbourhood}")
```

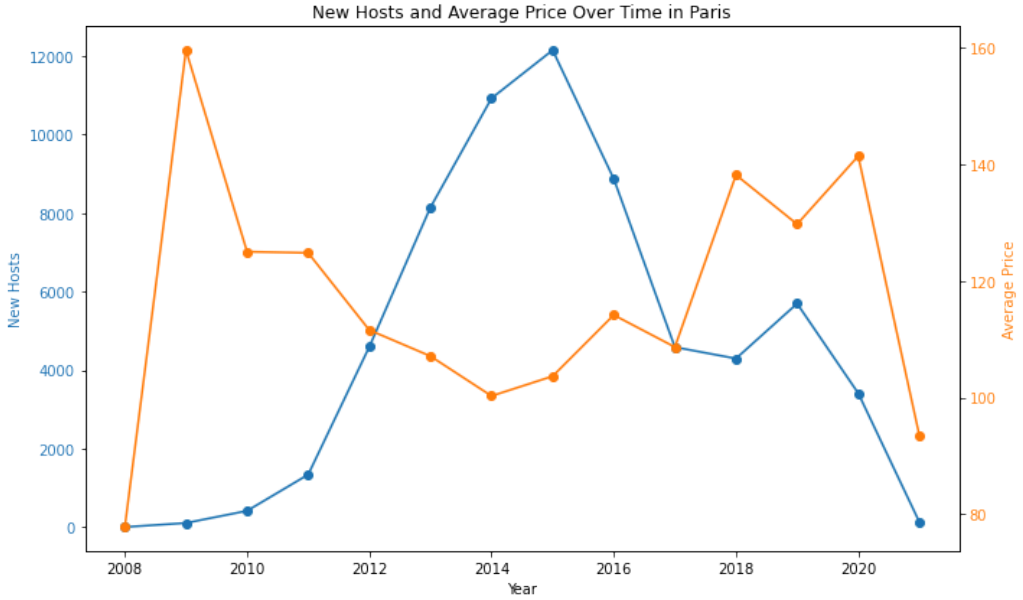
```
<class 'pandas.core.frame.DataFrame'>
Index: 64690 entries, 0 to 279711
Data columns (total 5 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   host_since     64657 non-null  datetime64[ns]
 1   neighbourhood  64690 non-null  object
 2   city           64690 non-null  object
 3   accommodates   64690 non-null  int64
 4   price          64690 non-null  int64
dtypes: datetime64[ns](1), int64(2), object(2)
memory usage: 3.0+ MB
Data Info:
 None

Data Description:
                           host_since  accommodates         price
count                           64657  64690.000000  64690.000000
mean   2015-11-01 11:06:05.528867584      3.037997    113.096445
min              2008-08-30 00:00:00      0.000000      0.000000
25%              2014-03-09 00:00:00      2.000000     59.000000
50%              2015-07-07 00:00:00      2.000000     80.000000
75%              2017-05-29 00:00:00      4.000000    120.000000
max              2021-02-07 00:00:00     16.000000  12000.000000
std                              NaN      1.588766    214.433668

Missing Values:
 host_since       33
neighbourhood     0
city              0
accommodates      0
price             0
dtype: int64
```




Average Price by Accommodates in Elysee
Accommodates
15.0
12.5
10.0
7.5
5.0
2.5
0.0
0
200
400
600
800
1000
Average Price


New Hosts Over Time in Paris
New Hosts
Number of New Hosts
12000
10000
8000
6000
4000
2000
0
2008
2010
2012
2014
2016
2018
2020
Year





The neighborhood in Paris with the highest average AirBnB listing price is: Elysee