In [1]:
```
pip install scikit-learn
```

```
Requirement already satisfied: scikit-learn in c:\users\lenovo\anaconda3\lib\site
-packages (1.5.1)Note: you may need to restart the kernel to use updated package
s.

Requirement already satisfied: numpy>=1.19.5 in c:\users\lenovo\anaconda3\lib\sit
e-packages (from scikit-learn) (1.26.4)
Requirement already satisfied: scipy>=1.6.0 in c:\users\lenovo\anaconda3\lib\site
-packages (from scikit-learn) (1.13.1)
Requirement already satisfied: joblib>=1.2.0 in c:\users\lenovo\anaconda3\lib\sit
e-packages (from scikit-learn) (1.4.2)
Requirement already satisfied: threadpoolctl>=3.1.0 in c:\users\lenovo\anaconda3
\lib\site-packages (from scikit-learn) (3.5.0)
```

In [6]:
```
df = pd.read_csv(r'C:\Users\lenovo\Desktop\NIT FILES\20th- slr\SLR - House price
df
```

Out[6]:

	id	date	price	bedrooms	bathrooms	sqft_living	sqft
0	7129300520	20141013T000000	221900.0	3	1.00	1180	5
1	6414100192	20141209T000000	538000.0	3	2.25	2570	7
2	5631500400	20150225T000000	180000.0	2	1.00	770	10
3	2487200875	20141209T000000	604000.0	4	3.00	1960	5
4	1954400510	20150218T000000	510000.0	3	2.00	1680	8
...	...	...	...	...	...	...	
21608	263000018	20140521T000000	360000.0	3	2.50	1530	1
21609	6600060120	20150223T000000	400000.0	4	2.50	2310	5
21610	1523300141	20140623T000000	402101.0	2	0.75	1020	1
21611	291310100	20150116T000000	400000.0	3	2.50	1600	2
21612	1523300157	20141015T000000	325000.0	2	0.75	1020	1

21613 rows × 21 columns

In [3]:
```
import pandas as pd
```

In [5]:
```
import matplotlib.pyplot as plt
```

In [7]:
```
import numpy as np
```

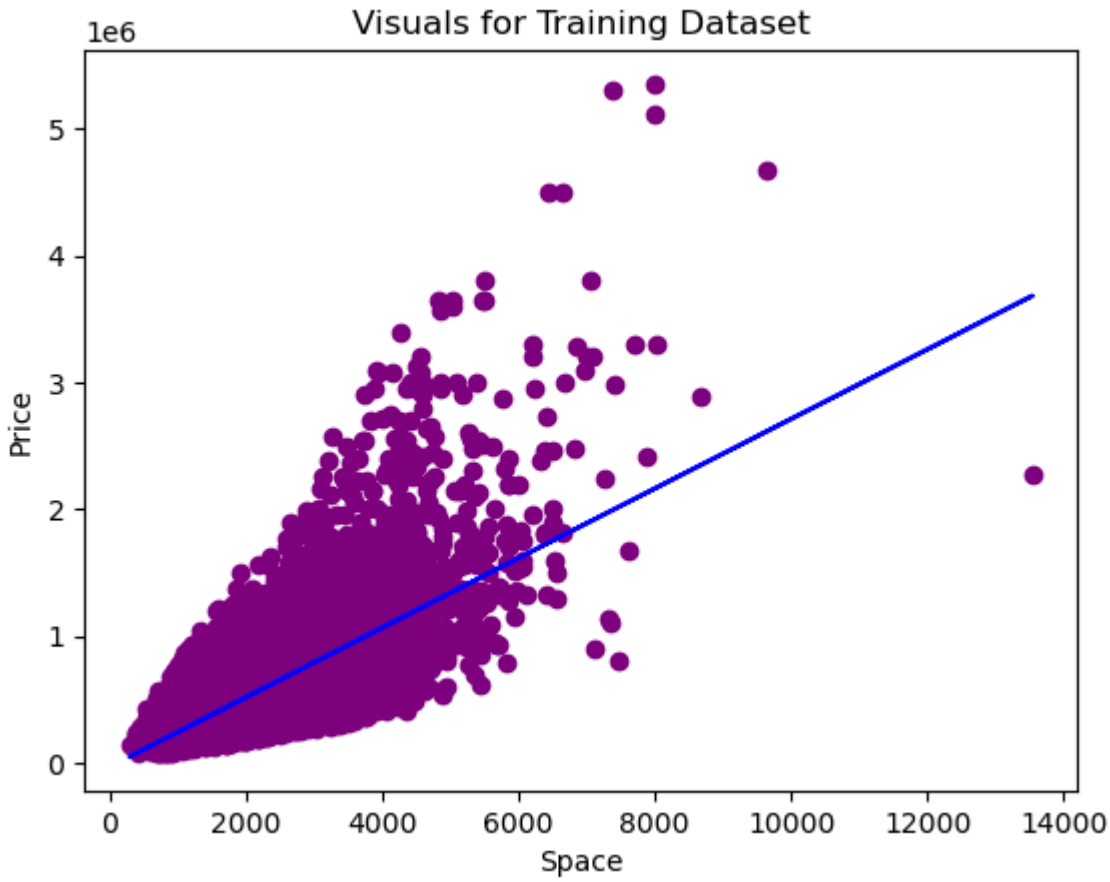
In [9]:
```
df = pd.read_csv(r'C:\Users\lenovo\Desktop\NIT FILES\20th- slr\SLR - House price
space=df['sqft_living']
price=df['price']
x = np.array(space).reshape(-1, 1)
y = np.array(price)
from sklearn.model_selection import train_test_split
xtrain, xtest, ytrain, ytest = train_test_split(x,y,test_size=1/3, random_state=
```

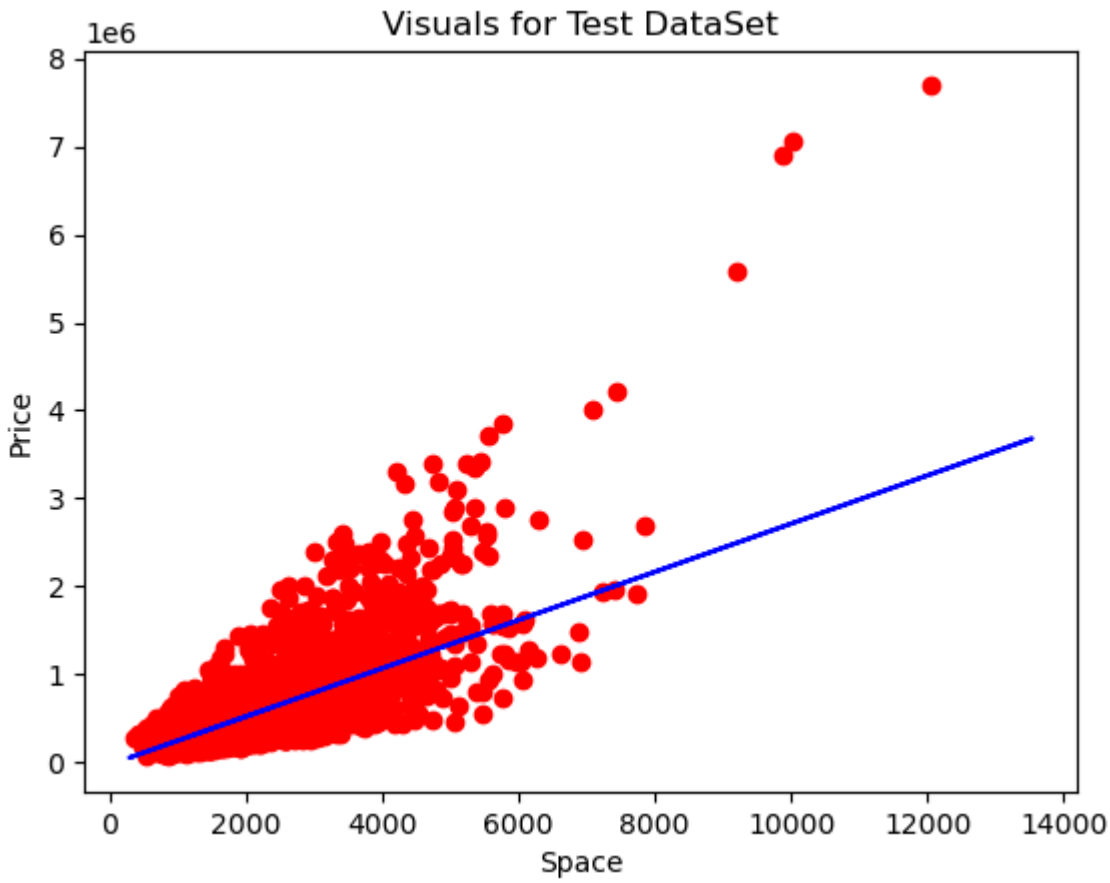
```
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(xtrain, ytrain)

pred = regressor.predict(xtest)

plt.scatter(xtrain, ytrain, color= 'purple')
plt.plot(xtrain, regressor.predict(xtrain), color = 'blue')
plt.title ("Visuals for Training Dataset")
plt.xlabel("Space")
plt.ylabel("Price")
plt.show()

plt.scatter(xtest, ytest, color= 'red')
plt.plot(xtrain, regressor.predict(xtrain), color = 'blue')
plt.title("Visuals for Test DataSet")
plt.xlabel("Space")
plt.ylabel("Price")
plt.show()
```




Visuals for Test DataSet
1e6
8
7
6
5
4
3
2
1
0
Price
0
2000
4000
6000
8000
10000
12000
14000
Space