In [51]:

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

from sklearn.model_selection import train_test_split
```

In [52]:

```
df = pd.read_csv("C:\\Users\\91805\\Downloads\\archive (10).zip")
df.head(10)
```

Out[52]:

	Unnamed: 0	Address	Zip	Price	Area	Room	Lon	Lat
0	1	Blasiusstraat 8 2, Amsterdam	1091 CR	685000.0	64	3	4.907736	52.356157
1	2	Kromme Leimuidenstraat 13 H, Amsterdam	1059 EL	475000.0	60	3	4.850476	52.348586
2	3	Zaaiersweg 11 A, Amsterdam	1097 SM	850000.0	109	4	4.944774	52.343782
3	4	Tenerifestraat 40, Amsterdam	1060 TH	580000.0	128	6	4.789928	52.343712
4	5	Winterjanpad 21, Amsterdam	1036 KN	720000.0	138	5	4.902503	52.410538
5	6	De Wittenkade 134 I, Amsterdam	1051 AM	450000.0	53	2	4.875024	52.382228
6	7	Pruimenstraat 18 B, Amsterdam	1033 KM	450000.0	87	3	4.896536	52.410585
7	8	Da Costakade 32 II, Amsterdam	1053 WL	590000.0	80	2	4.871555	52.371041
8	9	Postjeskade 41 2, Amsterdam	1058 DG	399000.0	49	3	4.854671	52.363471
9	10	Van Ostadestraat 193 H, Amsterdam	1073 TM	300000.0	33	2	4.897142	52.353111

In [53]:

```
df= df.drop('Unnamed: 0',axis=1)
df.head()
```

Out[53]:

	Address	Zip	Price	Area	Room	Lon	Lat
0	Blasiusstraat 8 2, Amsterdam	1091 CR	685000.0	64	3	4.907736	52.356157
1	Kromme Leimuidenstraat 13 H, Amsterdam	1059 EL	475000.0	60	3	4.850476	52.348586
2	Zaaiersweg 11 A, Amsterdam	1097 SM	850000.0	109	4	4.944774	52.343782
3	Tenerifestraat 40, Amsterdam	1060 TH	580000.0	128	6	4.789928	52.343712
4	Winterjanpad 21, Amsterdam	1036 KN	720000.0	138	5	4.902503	52.410538

In [55]:

```
df.shape
```

Out[55]:

```
(924, 7)
```

In [58]:

```
df.isnull().sum()
```

Out[58]:

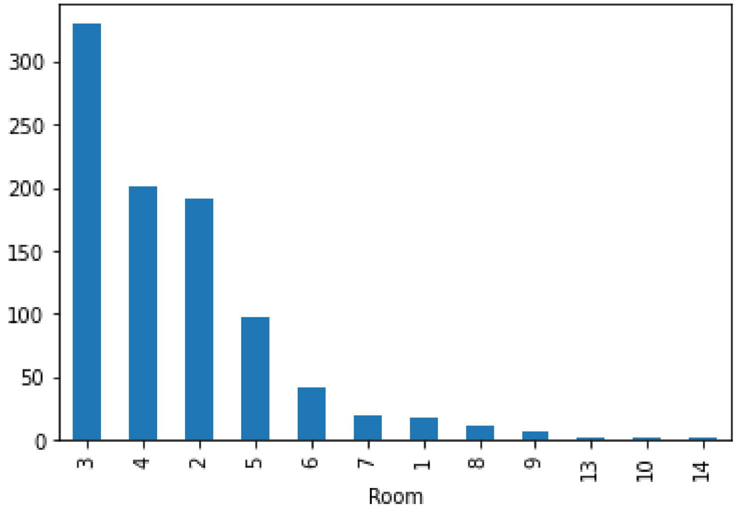
```
Address    0
Zip        0
Price      4
Area       0
Room       0
Lon        0
Lat        0
dtype: int64
```

```
In [59]: for i in df.index[df.isnull().any(axis=1)]:
             df=df.drop(i)
         df.isnull().sum()
```

```
Out[59]: Address    0
         Zip        0
         Price      0
         Area       0
         Room       0
         Lon        0
         Lat        0
         dtype: int64
```

```
In [60]: df.Room.value_counts().plot.bar()
```

```
Out[60]: <AxesSubplot:xlabel='Room'>
```



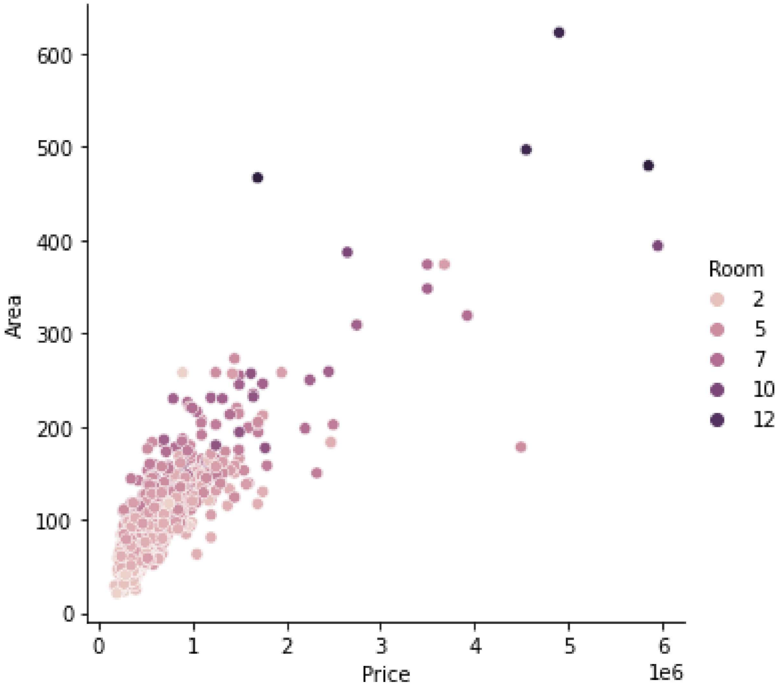
In [62]:

```
df_low_price=df[df['Price']<= df.Price.mean()]
print(df_low_price)
df_high_price=df[df['Price']> df.Price.mean()]
print(df_high_price)
```

```
                                    Address      Zip     Price  Area  Room  \
1    Kromme Leimuidenstraat 13 H, Amsterdam  1059 EL  475000.0    60     3
3              Tenerifestraat 40, Amsterdam  1060 TH  580000.0   128     6
5            De Wittenkade 134 I, Amsterdam  1051 AM  450000.0    53     2
6             Pruimenstraat 18 B, Amsterdam  1033 KM  450000.0    87     3
7             Da Costakade 32 II, Amsterdam  1053 WL  590000.0    80     2
..                                      ...      ...       ...   ...   ...
918                     Ringdijk, Amsterdam  1097 AE  295000.0    41     1
920         Kleine Beerstraat 31, Amsterdam  1033 CP  350000.0    72     3
921       Stuyvesantstraat 33 II, Amsterdam  1058 AK  350000.0    51     3
922   John Blankensteinstraat 51, Amsterdam  1095 MB  599000.0   113     4
923      S. F. van Ossstraat 334, Amsterdam  1068 JS  300000.0    79     4

          Lon        Lat
1    4.850476  52.348586
3    4.789928  52.343712
5    4.875024  52.382228
6    4.896536  52.410585
7    4.871555  52.371041
..        ...        ...
918  4.927757  52.354173
920  4.890612  52.414587
921  4.856935  52.363256
922  4.965731  52.375268
923  4.810678  52.355493

[635 rows x 7 columns]
                               Address      Zip      Price  Area  Room  \
0        Blasiusstraat 8 2, Amsterdam  1091 CR   685000.0    64     3
2          Zaaiersweg 11 A, Amsterdam  1097 SM   850000.0   109     4
4          Winterjanpad 21, Amsterdam  1036 KN   720000.0   138     5
14     Blasiusstraat 50 II, Amsterdam  1091 CT   650000.0    86     3
16   Paramaribostraat 122 3, Amsterdam  1058 VP   700000.0   102     6
..                                 ...      ...        ...   ...   ...
910     Valeriusstraat 193, Amsterdam  1075 EW  1698000.0   205     5
913   Bastenakenstraat 122, Amsterdam  1066 JG   675000.0   129     4
914      Hagedoornplein 24, Amsterdam  1031 BV   849000.0   111     5
917         Kromme Waal 18, Amsterdam  1011 BS  1500000.0   194     9
919               Ringdijk, Amsterdam  1097 AE   750000.0   117     1

          Lon        Lat
0    4.907736  52.356157
2    4.944774  52.343782
```

```
4    4.902503  52.410538
14   4.909361  52.356532
16   4.854520  52.362090
..        ...        ...
910  4.860637  52.351894
913  4.819965  52.343878
914  4.913690  52.388160
917  4.902614  52.374848
919  4.927757  52.354173

[285 rows x 7 columns]
```

In [64]:
```
sns.relplot(x="Price", y="Area", hue="Room", data=df)
plt.show()
```



In [65]:
```
df.columns
```

Out[65]:
```
Index(['Address', 'Zip', 'Price', 'Area', 'Room', 'Lon', 'Lat'], dtype='object')
```

In [66]:

```
continuous_features = ['Area','Lon','Lat','Price']
discrete_features = ['Address','Zip','Room']
```

In [67]:

```
df1 = df[df.Price<3000000].drop(discrete_features, axis=1)
```

In [68]:

```
X = df1.drop(['Price'], axis=1)
y = df1.Price
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.1, random_state=1)
```

In [69]:

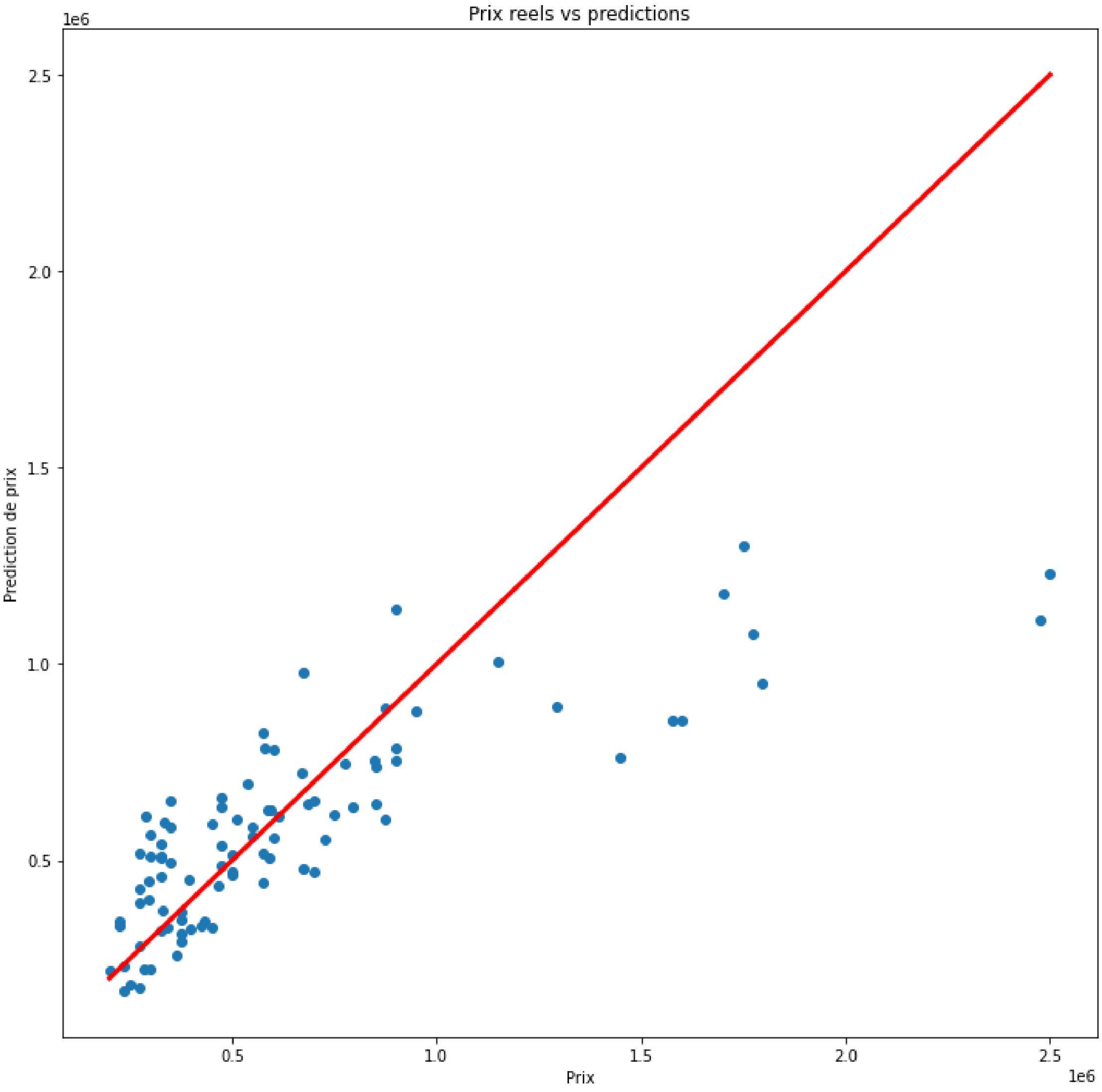
```
from sklearn.linear_model import LinearRegression
lm = LinearRegression()
lm.fit(X_train, y_train)
y_pred = lm.predict(X_test)
```

In [70]:

```
plt.figure(figsize=(12,12))
plt.scatter(y_test, y_pred)
plt.plot([y_test.min(),y_test.max()],[y_test.min(),y_test.max()], color='red', linewidth=3)
plt.xlabel("Prix")
plt.ylabel("Prediction de prix")
plt.title("Prix reels vs predictions")
```

Out[70]:

```
Text(0.5, 1.0, 'Prix reels vs predictions')
```


Prix reels vs predictions
1e6
2.5
2.0
1.5
1.0
0.5
Prediction de prix
0.5
1.0
1.5
2.0
2.5
1e6
Prix

In [71]:

```
lm.score(X_test,y_test)
```

Out[71]:

```
0.5758162339785525
```

In [72]:

```
X = df.drop(['Price','Address','Zip'], axis=1)
y = df.Price
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.1, random_state=1)
```

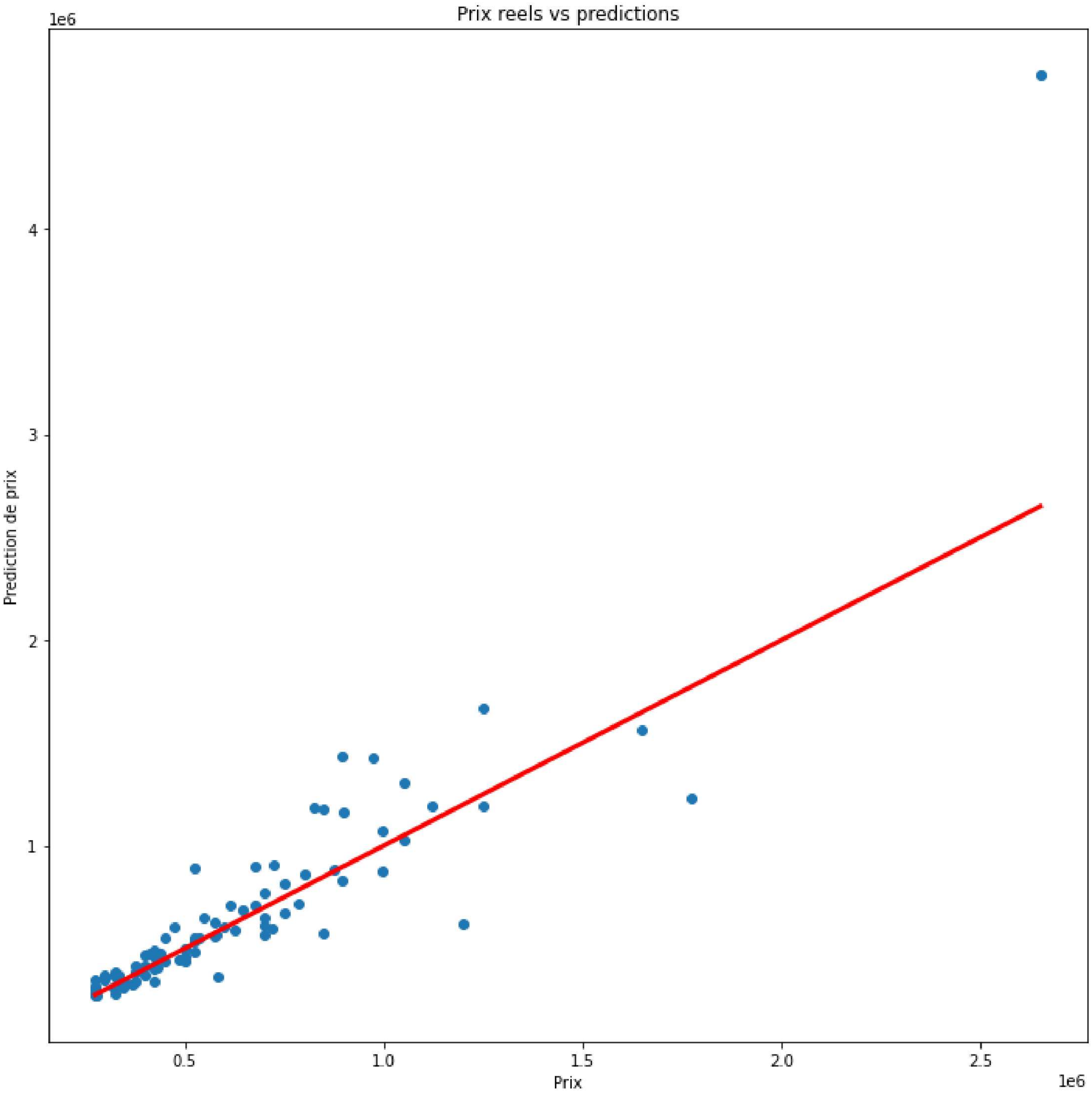
In [73]:

```
from sklearn import ensemble
rf = ensemble.RandomForestRegressor()
rf.fit(X_train, y_train)
y_rf = rf.predict(X_test)
print(rf.score(X_test,y_test))
```

```
0.4655794775058735
```

```
In [74]: plt.figure(figsize=(12,12))
         plt.scatter(y_test, y_rf)
         plt.plot([y_test.min(),y_test.max()],[y_test.min(),y_test.max()], color='red', linewidth=3)
         plt.xlabel("Prix")
         plt.ylabel("Prediction de prix")
         plt.title("Prix reels vs predictions")
```

```
Out[74]: Text(0.5, 1.0, 'Prix reels vs predictions')
```


Prix reels vs predictions
1e6
4
3
2
1
Prediction de prix
0.5
1.0
1.5
2.0
2.5
1e6
Prix

In [ ]: