

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
In [2]: data=pd.read_csv("housing_prices_SLR.csv")
```

```
In [3]: data.head()
```

```
Out[3]:
```

	AREA	PRICE
0	1000	5618
1	1030	5201
2	1060	4779
3	1090	5425
4	1120	5657

```
In [4]: x=data.iloc[:, :-1].values
y=data.iloc[:, 1].values
```

```
In [5]: print(x)
print(y)
```

```
[[1000]
 [1030]
 [1060]
 [1090]
 [1120]
 [1150]
 [1180]
 [1210]
 [1240]
 [1270]
 [1300]
 [1330]
 [1360]
 [1390]
 [1420]
 [1450]
 [1480]
 [1510]
 [1540]
 [1570]
 [1600]
 [1630]
 [1660]
 [1690]
 [1720]
 [1750]
 [1780]
 [1810]
 [1840]
 [1870]
 [1900]
 [1930]
 [1960]
 [1990]
 [2020]
 [2050]
 [2080]
 [2110]
 [2140]
 [2170]
 [2200]
 [2230]
 [2260]
 [2290]
 [2320]
 [2350]
 [2380]
 [2410]
 [2440]
 [2470]]
[ 5618  5201  4779  5425  5657  4945  6475  5352  6290  6147  7718  7035
  8712  9457  7304  7535  9968  7614  9762 10067  8074 10201 10051  9775
 10229  8314  9288 10503  9763 11527  9912 10204  9085 10240 15220 12278
 22270 12574 12618 15457 12793 12641 17207 16969 16293 16845 17604 17310
14036 13699]
```

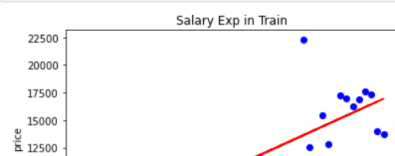
```
In [6]: from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

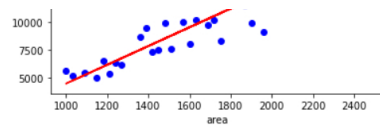
```
In [7]: x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=0)
```

```
In [8]: lreg=LinearRegression()
lreg.fit(x_train,y_train)
```

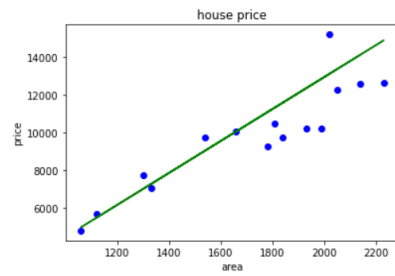
```
Out[8]: LinearRegression()
```

```
In [9]: plt.scatter(x_train,y_train,color='blue')
plt.plot(x_train,lreg.predict(x_train),color="red")
plt.title("Salary Exp in Train")
plt.xlabel("area")
plt.ylabel("price")
plt.show()
```





```
In [10]: plt.scatter(x_test,y_test,color="blue")
plt.plot(x_test,lireg.predict(x_test),color="green")
plt.title("house price")
plt.xlabel("area")
plt.ylabel("price")
plt.show()
```



```
In [11]: from sklearn import metrics
```

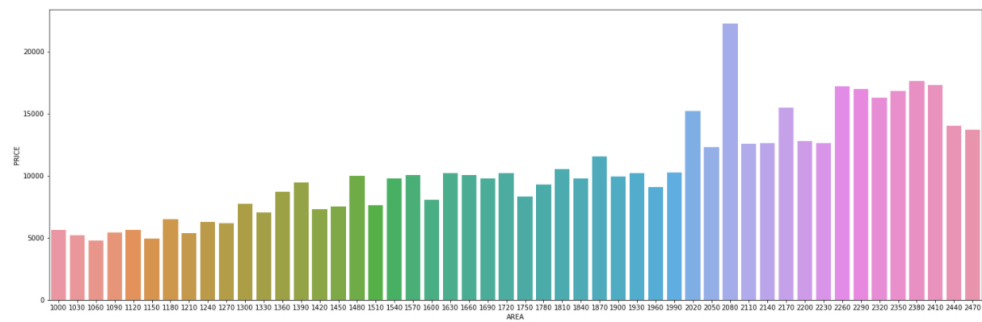
```
In [12]: ypre=lireg.predict(x_test)
```

```
In [13]: print("RMSE:",np.sqrt(metrics.mean_absolute_error(y_test,ypre)))
```

RMSE: 34.87213571320669

```
In [21]: plt.figure(figsize=(25, 8))
sns.barplot(x='AREA',y='PRICE',data=data)
```

Out[21]: <AxesSubplot:xlabel='AREA', ylabel='PRICE'>



```
In [ ]:
```