LONDON HOUSE PRICES ANALYSIS:

INTRODUCTION:

The housing market in London is known for its dynamic nature, with a wide range of property prices influenced by factors such as location, property size, and the number of bedrooms and bathrooms. Understanding these factors can provide valuable insights into the pricing trends of houses, helping potential buyers, investors, and policymakers make informed decisions. In this project, we explore the house prices in London by analyzing various aspects of the dataset, such as price distributions, house types, area size, and the number of bedrooms and bathrooms. Through this analysis, we aim to uncover trends, detect outliers, and potentially predict house prices based on key features.

PROBLEM STATEMENT:

The goal of this project is to analyze the factors affecting house prices in London and identify key variables that contribute to high or low prices. By conducting exploratory data analysis (EDA), we will answer several critical questions:

What is the distribution of house prices in London, and are there significant outliers?

How do property size (area in square feet), the number of bedrooms, and other features correlate with house prices?

Are there any trends that indicate which areas or property types have the highest or lowest prices?

Can normalization and outlier detection improve the quality of data for further predictive modelling?

Overview of the Analysis:

1. Data Collection:

The dataset used for this analysis contains house price information for various properties in London, including details such as the property name, price, house type, area, and number of bedrooms and bathrooms.

2. Exploratory Data Analysis (EDA):

We begin by exploring the dataset through summary statistics and visualizations to understand the distribution of key features such as price, area in square feet, and the number of bedrooms.

We analyze the correlation between these features and house prices to identify which variables have the most influence on pricing.

Outlier detection is performed using boxplots and statistical methods, as extreme values can distort the analysis. These outliers are addressed by either removing or capping them.

3. Data Normalization:

To prepare the dataset for further machine learning models or deeper statistical analysis, continuous variables like price and area are normalized. This ensures that the scales of different features are comparable.

4. Visual Analysis:

Visualizations such as scatterplots, boxplots, and heatmaps are employed to uncover relationships between different features and how they impact house prices. These visual insights guide the interpretation of the dataset and offer a clearer picture of housing trends in London.

Analyzing London house prices using Python exploring various aspects of the dataset to derive meaningful insights.

```
# importing required libraries
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

Importing Data from CSV File. Data source is Kaggle.

```
data = pd.read_csv("London.csv")

# Checking the DataFrame
data.head(2)
```

```
   Unnamed: 0  Property Name    Price        House Type  Area in sq ft
\
0           0    Queens Road  1675000             House           2716

1           1  Seward Street   650000  Flat / Apartment            814


   No. of Bedrooms  No. of Bathrooms  No. of Receptions
Location  \
0                5                 5                  5    Wimbledon

1                2                 2                  2  Clerkenwell


  City/County Postal Code
0      London    SW19 8NY
1      London    EC1V 3PA
```

```
#  This method returns the information about the dataframe including
index dtype and columns,
#  non nulls values and memory usage
data.info
```

```
<bound method DataFrame.info of       Unnamed: 0       Property Name
Price        House Type  \
0              0         Queens Road  1675000             House
1              1       Seward Street   650000  Flat / Apartment
2              2         Hotham Road   735000  Flat / Apartment
3              3        Festing Road  1765000             House
4              4        Spencer Walk   675000  Flat / Apartment
...          ...                 ...      ...               ...
3475        3475   One Lillie Square  3350000   New development
3476        3476  St. James's Street  5275000  Flat / Apartment
3477        3477       Ingram Avenue  5995000             House
3478        3478         Cork Street  6300000   New development
```

```
3479        3479    Courtenay Avenue  8650000             House

      Area in sq ft  No. of Bedrooms  No. of Bathrooms  No. of
Receptions  \
0               2716                 5                   5
5
1                814                 2                   2
2
2                761                 2                   2
2
3               1986                 4                   4
4
4                700                 2                   2
2
...              ...               ...                 ...
...
3475            1410                 3                   3
3
3476            1749                 3                   3
3
3477            4435                 6                   6
6
3478            1506                 3                   3
3
3479            5395                 6                   6
6

                      Location    City/County Postal Code
0                    Wimbledon         London    SW19 8NY
1                   Clerkenwell         London    EC1V 3PA
2                        Putney         London    SW15 1QL
3                        Putney         London    SW15 1LP
4                        Putney         London    SW15 1PL
...                         ...            ...         ...
3475                       NaN  Lillie Square     SW6 1UE
3476                St James's         London    SW1A 1JT
3477  Hampstead Garden Suburb         London    NW11 6TG
3478                   Mayfair         London     W1S 3AR
3479                  Highgate         London      N6 4LP

[3480 rows x 11 columns]>
```

```
# counting total number of rows and columns
data.shape
```

```
(3480, 11)
```

```
# counting total non null values in each column
data.count()
```

```
Unnamed: 0            3480
Property Name         3480
Price                 3480
House Type            3480
Area in sq ft         3480
No. of Bedrooms       3480
No. of Bathrooms      3480
No. of Receptions     3480
Location              2518
City/County           3480
Postal Code           3480
dtype: int64
```

It seems that "Location" has missing values, as it only contains 2518 values whereas other columns has 3480 columns

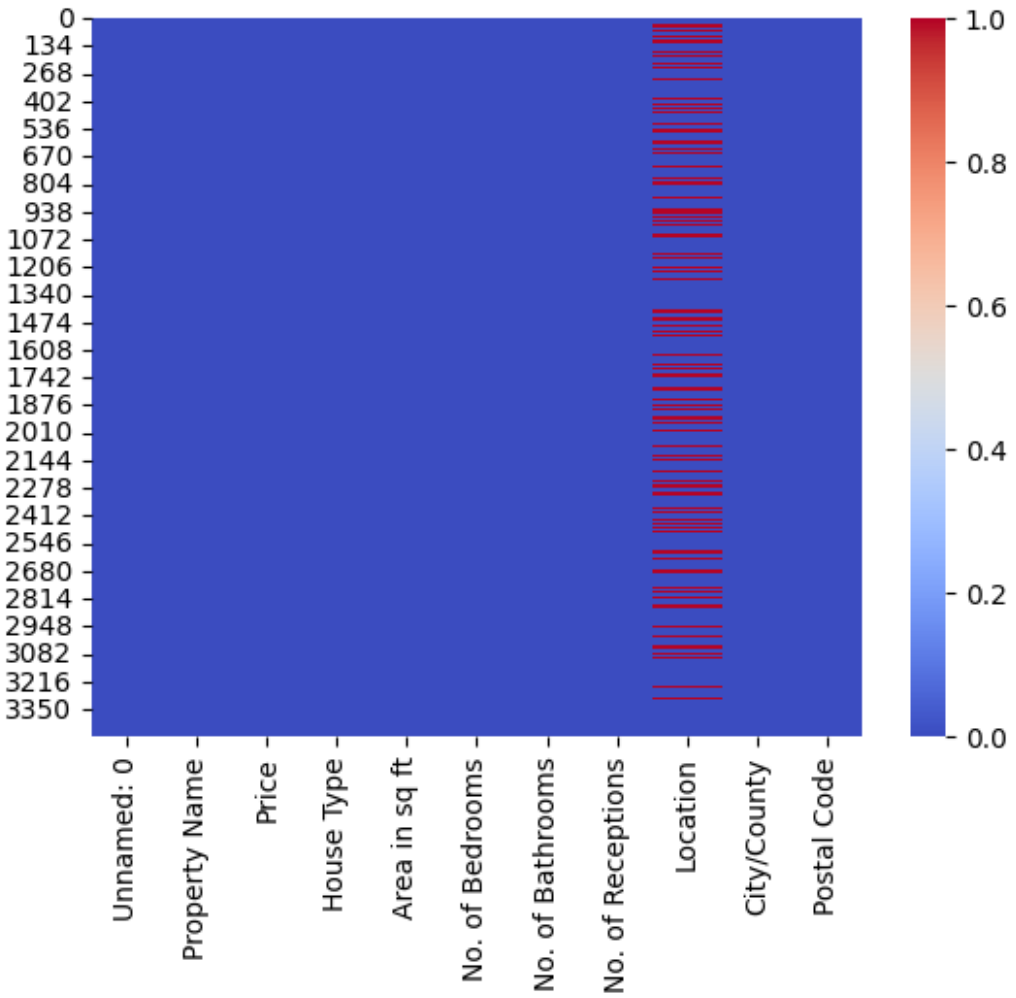
```
# Check for missing values:
data.isnull().sum()
```

```
Unnamed: 0              0
Property Name           0
Price                   0
House Type              0
Area in sq ft           0
No. of Bedrooms         0
No. of Bathrooms        0
No. of Receptions       0
Location              962
City/County             0
Postal Code             0
dtype: int64
```

Location has 962 null values

Heatmap of Missing Values in London House Price Dataset

```
sns.heatmap(data.isnull(), cmap='coolwarm')
plt.show()
```



Findings:

- Location column contains missing values: As per the heatmap, only the "Location" column shows missing data. The red and blue stripes in this column indicate the presence of both null and non-null values.
- No missing data in other columns: All other columns, such as "Property Name," "Price," "House Type," "Area in sq ft," and so on, are fully populated, represented by the solid blue color across these variables. This heatmap confirms that the dataset is mostly complete except for some missing data in the "Location" column.

```
# Find distinct values in 'House Type'
distinct_House_Type = data['House Type'].unique()

print("Distinct values in 'House Type':", distinct_House_Type)
# Find frequency of each distinct value in 'House Type'
distinct_House_Type = data['House Type'].value_counts()

print("Frequency of each value in 'No. of Bedrooms':\
n",distinct_House_Type)
```

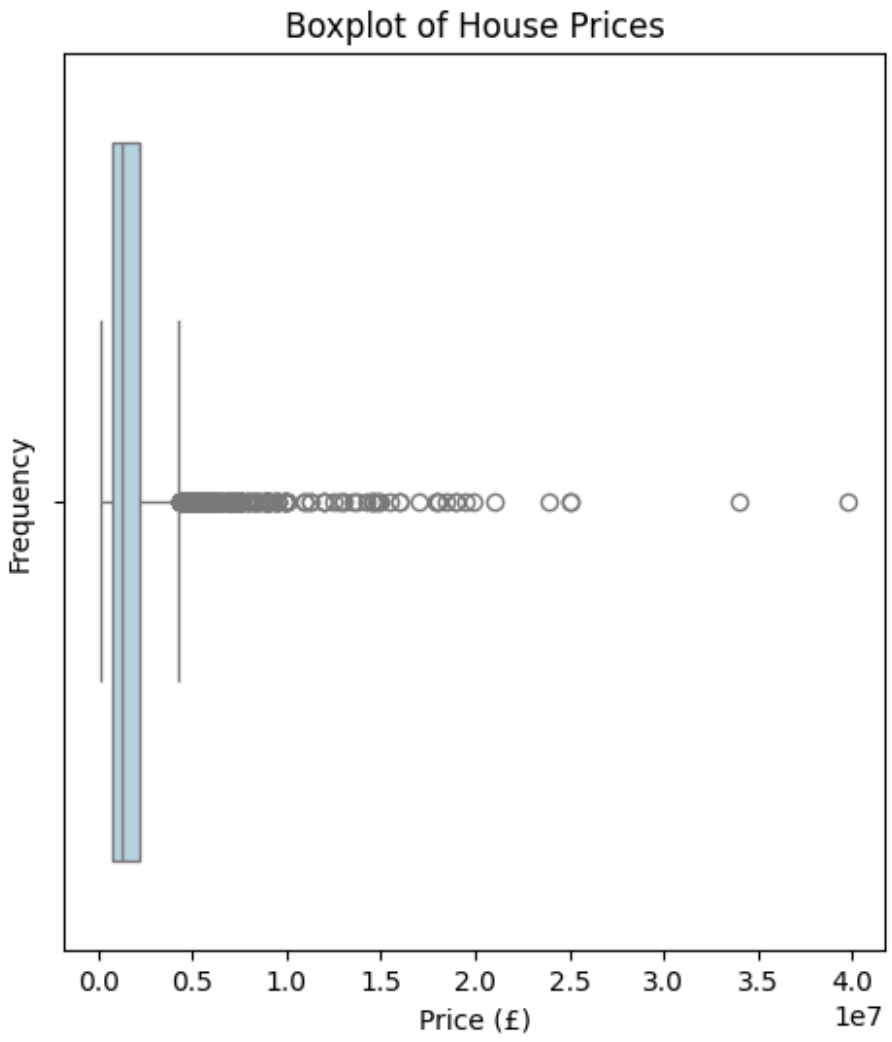
```
print(data['Price'].max())
print(data['Price'].min())
print(data['Price'].mean())
print(data['Price'].describe())
```

```
39750000
180000
1864172.5399425288
count    3.480000e+03
mean     1.864173e+06
std      2.267283e+06
min      1.800000e+05
25%      7.500000e+05
50%      1.220000e+06
75%      2.150000e+06
max      3.975000e+07
Name: Price, dtype: float64
```

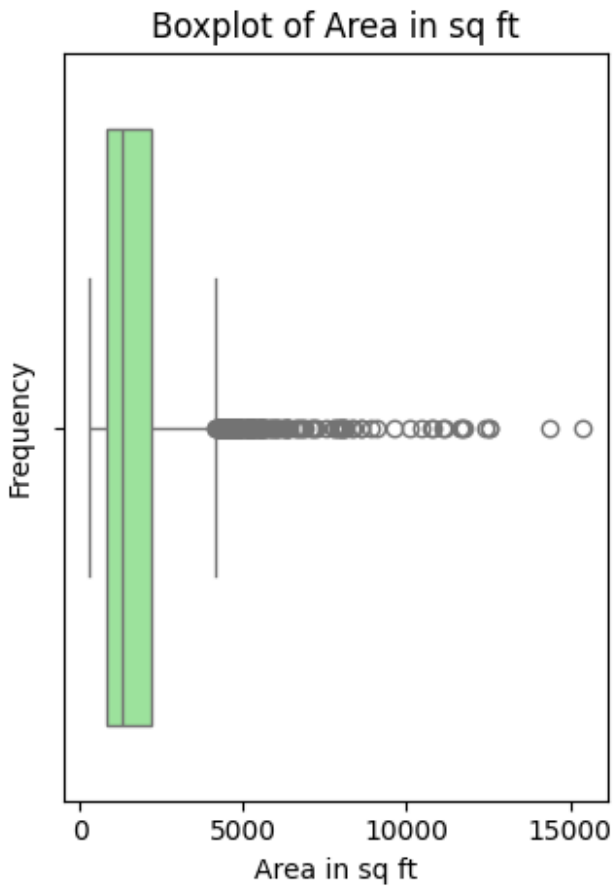
Outlier Detection A. Using Boxplots A boxplot is an effective way to visualize outliers. Data points outside the whiskers of the boxplot are considered potential outliers.

```
 #Boxplot for 'Price'
plt.figure(figsize=(12, 6))
plt.subplot(1, 2, 1)
sns.boxplot(x=data['Price'], color='lightblue')
plt.title('Boxplot of House Prices')         #Title of the boxplot
plt.xlabel('Price (£)')                       # Xaxis label
plt.ylabel('Frequency')                       # Yaxis label
```

```
Text(0, 0.5, 'Frequency')
```




```
# Boxplot for 'Area in sq ft'
plt.subplot(1, 2, 2)
sns.boxplot(x=data['Area in sq ft'], color='lightgreen')
plt.title('Boxplot of Area in sq ft')        # Title of the boxplot
plt.xlabel('Area in sq ft')                  # Xaxis label
plt.ylabel('Frequency')                       # Yaxis label (optional)


plt.tight_layout()
plt.show()
```



1. Boxplot of House Prices

In the first boxplot for house prices:

X-axis (Price): The house prices are represented on the x-axis in pounds (£). The scale shows a range up to around 40 million (£).

Outliers: There are a large number of outliers (represented by circles) that are located far from the main distribution of the data. These represent properties that are priced significantly higher than the majority of the dataset. In this case, anything beyond the whiskers (around the upper quartile) is considered an outlier.

Whiskers and the Box: The majority of the house prices fall within a very small range towards the left of the boxplot. This indicates that most house prices are lower (likely clustered near the lower end). The box (between the first and third quartile) shows the interquartile range (IQR), and the whiskers extend to the minimum and maximum non-outlier values.

Right Skewness: The plot is highly right-skewed, meaning a few properties are priced much higher than the majority. The presence of many outliers indicates that a few very expensive properties are pulling the mean price upwards.

2. Boxplot of Area in sq ft

In the second boxplot for area in square feet:

X-axis (Area in sq ft): The area in square feet is plotted along the x-axis, with values ranging from 0 to 15,000 sq ft.

Outliers: Similar to the price boxplot, there are several outliers (represented by circles) for larger properties, which are significantly bigger than the majority. These are homes with a much larger area, pulling the right side of the plot.

Whiskers and the Box: Most properties have a relatively smaller area, as shown by the narrow box and the position of the whiskers (the minimum and maximum non-outlier values). The box and whiskers cover a tight range, with most homes having an area far smaller than 15,000 sq ft.

Right Skewness: This boxplot is also right-skewed, indicating a small number of large properties (high area) are outliers compared to the majority of homes.

General Observations: Outliers: Both plots show a significant number of outliers on the right, indicating there are several properties priced much higher and with a much larger area than most houses in the dataset. Distribution: Both datasets (price and area) are right-skewed, meaning that while most houses are priced lower and are smaller, there are a few very large or expensive properties pulling the data to the right. IQR: The majority of the data for both price and area is clustered close to the lower quartiles. This suggests that most houses in the dataset are relatively affordable (compared to the highest price outliers) and smaller in size.

General Observations:

Outliers: Both plots show a significant number of outliers on the right, indicating there are several properties priced much higher and with a much larger area than most houses in the dataset.

Distribution: Both datasets (price and area) are right-skewed, meaning that while most houses are priced lower and are smaller, there are a few very large or expensive properties pulling the data to the right.

IQR: The majority of the data for both price and area is clustered close to the lower quartiles. This suggests that most houses in dataset are relatively affordable (compared to the highest price outliers) and smaller in size.

Analysis of Boxplot chars and output of describing the price column

```
count    3.480000e+03
mean     1.864173e+06
std      2.267283e+06
min      1.800000e+05
25%      7.500000e+05
50%      1.220000e+06
75%      2.150000e+06
max      3.975000e+07
```

This summary gives the descriptive statistics of the `Price` column in the dataset.

1. **Count**:

- `count = 3480`
- This tells that there are **3,480 entries** (or houses) in dataset for the `Price` column.

2. **Mean**:

- `mean = 1.864173e+06`
- The **mean (average) price** of the houses is approximately **£1,864,173**. This is computed by summing all the prices and dividing by the number of houses (3,480).
- The mean can be influenced by **outliers**, especially since boxplot showed some very high house prices.

3. **Standard Deviation (std)**:

- `std = 2.267283e+06`
- The **standard deviation** of the prices is approximately **£2,267,283**.
- This measures how much house prices **vary** from the **mean**. A high standard deviation indicates that there is **wide variability** in the house prices, meaning some prices are much higher or lower than the average.

4. **Minimum Price (min)**:

- `min = 1.800000e+05`
- The **cheapest** house in dataset is priced at **£180,000**. This is the **minimum** value in the price column.

5. **25th Percentile (25%)**:

- `25% = 7.500000e+05`
- The **25th percentile** (or first quartile, **Q1**) price is **£750,000**.
- This means that **25%** of the houses in dataset are priced **below £750,000**, and **75%** are priced **above £750,000**.

6. **50th Percentile (50% or Median)**:

- `50% = 1.220000e+06`
- The **50th percentile**, or **median** price, is **£1,220,000**.
- This means that **half of the houses** in dataset are priced **below £1.22 million** and the other half are priced above this value.
- The **median** is a better measure of central tendency in the presence of outliers, as it is not influenced by extremely high values (like the ones saw in the boxplot).

7. **75th Percentile (75%)**:

- `75% = 2.150000e+06`
- The **75th percentile** (or third quartile, **Q3**) price is **£2.15 million**.
- This means that **25% of the houses** are priced **above £2.15 million**, and **75%** are priced **below £2.15 million**.

8. **Maximum Price (max)**:

- `max = 3.975000e+07`
- The **most expensive** house in dataset is priced at **£39.75 million**.
- This extremely high value is why the dataset is **right-skewed**, as shown in the boxplot with many **outliers** on the higher end of prices.

Insights from this summary:

- **Right-Skewed Distribution**:
  - The fact that the **mean** (£1.86M) is much higher than the **median** (£1.22M) indicates that the data is **right-skewed**. This is confirmed by the **boxplot** with many high-priced outliers.
- **Wide Range of Prices**:
  - The prices vary widely, from **£180,000** to **£39.75 million**. This suggests that dataset contains both affordable homes and extremely expensive properties.
- **Outliers Affecting the Mean**:
  - Since the **mean** is higher than the **median**, the **expensive houses (outliers)** are pulling the mean up. For a better understanding of central tendency, may want to rely more on the **median**.

```
# Find distinct values in 'House Type'
distinct_House_Type = data['House Type'].unique()

print("Distinct values in 'House Type':", distinct_House_Type)
# Find frequency of each distinct value in 'House Type'
distinct_House_Type = data['House Type'].value_counts()

print("Frequency of each value in 'No. of Bedrooms':\
n",distinct_House_Type)
```

```
Distinct values in 'House Type': ['House' 'Flat / Apartment' 'New
development' 'Duplex' 'Penthouse'
 'Studio' 'Bungalow' 'Mews']
Frequency of each value in 'No. of Bedrooms':
 House Type
Flat / Apartment    1565
House               1430
New development      357
Penthouse            100
Studio                10
Bungalow               9
Duplex                 7
Mews                   2
Name: count, dtype: int64
```

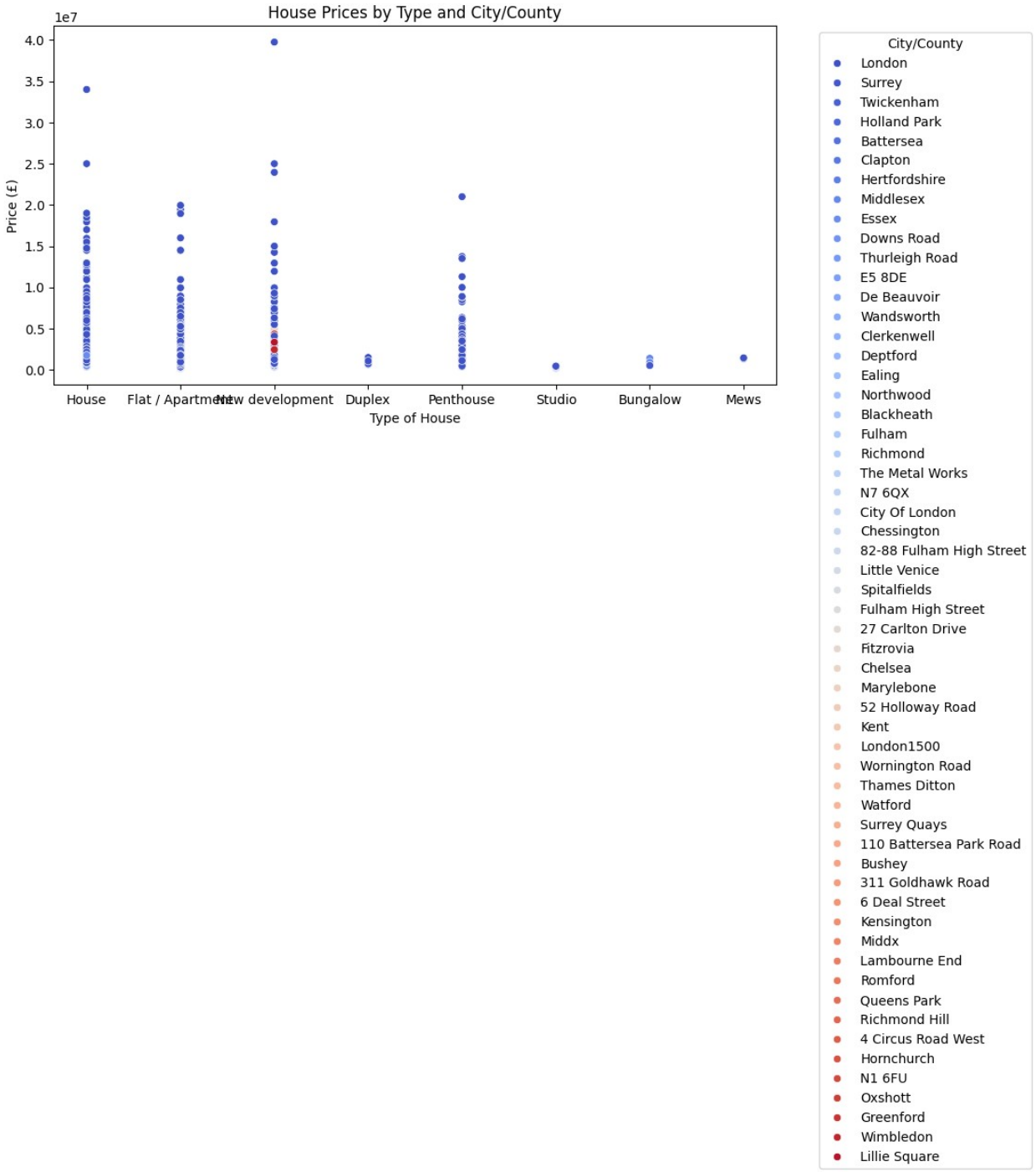
```
# Find distinct values in 'No. of Bedrooms'
distinct_bedrooms = data['No. of Bedrooms'].unique()
```

```
print("Distinct values in 'No. of Bedrooms':", distinct_bedrooms)
# Find frequency of each distinct value in 'No. of Bedrooms'
bedroom_counts = data['No. of Bedrooms'].value_counts()

print("Frequency of each value in 'No. of Bedrooms':\n",
bedroom_counts)
```

```
Distinct values in 'No. of Bedrooms': [ 5  2  4  1  3  6 10  7  0  8
9]
Frequency of each value in 'No. of Bedrooms':
 No. of Bedrooms
2     1078
3      706
4      576
5      453
1      414
6      176
7       53
8       10
0       10
9        3
10       1
Name: count, dtype: int64
```

```
import seaborn as sns
import matplotlib.pyplot as plt


# Create figure size
plt.figure(figsize=(10, 5))

# Create a boxplot by 'House Type' with colors for each 'City/County'
boxplot = sns.scatterplot(x='House Type', y='Price', data=data,
hue='City/County', palette='coolwarm')

# Add title and labels
plt.title('House Prices by Type and City/County')
plt.xlabel('Type of House')
plt.ylabel('Price (£)')

# Add the legend outside the plot
plt.legend(title='City/County', bbox_to_anchor=(1.05, 1), loc='upper
left')

# Adjust layout so the plot fits well
plt.tight_layout()

# Display the plot
plt.show()
```

```
C:\Users\anasu\AppData\Local\Temp\ipykernel_10896\519189339.py:21:
UserWarning: Tight layout not applied. The bottom and top margins
cannot be made large enough to accommodate all Axes decorations.
  plt.tight_layout()
```




```
one_bedroom_count = data[data['No. of Bedrooms'] == 2].shape[0]
print(f"Total number of 2bedroom houses: {one_bedroom_count}")
```

```
two_bedroom_count = data[data['No. of Bedrooms'] == 2].shape[0]
print(f"Total number of 2bedroom houses: {two_bedroom_count}")

three_bedroom_count = data[data['No. of Bedrooms'] == 3].shape[0]
print(f"Total number of 3bedroom houses: {three_bedroom_count}")
```

```
Total number of 2-bedroom houses: 1078
Total number of 3-bedroom houses: 706
```

Scatter plot of **house prices** by **type of house** across different **parts of London** .

X-Axis: Type of House

- The x-axis lists various types of houses, including:
  - **House**
  - **Flat/Apartment**
  - **New Development**
  - **Duplex**
  - **Penthouse**
  - **Studio**
  - **Bungalow**
  - **Mews**

  These categories represent different housing types.

Y-Axis: Price (£)

- The y-axis represents **house prices in pounds (£)**, ranging from 0 up to **£40 million (4×10^7)**.

  Most prices seem to be concentrated below **£10 million**, but there are a few high-value properties.

Data Points:

- Each **circle** on the plot represents a **house price** for a specific **type of house** in a certain **part of the city**.
- The vertical spread of dots within each house type category indicates the **range of prices** for that particular type of house.

Legend (on the right):

- The colors of the data points correspond to different **area**, with each location having its own color.
  - Example locations include **London**, **Surrey**, **Twickenham**, **Richmond**, etc.
  - Each location has a distinctive color, helping to identify how house prices vary by region.

Observations:

1. **Houses and Flats/Apartments** have the most price points, suggesting they are the most common property types.

2. **Penthouses** have some of the highest prices, with a few exceeding £10 million.
3. **New Developments** also seem to have a wide price range, with a significant number of high-value properties.
4. Other categories like **Bungalows**, **Studios**, and **Mews** show much lower price ranges, with fewer extreme values.

Key Takeaway:

This chart provides an overview of the **price distribution of different types of housing** across various parts of the city . We can see which areas have higher concentrations of expensive properties and how certain property types (like penthouses and new developments) tend to have higher prices compared to others (like studios and bungalows).

Outlier Analysis

Using Z-scores A z-score measures how many standard deviations a data point is from the mean. A common threshold to consider a value an outlier is if its z-score is greater than 3 or less than -3.

Here's the z-scores to detect outliers in the "Price" and "Area in sq ft" columns:

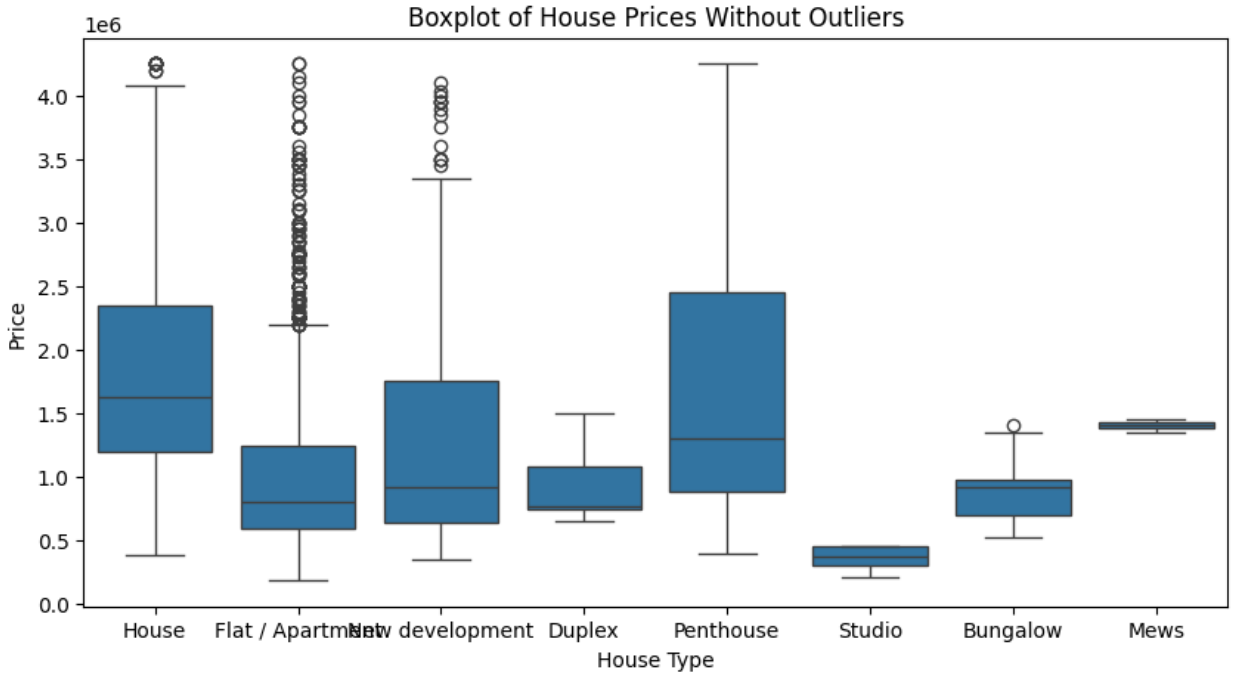
```
# Import necessary library for z-score
from scipy import stats

# Calculate z-scores for 'Price' and 'Area in sq ft'
data['Price_zscore'] = stats.zscore(data['Price'])
data['Area_zscore'] = stats.zscore(data['Area in sq ft'])

# Find outliers based on z-score
price_outliers = data[data['Price_zscore'].abs() > 3]
area_outliers = data[data['Area_zscore'].abs() > 3]

print("Number of price outliers:", price_outliers.shape[0])
print("Number of area outliers:", area_outliers.shape[0])
```

```
Number of price outliers: 64
Number of area outliers: 58
```

1. **Removing Outliers and Analysing Again**
   - Appling the **IQR method** to remove outliers:

```
Q1 = data['Price'].quantile(0.25)
Q3 = data['Price'].quantile(0.75)
IQR = Q3 - Q1
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR
```

```
# Remove outliers
data_no_outliers = data[(data['Price'] >= lower_bound) &
(data['Price'] <= upper_bound)]

# Boxplot without outliers
plt.figure(figsize=(10, 5))
sns.boxplot(x='House Type', y='Price', data=data_no_outliers)
plt.title('Boxplot of House Prices Without Outliers')
plt.show()
```



From the boxplot of house prices without outliers for different house types, several key findings can be observed:

1. **Price Variation Across House Types:**

- **Penthouse** properties have the highest median price compared to other house types. The interquartile range (IQR) is also quite large, indicating significant price variation for this type of property.
- **Houses** and **Mews** properties follow as the next most expensive categories, with relatively higher median prices, although Mews have very little price variation as seen by the small IQR.
- **Flat/Apartments** and **New Developments** have more moderate prices, with the former having a wider spread in prices, suggesting more variety within this category.
- **Bungalows** and **Studios** tend to be on the lower end in terms of price, with studios being the least expensive overall.

2. **Price Consistency**:

- **Mews** and **Duplex** properties show less price variation (narrow IQR), indicating that prices are more consistent for these house types.
- **Flat/Apartments** and **New Developments** show higher price variability, especially Flat/Apartments, which have a much wider IQR, reflecting a range of pricing within these categories.

3. **Lower-Price Categories**:

- **Studio** properties show the lowest median price and the smallest overall range in prices. This reflects their relatively lower market value and possibly more affordability.
- **Bungalows** also show lower prices compared to most other house types, though the variability in pricing is slightly higher than studios.

4. **Penthouse as a Luxury Segment**:

- The **Penthouse** category's high median and wide IQR point to its status as a premium housing type. These properties may cater to the luxury market, explaining the significant price differences compared to other categories.

5. **Insights on Missing House Types**:

- Some house types like **Mews** and **Studios** show fewer data points, as seen by the limited number of outliers and smaller IQR. This could indicate fewer listings of these types or a more homogeneous pricing pattern.

Overall, the boxplot shows a clear distinction in the pricing trends across different house types, with Penthouses leading the premium segment, followed by Houses and Mews, while Bungalows and Studios are at the lower end of the price spectrum.

Comparison Between Boxplots (With and Without Outliers)

1. **Impact of Outliers on the Distribution**:

- **With outliers** (the new plot): The price distribution is highly skewed due to the presence of extreme values. There are a significant number of outliers on the higher end, with some properties exceeding £10 million, and even going up to almost £40 million. These outliers stretch the plot horizontally, making it harder to analyze the majority of the data.
- **Without outliers** (the previous plot): By removing outliers, the boxplot shows a much clearer and compact distribution. This helps focus on the central tendency and variability of property prices, making it easier to analyze trends for the typical range of house prices.

2. **Median and Quartile Comparisons**:

- **With outliers**: The boxplot shows a narrower box (the interquartile range, or IQR), suggesting that the majority of properties are priced within a lower range (below £1 million), but the overall visual is dominated by the extreme values on the higher end.
- **Without outliers**: The medians are clearer for each house type, and the IQR is wider, giving more insight into how prices vary for the bulk of properties. This helps to avoid the distortion that the extreme luxury properties caused.

3. **Range of Prices**:
   - **With outliers**: The total range of house prices extends dramatically, from below £100,000 to close to £40 million. This extreme range distorts the visual representation, making it difficult to discern pricing trends for the majority of properties.
   - **Without outliers**: The range is significantly narrower, focusing on properties that fall within a more typical range. This makes the price trends much more interpretable.

4. **Price Anomalies**:
   - **With outliers**: The outliers likely represent luxury properties, which distort the overall view of the housing market. These outliers can be interesting to analyze separately but tend to obscure the general trends.
   - **Without outliers**: By removing these anomalies, we can see a clearer picture of the pricing patterns for most properties, giving a better sense of what buyers can typically expect in the London housing market.

Key Findings from the Comparison:

- **Without outliers**, the boxplot is easier to interpret, focusing on the majority of properties that fall within the typical price range. This gives a more accurate picture of price trends and variation across house types.
- **With outliers**, the luxury property market skews the data, making it difficult to analyze the core trends. While it's valuable to acknowledge that such high-priced properties exist, they should be analyzed separately from the bulk of the data.

Suggested Conclusion:

Including both plots in your presentation will allow you to highlight the difference between the **overall housing market** (with outliers) and the **typical market trends** (without outliers). This shows that while London has extremely high-priced properties, most properties fall within a much more accessible price range.

Summary statistics without outliers

```
 # Summary statistics without outliers
print(data_no_outliers['Price'].describe())
```

```
count    3.223000e+03
mean     1.390577e+06
std      8.716327e+05
min      1.800000e+05
25%      7.100000e+05
50%      1.135000e+06
75%      1.850000e+06
max      4.250000e+06
Name: Price, dtype: float64
```

The **summary statistics** give a detailed overview of the **price distribution** after outliers have been removed.

1. **Count**:

- `3,223`: This is the number of data points (or houses) included in the analysis **after removing outliers**.
- It tells that the dataset has **3,223 house prices** that fall within the reasonable range (without extreme outliers).

2. **Mean (Average Price)**:

- `1.39 million (£1,390,577)`: This is the **average house price** in the dataset.
- The **mean** represents the "typical" price, but since house prices can vary widely, it can still be influenced by somewhat high or low values, even after outliers are removed.

3. **Standard Deviation (std)**:

- `871,632.7 (£871,633)`: This measures the **spread or variability** of house prices around the mean.
- A **high standard deviation** like this means that house prices vary significantly, with a large range of prices. Even after outliers are removed, there are still expensive properties pushing the distribution wider.

4. **Minimum (min)**:

- `180,000 (£180,000)`: This is the **lowest house price** in the dataset after outliers were removed.
- This would represent the least expensive property in the dataset, indicating that there are still more affordable homes even though extreme outliers have been eliminated.

5. **25th Percentile (25%)**:

- `710,000 (£710,000)`: This is the **first quartile (Q1)**, meaning that **25% of the house prices** in the dataset are **less than or equal to** £710,000.
- It helps to understand the lower range of prices—**25% of the houses are cheaper** than this amount.

6. **Median (50%)**:

- `1.135 million (£1,135,000)`: The **median price** is the middle value of the dataset, meaning **50% of the house prices are below** this value, and **50% are above**.
- The **median** is often a better measure than the mean for skewed distributions because it isn't influenced by extreme values as much.

7. **75th Percentile (75%)**:

- `1.85 million (£1,850,000)`: This is the **third quartile (Q3)**, meaning **75% of house prices** are **less than or equal to** £1.85 million.
- This indicates that the upper 25% of houses are priced above this, showing that there are many high-value properties, but most homes are priced below this point.

8. **Maximum (max)**:

- `4.25 million (£4,250,000)`: This is the **highest house price** in the dataset (after outliers were removed).

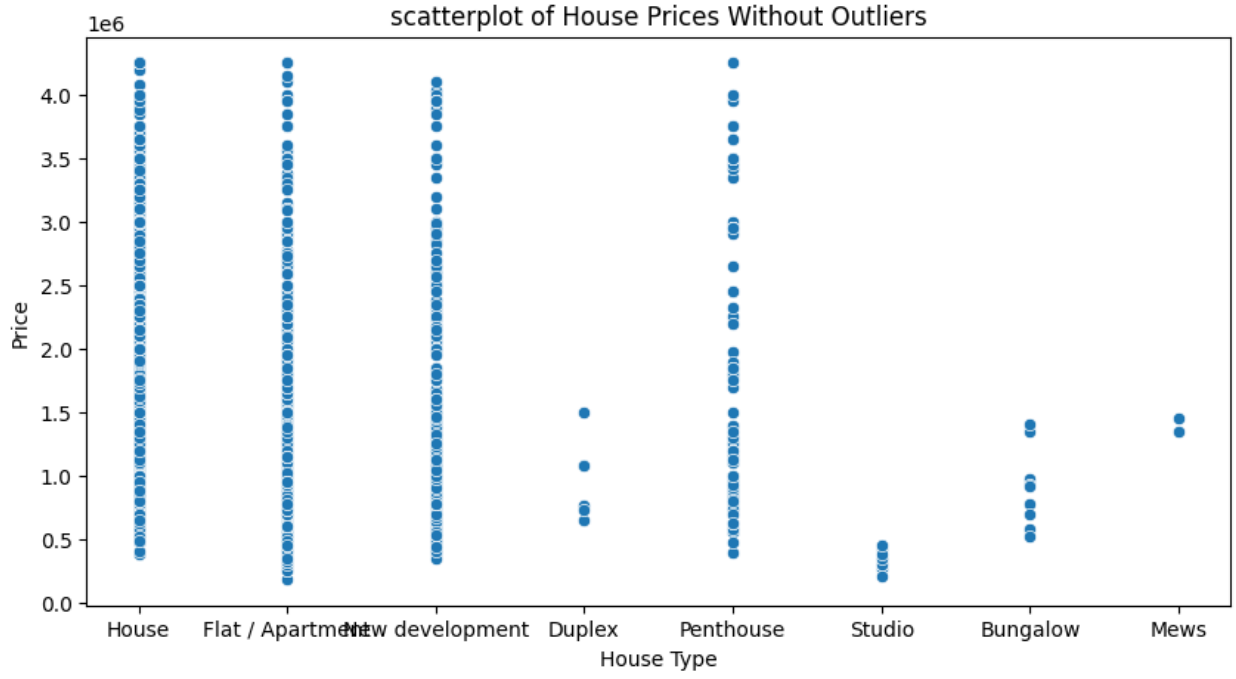
- It indicates that even after removing outliers, there are still high-end properties that cost several million pounds, but prices beyond this were considered outliers and removed.

Summary of Insights:

- **Average price** is around £1.39 million, but the **median price** is £1.135 million. The fact that the **mean is higher than the median** suggests that the distribution is **right-skewed**, meaning there are still expensive properties pulling the average higher.
- House prices vary a lot, with a **standard deviation** of over **£871,000**, indicating a wide range of prices, even after removing outliers.
- The **25th percentile** tells us that 25% of properties are priced at or below **£710,000**, while the **75th percentile** shows that 75% are priced below **£1.85 million**, giving a range of typical prices.
- The **maximum price** is still quite high at £4.25 million, but significantly lower than the extreme outliers that were removed from the dataset.

This analysis shows that the data, even without outliers, has significant variation, including both relatively affordable homes and high-end properties.

```
 # Boxplot without outliers
plt.figure(figsize=(10, 5))
sns.scatterplot(x='House Type', y='Price', data=data_no_outliers)
plt.title('scatterplot of House Prices Without Outliers')
plt.show()
```



2. Data Normalization

Normalization is typically done to scale numerical features for machine learning models. This ensures that features like "Price" and "Area in sq ft" are on a comparable scale. We can normalize using methods like min-max scaling or standardization (z-score normalization).

A. Min-Max Normalization: This scales the data to a range between 0 and 1.

```
from sklearn.preprocessing import MinMaxScaler

# Initialize the MinMaxScaler
scaler = MinMaxScaler()

# Normalize the 'Price' and 'Area in sq ft' columns
data[['Price', 'Area in sq ft']] = scaler.fit_transform(data[['Price',
'Area in sq ft']])

print(data[['Price', 'Area in sq ft']].head())
```

```
      Price  Area in sq ft
0  0.037781       0.161391
1  0.011878       0.035688
2  0.014026       0.032186
3  0.040056       0.113145
4  0.012509       0.028154
```

B. Z-Score Normalization (Standardization): Z-score normalization (standardization) scales the data such that the mean becomes 0 and the standard deviation becomes 1.

```
from sklearn.preprocessing import StandardScaler

# Initialize the StandardScaler
scaler = StandardScaler()

# Standardize the 'Price' and 'Area in sq ft' columns
data[['Price', 'Area in sq ft']] = scaler.fit_transform(data[['Price',
'Area in sq ft']])

print(data[['Price', 'Area in sq ft']].head())
```

```
      Price  Area in sq ft
0 -0.083448       0.735322
1 -0.535596      -0.659041
2 -0.498101      -0.697895
3 -0.043747       0.200157
4 -0.524568      -0.742615
```

1. **Price vs. Area (Price vs. "Area in sq ft")**

- **Scatter Plot with Regression Line**: Visualising how price changes with square footage. And a **regression line** to see if there's a linear relationship (i.e., higher area leads to higher prices).

```
import seaborn as sns
import matplotlib.pyplot as plt

# Scatter plot of Price vs Area with regression line
# plt.figure(figsize=(10, 5))
sns.regplot(x='Area in sq ft', y='Price', data=data)
plt.title('Price vs. Area (with Regression Line)')
plt.show()
```

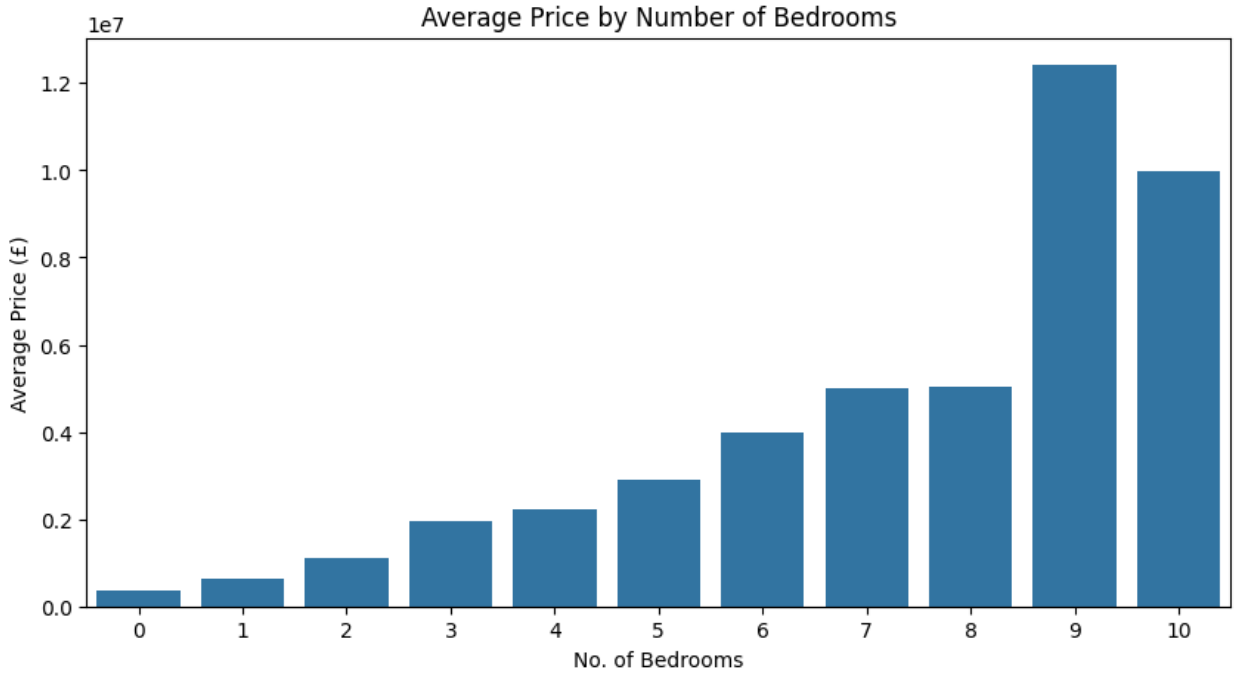
- The **regression line** shows the correlation between price and area, showing whether prices generally increase as square footage increases.

2. **Price by Number of Bedrooms/Bathrooms**

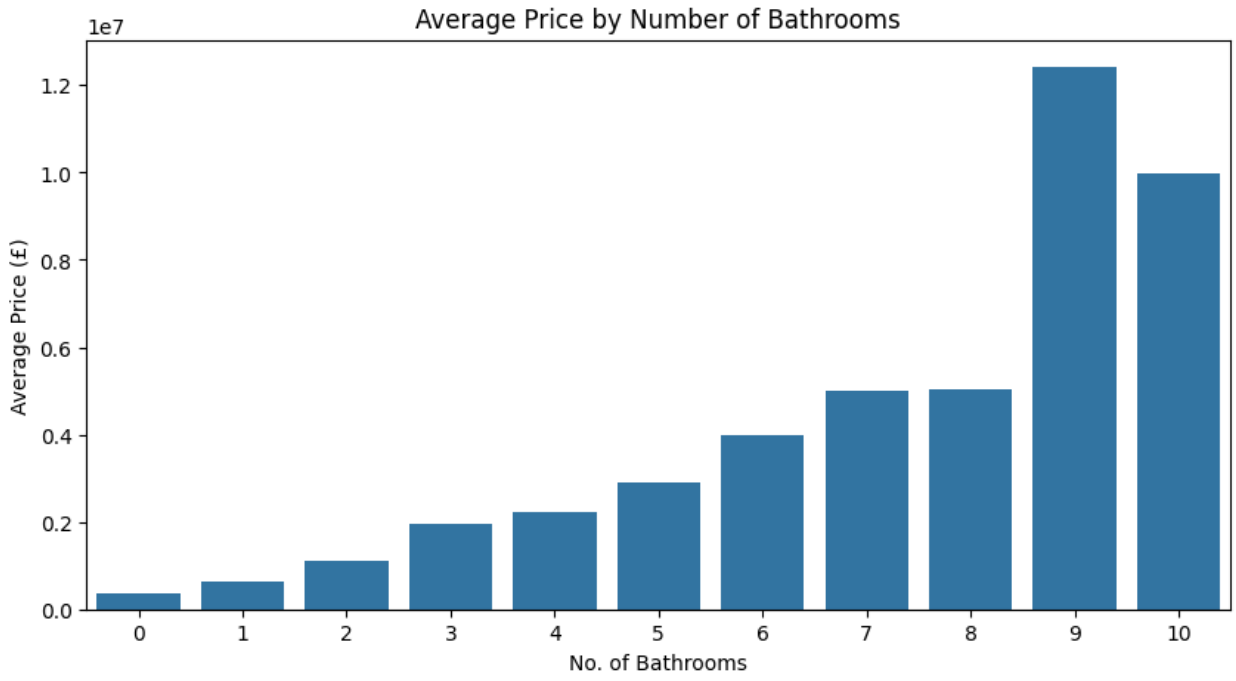
- **Column Chart**: Each category (e.g., 1 bedroom, 2 bedrooms, 3 bedrooms) displayed on the x-axis, and the average price plotted on the y-axis.

```
# Group by number of bedrooms, calculate the mean price
avg_price_by_bedrooms = data.groupby('No. of Bedrooms')
```

```
['Price'].mean().reset_index()

# Bar plot for average price by number of bedrooms
plt.figure(figsize=(10, 5))
sns.barplot(x='No. of Bedrooms', y='Price',
data=avg_price_by_bedrooms)
plt.title('Average Price by Number of Bedrooms')
plt.ylabel('Average Price (£)')
plt.show()
```

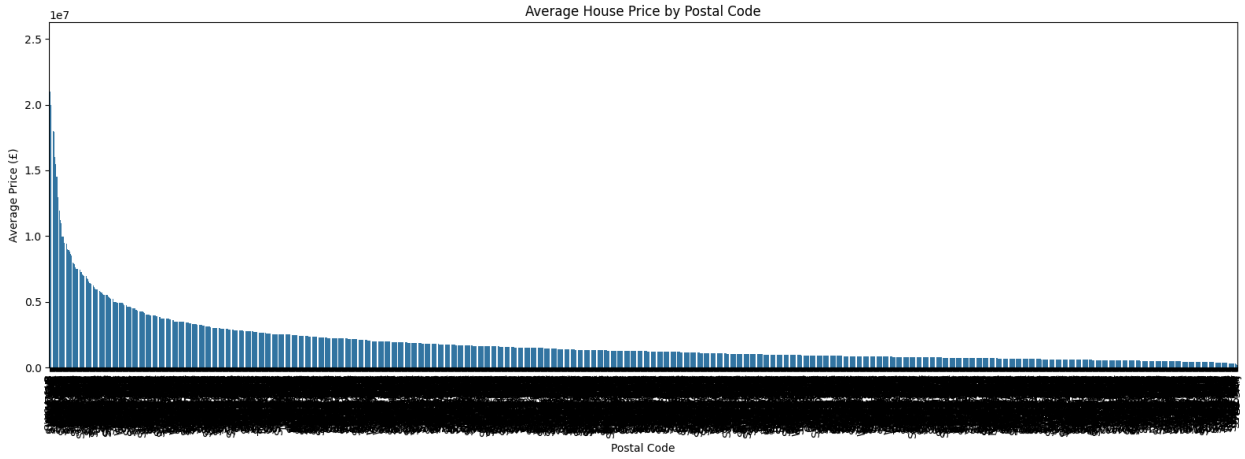



```
# Group by number of bathrooms, calculate the mean price
avg_price_by_bathrooms = data.groupby('No. of Bathrooms')
['Price'].mean().reset_index()

# Bar plot for average price by number of bathrooms
plt.figure(figsize=(10, 5))
sns.barplot(x='No. of Bathrooms', y='Price',
data=avg_price_by_bathrooms)
plt.title('Average Price by Number of Bathrooms')
plt.ylabel('Average Price (£)')
plt.show()
```



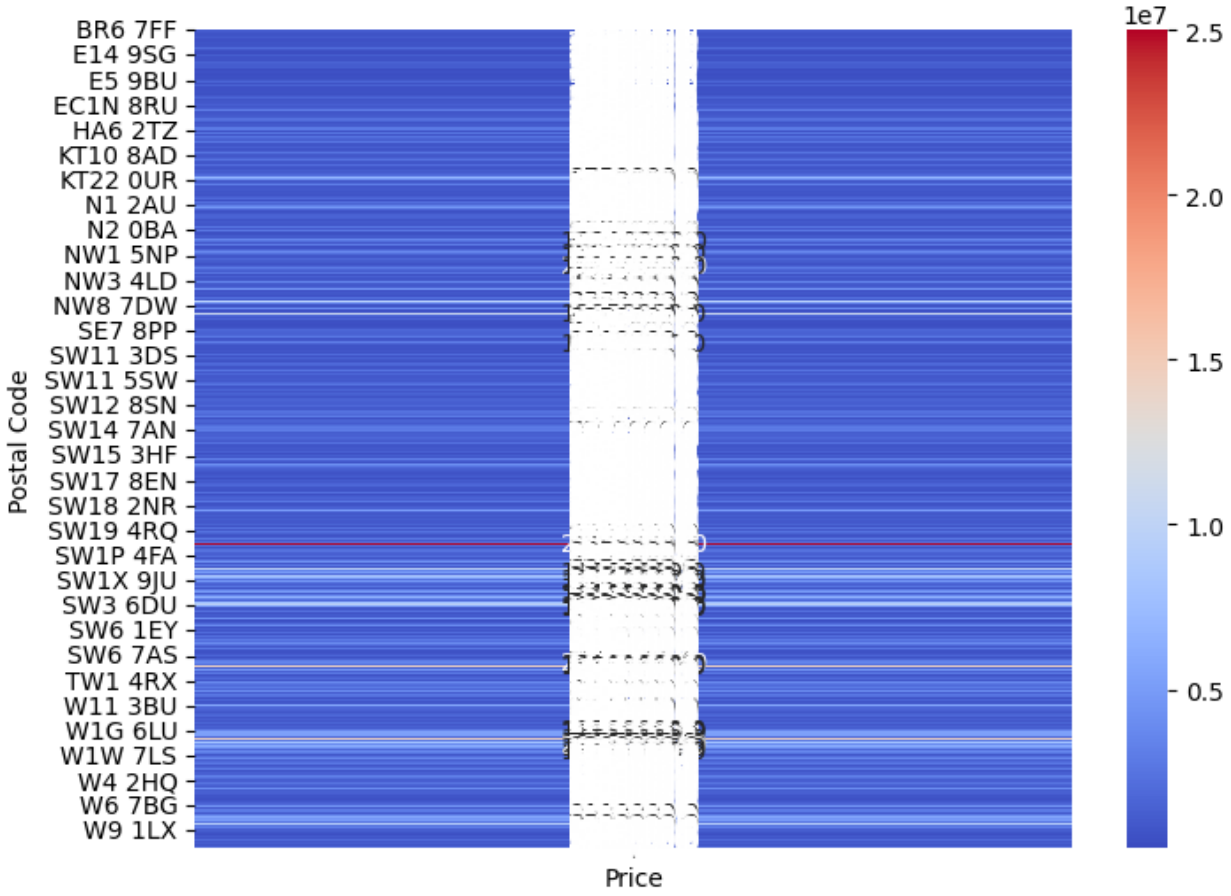
1. Bar Chart for Average Price by Postal Code

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Group by 'Postal Code' and calculate the mean price
avg_price_by_postal_code = data.groupby('Postal Code')
['Price'].mean().reset_index()

# Sort by price to make the chart more readable
avg_price_by_postal_code =
avg_price_by_postal_code.sort_values(by='Price', ascending=False)

# Create the bar plot
plt.figure(figsize=(16, 6))
sns.barplot(x='Postal Code', y='Price', data=avg_price_by_postal_code)

# Title and labels
plt.title('Average House Price by Postal Code')
plt.xlabel('Postal Code')
plt.ylabel('Average Price (£)')
plt.xticks(rotation=90)  # Rotate the x labels for better readability

plt.tight_layout()
plt.show()
```



The bar chart depicts **average house prices by postal code** in London. The distribution is highly skewed, with a few postal codes showing significantly higher average prices, potentially exceeding £10 million, while the majority of postal codes have much lower average prices. This indicates that a small number of highly expensive areas dramatically elevate the upper range of property prices, while most areas have more moderate pricing. The chart highlights the significant disparity in house prices across different regions of London.

```
# Group by postal code and calculate the mean price for each
price_by_postal_code = data.groupby('Postal Code')
['Price'].mean().reset_index()

# Sort values by Price (optional, for better visual comparison)
price_by_postal_code = price_by_postal_code.sort_values('Price',
ascending=False)

# Convert to pivot table (for use in heatmap)
price_pivot = pd.pivot_table(price_by_postal_code, index='Postal
Code', values='Price')

# Create the heatmap
plt.figure(figsize=(8, 6))
sns.heatmap(price_pivot, annot=True, fmt=".1f", cmap='coolwarm')


<Axes: ylabel='Postal Code'>
```



Price Distribution: The majority of the price points appear to be centered around a specific range (likely between 0 and 1 million), as shown by the denser regions (dark blue areas) in the center of the plot.

Postal Code Variations: The price distribution seems fairly consistent across most postal codes. No clear outliers or extreme differences are visually evident between the postal codes listed on the y-axis.

Heatmap Colors: The color scale indicates prices ranging up to 25 million (in red), though it seems that most prices are concentrated in the lower range (blues), with fewer higher-priced properties reaching above 10 million.

The white zone in the middle of the graph represents a **price gap**, where there are few or no properties within that price range across the postal codes. Most properties either fall in the **lower price range** (shown by dark blue) or in **higher price ranges** (shades of red), with very few in between. This gap likely indicates that the dataset has mostly affordable or expensive properties but lacks representation in the middle price tier.

Conclusion

In this project, I conducted an in-depth analysis of house prices in London, examining various factors that influence property values. Through exploratory data analysis, statistical methods, and visualizations (such as boxplots), uncovered several important trends and insights regarding the London housing market.

Key Findings:

- **House Type**: The type of property plays a significant role in determining price. For instance, penthouses and houses generally have higher median prices, whereas studios and flats tend to be more affordable.
- **Price Variation**: There is considerable price variation within certain property types, particularly in flats and new developments, as evidenced by the spread of values in the boxplots. These variations may be driven by location, property size, and other amenities.
- **Outliers**: The presence of outliers, such as luxury homes or penthouses, heavily skews the price distribution, particularly in high-end areas of London. After removing outliers, the majority of properties fell within a more reasonable price range, providing a clearer picture of the typical housing market.
- **Location**: Although not fully explored in this particular analysis, it is evident from the data that location has a significant impact on prices. Central and premium neighborhoods tend to command higher property prices, further skewing the data when considering outliers.

Challenges and Limitations:

- **Missing Data**: Some properties lacked specific details, such as location data, which may have limited the accuracy of certain analyses. Future work could benefit from a more complete dataset to better understand the role of neighborhood in price determination.
- **Outliers**: While removing outliers helped in understanding the general trends, the presence of these extreme values suggests a need for separate analysis of the luxury property segment.

Future Work: This project opens the door to several opportunities for further analysis. A deeper dive into the geographical distribution of property prices across London boroughs or neighborhoods would provide additional insights. Furthermore, expanding the dataset to include factors such as proximity to transport hubs, school districts, and other amenities could offer a more comprehensive understanding of what drives house prices in London.

In conclusion, this project has successfully explored the key factors affecting house prices in London. It provides a solid foundation for further research and can be used to guide property investment decisions, policy planning, or more detailed market analysis. By continuing to expand the scope of data and analytical methods, future work can shed more light on the complex and dynamic nature of the London housing market.

Thank you for taking the time to review my project. I hope you found the analysis insightful, and I appreciate your interest in London house price market analysis. Your feedback is valuable, and I look forward to any thoughts or suggestions you might have.