In [1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn import preprocessing, svm
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

In [3]:
```
df=pd.read_csv(r"C:\Users\manis\OneDrive\Desktop\ParisHousing.csv")
df
```

Out[3]:

	squareMeters	numberOfRooms	hasYard	hasPool	floors	cityCode	cityPartRang
0	75523	3	0	1	63	9373	
1	80771	39	1	1	98	39381	
2	55712	58	0	1	19	34457	
3	32316	47	0	0	6	27939	1
4	70429	19	1	1	90	38045	
...	...	...	...	...	...	...	
9995	1726	89	0	1	5	73133	
9996	44403	29	1	1	12	34606	
9997	83841	3	0	0	69	80933	1
9998	59036	70	0	0	96	55856	
9999	1440	84	0	0	49	18412	

10000 rows × 20 columns

In [4]:
```
df=df[['squareMeters','price']]
df.columns=['sm','pr']
```

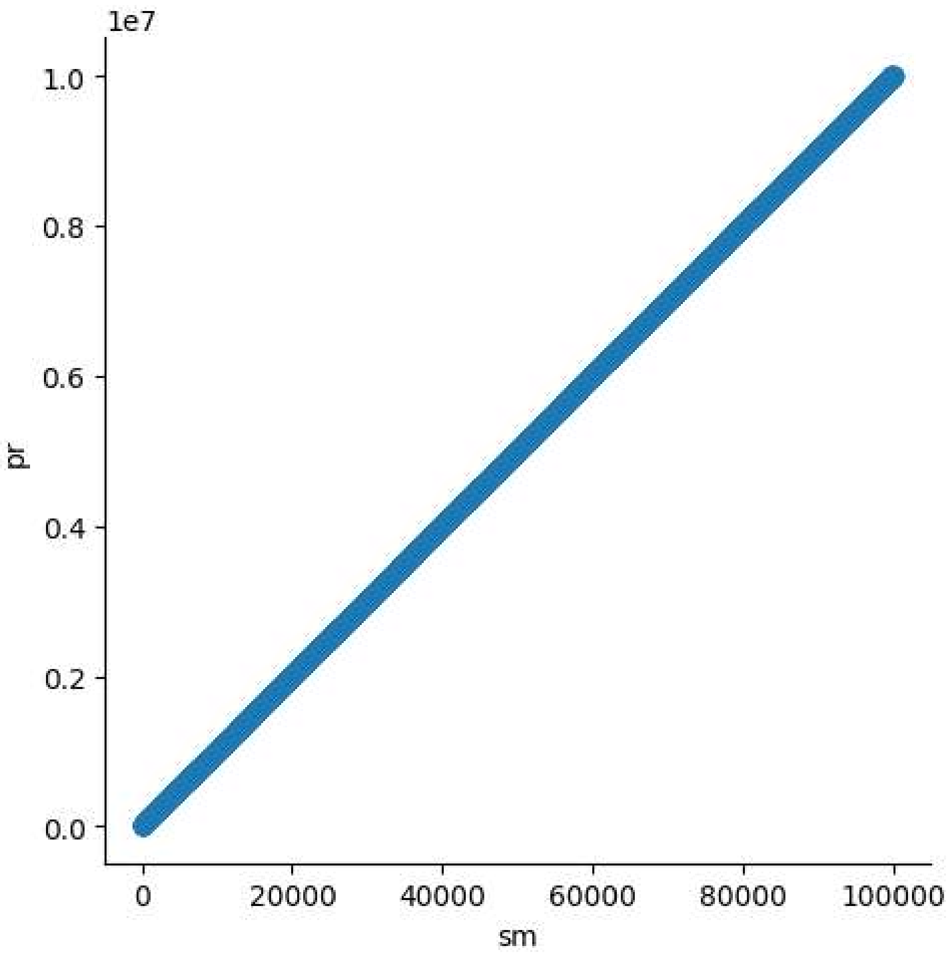
In [5]:
```
df.head(10)
```

Out[5]:

	sm	pr
0	75523	7559081.5
1	80771	8085989.5
2	55712	5574642.1
3	32316	3232561.2
4	70429	7055052.0
5	39223	3926647.2
6	58682	5876376.5
7	86929	8696869.3
8	51522	5154055.2
9	39686	3970892.1

In [6]:
```
sns.lmplot(x='sm',y='pr',data=df,order=2,ci=None)
```

Out[6]:
```
<seaborn.axisgrid.FacetGrid at 0x205f3960ed0>
```



In [7]:
```
df.describe()
```

Out[7]:

	sm	pr
count	10000.00000	1.000000e+04
mean	49870.13120	4.993448e+06
std	28774.37535	2.877424e+06
min	89.00000	1.031350e+04
25%	25098.50000	2.516402e+06
50%	50105.50000	5.016180e+06
75%	74609.75000	7.469092e+06
max	99999.00000	1.000677e+07

In [8]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   sm      10000 non-null  int64  
 1   pr      10000 non-null  float64
dtypes: float64(1), int64(1)
memory usage: 156.4 KB
```

In [9]:
```
df.fillna(method='ffill',inplace=True)
```

```
C:\Users\manis\AppData\Local\Temp\ipykernel_16088\4116506308.py:1: SettingWithCop
yWarning: 
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stabl
e/user_guide/indexing.html#returning-a-view-versus-a-copy
  df.fillna(method='ffill',inplace=True)
```

In [11]:
```
X=np.array(df['sm']).reshape(-1,1)
y=np.array(df['pr']).reshape(-1,1)
df.dropna(inplace=True)
```

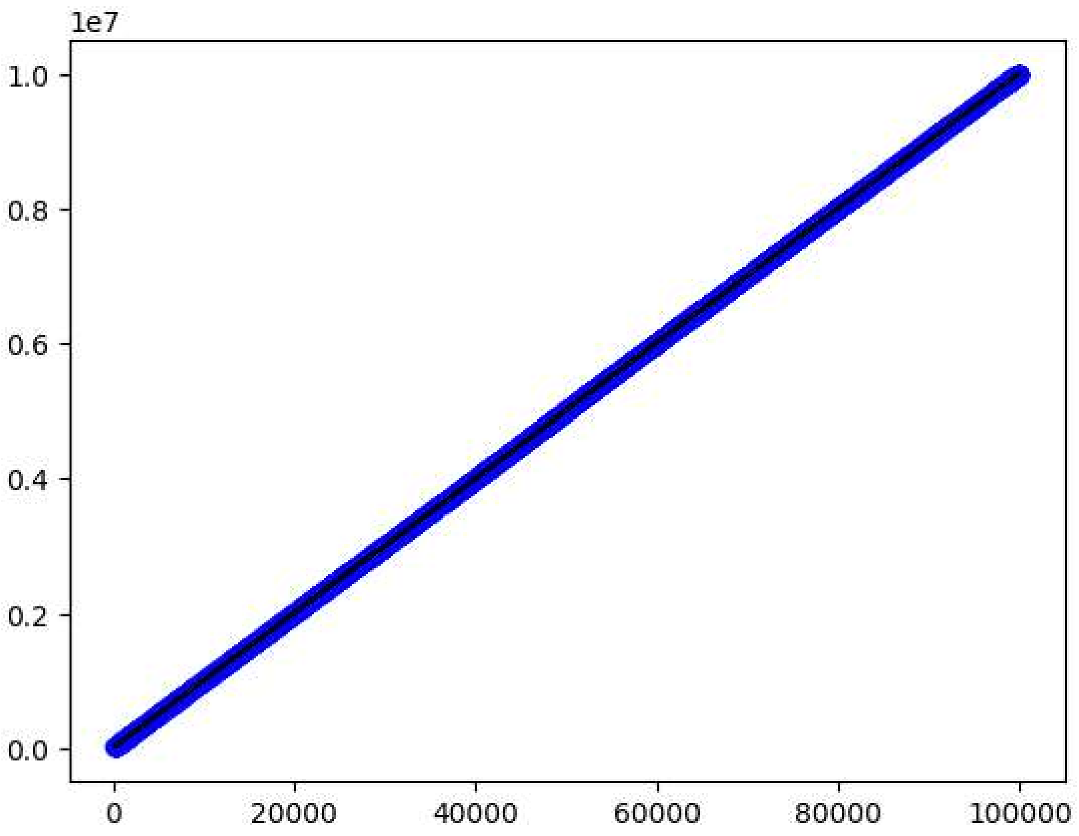
```
C:\Users\manis\AppData\Local\Temp\ipykernel_16088\2340689882.py:3: SettingWithCop
yWarning: 
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stabl
e/user_guide/indexing.html#returning-a-view-versus-a-copy
  df.dropna(inplace=True)
```

In [12]:
```
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.25)
regr=LinearRegression()
regr.fit(X_train,y_train)
print(regr.score(X_test,y_test))
```

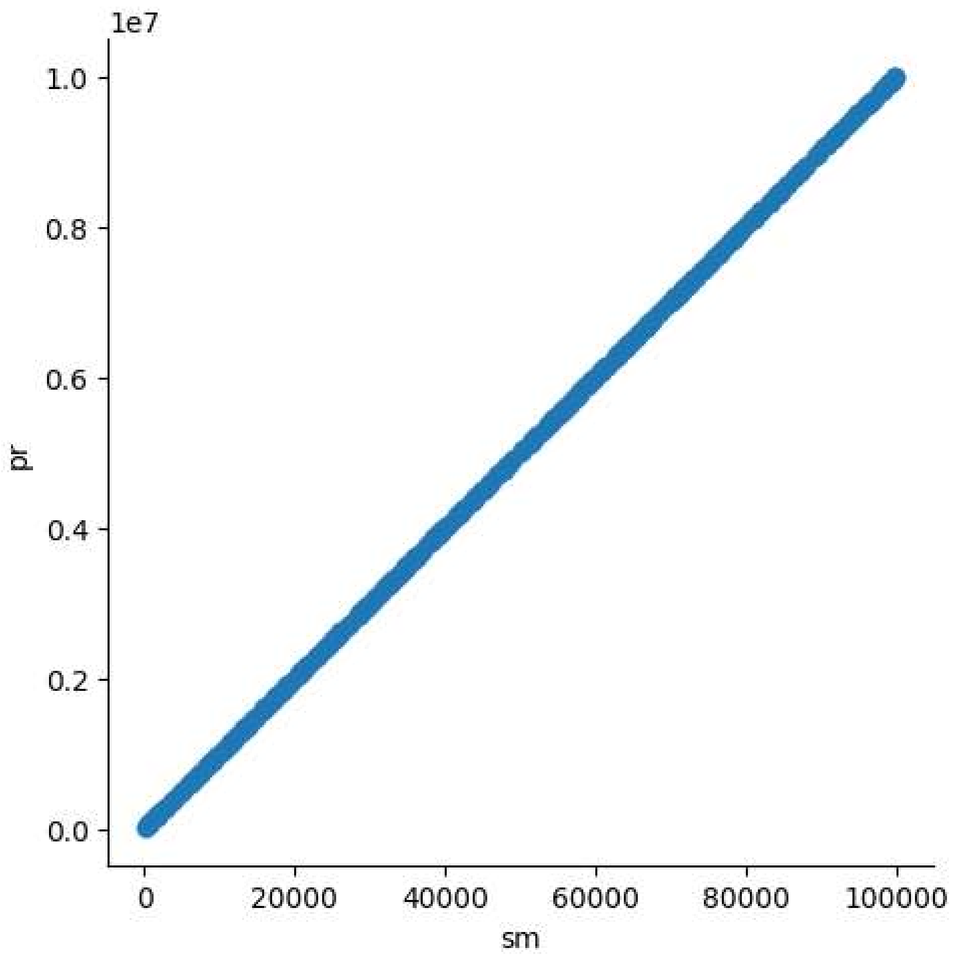
```
0.9999987468023971
```

In [13]:
```
y_pred=regr.predict(X_test)
plt.scatter(X_test,y_test,color='b')
plt.plot(X_test,y_pred,color='k')
plt.show()
```



```
In [14]: df500=df[:][:500]
         sns.lmplot(x='sm',y='pr',data=df500,order=1,ci=None)
```

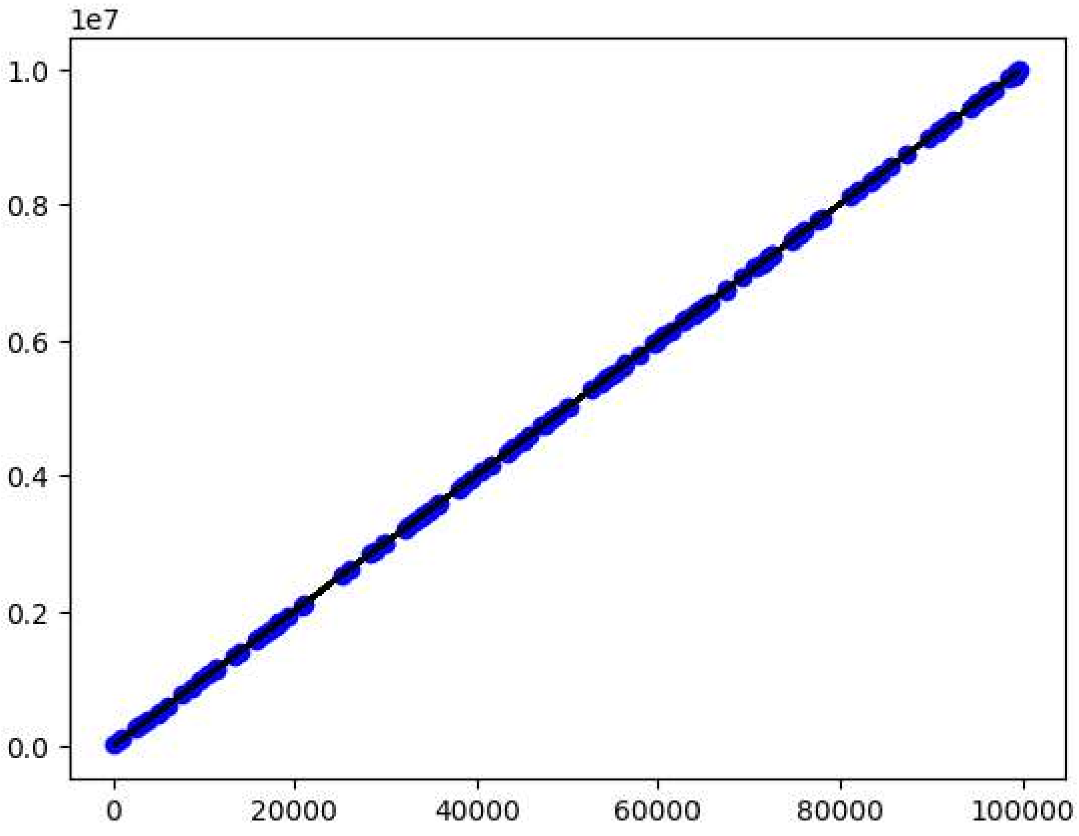
```
Out[14]: <seaborn.axisgrid.FacetGrid at 0x2058a501790>
```



```
In [15]: df500.fillna(method='ffill',inplace=True)
         X=np.array(df500['sm']).reshape(-1,1)
         y=np.array(df500['pr']).reshape(-1,1)
         df500.dropna(inplace=True)
         X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.25)
         regr=LinearRegression()
         regr.fit(X_train,y_train)
         print("Regression: ",regr.score(X_test,y_test))
```

```
Regression:  0.9999986053702787
```

```
In [16]: y_pred=regr.predict(X_test)
         plt.scatter(X_test,y_test,color='b')
         plt.plot(X_test,y_pred,color='k')
         plt.show()
```



In [17]:

```
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score
model=LinearRegression()
model.fit(X_train,y_train)
y_pred=model.predict(X_test)
r2=r2_score(y_test,y_pred)
print("R2 score: ",r2)
```

```
R2 score:  0.9999986053702787
```

In [ ]: