In [2]:

```
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\91628\Downloads\data (1).csv")
print(df)
```

```
                     date         price  bedrooms  bathrooms  sqft_living
0     2014-05-02 00:00:00  3.130000e+05       3.0       1.50         1340
\
1     2014-05-02 00:00:00  2.384000e+06       5.0       2.50         3650
2     2014-05-02 00:00:00  3.420000e+05       3.0       2.00         1930
3     2014-05-02 00:00:00  4.200000e+05       3.0       2.25         2000
4     2014-05-02 00:00:00  5.500000e+05       4.0       2.50         1940
...                   ...           ...       ...        ...          ...
4595  2014-07-09 00:00:00  3.081667e+05       3.0       1.75         1510
4596  2014-07-09 00:00:00  5.343333e+05       3.0       2.50         1460
4597  2014-07-09 00:00:00  4.169042e+05       3.0       2.50         3010
4598  2014-07-10 00:00:00  2.034000e+05       4.0       2.00         2090
4599  2014-07-10 00:00:00  2.206000e+05       3.0       2.50         1490

      sqft_lot  floors  waterfront  view  condition  sqft_above  \
0         7912     1.5           0     0          3        1340
1         9050     2.0           0     4          5        3370
2        11947     1.0           0     0          4        1930
3         8030     1.0           0     0          4        1000
4        10500     1.0           0     0          4        1140
...        ...     ...         ...   ...        ...         ...
4595      6360     1.0           0     0          4        1510
4596      7573     2.0           0     0          3        1460
4597      7014     2.0           0     0          3        3010
4598      6630     1.0           0     0          3        1070
4599      8102     2.0           0     0          4        1490

      sqft_basement  yr_built  yr_renovated                    street  \
0                 0      1955          2005      18810 Densmore Ave N
1               280      1921             0           709 W Blaine St
2                 0      1966             0  26206-26214 143rd Ave SE
3              1000      1963             0           857 170th Pl NE
4               800      1976          1992         9105 170th Ave NE
...             ...       ...           ...                       ...
4595              0      1954          1979          501 N 143rd St
4596              0      1983          2009         14855 SE 10th Pl
4597              0      2009             0          759 Ilwaco Pl NE
4598           1020      1974             0         5148 S Creston St
4599              0      1990             0         18717 SE 258th St

           city  statezip country
0     Shoreline  WA 98133     USA
1       Seattle  WA 98119     USA
2          Kent  WA 98042     USA
3      Bellevue  WA 98008     USA
4       Redmond  WA 98052     USA
...         ...       ...     ...
4595    Seattle  WA 98133     USA
4596   Bellevue  WA 98007     USA
4597     Renton  WA 98059     USA
4598    Seattle  WA 98178     USA
4599  Covington  WA 98042     USA

[4600 rows x 18 columns]
```

In [4]:

```
df=df[['sqft_living','yr_built']]
df.columns=['living','built']
```

In [5]:

```
df.head(10)
```

Out[5]:

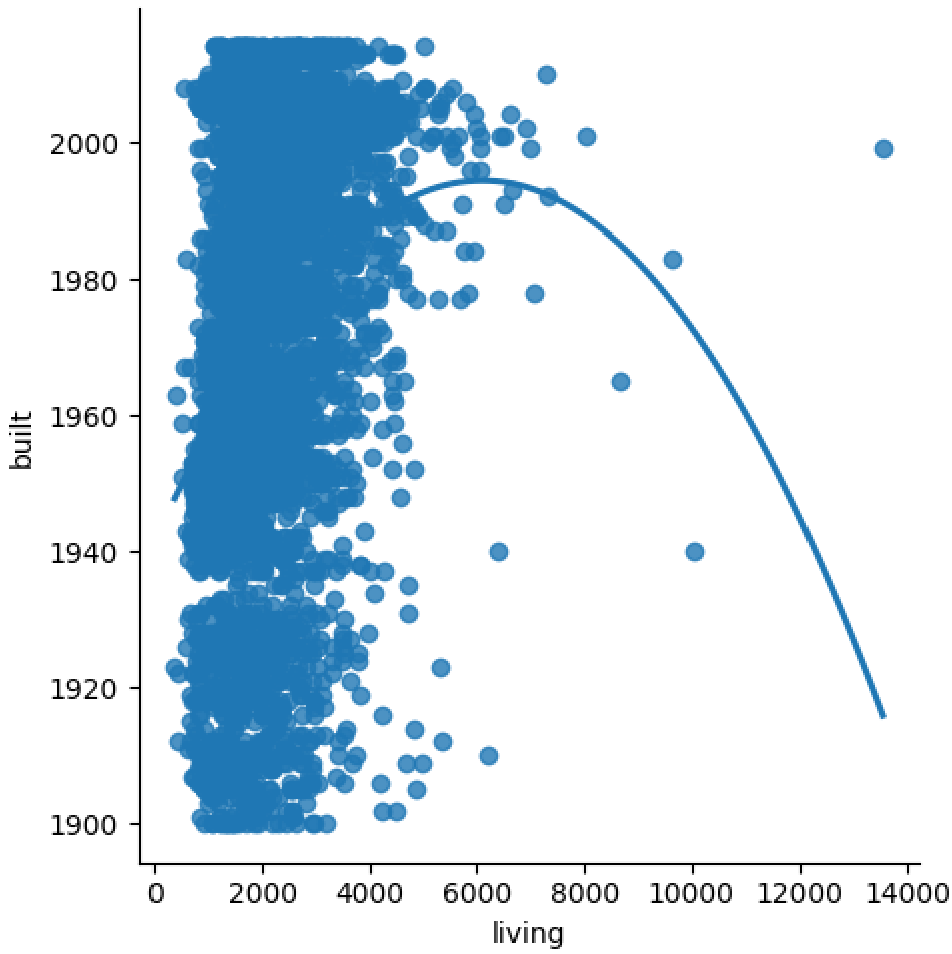
	living	built
0	1340	1955
1	3650	1921
2	1930	1966
3	2000	1963
4	1940	1976
5	880	1938
6	1350	1976
7	2710	1989
8	2430	1985
9	1520	1945

In [6]:

```
sns.lmplot(x="living",y="built",data=df,order=2,ci=None)
```

Out[6]:

```
<seaborn.axisgrid.FacetGrid at 0x24354b52e90>
```



In [7]:

```
df.describe()
```

Out[7]:

	living	built
count	4600.000000	4600.000000
mean	2139.346957	1970.786304
std	963.206916	29.731848
min	370.000000	1900.000000
25%	1460.000000	1951.000000
50%	1980.000000	1976.000000
75%	2620.000000	1997.000000
max	13540.000000	2014.000000

In [8]:

```
df.fillna(method='ffill',inplace=True)
```

In [9]:

```
x=np.array(df['living']).reshape(-1,1)
y=np.array(df['built']).reshape(-1,1)
```

In [10]:

```
df.dropna(inplace=True)
```

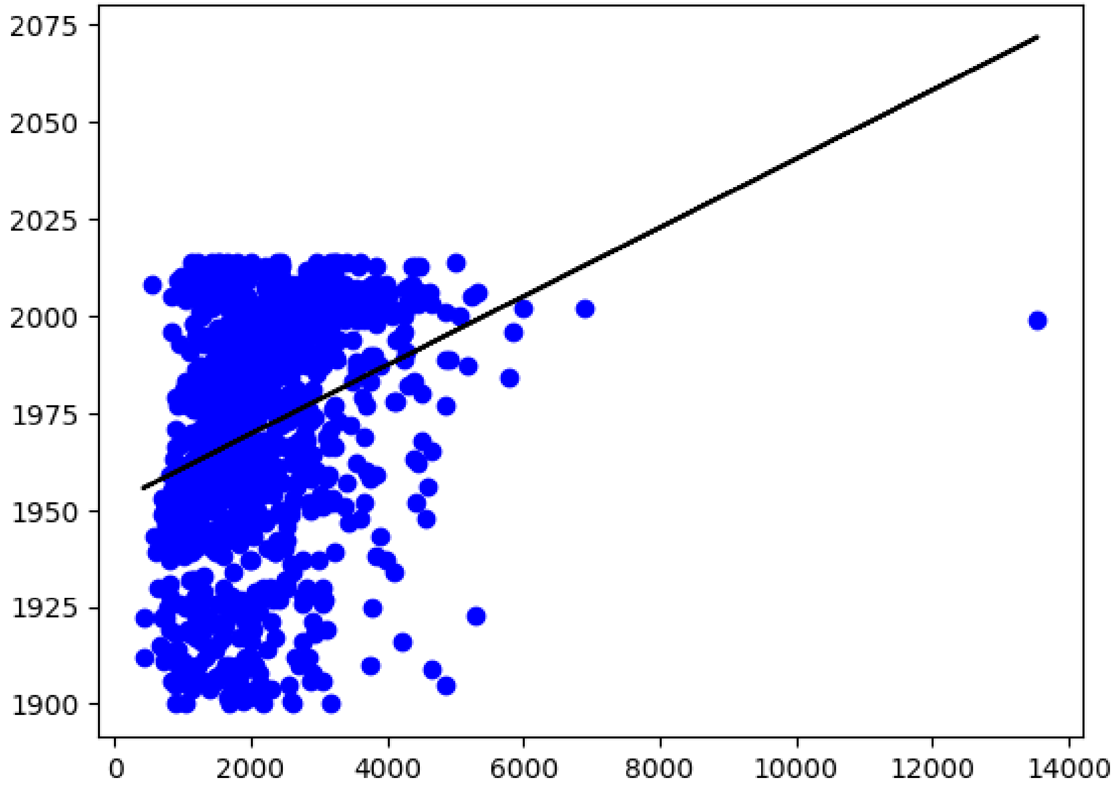
In [11]:

```
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))
```

```
0.08578920038865767
```

In [12]:

```
y_pred=regr.predict(x_test)
plt.scatter (x_test,y_test,color='b')
plt.plot(x_test,y_pred,color='k')
plt.show()
```

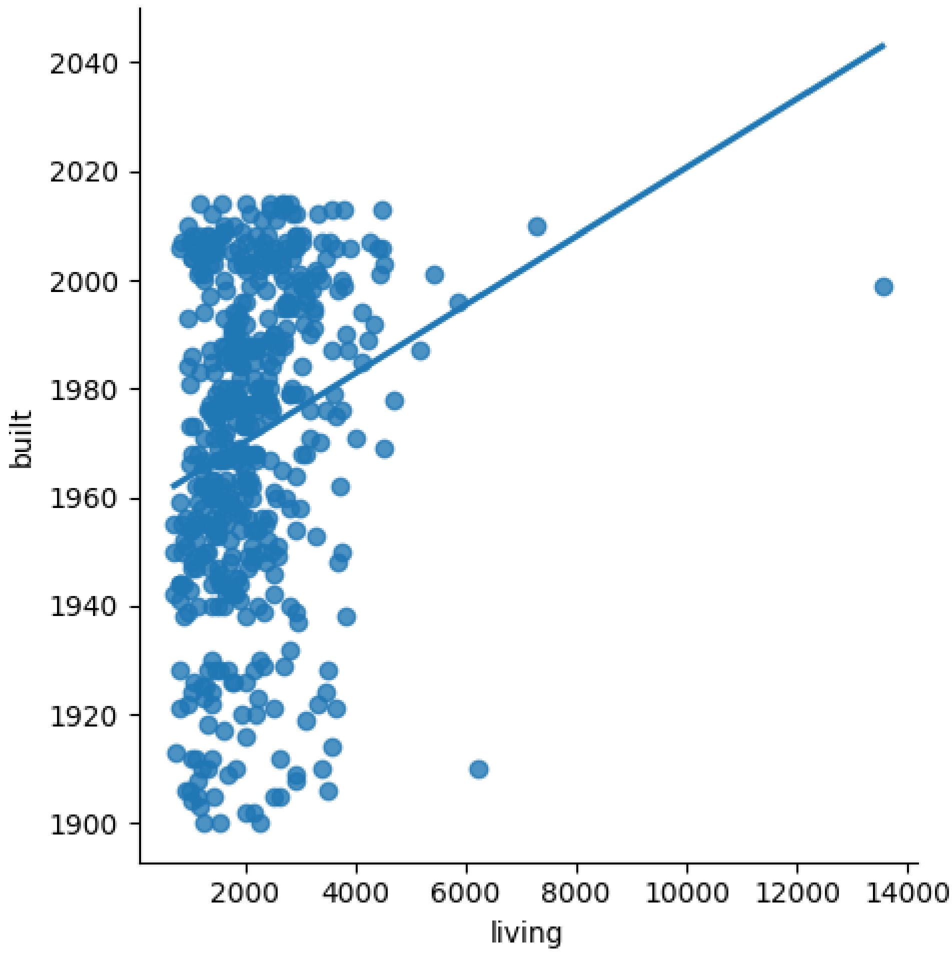


In [13]:

```
df500=df[:][:500]
sns.lmplot(x="living",y="built",data=df500,order=1,ci=None)
```

Out[13]:

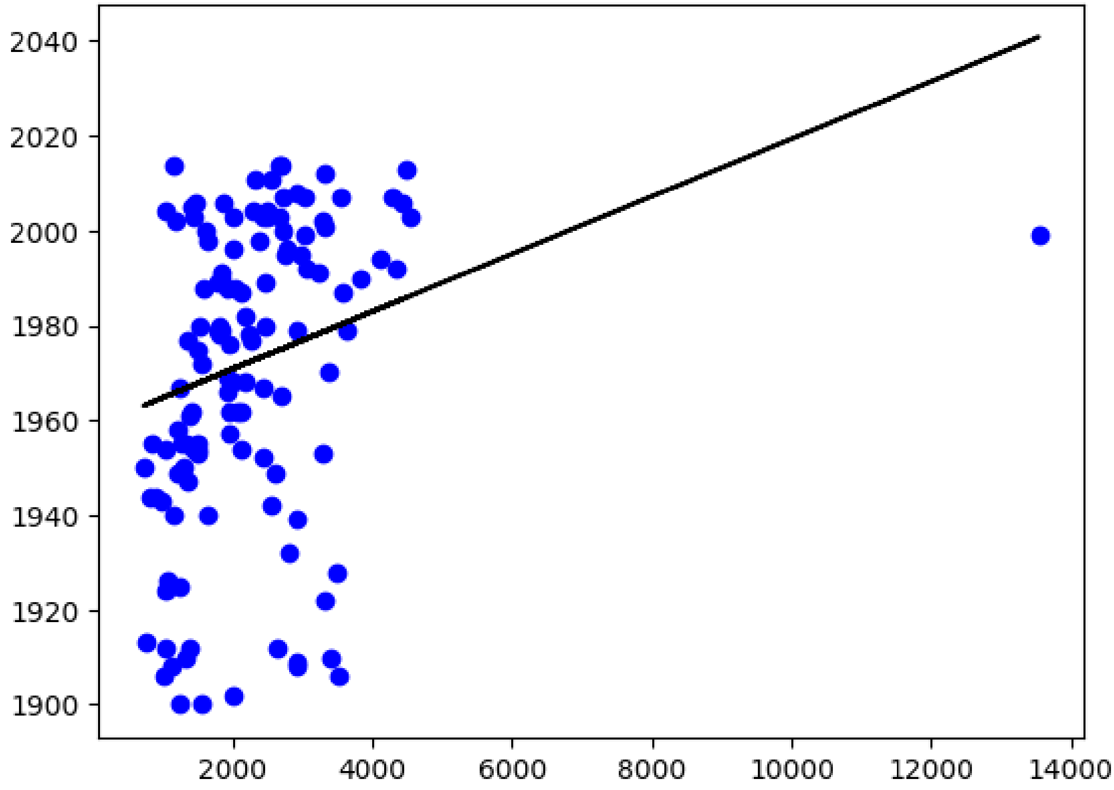
<seaborn.axisgrid.FacetGrid at 0x24339bbae90>



In [14]:

```
df500.fillna(method='ffill',inplace=True)
x=np.array(df500['living']).reshape(-1,1)
y=np.array(df500['built']).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))
y_pred=regr.predict(x_test)
plt.scatter(x_test,y_test,color='b')
plt.plot(x_test,y_pred,color='k')
plt.show()
```

```
Regression: 0.07599677032427032
```



In [15]:

```
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score
mode1=LinearRegression()
mode1.fit(x_train,y_train)
y_pred=mode1.predict(x_test)
r2=r2_score(y_test,y_pred)
print("R2 score:",r2)
```

```
R2 score: 0.07599677032427032
```

In [ ]:

```

```