

Data Visualization

(1) Distribution Of Airbnb Bookings Price Range Using Histogram

```
# Creating a figure with a custom size
plt.figure(figsize=(12, 5))

# Setting the seaborn theme to darkgrid (Other options: white, whitegrid,
dark, ticks)
sns.set_theme(style='darkgrid')

# Creating a histogram of the 'price' column of the Airbnb_df dataframe
# using sns distplot function and specifying the color as red
sns.distplot(Airbnb_df['price'],color=('red'))

# Adding labels to the x-axis and y-axis
plt.xlabel('Price', fontsize=14)
plt.ylabel('Density', fontsize=14)

# Adding a title to the plot
plt.title('Distribution of Airbnb Prices',fontsize=15)
```

Observations -->

The range of prices being charged on Airbnb appears to be from 20 to 330 dollars, with majority of listings falling in the price range of 50 to 150 dollars.

The distribution of prices appears to have a peak in the 50 to 150 dollars range, with a relatively lower density of listings in higher and lower price ranges.

There may be fewer listings available at prices above 250 dollars, as the density of listings drops significantly in this range.

(2) Total Listing/Property count in Each Neighborhood Group using Catplot

```
# Counting the number of listings in each neighborhood group
counts = Airbnb_df['neighbourhood_group'].value_counts()

# Resetting the index of the series so that the neighborhood groups become
columns in the resulting dataframe

Top_Neighborhood_group = counts.reset_index()

# Renaming the columns of the dataframe to be more descriptive

Top_Neighborhood_group.columns = ['Neighborhood_Groups', 'Listing_Counts']

Top_Neighborhood_group
```

```
# Setting the figure size

plt.figure(figsize=(12, 8))

# Creating a catplot of the neighbourhood group data

sns.catplot(data=Top_Neighborhood_group, x = 'Neighborhood_Groups', y =
'Listing_Counts')

# Setting the title of the plot

plt.title('Neighbourhood_group Listing Counts in NYC', fontsize=15)

# Setting the x-axis label

plt.xlabel('Neighbourhood_Group', fontsize=14)

# Setting the y-axis label

plt.ylabel('Total listings counts', fontsize=14)
```

Observations -->

Manhattan and Brooklyn have the highest number of listings on Airbnb, with over 19,000 listings each.

Queens and the Bronx have significantly fewer listings compared to Manhattan and Brooklyn, with 5,567 and 1,070 listings, respectively

Staten Island has the fewest number of listings, with only 365.

The distribution of listings across the different neighborhood groups is skewed, with a concentration of listings in Manhattan and Brooklyn.

Despite being larger in size, the neighborhoods in Queens, the Bronx, and Staten Island have fewer listings on Airbnb compared to Manhattan, which has a smaller geographical area.

This could suggest that the demand for Airbnb rentals is higher in Manhattan compared to the other neighborhoods, leading to a higher concentration of listings in this area.

Alternatively, it could be that the supply of listings is higher in Manhattan due to a higher number of homeowners or property owners in this neighborhood who are willing to list their properties on Airbnb.

(3) Price Distribution Of Each Neighborhood Group using Violin Plot

```
# Grouping the Airbnb dataset by neighborhood group and calculating the
mean of each group

numeric_columns = Airbnb_df.select_dtypes(include=['number'])

grouped =
Airbnb_df.groupby(["neighbourhood_group"])[numeric_columns].mean()

# Resetting the index of the grouped dataframe so that the neighborhood
group becomes a column

neighbourhood_group_avg_price = grouped.reset_index()

# Renaming the "price" column to "avg_price"

neighbourhood_group_avg_price =
round(neighbourhood_group_avg_price.rename(columns={"price":
"avg_price"}),2)

# Selecting only the "neighbourhood_group" and "avg_price" columns

neighbourhood_group_avg_price[['neighbourhood_group', 'avg_price']].head()
```

Observations

The average price of a listing in New York City varies significantly across different neighborhoods, with Manhattan having the highest 146 dollars/day average price and the Bronx having the lowest near 77 dollars/day.

The average price increases as you move from the outer boroughs (Bronx, Brooklyn, Queens, and Staten Island) towards the center of the city (Manhattan).

The average price in Queens and Staten Island is relatively similar, despite being in different parts of the city.

The data suggests that the overall cost of living in New York City is higher in the center of the city (Manhattan) compared to the outer boroughs. This is likely due to the fact that Manhattan is the most densely populated and commercially important borough, and therefore has higher demand for housing.

Price distribution is very high in Manhattan and Brooklyn. but Manhattan has more diversity in price range as you can see in the plot.

Queens and Bronx have similar price distributions and more distribution is in 50 to 100 but there is little diversity in price unlike Manhattan and Brooklyn.

(4) Average Minimum Price In Neighborhoods using Scatter chart

```
# Creating a new DataFrame that displays the average price of Airbnb
rentals in each neighborhood
neighbourhood_avg_price =
Airbnb_df.groupby("neighbourhood")[numeric_columns.columns].mean().reset_i
ndex().rename(columns={"price": "avg_price"})[['neighbourhood',
'avg_price']]

# Selecting the top 10 neighborhoods with the lowest average prices
neighbourhood_avg_price =
neighbourhood_avg_price.sort_values("avg_price").head(10)

# Joining the resulting DataFrame with the 'neighbourhood_group' column
from the Airbnb NYC dataset, dropping any duplicate entries
neighbourhood_avg_price_sorted_with_group =
neighbourhood_avg_price.join(Airbnb_df[['neighbourhood',
'neighbourhood_group']]).drop_duplicates().set_index('neighbourhood'),

on='neighbourhood')

neighbourhood_avg_price_sorted_with_group
```

```
neighbourhood_avg_price =
(Airbnb_df.groupby("neighbourhood")[numeric_columns.columns].mean().reset_
index().rename(columns={"price": "avg_price"}))[['neighbourhood',
'avg_price']]
neighbourhood_avg_price =
(neighbourhood_avg_price.sort_values("avg_price"))

# Grouping the data by neighborhood and calculate the average price
neighbourhood_avg_price =
Airbnb_df.groupby("neighbourhood")["price"].mean()

# Creating a new DataFrame with the average price for each neighborhood
neighbourhood_prices = pd.DataFrame({"neighbourhood":
neighbourhood_avg_price.index, "avg_price":
neighbourhood_avg_price.values})
```

```
# Merging the average price data with the original DataFrame#trying to
find where the coordinates belong from the latitude and longitude
df = Airbnb_df.merge(neighbourhood_prices, on="neighbourhood")

# Creating the scattermapbox plot
fig = df.plot.scatter(x="longitude", y="latitude", c="avg_price",
title="Average Airbnb Price by Neighborhoods in New York City",
figsize=(12,6), cmap="plasma")
fig
```

Observations -->

All of the neighborhoods listed are located in the outer boroughs of New York City (Bronx, Queens, and Staten Island). This suggests that these neighborhoods may have a lower overall cost of living compared to neighborhoods in Manhattan and Brooklyn.

Most of these neighborhoods are located in the Bronx and Staten Island. These boroughs tend to have a lower overall cost of living compared to Manhattan and Brooklyn.

These neighborhoods may be attractive to renters or buyers looking for more affordable housing options in the New York City area.

(5) Number of listings of each room type in the Airbnb NYC dataset

```
# Creating a new DataFrame that displays the number of listings of each
room type in the Airbnb NYC dataset
top_room_type = Airbnb_df['room_type'].value_counts().reset_index()

# Renaming the columns of the resulting DataFrame to 'Room_Type' and
'Total_counts'
top_room_type.columns = ['Room_Type', 'Total_counts']

top_room_type
```

```
# Setting the figure size
plt.figure(figsize=(10, 6))

# Getting the room type counts
room_type_counts = Airbnb_df['room_type'].value_counts()

# Setting the labels and sizes for the pie chart
labels = room_type_counts.index
sizes = room_type_counts.values

# Creating the pie chart
plt.pie(sizes, labels=labels, autopct='%1.1f%%')

# Adding a legend to the chart
plt.legend(title='Room Type', bbox_to_anchor=(0.8, 0, 0.5, 1),
fontsize='12')

plt.show()
```

Observations -->

The majority of listings on Airbnb are for entire homes or apartments with 22784 listings, followed by private rooms with 21996 listings, and shared rooms with 1138 listings.

There is a significant difference in the number of listings for each room type. For example, there are almost 20 times as many listings for entire homes or apartments as there are for shared rooms.

The data suggests that travelers using Airbnb have a wide range of accommodation options to choose from, including private rooms and entire homes or apartments

(6) Using latitude and longitude in scatterplot map and finding neighbourhood_groups and room types in map

```
# Trying to find where the coordinates belong from the latitude and longitude

# Setting the default figure size for the seaborn library
sns.set(rc={"figure.figsize": (10, 8)})

# Creating a scatter plot that displays the longitude and latitude of the listings in the Airbnb NYC dataset
ax = sns.scatterplot(data=Airbnb_df, x="longitude", y="latitude",
hue='neighbourhood_group', palette='bright')

# Setting the title of the plot
ax.set_title('Location Co-ordinates', fontsize='14')
```

```
# Let's observe the type of room_types

# Setting the default figure size for the seaborn library
sns.set(rc={"figure.figsize": (10, 8)})

# Creating a scatter plot that displays the longitude and latitude of the listings in the Airbnb NYC dataset with room_types.
ax = sns.scatterplot(x=Airbnb_df.longitude, y=Airbnb_df.latitude,
hue=Airbnb_df.room_type, palette='muted')

# Setting the title of the plot
ax.set_title('Distribution of type of rooms across NYC', fontsize='14')
```

```
# Let's have an idea of the price variations in neighborhood_groups
# Creating a scatter plot that displays the longitude and latitude of the listings in the Airbnb NYC dataset, with the color of each point indicating the price of the listing
lat_long = Airbnb_df.plot(kind='scatter', x='longitude', y='latitude',
label='price_variations', c='price',
cmap=plt.get_cmap('jet'), colorbar=True, alpha=0.4,
figsize=(10, 8))

# Adding a legend to the plot
lat_long.legend()
```

