In [84]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

In [85]:
```
df=pd.read_csv("bengaluruPrices.csv")
df
```

Out[85]:

	area_type	availability	location	size	society	total_sqft	bath	balcor
0	Super built-up Area	19-Dec	Electronic City Phase II	2 BHK	Coomee	1056	2.0	1
1	Plot Area	Ready To Move	Chikka Tirupathi	4 Bedroom	Theanmp	2600	5.0	3
2	Built-up Area	Ready To Move	Uttarahalli	3 BHK	NaN	1440	2.0	3
3	Super built-up Area	Ready To Move	Lingadheeranahalli	3 BHK	Soiewre	1521	3.0	1
4	Super built-up Area	Ready To Move	Kothanur	2 BHK	NaN	1200	2.0	1
...	...	...	...	...	...	...	...	
13315	Built-up Area	Ready To Move	Whitefield	5 Bedroom	ArsiaEx	3453	4.0	0
13316	Super built-up Area	Ready To Move	Richards Town	4 BHK	NaN	3600	5.0	Na
13317	Built-up Area	Ready To Move	Raja Rajeshwari Nagar	2 BHK	Mahla T	1141	2.0	1
13318	Super built-up Area	18-Jun	Padmanabhanagar	4 BHK	SollyCl	4689	4.0	1
13319	Super built-up Area	Ready To Move	Doddathoguru	1 BHK	NaN	550	1.0	1

13320 rows × 9 columns

In [86]:
```
df.shape
```

Out[86]:
```
(13320, 9)
```

In [87]:
```
df.columns
```

Out[87]:
```
Index(['area_type', 'availability', 'location', 'size', 'society',
       'total_sqft', 'bath', 'balcony', 'price'],
      dtype='object')
```

In [24]:
```
df['area_type'].unique()
```

Out[24]:
```
array(['Super built-up  Area', 'Plot  Area', 'Built-up  Area',
       'Carpet  Area'], dtype=object)
```

In [25]:
```
df['availability'].unique()
```

Out[25]:
```
array(['19-Dec', 'Ready To Move', '18-May', '18-Feb', '18-Nov', '20-Dec',
       '17-Oct', '21-Dec', '19-Sep', '20-Sep', '18-Mar', '20-Feb',
       '18-Apr', '20-Aug', '18-Oct', '19-Mar', '17-Sep', '18-Dec',
       '17-Aug', '19-Apr', '18-Jun', '22-Dec', '22-Jan', '18-Aug',
       '19-Jan', '17-Jul', '18-Jul', '21-Jun', '20-May', '19-Aug',
       '18-Sep', '17-May', '17-Jun', '21-May', '18-Jan', '20-Mar',
       '17-Dec', '16-Mar', '19-Jun', '22-Jun', '19-Jul', '21-Feb',
       'Immediate Possession', '19-May', '17-Nov', '20-Oct', '20-Jun',
       '19-Feb', '21-Oct', '21-Jan', '17-Mar', '17-Apr', '22-May',
       '19-Oct', '21-Jul', '21-Nov', '21-Mar', '16-Dec', '22-Mar',
       '20-Jan', '21-Sep', '21-Aug', '14-Nov', '19-Nov', '15-Nov',
       '16-Jul', '15-Jun', '17-Feb', '20-Nov', '20-Jul', '16-Sep',
       '15-Oct', '15-Dec', '16-Oct', '22-Nov', '15-Aug', '17-Jan',
       '16-Nov', '20-Apr', '16-Jan', '14-Jul'], dtype=object)
```

In [26]:
```
df['location'].unique()
```

Out[26]:
```
array(['Electronic City Phase II', 'Chikka Tirupathi', 'Uttarahalli', ...,
       '12th cross srinivas nagar banshankari 3rd stage',
       'Havanur extension', 'Abshot Layout'], dtype=object)
```

In [27]:
```
df['size'].unique()
```

Out[27]:
```
array(['2 BHK', '4 Bedroom', '3 BHK', '4 BHK', '6 Bedroom', '3 Bedroom',
       '1 BHK', '1 RK', '1 Bedroom', '8 Bedroom', '2 Bedroom',
       '7 Bedroom', '5 BHK', '7 BHK', '6 BHK', '5 Bedroom', '11 BHK',
       '9 BHK', nan, '9 Bedroom', '27 BHK', '10 Bedroom', '11 Bedroom',
       '10 BHK', '19 BHK', '16 BHK', '43 Bedroom', '14 BHK', '8 BHK',
       '12 Bedroom', '13 BHK', '18 Bedroom'], dtype=object)
```

In [28]:
```
df['society'].unique()
```

Out[28]:
```
array(['Coomee ', 'Theanmp', nan, ..., 'SJovest', 'ThhtsV ', 'RSntsAp'],
      dtype=object)
```

In [29]:
```
df['total_sqft'].unique()
```

Out[29]:
```
array(['1056', '2600', '1440', ..., '1133 - 1384', '774', '4689'],
      dtype=object)
```

In [30]:
```
df['bath'].unique()
```

Out[30]:
```
array([ 2.,  5.,  3.,  4.,  6.,  1.,  9., nan,  8.,  7., 11., 10., 14.,
       27., 12., 16., 40., 15., 13., 18.])
```

In [31]:
```
df['balcony'].unique()
```

Out[31]:
```
array([ 1.,  3., nan,  2.,  0.])
```

In [32]:
```
df['price'].unique()
```

Out[32]:
```
array([ 39.07, 120.  ,  62.  , ...,  40.14, 231.  , 488.  ])
```

In [33]:
```
df['area_type'].value_counts()
```

Out[33]:
```
Super built-up  Area    8790
Built-up  Area          2418
Plot  Area              2025
Carpet  Area              87
Name: area_type, dtype: int64
```

In [34]:
```
df['ready to move'] = df['availability'].apply(lambda x: 'Yes' if x == 'Rea
```

In [35]:
```
df
```

Out[35]:

	area_type	availability	location	size	society	total_sqft	bath	balco
0	Super built-up Area	19-Dec	Electronic City Phase II	2 BHK	Coomee	1056	2.0	1
1	Plot Area	Ready To Move	Chikka Tirupathi	4 Bedroom	Theanmp	2600	5.0	3
2	Built-up Area	Ready To Move	Uttarahalli	3 BHK	NaN	1440	2.0	3
3	Super built-up Area	Ready To Move	Lingadheeranahalli	3 BHK	Soiewre	1521	3.0	1
4	Super built-up Area	Ready To Move	Kothanur	2 BHK	NaN	1200	2.0	1
...	...	...	...	...	...	...	...	
13315	Built-up Area	Ready To Move	Whitefield	5 Bedroom	ArsiaEx	3453	4.0	0
13316	Super built-up Area	Ready To Move	Richards Town	4 BHK	NaN	3600	5.0	Na
13317	Built-up Area	Ready To Move	Raja Rajeshwari Nagar	2 BHK	Mahla T	1141	2.0	1
13318	Super built-up Area	18-Jun	Padmanabhanagar	4 BHK	SollyCl	4689	4.0	1
13319	Super built-up Area	Ready To Move	Doddathoguru	1 BHK	NaN	550	1.0	1

13320 rows × 10 columns

In [36]:
```
print("Before drop:", df.shape)
df.drop(['availability', 'society'], axis='columns', inplace=True)
print("After drop:", df.shape)
```

```
Before drop: (13320, 10)
After drop: (13320, 8)
```

In [37]:
```
df.isnull().sum()
```

Out[37]:
```
area_type          0
location           1
size              16
total_sqft         0
bath              73
balcony          609
price              0
ready to move      0
dtype: int64
```

In [38]:
```
df.shape
```

Out[38]:
```
(13320, 8)
```

In [39]:
```
df.dropna()
```

Out[39]:

	area_type	location	size	total_sqft	bath	balcony	price	ready to move
0	Super built-up Area	Electronic City Phase II	2 BHK	1056	2.0	1.0	39.07	No
1	Plot Area	Chikka Tirupathi	4 Bedroom	2600	5.0	3.0	120.00	Yes
2	Built-up Area	Uttarahalli	3 BHK	1440	2.0	3.0	62.00	Yes
3	Super built-up Area	Lingadheeranahalli	3 BHK	1521	3.0	1.0	95.00	Yes
4	Super built-up Area	Kothanur	2 BHK	1200	2.0	1.0	51.00	Yes
...	...	...	...	...	...	...	...	...
13314	Super built-up Area	Green Glen Layout	3 BHK	1715	3.0	3.0	112.00	Yes
13315	Built-up Area	Whitefield	5 Bedroom	3453	4.0	0.0	231.00	Yes
13317	Built-up Area	Raja Rajeshwari Nagar	2 BHK	1141	2.0	1.0	60.00	Yes
13318	Super built-up Area	Padmanabhanagar	4 BHK	4689	4.0	1.0	488.00	No
13319	Super built-up Area	Doddathoguru	1 BHK	550	1.0	1.0	17.00	Yes

12710 rows × 8 columns

In [89]:
```
df.isnull().sum()
```

Out[89]:
```
area_type          0
availability       0
location           1
size              16
society         5502
total_sqft         0
bath              73
balcony          609
price              0
dtype: int64
```

In [41]:
```
df.shape
```

Out[41]:
```
(13320, 8)
```

In [42]:
```
df.dropna(inplace=True)
```

In [43]:
```
df.shape
```

Out[43]:
```
(12710, 8)
```

In [44]:
```
df['size'].unique()
```

Out[44]:
```
array(['2 BHK', '4 Bedroom', '3 BHK', '3 Bedroom', '1 BHK', '1 RK',
       '4 BHK', '1 Bedroom', '2 Bedroom', '6 Bedroom', '8 Bedroom',
       '7 Bedroom', '5 BHK', '7 BHK', '6 BHK', '5 Bedroom', '11 BHK',
       '9 BHK', '9 Bedroom', '27 BHK', '11 Bedroom', '43 Bedroom',
       '14 BHK', '8 BHK', '12 Bedroom', '10 Bedroom', '13 BHK'],
      dtype=object)
```

In [45]:
```
df['bhk'] = df['size'].apply(lambda x: int(x.split(' ')[0]))
```

In [46]:
```
df
```

Out[46]:

	area_type	location	size	total_sqft	bath	balcony	price	ready to move	bhk
0	Super built-up Area	Electronic City Phase II	2 BHK	1056	2.0	1.0	39.07	No	2
1	Plot Area	Chikka Tirupathi	4 Bedroom	2600	5.0	3.0	120.00	Yes	4
2	Built-up Area	Uttarahalli	3 BHK	1440	2.0	3.0	62.00	Yes	3
3	Super built-up Area	Lingadheeranahalli	3 BHK	1521	3.0	1.0	95.00	Yes	3
4	Super built-up Area	Kothanur	2 BHK	1200	2.0	1.0	51.00	Yes	2
...	...	...	...	...	...	...	...	...	...
13314	Super built-up Area	Green Glen Layout	3 BHK	1715	3.0	3.0	112.00	Yes	3
13315	Built-up Area	Whitefield	5 Bedroom	3453	4.0	0.0	231.00	Yes	5
13317	Built-up Area	Raja Rajeshwari Nagar	2 BHK	1141	2.0	1.0	60.00	Yes	2
13318	Super built-up Area	Padmanabhanagar	4 BHK	4689	4.0	1.0	488.00	No	4
13319	Super built-up Area	Doddathoguru	1 BHK	550	1.0	1.0	17.00	Yes	1

12710 rows × 9 columns

In [47]:
```
df[df.bhk>20]
```

Out[47]:

	area_type	location	size	total_sqft	bath	balcony	price	ready to move	bhk
1718	Super built-up Area	2Electronic City Phase II	27 BHK	8000	27.0	0.0	230.0	Yes	27
4684	Plot Area	Munnekollal	43 Bedroom	2400	40.0	0.0	660.0	Yes	43

In [48]:
```
df.drop(['size'], axis='columns', inplace=True)
```

In [49]:
```
df
```

Out[49]:

	area_type	location	total_sqft	bath	balcony	price	ready to move	bhk
0	Super built-up Area	Electronic City Phase II	1056	2.0	1.0	39.07	No	2
1	Plot Area	Chikka Tirupathi	2600	5.0	3.0	120.00	Yes	4
2	Built-up Area	Uttarahalli	1440	2.0	3.0	62.00	Yes	3
3	Super built-up Area	Lingadheeranahalli	1521	3.0	1.0	95.00	Yes	3
4	Super built-up Area	Kothanur	1200	2.0	1.0	51.00	Yes	2
...	...	...	...	...	...	...	...	...
13314	Super built-up Area	Green Glen Layout	1715	3.0	3.0	112.00	Yes	3
13315	Built-up Area	Whitefield	3453	4.0	0.0	231.00	Yes	5
13317	Built-up Area	Raja Rajeshwari Nagar	1141	2.0	1.0	60.00	Yes	2
13318	Super built-up Area	Padmanabhanagar	4689	4.0	1.0	488.00	No	4
13319	Super built-up Area	Doddathoguru	550	1.0	1.0	17.00	Yes	1

12710 rows × 8 columns

In [50]:
```
df.dtypes
```

Out[50]:
```
area_type          object
location           object
total_sqft         object
bath              float64
balcony           float64
price             float64
ready to move      object
bhk                 int64
dtype: object
```

In [51]:
```
df['total_sqft'].unique()
```

Out[51]:
```
array(['1056', '2600', '1440', ..., '1133 - 1384', '774', '4689'],
      dtype=object)
```

In [52]:
```
def is_float(x):
    try:
        float(x)
    except:
        return False
    return True
```

In [53]:
```
df[~df['total_sqft'].apply(is_float)].head(10)
```

Out[53]:

	area_type	location	total_sqft	bath	balcony	price	ready to move	bhk
30	Super built-up Area	Yelahanka	2100 - 2850	4.0	0.0	186.000	No	4
122	Super built-up Area	Hebbal	3067 - 8156	4.0	0.0	477.000	No	4
137	Super built-up Area	8th Phase JP Nagar	1042 - 1105	2.0	0.0	54.005	No	2
165	Super built-up Area	Sarjapur	1145 - 1340	2.0	0.0	43.490	No	2
188	Super built-up Area	KR Puram	1015 - 1540	2.0	0.0	56.800	Yes	2
410	Super built-up Area	Kengeri	34.46Sq. Meter	1.0	0.0	18.500	Yes	1
549	Super built-up Area	Hennur Road	1195 - 1440	2.0	0.0	63.770	No	2
661	Super built-up Area	Yelahanka	1120 - 1145	2.0	0.0	48.130	Yes	2
672	Built-up Area	Bettahalsoor	3090 - 5002	4.0	0.0	445.000	No	4
772	Super built-up Area	Banashankari Stage VI	1160 - 1195	2.0	0.0	59.935	No	2

In [54]:
```
def convertSQFTtoNum(x):
    tokens=x.split('-')
    if len(tokens)==2:
        return (float(tokens[0])+float(tokens[1]))/2
    try:
        return float(x)
    except:
        return None
```

In [55]:
```
#testing the function
convertSQFTtoNum('2100 - 2850')
```

Out[55]:
```
2475.0
```

In [56]:
```
df['total_sqft']= df['total_sqft'].apply(convertSQFTtoNum)
```

In [57]:
```
df.head()
```

Out[57]:

	area_type	location	total_sqft	bath	balcony	price	ready to move	bhk
0	Super built-up Area	Electronic City Phase II	1056.0	2.0	1.0	39.07	No	2
1	Plot Area	Chikka Tirupathi	2600.0	5.0	3.0	120.00	Yes	4
2	Built-up Area	Uttarahalli	1440.0	2.0	3.0	62.00	Yes	3
3	Super built-up Area	Lingadheeranahalli	1521.0	3.0	1.0	95.00	Yes	3
4	Super built-up Area	Kothanur	1200.0	2.0	1.0	51.00	Yes	2

In [58]:
```
df.loc[30]
```

Out[58]:
```
area_type        Super built-up  Area
location                    Yelahanka
total_sqft                     2475.0
bath                              4.0
balcony                           0.0
price                           186.0
ready to move                      No
bhk                                 4
Name: 30, dtype: object
```

In [59]:
```
#creating new columns
#created price per sqft
df['Price Per sqft']=df['price']*100000/df['total_sqft']
```

In [60]:
```
df.head()
```

Out[60]:

	area_type	location	total_sqft	bath	balcony	price	ready to move	bhk	Price Per sqft
0	Super built-up Area	Electronic City Phase II	1056.0	2.0	1.0	39.07	No	2	3699.810606
1	Plot Area	Chikka Tirupathi	2600.0	5.0	3.0	120.00	Yes	4	4615.384615
2	Built-up Area	Uttarahalli	1440.0	2.0	3.0	62.00	Yes	3	4305.555556
3	Super built-up Area	Lingadheeranahalli	1521.0	3.0	1.0	95.00	Yes	3	6245.890861
4	Super built-up Area	Kothanur	1200.0	2.0	1.0	51.00	Yes	2	4250.000000

In [61]:
```
#working on location
```

In [62]:
```
len(df.location.unique())
```

Out[62]:
```
1265
```

In [63]:
```
df.location.apply(lambda x: x.strip())
```

Out[63]:
```
0        Electronic City Phase II
1                Chikka Tirupathi
2                     Uttarahalli
3              Lingadheeranahalli
4                        Kothanur
                    ...
13314            Green Glen Layout
13315                   Whitefield
13317       Raja Rajeshwari Nagar
13318              Padmanabhanagar
13319                 Doddathoguru
Name: location, Length: 12710, dtype: object
```

In [64]:
```
location_counts = df['location'].value_counts()
```

In [65]:
```
location_counts
```

Out[65]:
```
Whitefield            514
Sarjapur  Road        372
Electronic City       300
Kanakpura Road        261
Thanisandra           231
                     ...
Milk Colony             1
Sundara Nagar           1
Jaladarsini Layout      1
Madanayakahalli         1
Abshot Layout           1
Name: location, Length: 1265, dtype: int64
```

In [66]:
```
len(location_counts[location_counts<=30])
```

Out[66]:
```
1174
```

In [67]:
```
LocationCount_lessthan10=location_counts[location_counts<=30]
```

In [68]:
```
LocationCount_lessthan10
```

Out[68]:
```
Ananth Nagar          30
Doddathoguru          30
R.T. Nagar            30
Chikkalasandra        30
Kudlu                 29
                      ..
Milk Colony            1
Sundara Nagar          1
Jaladarsini Layout     1
Madanayakahalli        1
Abshot Layout          1
Name: location, Length: 1174, dtype: int64
```

In [69]:
```
df.location=df.location.apply(lambda x: 'other' if x in LocationCount_lesstl
```

In [70]:
```
len(df.location.unique())
```

Out[70]:
```
92
```

In [71]:
```
df['total_sqft'].describe()
```

Out[71]:
```
count    12668.000000
mean      1511.835167
std       1162.097276
min          5.000000
25%       1100.000000
50%       1260.000000
75%       1640.000000
max      52272.000000
Name: total_sqft, dtype: float64
```

In [72]:
```
# inference- there are houses of same sqft with varied prices since we have
```

In [ ]:
```

```

In [73]:
```
least_expensive_house = df.sort_values('price').head(1)

print("Details of the Least Expensive House:")
print(least_expensive_house)
```

```
Details of the Least Expensive House:
                area_type            location  total_sqft  bath  balcony
\
10526  Super built-up  Area  Yelahanka New Town       284.0   1.0      1.0

       price ready to move  bhk  Price Per sqft
10526    8.0           Yes    1     2816.901408
```

In [74]:
```
df.shape
```

Out[74]:
```
(12710, 9)
```

In [75]:
```
most_expensive_house = df.sort_values('price', ascending=False).head(1)

print("Details of the Most Expensive House:")
print(most_expensive_house)
```

```
Details of the Most Expensive House:
                area_type location  total_sqft  bath  balcony   price  \
11080  Super built-up  Area    other      8321.0   5.0      2.0  2912.0

      ready to move  bhk  Price Per sqft
11080            No    4    34995.793775
```

In [76]:
```
df
```

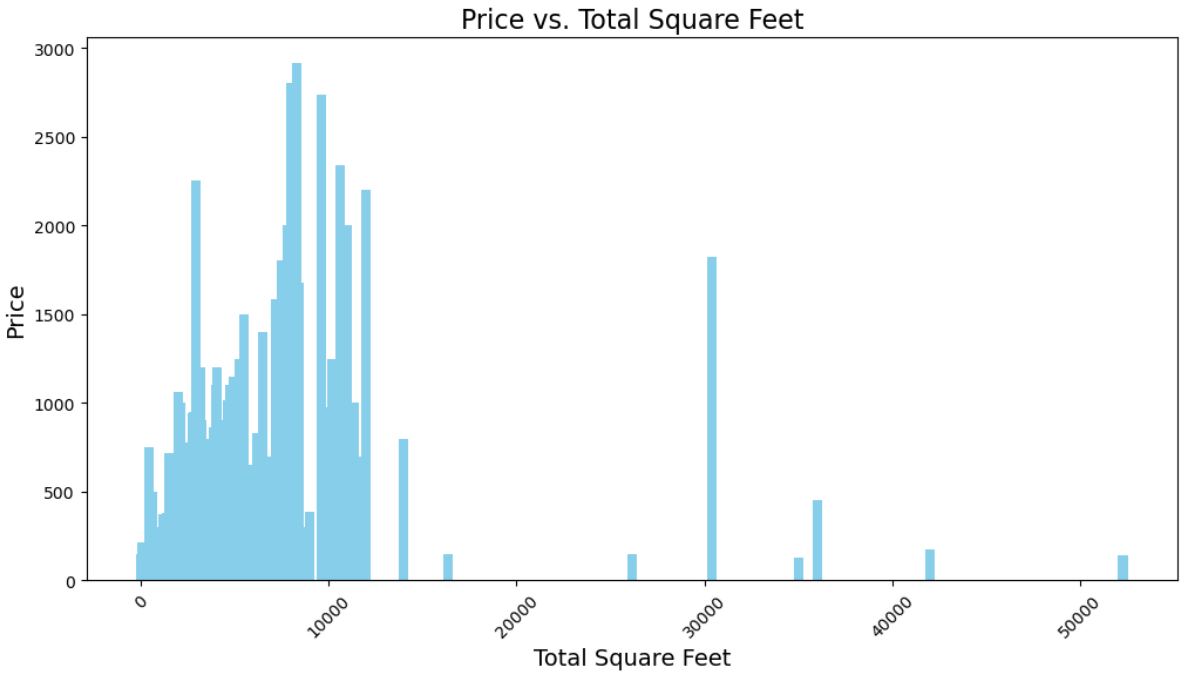
Out[76]:

	area_type	location	total_sqft	bath	balcony	price	ready to move	bhk	Price Per sqft
0	Super built-up Area	Electronic City Phase II	1056.0	2.0	1.0	39.07	No	2	3699.810606
1	Plot Area	other	2600.0	5.0	3.0	120.00	Yes	4	4615.384615
2	Built-up Area	Uttarahalli	1440.0	2.0	3.0	62.00	Yes	3	4305.555556
3	Super built-up Area	other	1521.0	3.0	1.0	95.00	Yes	3	6245.890861
4	Super built-up Area	Kothanur	1200.0	2.0	1.0	51.00	Yes	2	4250.000000
...	...	...	...	...	...	...	...	...	...
13314	Super built-up Area	Green Glen Layout	1715.0	3.0	3.0	112.00	Yes	3	6530.612245
13315	Built-up Area	Whitefield	3453.0	4.0	0.0	231.00	Yes	5	6689.834926
13317	Built-up Area	Raja Rajeshwari Nagar	1141.0	2.0	1.0	60.00	Yes	2	5258.545136
13318	Super built-up Area	other	4689.0	4.0	1.0	488.00	No	4	10407.336319
13319	Super built-up Area	other	550.0	1.0	1.0	17.00	Yes	1	3090.909091

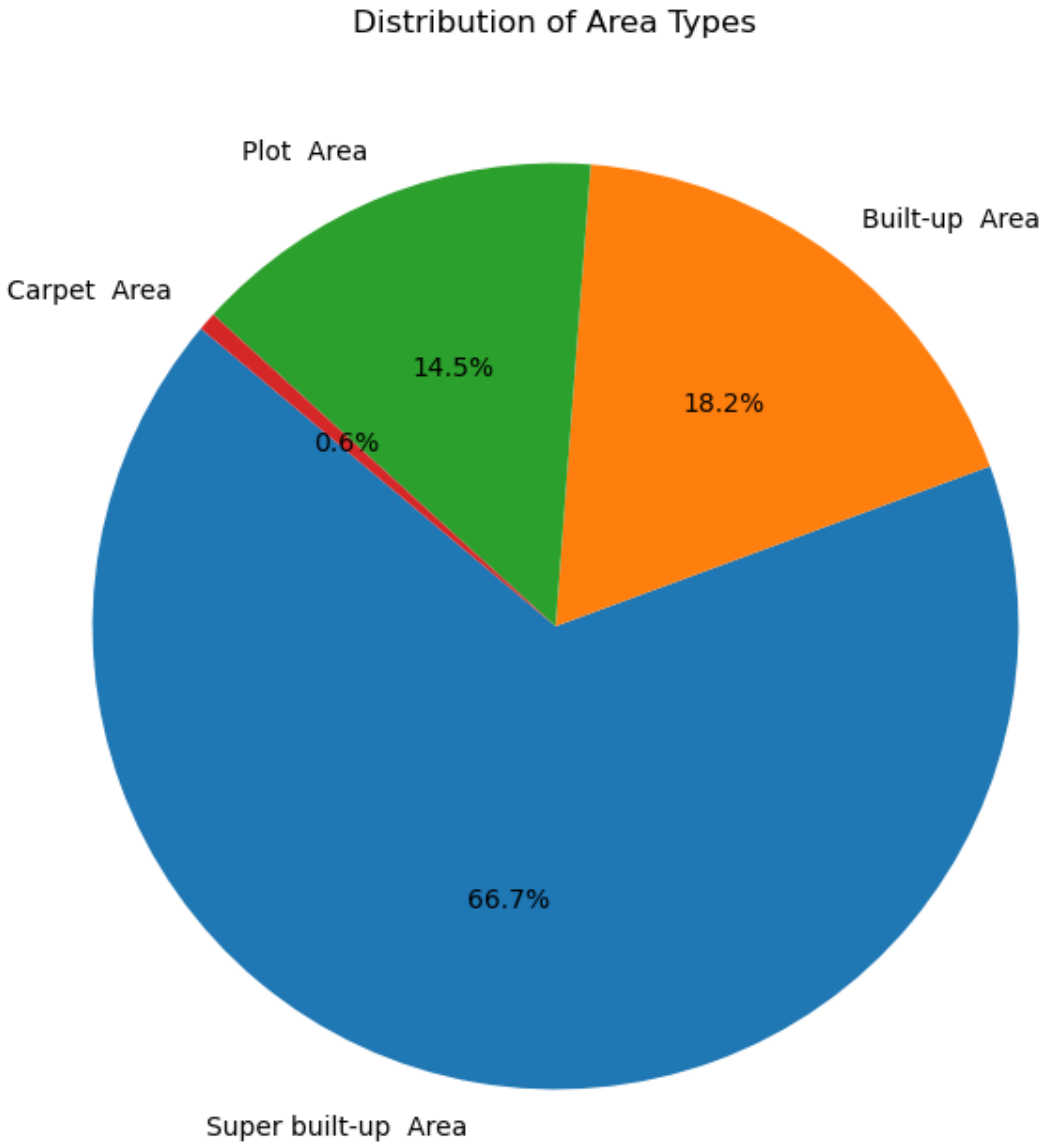
12710 rows × 9 columns

In [77]:

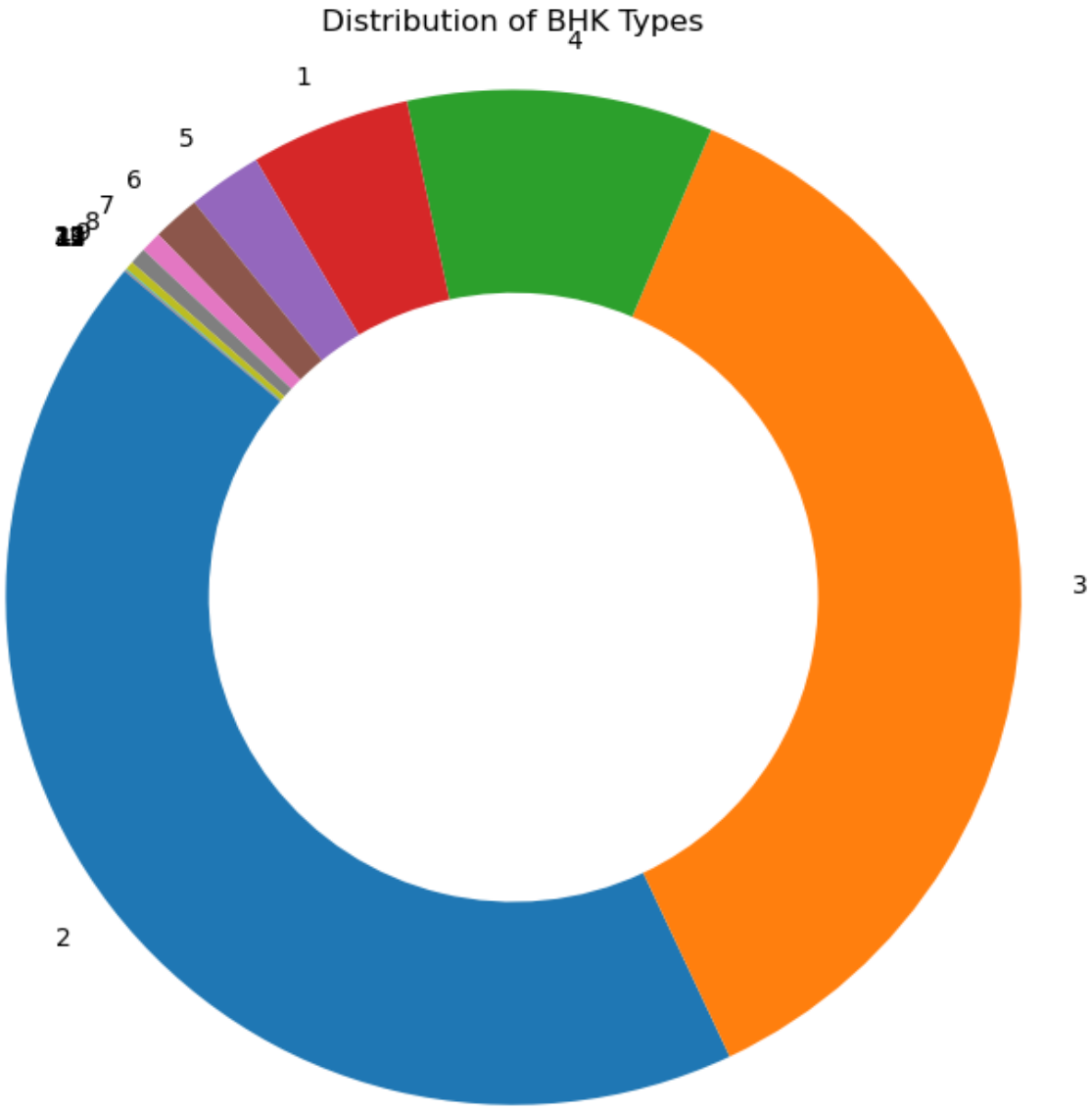
```
plt.figure(figsize=(12, 6))
bars = plt.bar(df['total_sqft'], df['price'], width=500, color='skyblue')
# Adding labels and title
plt.xlabel('Total Square Feet', fontsize=14)
plt.ylabel('Price', fontsize=14)
plt.title('Price vs. Total Square Feet', fontsize=16)
plt.xticks(rotation=45)
plt.show()
```



In [78]:
```
area_type_counts = df['area_type'].value_counts()
# Plotting a pie chart
plt.figure(figsize=(8, 8))
plt.pie(area_type_counts, labels=area_type_counts.index, autopct='%1.1f%%',
plt.title('Distribution of Area Types')
plt.show()
```

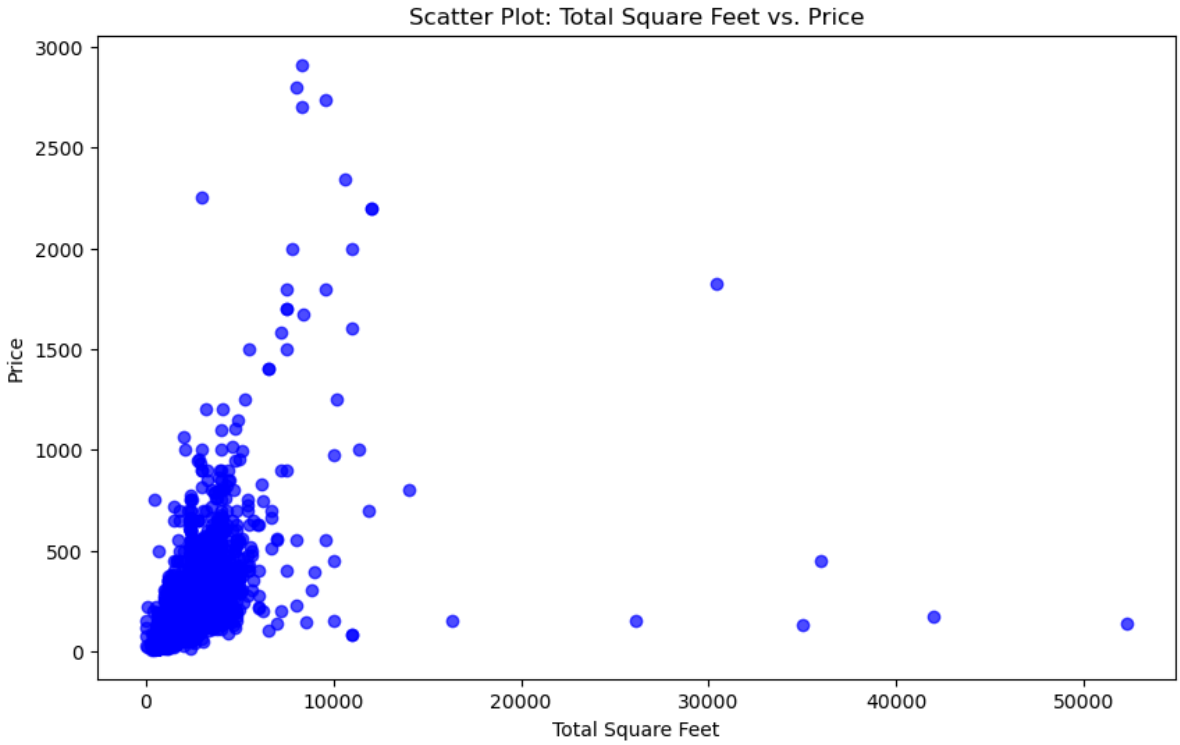


In [79]:
```
bhk_counts = df['bhk'].value_counts()
plt.figure(figsize=(8, 8))
plt.pie(bhk_counts, labels=bhk_counts.index, startangle=140, wedgeprops=dict
plt.axis('equal')
plt.title('Distribution of BHK Types')
plt.show()
```

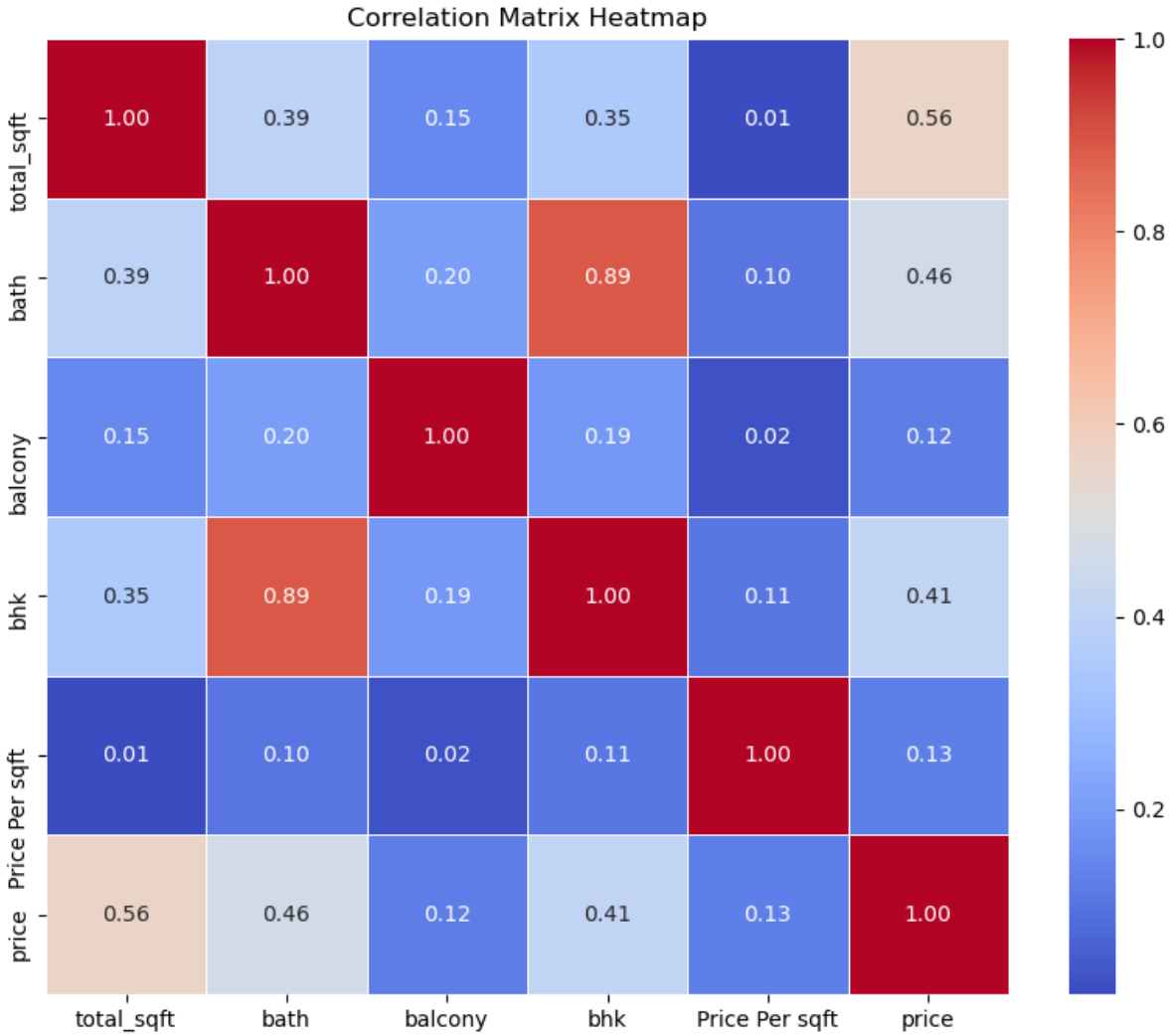


In [80]:
```
plt.figure(figsize=(10, 6))
plt.scatter(df['total_sqft'], df['price'], alpha=0.7, color='blue')
plt.title('Scatter Plot: Total Square Feet vs. Price')
plt.xlabel('Total Square Feet')
plt.ylabel('Price')

plt.show()
```



In [83]:
```
correlation_cols = ['total_sqft', 'bath', 'balcony', 'bhk', 'Price Per sqft
correlation_matrix = df[correlation_cols].corr()
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt='.2f', line
plt.title('Correlation Matrix Heatmap')
plt.show()
```



In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: