

```

import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns

def countplot(dataframe, x_val, plot_title='', figsize=(15,8)):
    plt.figure(figsize=figsize)
    plt.title(plot_title)
    sns.countplot(data=df, x=x_val)
    plt.show()

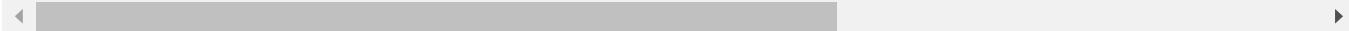
def barplot(dataframe, x_val, y_val):
    sns.barplot(data=dataframe, x=x_val, y=y_val)
    plt.title(x_val.title() + ' vs ' + y_val.title())
    plt.show()

df = pd.read_csv('House Price India.csv')
df.head()

```

	id	Date	number of bedrooms	number of bathrooms	living area	lot area	number of floors	waterfront present	number of views	condition of the house	...	Built Year	Renovation Year	Postal Code	Latitude
0	6762810145	42491	5	2.50	3650	9050	2.0	0	4	5	...	1921	0	122003	5
1	6762810635	42491	4	2.50	2920	4000	1.5	0	0	5	...	1909	0	122004	5
2	6762810998	42491	5	2.75	2910	9480	1.5	0	0	3	...	1939	0	122004	5
3	6762812605	42491	4	2.50	3310	42998	2.0	0	0	3	...	2001	0	122005	5
4	6762812919	42491	3	2.00	2710	4500	1.5	0	0	4	...	1929	0	122006	5

5 rows × 23 columns



```
pd.DataFrame(list(df.columns), columns=['Column Name'])
```

	Column Name
0	id
1	Date
2	number of bedrooms
3	number of bathrooms
4	living area
5	lot area
6	number of floors
7	waterfront present
8	number of views
9	condition of the house
10	grade of the house
11	Area of the house(excluding basement)
12	Area of the basement
13	Built Year
14	Renovation Year
15	Postal Code
16	Latitude
17	Longitude
18	living_area_renov
19	lot_area_renov
20	Number of schools nearby
21	Distance from the airport
22	Price

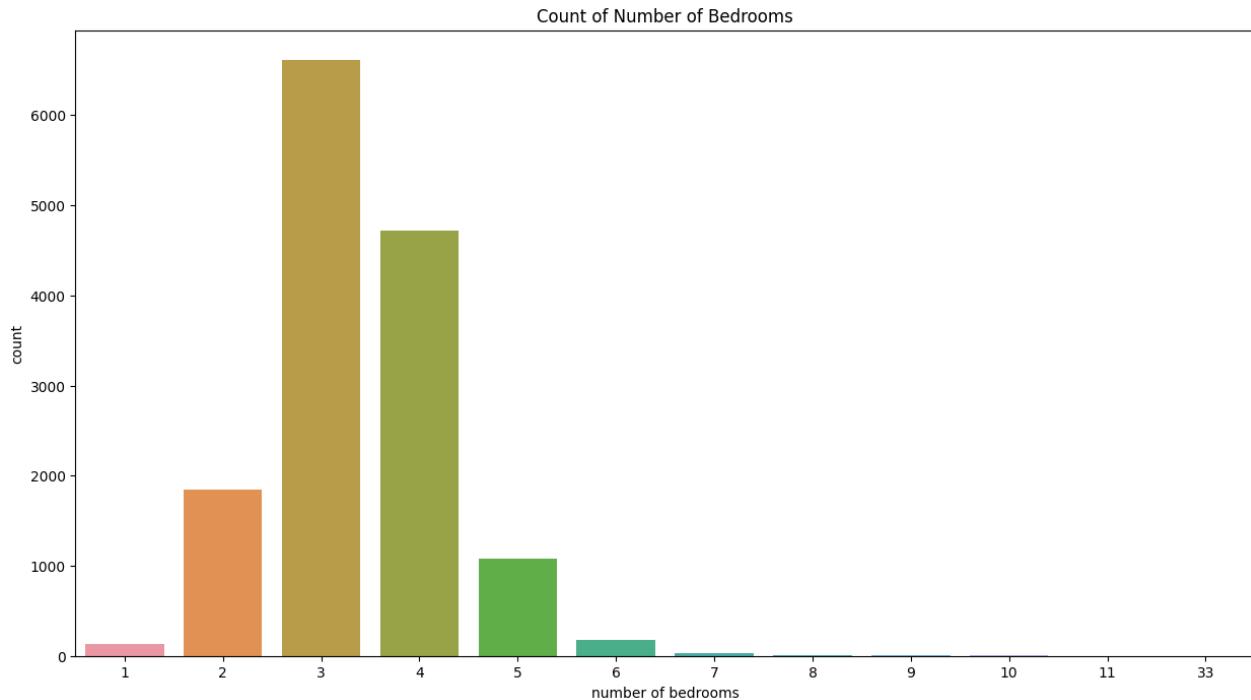
```
df.shape
```

```
(14620, 23)
```

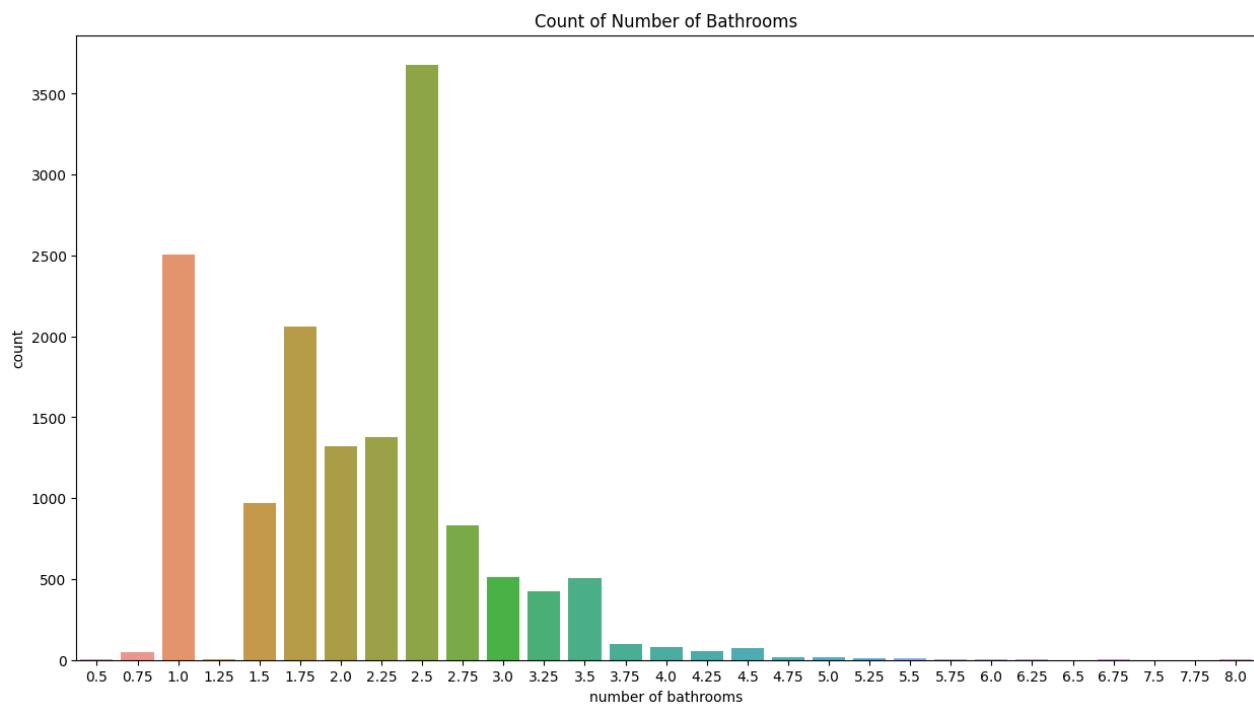
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14620 entries, 0 to 14619
Data columns (total 23 columns):
 #   Column           Non-Null Count  Dtype  
 ---  -- 
 0   id               14620 non-null   int64  
 1   Date              14620 non-null   int64  
 2   number of bedrooms 14620 non-null   int64  
 3   number of bathrooms 14620 non-null   float64 
 4   living area        14620 non-null   int64  
 5   lot area           14620 non-null   int64  
 6   number of floors   14620 non-null   float64 
 7   waterfront present 14620 non-null   int64  
 8   number of views    14620 non-null   int64  
 9   condition of the house 14620 non-null   int64  
 10  grade of the house 14620 non-null   int64  
 11  Area of the house(excluding basement) 14620 non-null   int64  
 12  Area of the basement 14620 non-null   int64  
 13  Built Year         14620 non-null   int64  
 14  Renovation Year    14620 non-null   int64  
 15  Postal Code        14620 non-null   int64  
 16  Latitude            14620 non-null   float64 
 17  Longitude           14620 non-null   float64 
 18  living_area_renov  14620 non-null   int64  
 19  lot_area_renov     14620 non-null   int64  
 20  Number of schools nearby 14620 non-null   int64  
 21  Distance from the airport 14620 non-null   int64  
 22  Price               14620 non-null   int64  
dtypes: float64(4), int64(19)
memory usage: 2.6 MB
```

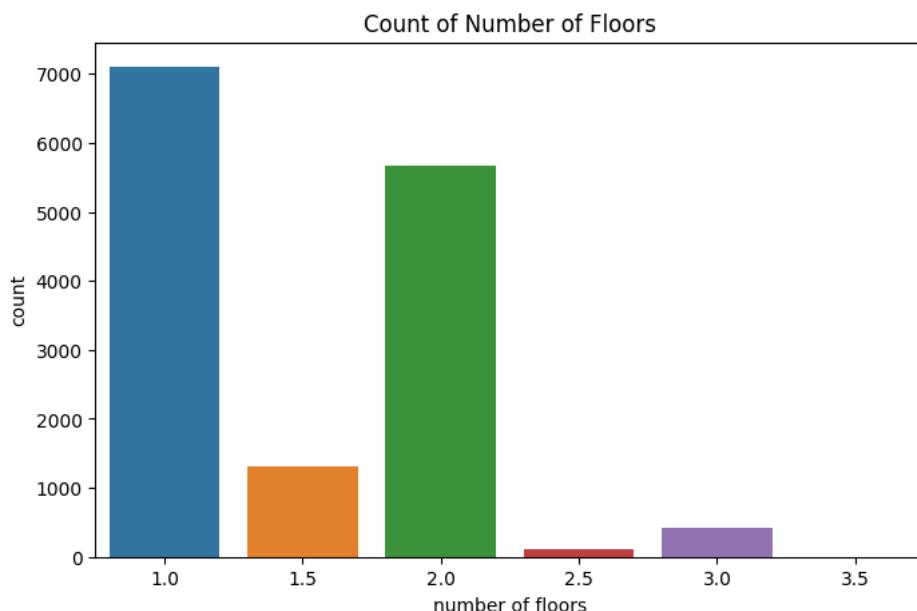
```
countplot(dataframe=df, x_val='number of bedrooms', plot_title='Count of Number of Bedrooms')
```



```
countplot(dataframe=df, x_val='number of bathrooms', plot_title='Count of Number of Bathrooms')
```



```
countplot(dataframe=df, x_val='number of floors', figsize=(8,5), plot_title='Count of Number of Floors')
```



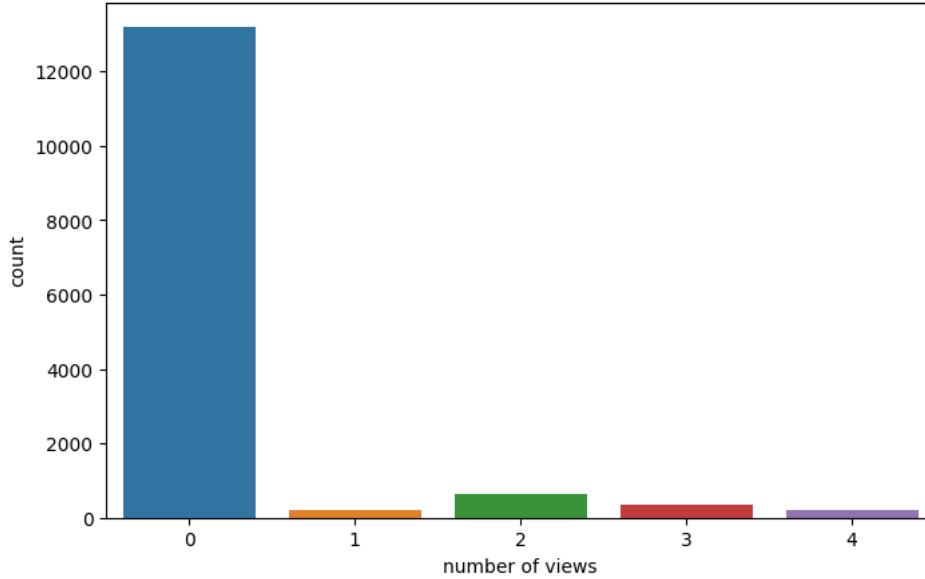
```
countplot(dataframe=df, x_val='waterfront present', figsize=(8,6), plot_title='Water present (1) or not (0)')
```

Water present (1) or not (0)



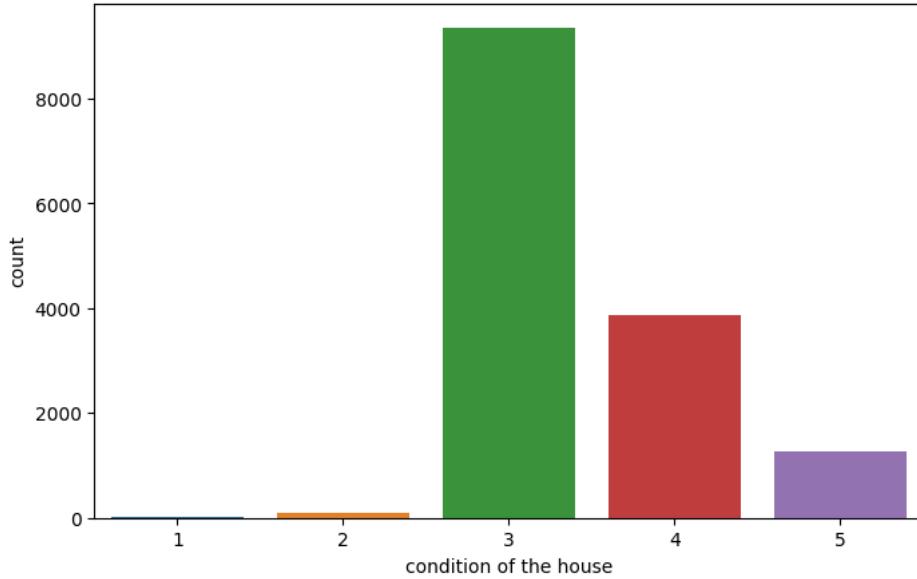
```
countplot(dataframe=df, x_val='number of views', figsize=(8,5), plot_title='Number of Views')
```

Number of Views



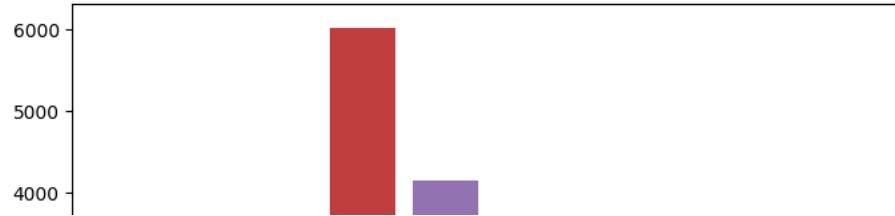
```
countplot(dataframe=df, x_val='condition of the house', figsize=(8,5), plot_title='House Condition')
```

House Condition



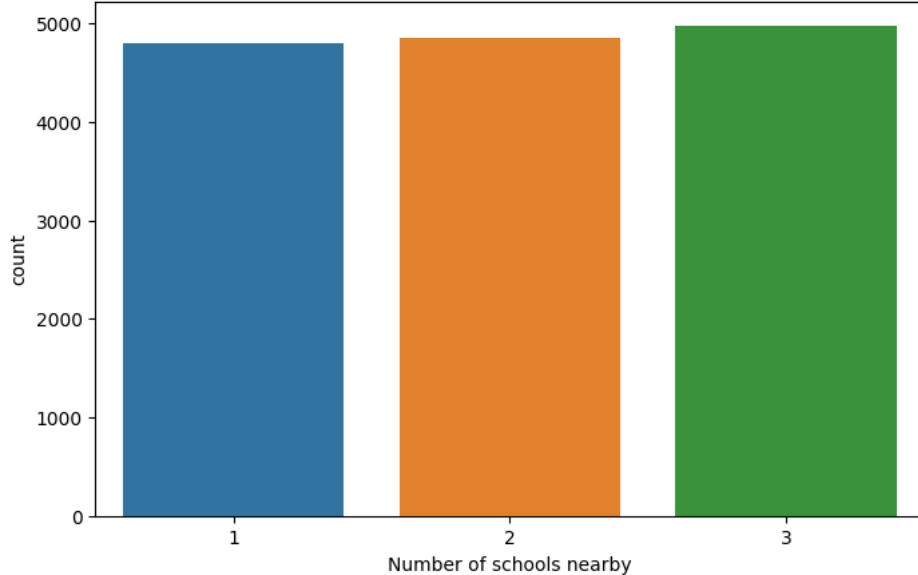
```
countplot(dataframe=df, x_val='grade of the house', figsize=(8,5), plot_title='Grade of the House')
```

Grade of the House

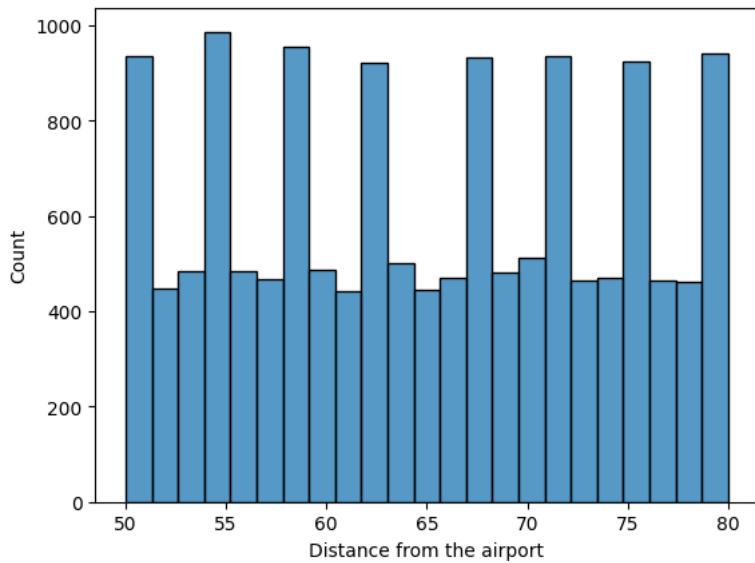


```
countplot(dataframe=df, x_val='Number of schools nearby', figsize=(8,5), plot_title='Number of Schools Nearby')
```

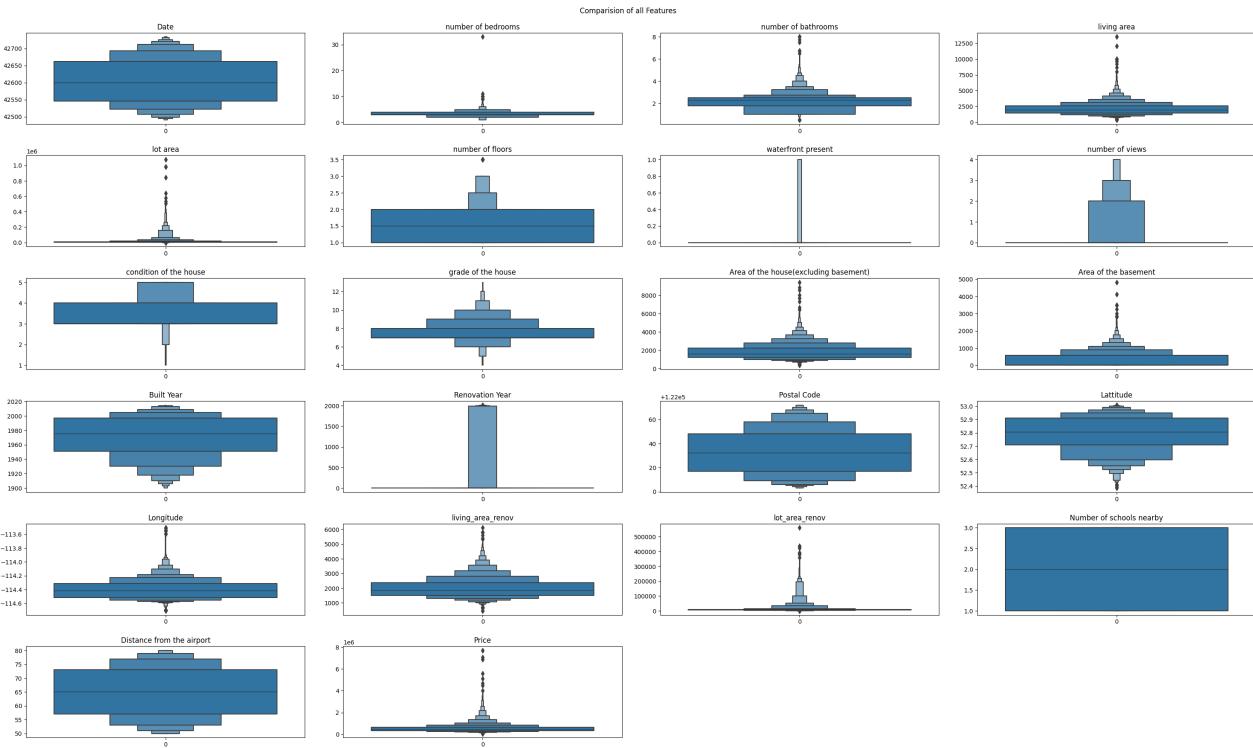
Number of Schools Nearby



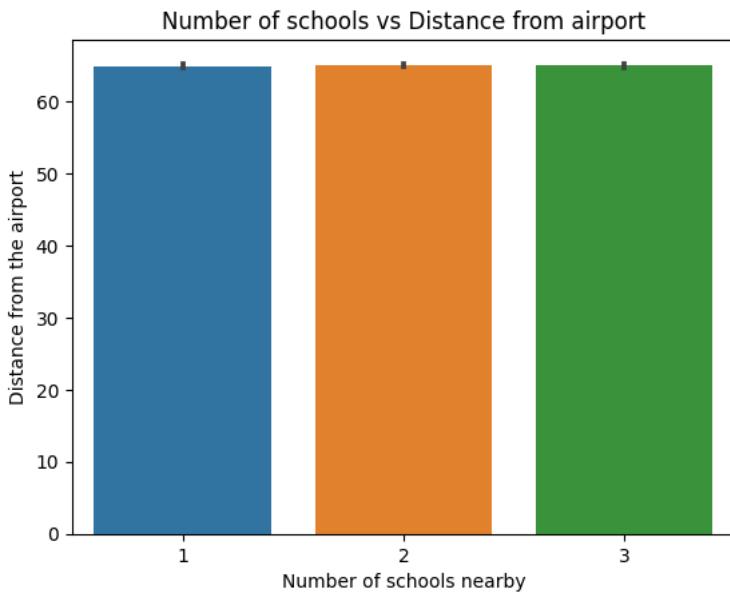
```
sns.histplot(data=df, x='Distance from the airport')
plt.show()
```



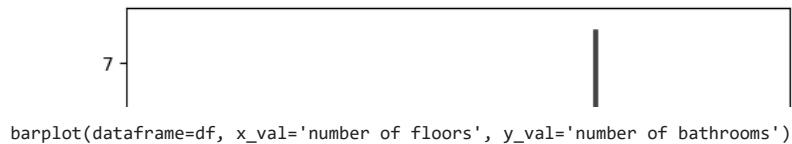
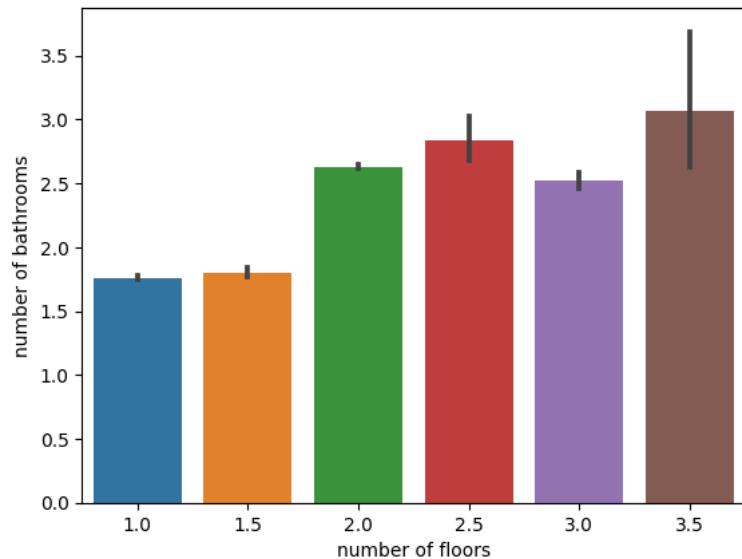
```
fig = plt.figure(figsize = [30, 18])
for i in range(1, len(list(df.columns))):  
    plt.subplot(6, 4, i)  
    fig.tight_layout(pad=2)  
    df_new = df.iloc[:,i]  
    plt.suptitle('Comparision of all Features')  
    plt.title(list(df.columns)[i])  
    sns.boxenplot(df_new)  
plt.show()
```



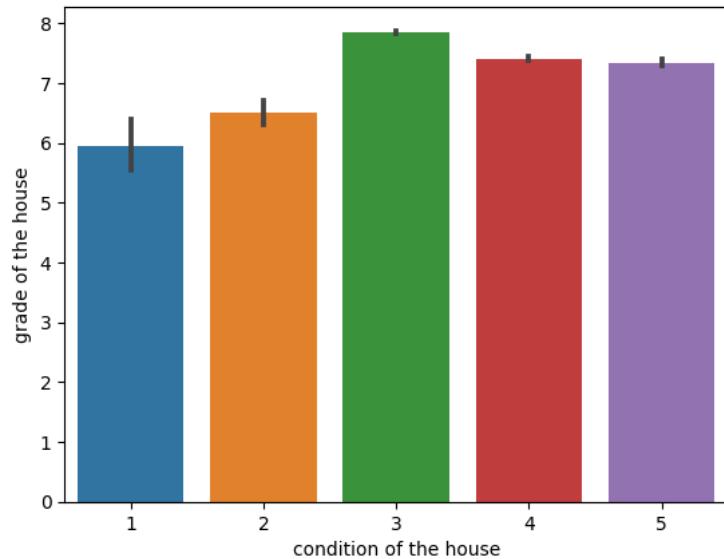
```
sns.barplot(data=df, x='Number of schools nearby', y='Distance from the airport')
plt.title('Number of schools vs Distance from airport')
plt.show()
```



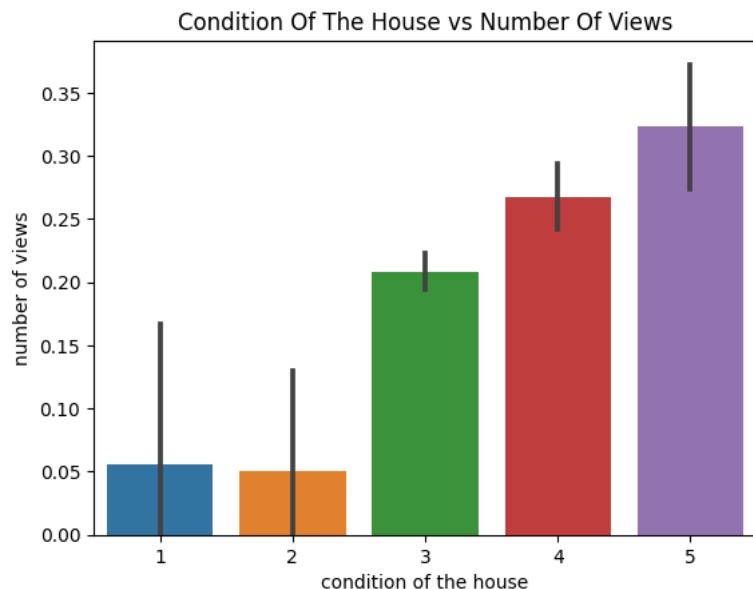
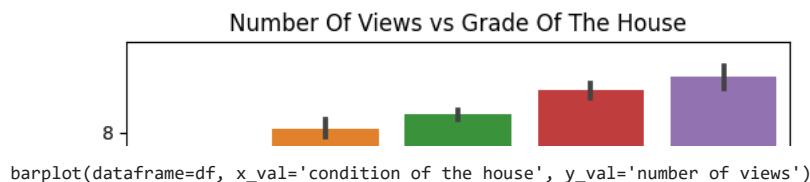
```
barplot(dataframe=df, x_val='number of bedrooms', y_val='number of bathrooms')
```

**Number Of Bedrooms vs Number Of Bathrooms****Number Of Floors vs Number Of Bathrooms**

```
barplot(dataframe=df, x_val='condition of the house', y_val='grade of the house')
```

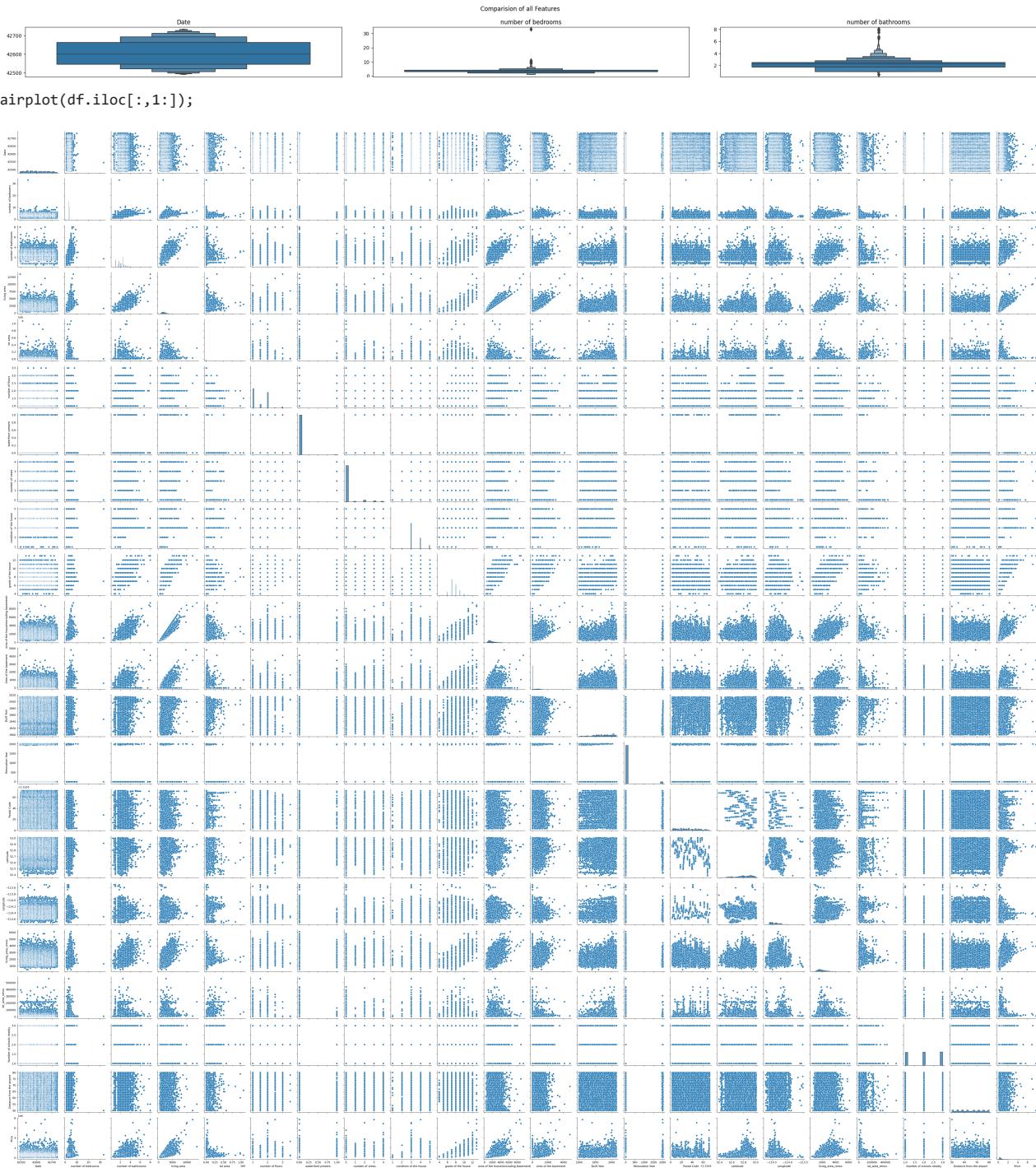
**Condition Of The House vs Grade Of The House**

```
barplot(dataframe=df, x_val='number of views', y_val='grade of the house')
```

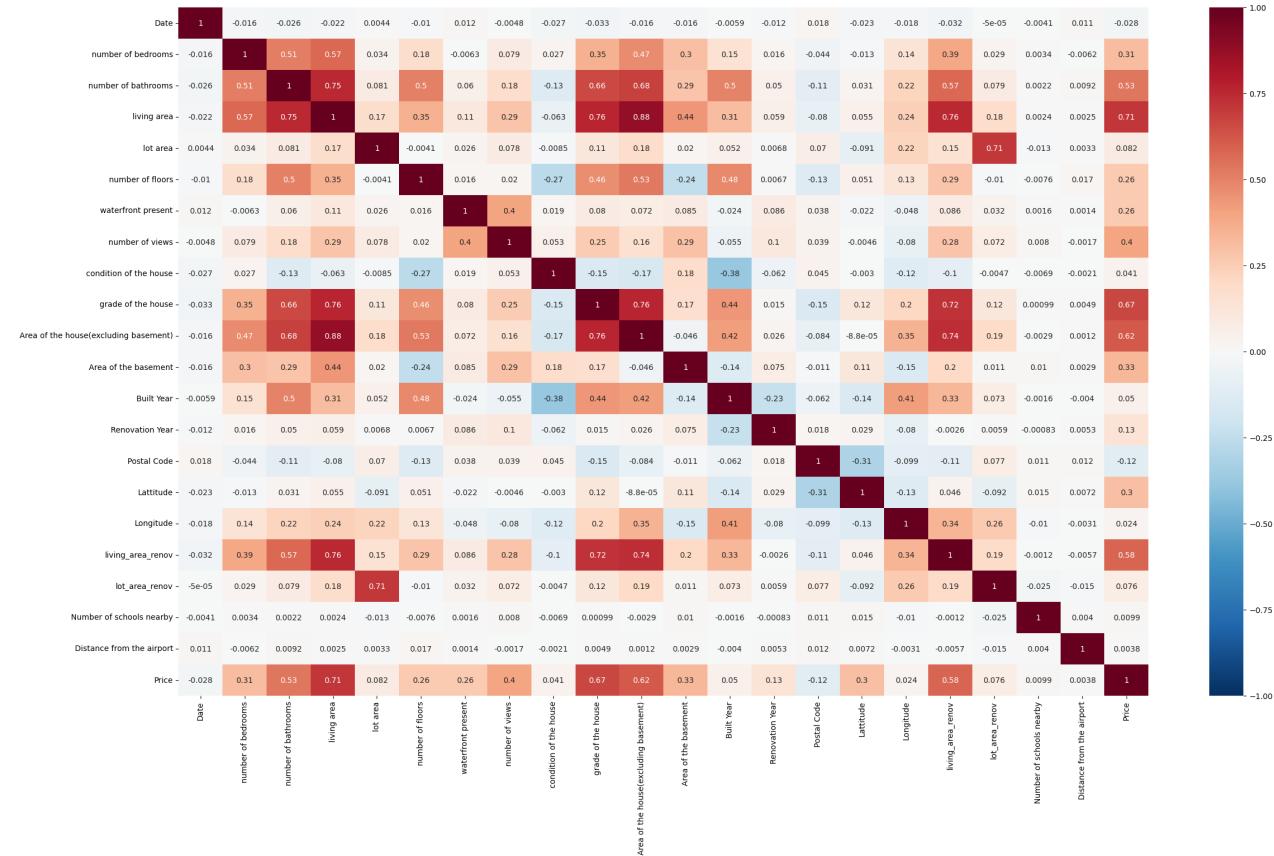


```
fig = plt.figure(figsize = [30, 18])

for i in range(1, len(list(df.columns))):  
    plt.subplot(8, 3, i)  
  
    fig.tight_layout(pad=2)  
    df_new = df.iloc[:,i]  
  
    plt.suptitle('Comparision of all Features')  
    plt.title(list(df.columns)[i])  
  
    sns.boxenplot(df_new)  
  
plt.show()
```



```
plt.figure(figsize=(28,16))
sns.heatmap(
    df.iloc[:,1:].corr(),
    cmap='RdBu_r',
    annot=True,
    vmin=-1, vmax=1);
```



```
df.iloc[:,2:].describe()
```

	number of bedrooms	number of bathrooms	living area	lot area	number of floors	waterfront present	number of views	condition of the house	grad the h
<b>count</b>	14620.000000	14620.000000	14620.000000	1.462000e+04	14620.000000	14620.000000	14620.000000	14620.000000	14620.000000
<b>mean</b>	3.379343	2.129583	2098.262996	1.509328e+04	1.502360	0.007661	0.233105	3.430506	7.68
<b>std</b>	0.938719	0.769934	928.275721	3.791962e+04	0.540239	0.087193	0.766259	0.664151	1.17
<b>min</b>	1.000000	0.500000	370.000000	5.200000e+02	1.000000	0.000000	0.000000	1.000000	4.00
<b>25%</b>	3.000000	1.750000	1440.000000	5.010750e+03	1.000000	0.000000	0.000000	3.000000	7.00
<b>50%</b>	3.000000	2.250000	1930.000000	7.620000e+03	1.500000	0.000000	0.000000	3.000000	7.00
<b>75%</b>	4.000000	2.500000	2570.000000	1.080000e+04	2.000000	0.000000	0.000000	4.000000	8.00
<b>max</b>	33.000000	8.000000	13540.000000	1.074218e+06	3.500000	1.000000	4.000000	5.000000	13.00

8 rows × 21 columns

```
df.isna().sum()
```

id	0
Date	0
number of bedrooms	0
number of bathrooms	0
living area	0
lot area	0
number of floors	0
waterfront present	0
number of views	0
condition of the house	0
grade of the house	0
Area of the house(excluding basement)	0
Area of the basement	0
Built Year	0
Renovation Year	0
Postal Code	0
Latitude	0
Longitude	0
living_area_renov	0
lot_area_renov	0

```
Number of schools nearby      0  
Distance from the airport    0  
Price                         0  
dtype: int64
```

[Colab paid products](#) - [Cancel contracts here](#)

