# The House EDA Project

## Introduction

### Objective

Is to view how the prices of the houses vary with the inputs provided in this dataset

Key Questions to be considered
1. What is the distribution of the house prices?
2. How different factors affect the price of a house, ie the furnishingstatus, area
3. Price against the conditions of the house. Are there any differences in prices based on their conditions.
4. How do nearby facilities like mainroads affect the price of the house.

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import warnings
warnings.filterwarnings('ignore')
df=pd.read_csv('housing.csv')
df.shape
```

Out[1]:
```
(545, 13)
```

In [2]:
```
df
```

Out[2]:

| | price | area | bedrooms | bathrooms | stories | mainroad | guestroom | basement | hotwaterheating | airconditioning | parking | prefare |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 13300000 | 7420 | 4 | 2 | 3 | yes | no | no | no | yes | 2 | ye |
| **1** | 12250000 | 8960 | 4 | 4 | 4 | yes | no | no | no | yes | 3 | n |
| **2** | 12250000 | 9960 | 3 | 2 | 2 | yes | no | yes | no | no | 2 | ye |
| **3** | 12215000 | 7500 | 4 | 2 | 2 | yes | no | yes | no | yes | 3 | ye |
| **4** | 11410000 | 7420 | 4 | 1 | 2 | yes | yes | yes | no | yes | 2 | n |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | |
| **540** | 1820000 | 3000 | 2 | 1 | 1 | yes | no | yes | no | no | 2 | n |
| **541** | 1767150 | 2400 | 3 | 1 | 1 | no | no | no | no | no | 0 | n |
| **542** | 1750000 | 3620 | 2 | 1 | 1 | yes | no | no | no | no | 0 | n |
| **543** | 1750000 | 2910 | 3 | 1 | 1 | no | no | no | no | no | 0 | n |
| **544** | 1750000 | 3850 | 3 | 1 | 2 | yes | no | no | no | no | 0 | n |

545 rows × 13 columns

In [3]:
```
df.columns
```

Out[3]:
```
Index(['price', 'area', 'bedrooms', 'bathrooms', 'stories', 'mainroad',
       'guestroom', 'basement', 'hotwaterheating', 'airconditioning',
       'parking', 'prefarea', 'furnishingstatus'],
      dtype='object')
```

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 545 entries, 0 to 544
Data columns (total 13 columns):
 #   Column            Non-Null Count  Dtype 
---  ------            --------------  ----- 
 0   price             545 non-null    int64 
 1   area              545 non-null    int64 
 2   bedrooms          545 non-null    int64 
 3   bathrooms         545 non-null    int64 
 4   stories           545 non-null    int64 
 5   mainroad          545 non-null    object
 6   guestroom         545 non-null    object
 7   basement          545 non-null    object
 8   hotwaterheating   545 non-null    object
 9   airconditioning   545 non-null    object
 10  parking           545 non-null    int64 
 11  prefarea          545 non-null    object
 12  furnishingstatus  545 non-null    object
dtypes: int64(6), object(7)
memory usage: 55.5+ KB
```

In [5]:
```
df.dtypes
```

Out[5]:
```
price              int64
area               int64
bedrooms           int64
bathrooms          int64
stories            int64
mainroad          object
guestroom         object
basement          object
hotwaterheating   object
airconditioning   object
parking            int64
prefarea          object
furnishingstatus  object
dtype: object
```

In [6]:
```
df.nunique()
```

Out[6]:
```
price              219
area               284
bedrooms             6
bathrooms            4
stories              4
mainroad             2
guestroom            2
basement             2
hotwaterheating      2
airconditioning      2
parking              4
prefarea             2
furnishingstatus     3
dtype: int64
```

In [7]:
```
#checking if there an null values
df.isnull().sum()
```

Out[7]:
```
price              0
area               0
bedrooms           0
bathrooms          0
stories            0
mainroad           0
guestroom          0
basement           0
hotwaterheating    0
airconditioning    0
parking            0
prefarea           0
furnishingstatus   0
dtype: int64
```

In [8]:
```
#checking the statistics summary of numeric values
df.describe().T
```

Out[8]:

| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| **price** | 545.0 | 4.766729e+06 | 1.870440e+06 | 1750000.0 | 3430000.0 | 4340000.0 | 5740000.0 | 13300000.0 |
| **area** | 545.0 | 5.150541e+03 | 2.170141e+03 | 1650.0 | 3600.0 | 4600.0 | 6360.0 | 16200.0 |
| **bedrooms** | 545.0 | 2.965138e+00 | 7.380639e-01 | 1.0 | 2.0 | 3.0 | 3.0 | 6.0 |
| **bathrooms** | 545.0 | 1.286239e+00 | 5.024696e-01 | 1.0 | 1.0 | 1.0 | 2.0 | 4.0 |
| **stories** | 545.0 | 1.805505e+00 | 8.674925e-01 | 1.0 | 1.0 | 2.0 | 2.0 | 4.0 |
| **parking** | 545.0 | 6.935780e-01 | 8.615858e-01 | 0.0 | 0.0 | 0.0 | 1.0 | 3.0 |

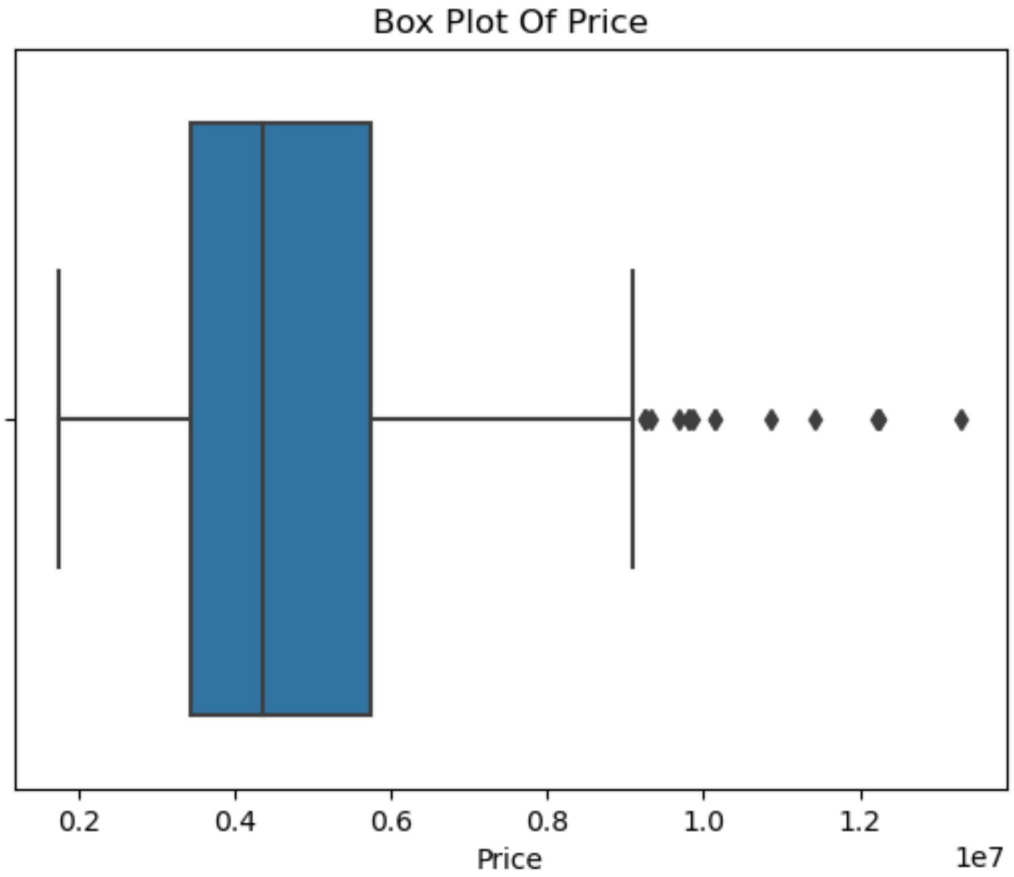
In [9]:
```
df.describe(include='object').T
```

Out[9]:

| | count | unique | top | freq |
|---|---|---|---|---|
| **mainroad** | 545 | 2 | yes | 468 |
| **guestroom** | 545 | 2 | no | 448 |
| **basement** | 545 | 2 | no | 354 |
| **hotwaterheating** | 545 | 2 | no | 520 |
| **airconditioning** | 545 | 2 | no | 373 |
| **prefarea** | 545 | 2 | no | 417 |
| **furnishingstatus** | 545 | 3 | semi-furnished | 227 |

# EXPLANATORY DATA ANALYSIS

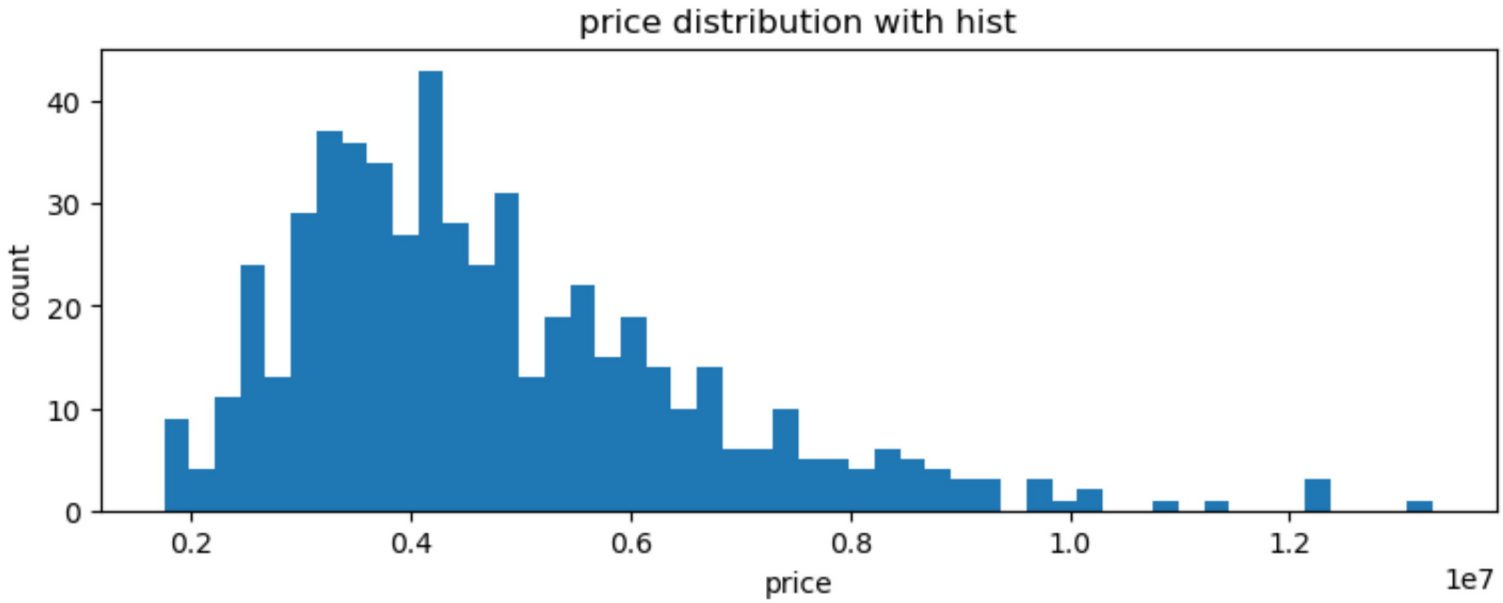
In [10]:
```
#how the price attribute is distributed
sns.boxplot(x='price', data=df)
plt.xlabel('Price')
plt.title('Box Plot Of Price')
plt.show()
```

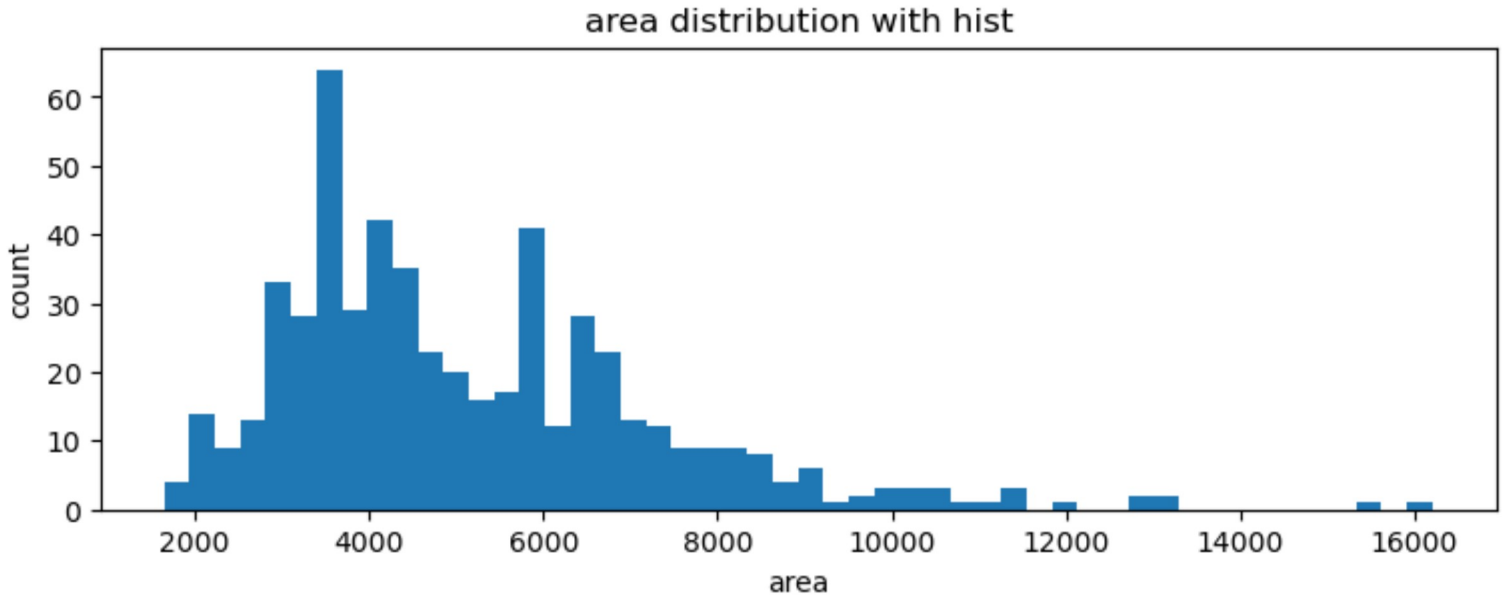



In [11]:
```
def plot_hist(variable):
    plt.figure(figsize=(9,3))
    plt.hist(df[variable], bins=50)
    plt.xlabel(variable)
    plt.ylabel('count')
    plt.title('{} distribution with hist'.format(variable))
    plt.show



numeric=['price', 'area']
for  n in numeric:
    plot_hist(n)
```




In [12]:
```
df.columns
```

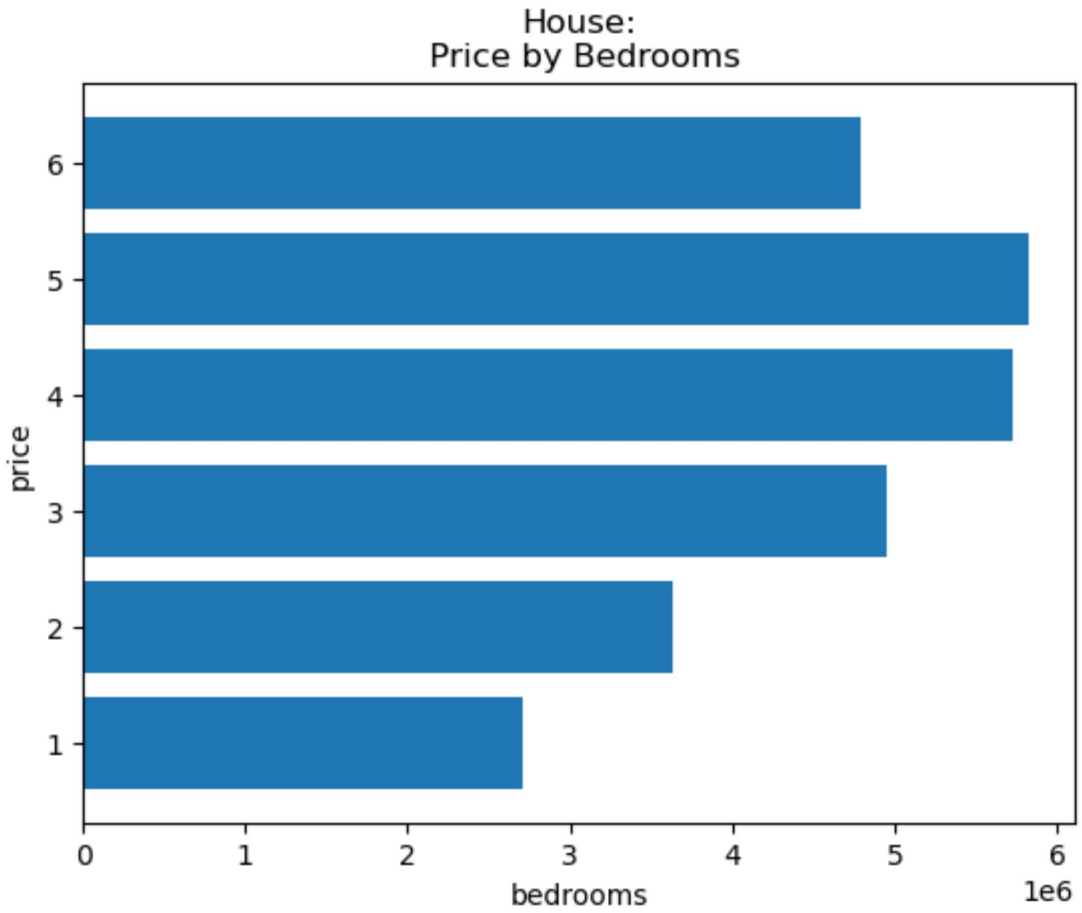
Out[12]:
```
Index(['price', 'area', 'bedrooms', 'bathrooms', 'stories', 'mainroad',
       'guestroom', 'basement', 'hotwaterheating', 'airconditioning',
       'parking', 'prefarea', 'furnishingstatus'],
      dtype='object')
```

In [13]:
```
#plotting the correlation matrix
plt.figure(figsize=(4,3))
corr=df.corr()
sns.heatmap(corr, cmap='summer_r', annot=True)
```

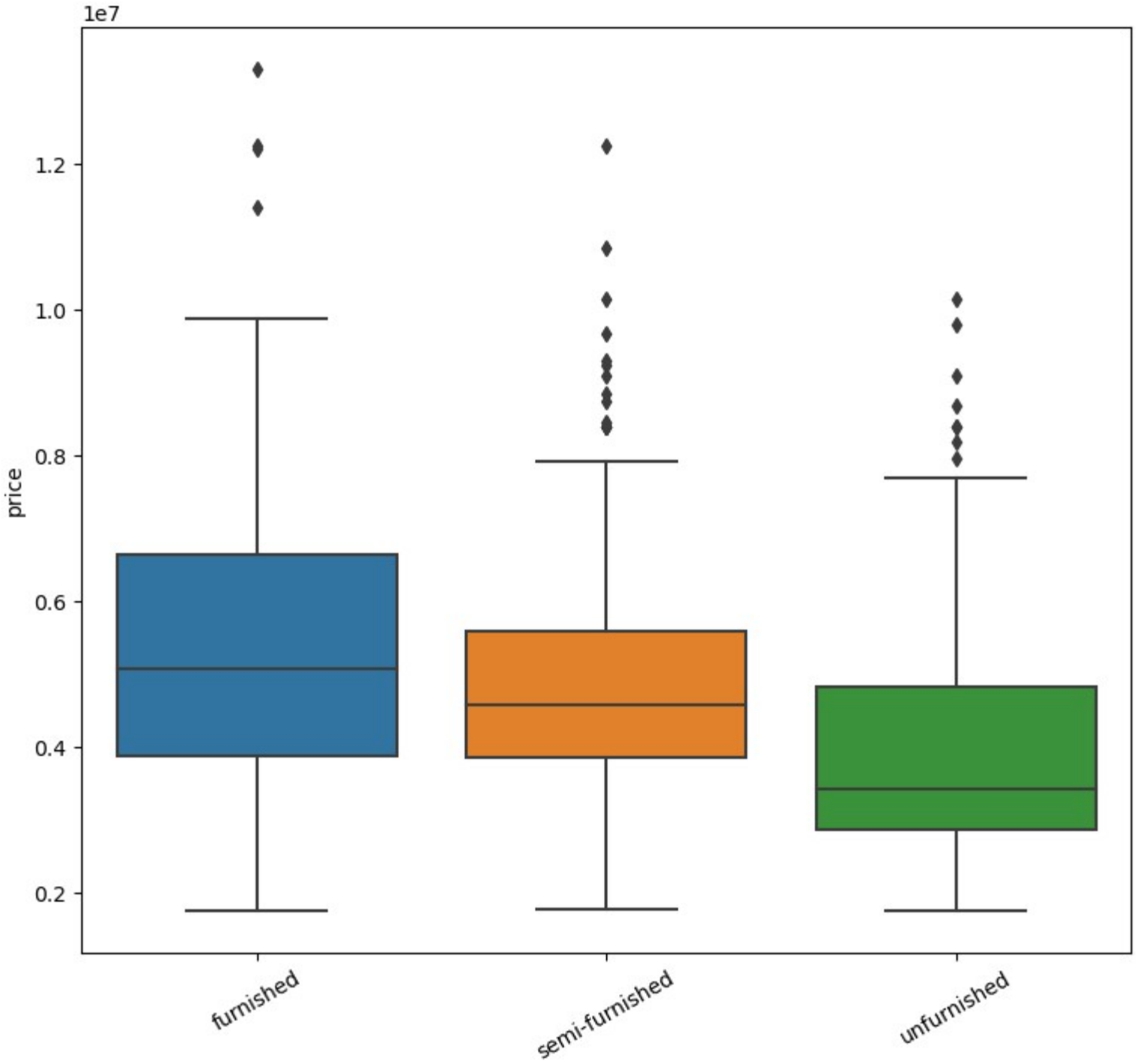
Out[13]:
```
<AxesSubplot:>
```


In [14]:
```
#price by number of bedrooms
price_by_bedroom=df.price.groupby(df.bedrooms).mean().sort_values(ascending=True)
fig1, ax=plt.subplots()
ax.barh(price_by_bedroom.index, price_by_bedroom, )
ax.set_title(f"House:\n Price by Bedrooms")
ax.set_xlabel('bedrooms')
ax.set_ylabel('price')
```

Out[14]:
```
Text(0, 0.5, 'price')
```




In [15]:
```
#finding how the prices are varying based on funishing status
plt.figure(figsize=(9,8))
ax=sns.boxplot(x='furnishingstatus', y='price', data=df)
ax.set_xticklabels(ax.get_xticklabels(), fontsize=10, rotation=30)
plt.show()
```


1e7
1.2
1.0
0.8
0.6
0.4
0.2
price
furnished
semi-furnished
unfurnished

furnishingstatus

In [ ]:

In [16]:
```
df.columns
```

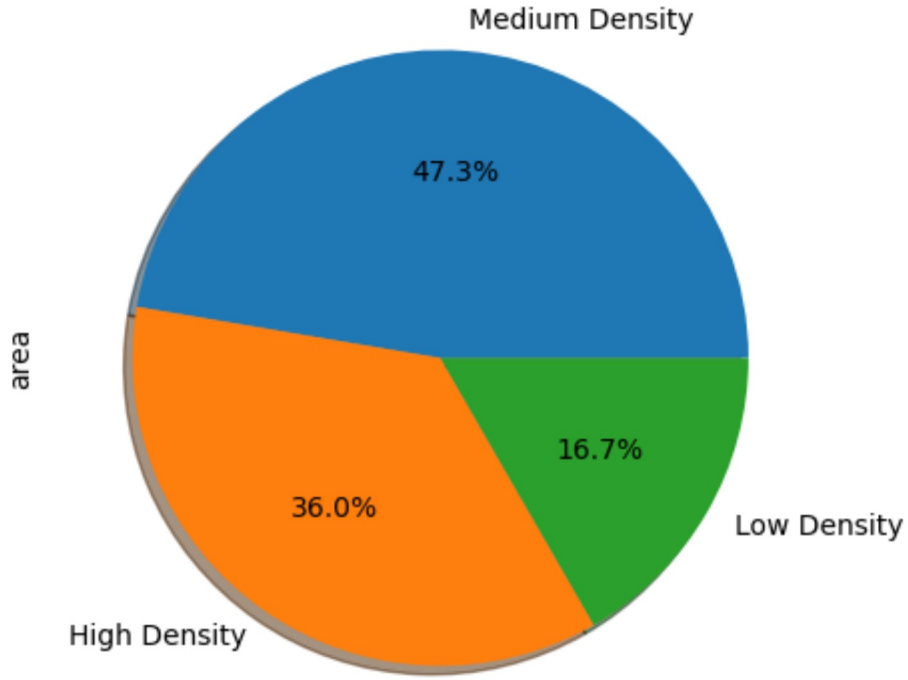
Out[16]:
```
Index(['price', 'area', 'bedrooms', 'bathrooms', 'stories', 'mainroad',
       'guestroom', 'basement', 'hotwaterheating', 'airconditioning',
       'parking', 'prefarea', 'furnishingstatus'],
      dtype='object')
```

In [17]:
```
df.area.value_counts()
```

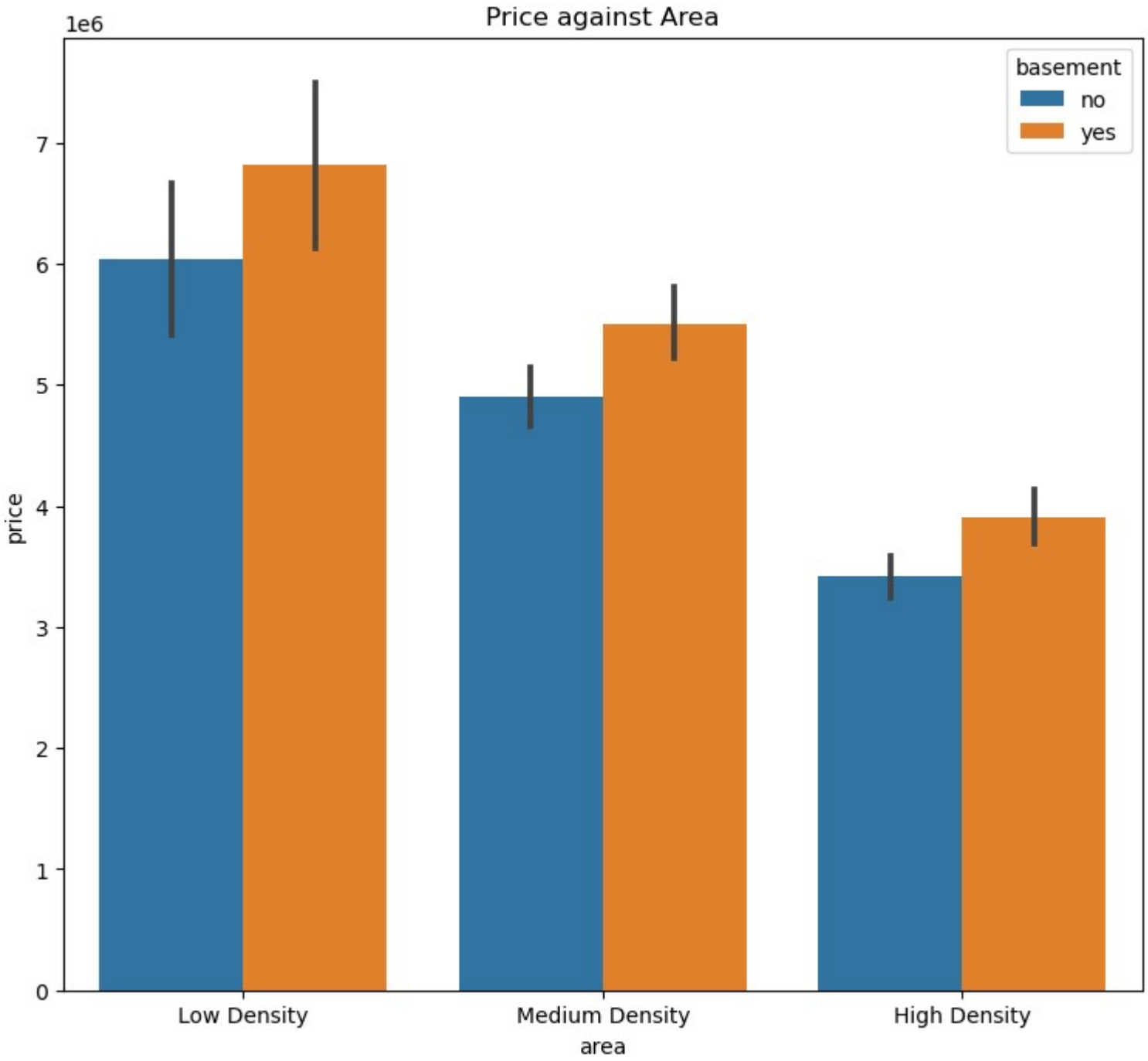
Out[17]:
```
6000    24
3000    14
4500    13
4000    11
5500     9
        ..
6862     1
4815     1
9166     1
6321     1
3620     1
Name: area, Length: 284, dtype: int64
```

In [18]:
```
#feature engineeirng on the area colum,
#changing it from int dtype to object dtype
df.area=df.area.replace(to_replace=range(1650,4000), value='High Density')
df.area=df.area.replace(to_replace=range(4000, 7000), value='Medium Density')
df.area=df.area.replace(to_replace=range(7000,16500), value='Low Density')

#plot a pie chart to view hoe this new area colmn in to be distributed
plt.figure(figsize=(6,5))
df['area'].value_counts().plot(kind='pie', autopct='%1.1f%%', shadow=True)
plt.show()
```

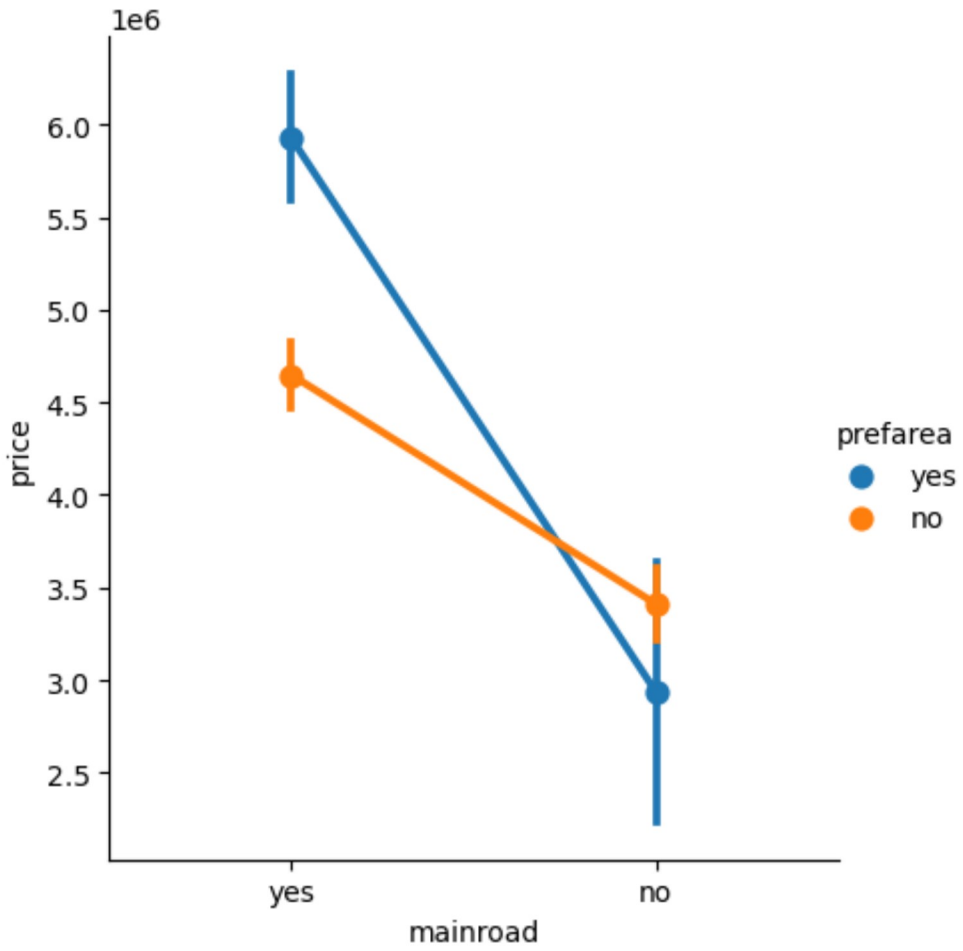



In [19]:
```
#how the price value ids varying with the area of the house
plt.figure(figsize=(9,8))
sns.barplot(x=df['area'], y=df['price'], hue=df['basement'])
plt.title('Price against Area')
plt.show()
```


Price against Area
1e6
basement
no
yes
price
0
1
2
3
4
5
6
7
Low Density
Medium Density
High Density
area

In [20]:
```
#how the price is distributed in accordance to the availability of guestrooms
fig=px.bar(x=df['guestroom'], y=df['price'])
fig.update_layout(
    title_text='Price Distribution against the Guestroom',
    xaxis_title='Guestroom',
    yaxis_title='Price')
fig.show()
```

In [21]:
```
#price versus mainroad and how the price vary with the house having the aircondition
sns.factorplot('mainroad','price', hue='prefarea', data=df)
fig=plt.gcf()
fig.set_size_inches(5,5)
plt.show()
```



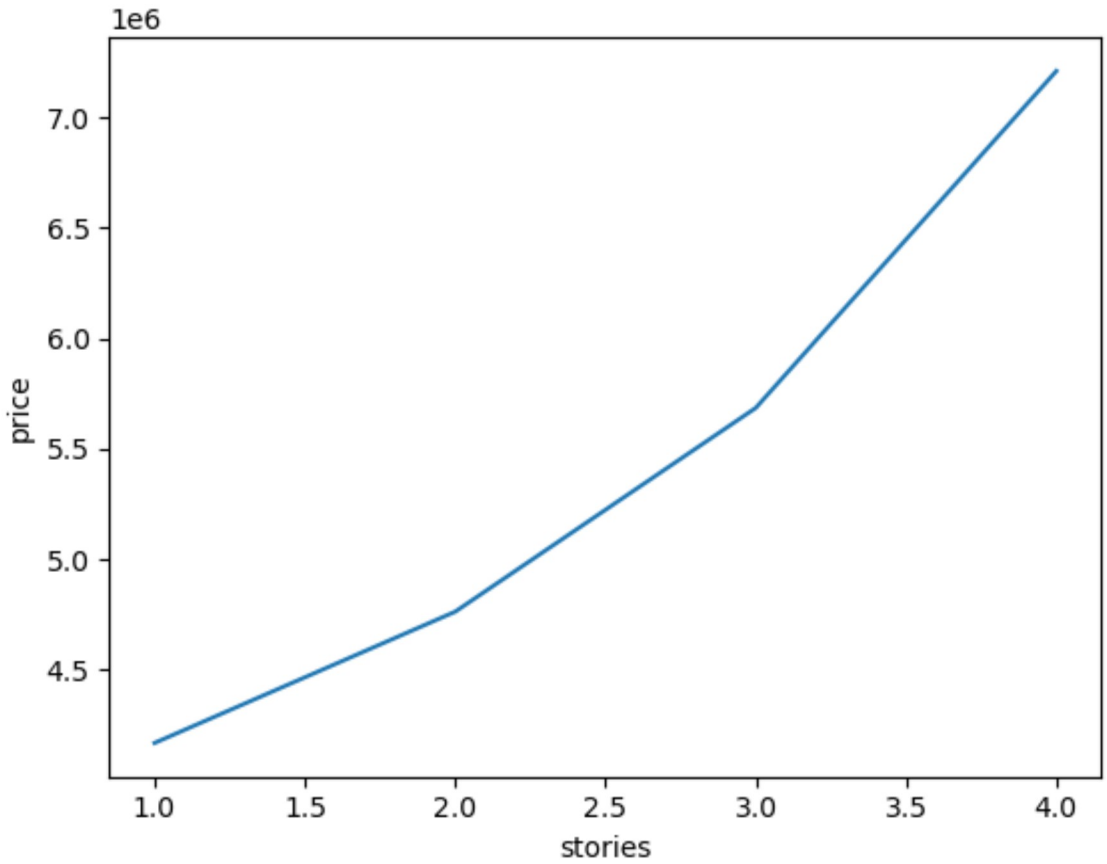

In [22]:
```
df.columns
```

Out[22]:
```
Index(['price', 'area', 'bedrooms', 'bathrooms', 'stories', 'mainroad',
       'guestroom', 'basement', 'hotwaterheating', 'airconditioning',
       'parking', 'prefarea', 'furnishingstatus'],
      dtype='object')
```

In [23]:
```
df.stories.value_counts()
```

Out[23]:
```
2    238
1    227
4     41
3     39
Name: stories, dtype: int64
```

In [24]:
```
#A time seies analysis
#mean price with stories
mean_price_stories=df.groupby(['stories'])['price'].mean().reset_index()
sns.lineplot(data=mean_price_stories, x='stories', y='price')
plt.show()
```



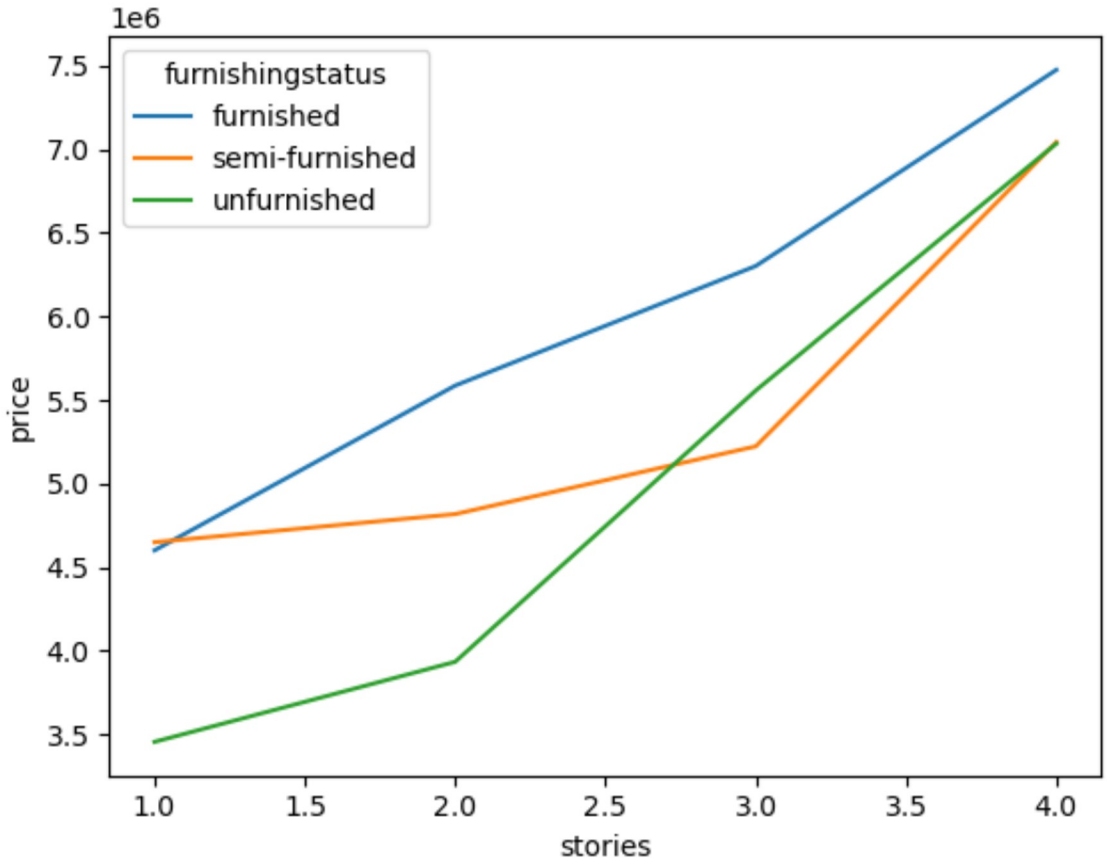
In [25]:
```
furnishing=pd.crosstab(df.stories, df.furnishingstatus).style.background_gradient(cmap='summer_r')
```

In [26]:
```
furnishing
```

Out[26]:

| stories | furnished | semi-furnished | unfurnished |
|---|---|---|---|
| **1** | 53 | 85 | 89 |
| **2** | 58 | 115 | 65 |
| **3** | 13 | 14 | 12 |
| **4** | 16 | 13 | 12 |

In [27]:
```
#with above code lets find out how the prices vary with the condition of furniture wih stories
furnishing=['furnished', 'semi-furnished', 'unfurnished']
df_furn=df.loc[df.furnishingstatus.isin(furnishing)]
mean_price_furnishingstatus=df.groupby(['stories', 'furnishingstatus'])['price'].mean().reset_index()
sns.lineplot(data=mean_price_furnishingstatus, x='stories', y='price', hue='furnishingstatus')
plt.show()
```




# Observations

1. Price Distribution
The prices of these houses follows a normal distribution with a right skewness.
There are some ootliers in this column in which prices were extremely high

2. Price with the influence of other factors
(i) price versus area: the higher the area the higher the price goes. Low dnsity surburbs are much expensive
(ii)price versus guestroom: houses with no guestroom were more expensive than the ones with the guestrooms.
(iii) price in multianalysis with mainroad, airconditiong, basemen: with all these, prices tend to go higher tho the houses that are near to the mainroad and also to yes having a basement and airconditioning. So fance houses are expensive.
(iv) Houses with many stories were alaso expensive as shown by the line ghaph.
(v) Prices vs Furnishing status: Furnished houses are exensive, seconded by semi-furnished. Last is unfirnished, they are less expensive

In [ ]: