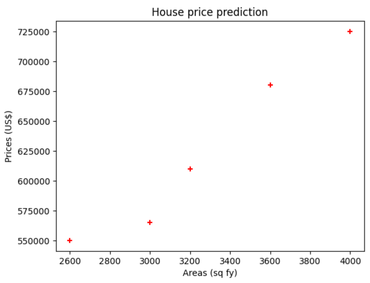
In [69]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn import linear_model
```

In [70]:
```
data = pd.read_csv('homeprices.csv')
data
```

Out[70]:

	area	price
0	2600	550000
1	3000	565000
2	3200	610000
3	3600	680000
4	4000	725000

In [71]:
```
%matplotlib inline
plt.scatter(data.area, data.price, marker="+", color="red")
plt.title("House price prediction")
plt.xlabel("Areas (sq fy)")
plt.ylabel("Prices (US$)")
plt.show()
```



In [72]:
```
model = linear_model.LinearRegression()
model.fit(data[['area']], data[['price']])
```

Out[72]:
```
LinearRegression()
```

In [73]:
```
model.predict([[3300]])
```

```
C:\Users\User\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\base.py:465: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
  warnings.warn(
```

Out[73]:
```
array([[628715.75342466]])
```

m is coefficient

In [74]:
```
model.coef_
```

Out[74]:
```
array([[135.78767123]])
```

b is intercept

In [75]:
```
model.intercept_
```

Out[75]:
```
array([180616.43835616])
```

Below is how it calculate and predict the price

In [76]:
```
# y = m*x+b
135.78767123*3300+180616.43835616
```

Out[76]:
```
628715.75341516
```

Now, lets predict the prices of given area from another CSV file

In [77]:
```
areas = pd.read_csv('areas.csv')
areas
```

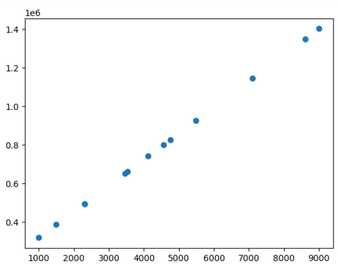
Out[77]:

	area
0	1000
1	1500
2	2300
3	3540
4	4120
5	4560
6	5490
7	3460
8	4750
9	2300
10	9000
11	8600
12	7100

In [78]:
```
predict = model.predict(areas)
predict
```

Out[78]:
```
array([[ 316404.10958904],
       [ 384297.94520548],
       [ 492928.08219178],
       [ 661304.79452055],
       [ 740061.64383562],
       [ 799808.21917808],
       [ 926090.75342466],
       [ 650441.78082192],
       [ 825607.87671233],
       [ 492928.08219178],
       [1402705.47945205],
       [1348390.4109589 ],
       [1144708.90410959]])
```

In [79]:
```
plt.scatter(areas, predict)
plt.show()
```



Adding the new price column to our areas.csv data

In [80]:
```
areas['prices'] = predict
```

In [81]:
```
areas
```

Out[81]:

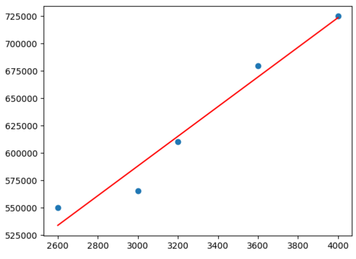
	area	prices
0	1000	3.164041e+05
1	1500	3.842979e+05
2	2300	4.929281e+05
3	3540	6.613048e+05
4	4120	7.400616e+05
5	4560	7.998082e+05
6	5490	9.260908e+05
7	3460	6.504418e+05
8	4750	8.256079e+05
9	2300	4.929281e+05
10	9000	1.402705e+06
11	8600	1.348390e+06
12	7100	1.144709e+06

Now, you can just export the new data

In [84]:
```
areas.to_csv('areas_predicted_price.csv', index=False)
```

Plotting the data with the best fit line

In [99]:
```
plt.scatter(data[['area']], data[['price']])
# Below is how you plot the best fit line after prediction
plt.plot(data[['area']], model.predict(data[['area']]), color='red')
plt.show()
```


725000
700000
675000
650000
625000
600000
575000
550000
525000
2600
2800
3000
3200
3400
3600
3800
4000

In [ ]: