In [10]:

```
import pandas as pd
import numpy as np
from sklearn import linear_model
import matplotlib.pyplot as plt
```

In [12]:

```
df=pd.read_csv("area and price.csv")
df
```

Out[12]:

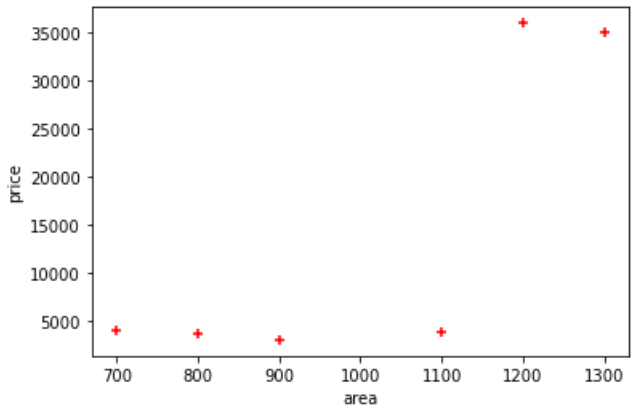
	area	price
0	1300	35000
1	1200	36000
2	1100	3750
3	900	3000
4	800	3600
5	700	4000

In [13]:

```
%matplotlib inline
plt.xlabel('area')
plt.ylabel('price')
plt.scatter(df.area,df.price,color='red',marker='+')
```

Out[13]:

<matplotlib.collections.PathCollection at 0x1b8466a06a0>



In [16]:

```
new_df= df.drop('price',axis='columns')
new_df
```

Out[16]:

	area
0	1300
1	1200
2	1100
3	900
4	800
5	700

In [26]:

```
model=linear_model.LinearRegression()
model.fit(new_df,df.price)
# Training the model
```

Out[26]:

LinearRegression()

In [18]:

```
model.predict([[4300]])
# Here I am predicting the price of a particular area, the area and prices  are not mentioned in the table.
# By using the linear regression, and especially by using the equation of it, it is possible to predict the prices of any area.
```

Out[18]:

array([201087.5])

In [13]:

```
model.predict([[6000]])
# Here I am predicting the price of a particular area, the area and prices  are not mentioned in the table.
# By using the linear regression, and especially by using the equation of it, it is possible to predict the prices of any area.
```

Out[13]:

array([297350.])

In [18]:

```
model.coef_
```

Out[18]:

array([56.625])

In [19]:

```
model.intercept_
```

Out[19]:

-42400.00000000001

In [21]:

```
# y= m*x + b
# y is price
# x is area
56.625*6000 + (-42400.00000000001)
```

Out[21]:

297350.0

In [27]:

```
area_df = pd.read_csv("area.csv")
area_df
```

Out[27]:

	area
0	1300
1	1200
2	1100
3	900
4	800
5	700
6	600

	area
7	500

In [28]:

```
model.predict(area_df)
```

Out[28]:

```
array([ 31212.5,  25550. ,  19887.5,   8562.5,   2900. ,  -2762.5,
        -8425. , -14087.5])
```

In [29]:

```
p=model.predict(area_df)
```

In [31]:

```
area_df['prices']=p
area_df
```

Out[31]:

	area	prices
0	1300	31212.5
1	1200	25550.0
2	1100	19887.5
3	900	8562.5
4	800	2900.0
5	700	-2762.5
6	600	-8425.0
7	500	-14087.5

In [32]:

```
area_df.to_csv("pridictions.csv")
```