


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

```
df=pd.read_csv("/content/Housing.csv")
df
```



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

545 rows × 13 columns

```
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 545 entries, 0 to 544
Data columns (total 13 columns):
#   Column          Non-Null Count  Dtype
---  -
#   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
```

df.shape

(545, 13)

df.size

7085

df.head()

	price	area	bedrooms	bathrooms	stories	mainroad	guestroom	basement	hotwaterheating	airconditioning	parking	prefarea	furnishingstatus
0	13300000	7420	4	2	3	yes	no	no	no	yes	2	yes	full
1	12250000	8960	4	4	4	yes	no	no	no	yes	3	no	full
2	12250000	9960	3	2	2	yes	no	yes	no	no	2	yes	semi-full
3	12215000	7500	4	2	2	yes	no	yes	no	yes	3	yes	full
4	11110000	7120	4	1	2	yes	yes	yes	no	yes	2	no	full

df.tail()

	price	area	bedrooms	bathrooms	stories	mainroad	guestroom	basement	hotwaterheating	airconditioning	parking	prefarea	furnishing
540	1820000	3000	2	1	1	yes	no	yes	no	no	2	no	unf
541	1767150	2400	3	1	1	no	no	no	no	no	0	no	semi-f
542	1750000	3620	2	1	1	yes	no	no	no	no	0	no	unf
543	1750000	2910	3	1	1	no	no	no	no	no	0	no	f
544	1750000	3850	3	1	2	yes	no	no	no	no	0	no	unf

df.dtypes

```
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
```

df.columns

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

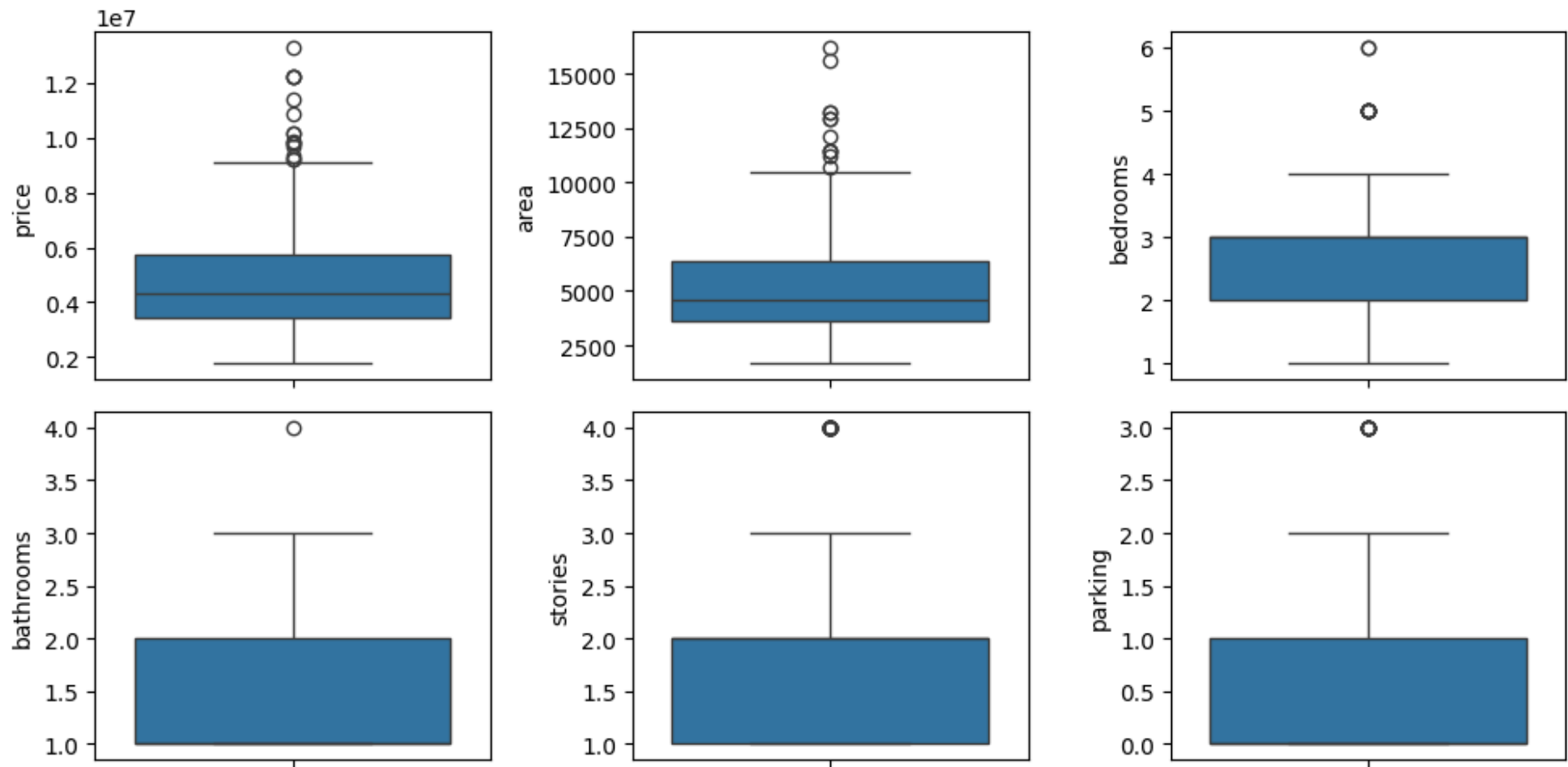
df.isna().sum()

```
price          0
area           0
bedrooms       0
```

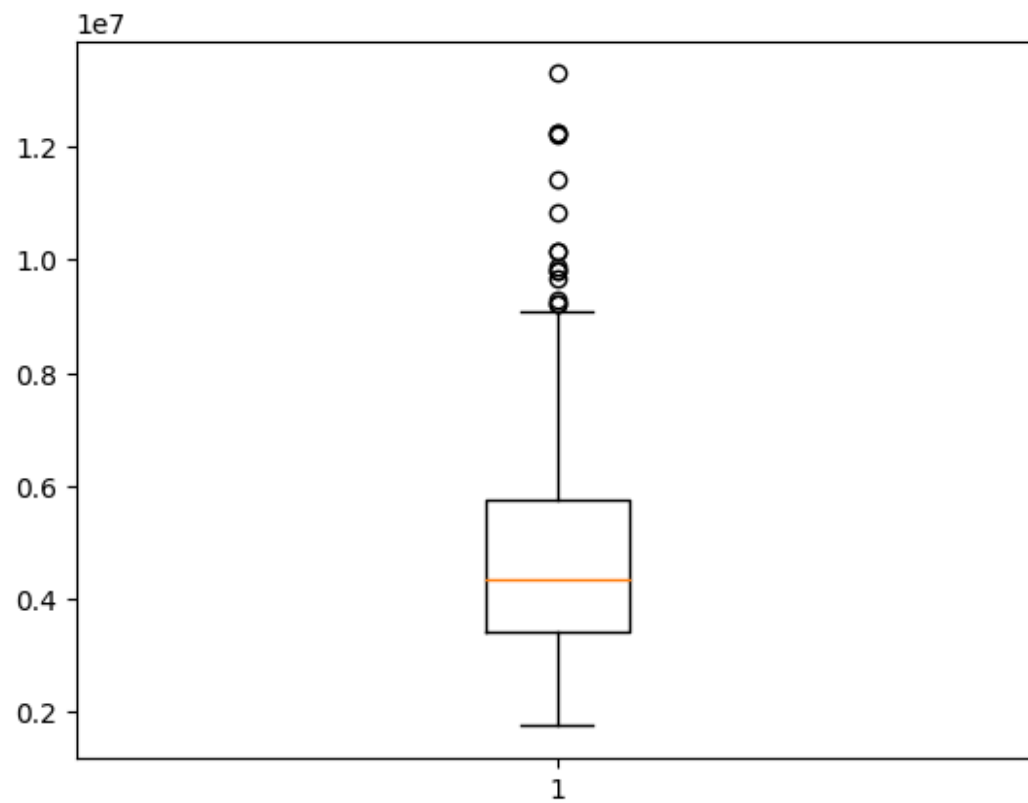
```
bathrooms      0
stories        0
mainroad       0
guestroom      0
basement       0
hotwaterheating 0
airconditioning 0
parking        0
prefarea       0
furnishingstatus 0
dtype: int64
```

```
# Outlier Analysis
fig, axs = plt.subplots(2,3, figsize = (10,5))
plt1 = sns.boxplot(df['price'], ax = axs[0,0])
plt2 = sns.boxplot(df['area'], ax = axs[0,1])
plt3 = sns.boxplot(df['bedrooms'], ax = axs[0,2])
plt1 = sns.boxplot(df['bathrooms'], ax = axs[1,0])
plt2 = sns.boxplot(df['stories'], ax = axs[1,1])
plt3 = sns.boxplot(df['parking'], ax = axs[1,2])

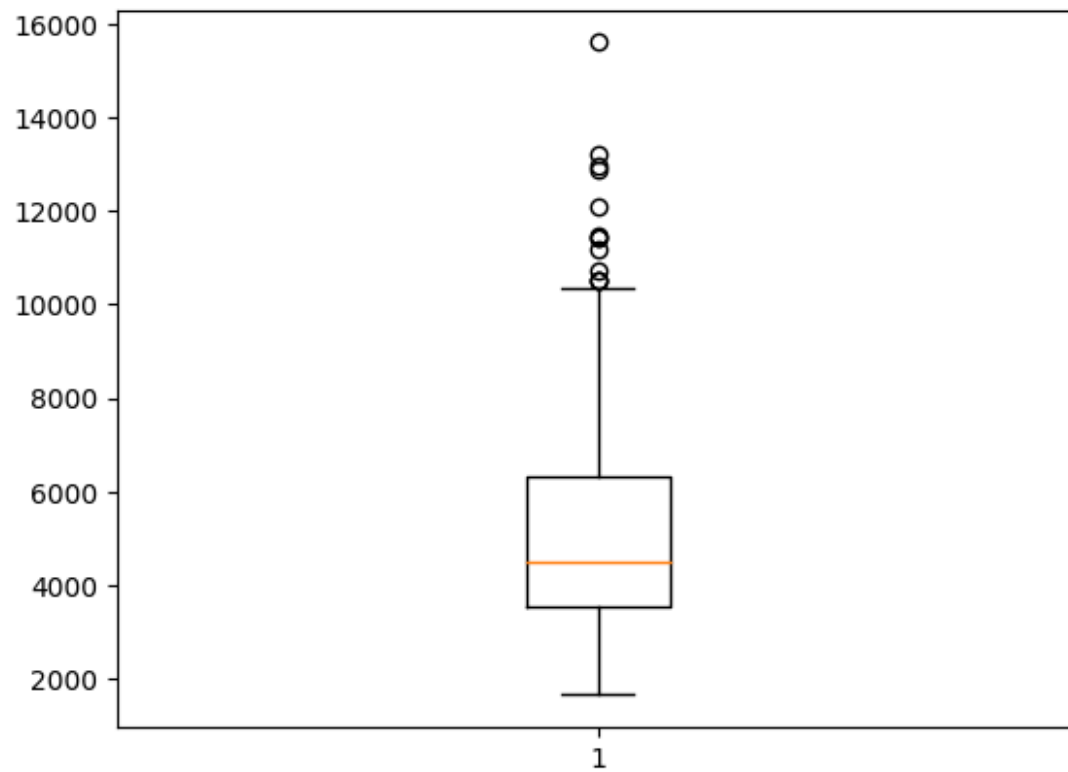
plt.tight_layout()
```



```
# outlier treatment for price
plt.boxplot(df.price)
Q1 = df.price.quantile(0.25)
Q3 = df.price.quantile(0.75)
IQR = Q3 - Q1
df = df[(df.price >= Q1 - 1.5*IQR) & (df.price <= Q3 + 1.5*IQR)]
```



```
# outlier treatment for area
plt.boxplot(df.area)
Q1 = df.area.quantile(0.25)
Q3 = df.area.quantile(0.75)
IQR = Q3 - Q1
df = df[(df.area >= Q1 - 1.5*IQR) & (df.area <= Q3 + 1.5*IQR)]
```



```
dummy=pd.get_dummies(df[['mainroad','guestroom', 'basement', 'hotwaterheating', 'airconditioning', 'prefarea', 'furnishingstatus']],drop_first=True)  
dummy
```

	mainroad_yes	guestroom_yes	basement_yes	hotwaterheating_yes	airconditioning_yes	prefarea_yes	furnishingstatus_semi-furnished	furnishing
15	1	0	1	0	0	0	1	
16	1	1	1	0	1	1	0	
17	1	0	0	0	1	0	0	
18	1	1	0	0	1	0	0	
19	1	0	0	0	1	1	1	
...	
540	1	0	1	0	0	0	0	
541	0	0	0	0	0	0	1	
542	1	0	0	0	0	0	0	
543	0	0	0	0	0	0	0	
544	1	0	0	0	0	0	0	

517 rows × 8 columns

```
df1=pd.concat([df,dummy],axis=1)
df1
```


	price	area	bedrooms	bathrooms	stories	mainroad	guestroom	basement	hotwaterheating	airconditioning	...	prefarea	furnishir
15	9100000	6000	4	1	2	yes	no	yes	no	no	...	no	semi-
16	9100000	6600	4	2	2	yes	yes	yes	no	yes	...	yes	un
17	8960000	8500	3	2	4	yes	no	no	no	yes	...	no	
18	8890000	4600	3	2	2	yes	yes	no	no	yes	...	no	
19	8855000	6420	3	2	2	yes	no	no	no	yes	...	yes	semi-
...	
540	1820000	3000	2	1	1	yes	no	yes	no	no	...	no	un
541	1767150	2400	3	1	1	no	no	no	no	no	...	no	semi-
542	1750000	3620	2	1	1	yes	no	no	no	no	...	no	un
543	1750000	2910	3	1	1	no	no	no	no	no	...	no	
544	1750000	3850	3	1	2	yes	no	no	no	no	...	no	un

517 rows × 21 columns

```
df1=df1.drop(['mainroad','guestroom', 'basement', 'hotwaterheating', 'airconditioning', 'prefarea', 'furnishingstatus'],axis=1)
df1
```

	price	area	bedrooms	bathrooms	stories	parking	mainroad_yes	guestroom_yes	basement_yes	hotwaterheating_yes	airconditioning
15	9100000	6000	4	1	2	2	1	0	1	0	
16	9100000	6600	4	2	2	1	1	1	1	0	
17	8960000	8500	3	2	4	2	1	0	0	0	
18	8890000	4600	3	2	2	2	1	1	0	0	
19	8855000	6420	3	2	2	1	1	0	0	0	
...	
540	1820000	3000	2	1	1	2	1	0	1	0	
541	1767150	2400	3	1	1	0	0	0	0	0	
542	1750000	3620	2	1	1	0	1	0	0	0	
543	1750000	2910	3	1	1	0	0	0	0	0	
544	1750000	3850	3	1	2	0	1	0	0	0	

517 rows × 14 columns

df1.dtypes

```

price          int64
area           int64
bedrooms       int64
bathrooms      int64
stories        int64
parking        int64
mainroad_yes   uint8
guestroom_yes  uint8
basement_yes   uint8
hotwaterheating_yes  uint8
airconditioning_yes  uint8
prefarea_yes   uint8
furnishingstatus_semi-furnished  uint8
furnishingstatus_unfurnished  uint8
dtype: object

```

```
x=df1.drop(['price'],axis=1)
x
```

	area	bedrooms	bathrooms	stories	parking	mainroad_yes	guestroom_yes	basement_yes	hotwaterheating_yes	airconditioning_yes	pre
15	6000	4	1	2	2	1	0	1	0	0	
16	6600	4	2	2	1	1	1	1	0	1	
17	8500	3	2	4	2	1	0	0	0	1	
18	4600	3	2	2	2	1	1	0	0	1	
19	6420	3	2	2	1	1	0	0	0	1	
...	
540	3000	2	1	1	2	1	0	1	0	0	
541	2400	3	1	1	0	0	0	0	0	0	
542	3620	2	1	1	0	1	0	0	0	0	
543	2910	3	1	1	0	0	0	0	0	0	
544	3850	3	1	2	0	1	0	0	0	0	

517 rows × 13 columns

```
y=df1['price']
y
```

```
15    9100000
16    9100000
17    8960000
18    8890000
19    8855000
...
540   1820000
541   1767150
542   1750000
```

```
543 1750000
544 1750000
Name: price, Length: 517, dtype: int64
```

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,random_state=42)
x_train
```

	area	bedrooms	bathrooms	stories	parking	mainroad_yes	guestroom_yes	basement_yes	hotwaterheating_yes	airconditioning_yes	pre
474	4352	4	1	2	1	0	0	0	0	0	
543	2910	3	1	1	0	0	0	0	0	0	
178	7085	3	1	1	2	1	1	1	0	0	
40	6550	3	1	2	0	1	0	1	0	1	
292	2747	4	2	2	0	0	0	0	0	0	
...	
91	6750	2	1	1	2	1	1	1	0	0	
127	6500	3	2	3	0	1	0	0	0	1	
297	3640	3	2	2	0	1	0	1	0	0	
463	3090	2	1	1	0	1	1	1	0	0	
122	6254	4	2	1	1	1	0	1	0	0	

387 rows × 13 columns

```
x_test
```

	area	bedrooms	bathrooms	stories	parking	mainroad_yes	guestroom_yes	basement_yes	hotwaterheating_yes	airconditioning_yes	pre
331	7260	3	2	1	3	1	1	1	0	0	
529	3970	3	1	2	0	1	0	1	0	0	
469	4600	2	1	1	0	1	0	0	0	0	
177	6050	3	1	1	0	1	0	1	0	0	
531	5300	3	1	1	0	0	0	0	0	1	
...	
34	6840	5	1	2	1	1	1	1	0	1	
533	2400	3	1	2	0	1	0	0	0	0	
367	3630	2	1	1	0	1	0	0	0	1	
75	4260	4	2	2	0	1	0	0	1	0	
305	3450	3	1	2	1	1	0	0	0	0	

130 rows × 13 columns

y_train

```

474    2975000
543    1750000
178    5229000
40     7875000
292    4200000
...
91     6419000
127    5880000
297    4200000
463    3080000
122    5950000

```

Name: price, Length: 387, dtype: int64

```
y_test
```

```
331    3920000
529    2275000
469    3010000
177    5243000
531    2233000
```

```
...
```

```
34     8120000
533    2100000
367    3675000
75     6650000
305    4193000
```

```
Name: price, Length: 130, dtype: int64
```

```
from sklearn.preprocessing import StandardScaler
scaler=StandardScaler()
scaler.fit(x_train)
```

▼ StandardScaler

```
StandardScaler()
```

```
x_train=scaler.transform(x_train)
x_test=scaler.transform(x_test)
```

```
from sklearn.linear_model import LinearRegression
model=LinearRegression()
model.fit(x_train,y_train)
```

▼ LinearRegression

```
LinearRegression()
```

```
y_pred=model.predict(x_test)
y_pred
```

```
array([6194128.36716976, 3708265.44191532, 3278149.9224309 ,
       4548078.41690674, 4259819.15023141, 5810311.38242633,
```

3787425.1319972 , 3413010.1523162 , 3755775.93391517,
4308741.98014449, 2920161.18725972, 3203662.95236367,
4852718.18418045, 6060809.50830749, 7711000.8138782 ,
2985882.87344751, 6558821.03792345, 6128719.13277128,
4988962.57949334, 6012760.81566001, 5997937.55702228,
4517016.06371847, 4210161.59609396, 5521345.42322105,
5160197.9893913 , 5390074.26625152, 5550131.72250513,
2897627.51447483, 3036172.05696851, 4246203.94237666,
3266504.15143808, 3487337.13290691, 6555242.31247906,
6892836.17158825, 6723888.72882142, 3084706.13577415,
2833049.15569326, 4597490.90334397, 6595774.39402495,
4054665.22140898, 4476068.24776444, 6365274.30704325,
4620492.65670607, 4885377.57224581, 3446690.53874322,
4447366.88987447, 6128523.66140305, 2821127.06307179,
3046377.33981311, 4235631.92768213, 5552709.92599634,
3878546.01165676, 3524362.2641639 , 6568288.13869669,
2213813.13678861, 4723186.21058468, 7699380.82380036,
3769851.55913972, 3098211.11904949, 6355107.31699222,
3417394.58478077, 4299535.09859223, 4952581.22704303,
7080913.47008744, 2786505.41463526, 4011266.73569139,
4750033.60728005, 4356620.86041333, 5177302.48653666,
5017742.3562727 , 7836496.16482978, 4082054.68213363,
7917196.34863343, 6121406.39308625, 4232057.88027172,
6531246.85456437, 5662052.55530476, 3172778.45635302,
2716108.66946111, 6271387.75990502, 3691672.36011089,
3866678.15124159, 3045492.62251385, 5149444.44770837,
6322764.39096781, 3220110.68063684, 6132270.00914305,
2872645.53594336, 5263726.37449914, 3861082.92903906,
5938087.41805516, 3280265.16009549, 4391503.64052012,
3006740.71812295, 7044401.39977643, 6127948.66909685,
6895101.54872653, 3847734.37060897, 2533911.97021292,
4013972.73482058, 4132614.2879905 , 2816944.10965609,
2411029.42705321, 3216207.58766904, 4850094.9135177 ,
3955683.79323657, 4669511.64853254, 2749985.57062585,
3809401.4221589 , 2832450.09306894, 3824125.26011381,
6407063.14926889, 3018581.00128911, 3075062.46707257,
2223492.25140711, 2837097.60219231, 5429050.7923989 ,
6085843.0679925 , 3017563.16014767, 6144820.02631941,
3349863.94664135, 7155487.91074025, 4205498.83943637,
5118089.32774551, 3753641.65838789, 6388013.69254177,
2986328.61547977, 3609704.51943376, 5129175.16540427,
3723911.21061878])

```
print(model.coef_)
```

```
[ 421274.20553618   69847.80099196  335519.49728808  382920.36146923
 193767.8716274   137064.9099503   149710.01107001  162730.10264607
 135340.16389145  400561.34939058  215809.60601501 -24587.66366152
-141010.24401814]
```

```
list(zip(x,model.coef_))
```

```
[('area', 421274.2055361774),
 ('bedrooms', 69847.80099196325),
 ('bathrooms', 335519.49728808226),
 ('stories', 382920.3614692315),
 ('parking', 193767.8716274008),
 ('mainroad_yes', 137064.9099503031),
 ('guestroom_yes', 149710.0110700127),
 ('basement_yes', 162730.1026460734),
 ('hotwaterheating_yes', 135340.16389144532),
 ('airconditioning_yes', 400561.3493905797),
 ('prefarea_yes', 215809.6060150095),
 ('furnishingstatus_semi-furnished', -24587.663661522216),
 ('furnishingstatus_unfurnished', -141010.24401813897)]
```

```
print(model.intercept_)
```

```
4543252.687338501
```

```
from sklearn.metrics import mean_absolute_error
print(mean_absolute_error(y_pred,y_test))
```

```
720789.0592011426
```

```
from sklearn.metrics import mean_absolute_percentage_error
print(mean_absolute_percentage_error(y_pred,y_test))
```

```
0.16370154111169832
```

```
from sklearn.metrics import mean_squared_error
print(mean_squared_error(y_pred,y_test))
```


910725548167.2351

```
mse=mean_squared_error(y_pred,y_test)
rmse=np.sqrt(mse)
rmse
```

954319.4162162034

```
from sklearn.metrics import r2_score
print(r2_score(y_pred,y_test))
```

0.5509313402870675

```
from sklearn.preprocessing import PolynomialFeatures
poly=PolynomialFeatures()
x_poly=poly.fit_transform(x)
x_poly
```

```
array([[1.00e+00, 6.00e+03, 4.00e+00, ..., 1.00e+00, 0.00e+00, 0.00e+00],
       [1.00e+00, 6.60e+03, 4.00e+00, ..., 0.00e+00, 0.00e+00, 1.00e+00],
       [1.00e+00, 8.50e+03, 3.00e+00, ..., 0.00e+00, 0.00e+00, 0.00e+00],
       ...,
       [1.00e+00, 3.62e+03, 2.00e+00, ..., 0.00e+00, 0.00e+00, 1.00e+00],
       [1.00e+00, 2.91e+03, 3.00e+00, ..., 0.00e+00, 0.00e+00, 0.00e+00],
       [1.00e+00, 3.85e+03, 3.00e+00, ..., 0.00e+00, 0.00e+00, 1.00e+00]])
```

```
model1=LinearRegression()
model1.fit(x_poly,y)
```

▾ LinearRegression

LinearRegression()

```
y_poly=model1.predict(x_poly)
y_poly
```

```
array([5212043.46412995, 7958165.1018231 , 7871580.2449039 ,
       7254161.92279949, 7591210.03811599, 7078274.73053734,
```

8075534.41363492, 6655796.43025099, 6973508.06943973,
7262500.3545027 , 8069633.12802633, 7855181.68434715,
4823953.33207059, 8650256.71149069, 7277923.55665832,
8139829.82662969, 6964903.78311496, 6632808.08992082,
6839153.47904213, 6877783.19297101, 7823314.49834665,
6833998.69879143, 7785759.35242216, 6866669.9702782 ,
7699649.32672868, 6023188.08170411, 7982217.77247437,
8083176.22763896, 6632035.97472083, 7500377.79090599,
6305921.11819862, 7583339.71726626, 7254015.97946179,
4930099.28507092, 7391294.33181166, 7338377.71343498,
6672820.15471011, 7789788.60925507, 7791968.21726042,
7301348.08615963, 4891330.08256173, 7690534.56505232,
7383270.96174543, 7587398.06276466, 5484425.30690609,
5312547.5785801 , 7023776.84006026, 7615121.50763191,
6906086.64329245, 5548009.18901891, 5648920.44382125,
6289908.78131224, 6464183.63583391, 6789743.80944074,
6827193.47085024, 7438613.33300808, 5910193.89731809,
7145906.56940692, 7162600.98000578, 6400918.22094498,
6378719.77617056, 7838001.31583158, 5741736.33549489,
6950932.29553033, 5804808.17068816, 6430512.58618441,
5933036.2423356 , 4158224.76905461, 6260996.29997177,
7013493.09967076, 4582568.07685421, 5124579.29991309,
6550539.14171991, 5903852.54361692, 6426370.15397618,
6521709.55469852, 5919871.3437011 , 6394136.67183848,
7032248.59083216, 5076496.72890513, 6814594.74973433,
5545985.40325069, 7620116.30129828, 6375712.03807781,
6171528.12790365, 5603280.92523459, 6661743.12870595,
5935167.50114324, 4984790.17990351, 6017645.93728299,
5651496.28190588, 5686605.09776249, 4705409.87273714,
5645720.92202947, 4786111.0959348 , 6289913.82476288,
5697354.008966 , 5406148.30384827, 5793518.30975962,
5905771.85642414, 5628542.46522015, 5599986.00726839,