In [1]:

```
import pandas as pd
```

In [2]:

```
df=pd.read_csv("Salary_Data.csv")
```

In [3]:

```
df
```

Out[3]:

	YearsExperience	Salary
0	1.1	39343.0
1	1.3	46205.0
2	1.5	37731.0
3	2.0	43525.0
4	2.2	39891.0
5	2.9	56642.0
6	3.0	60150.0
7	3.2	54445.0
8	3.2	64445.0
9	3.7	57189.0
10	3.9	63218.0
11	4.0	55794.0
12	4.0	56957.0
13	4.1	57081.0
14	4.5	61111.0
15	4.9	67938.0
16	5.1	66029.0
17	5.3	83088.0
18	5.9	81363.0
19	6.0	93940.0
20	6.8	91738.0
21	7.1	98273.0
22	7.9	101302.0
23	8.2	113812.0
24	8.7	109431.0
25	9.0	105582.0
26	9.5	116969.0
27	9.6	112635.0
28	10.3	122391.0
29	10.5	121872.0

In [4]:

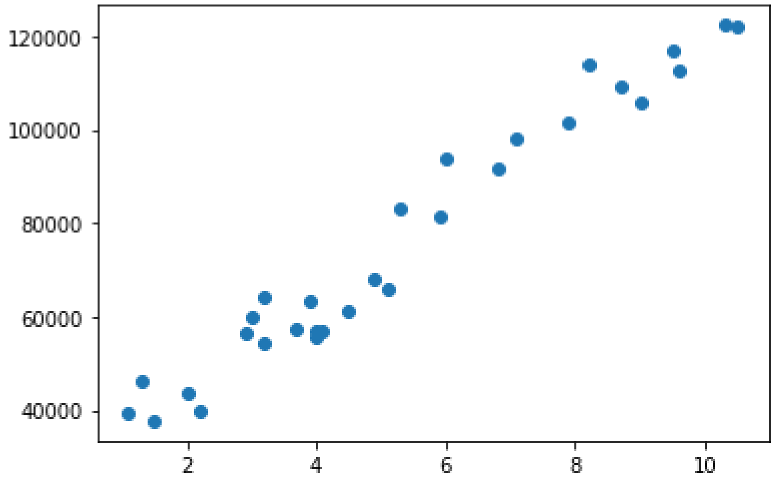
```
import matplotlib.pyplot as plt
```

In [5]:

```
plt.scatter(df.YearsExperience,df.Salary)
```

Out[5]:

```
<matplotlib.collections.PathCollection at 0x1786bcd4308>
```



In [6]:

```
#1.
X=df[['YearsExperience']]
y=df.Salary
```

In [7]:

```
X
```

Out[7]:

	YearsExperience
0	1.1
1	1.3
2	1.5
3	2.0
4	2.2
5	2.9
6	3.0
7	3.2
8	3.2
9	3.7
10	3.9
11	4.0
12	4.0
13	4.1
14	4.5
15	4.9
16	5.1
17	5.3
18	5.9
19	6.0
20	6.8
21	7.1
22	7.9
23	8.2
24	8.7
25	9.0
26	9.5
27	9.6
28	10.3
29	10.5

In [8]:

```
from sklearn.linear_model import LinearRegression
model=LinearRegression()
```

In [9]:

```
model.fit(X,y)
```

Out[9]:

```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=Fal
se)
```

In [10]:

```
model.score(X,y)
```

Out[10]:

```
0.9569566641435084
```

In [11]:

```
pre=model.predict(X)
```

In [12]:

```
y
```

Out[12]:

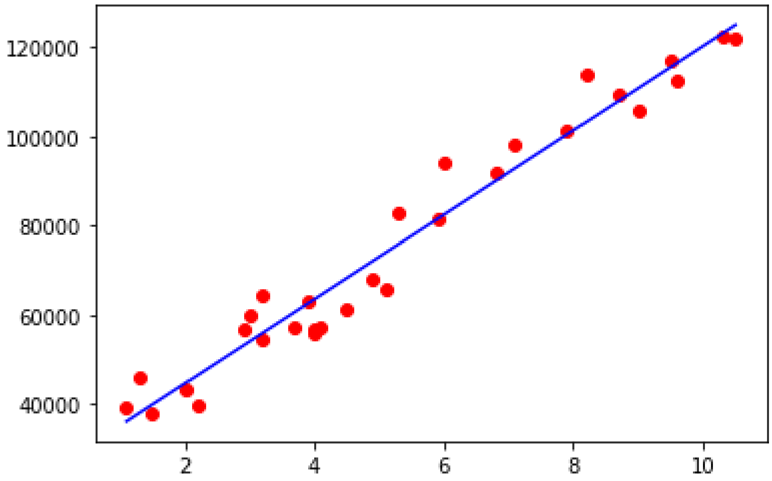
```
0      39343.0
1      46205.0
2      37731.0
3      43525.0
4      39891.0
5      56642.0
6      60150.0
7      54445.0
8      64445.0
9      57189.0
10     63218.0
11     55794.0
12     56957.0
13     57081.0
14     61111.0
15     67938.0
16     66029.0
17     83088.0
18     81363.0
19     93940.0
20     91738.0
21     98273.0
22    101302.0
23    113812.0
24    109431.0
25    105582.0
26    116969.0
27    112635.0
28    122391.0
29    121872.0
Name: Salary, dtype: float64
```

In [13]:

```
plt.scatter(X,y,c='r')
plt.plot(X,pre,c='b')
```

Out[13]:

```
[<matplotlib.lines.Line2D at 0x1786e036048>]
```



In [14]:

```
model.predict([[20]])
```

Out[14]:

```
array([214791.44662777])
```

In [15]:

```
model.intercept_
```

Out[15]:

```
25792.200198668717
```

In [16]:

```
model.coef_
```

Out[16]:

```
array([9449.96232146])
```

In [19]:

```
df=pd.read_csv("position_salaries.csv")
```

In [20]:

```
df
```

Out[20]:

	Position	Level	Salary
0	Business Analyst	1	45000
1	Junior Consultant	2	50000
2	Senior Consultant	3	60000
3	Manager	4	80000
4	Country Manager	5	110000
5	Region Manager	6	150000
6	Partner	7	200000
7	Senior Partner	8	300000
8	C-level	9	500000
9	CEO	10	1000000

In [21]:

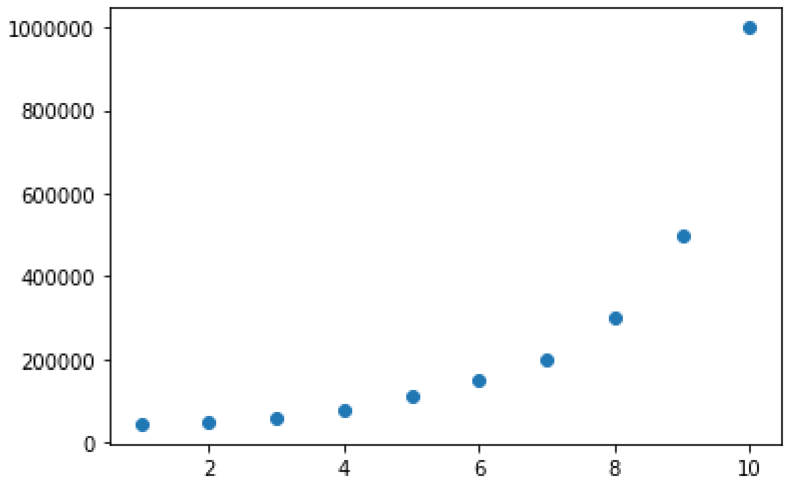
```
X=df[['Level']]
y=df.Salary
```

In [22]:

```
plt.scatter(X,y)
```

Out[22]:

```
<matplotlib.collections.PathCollection at 0x1786f628d48>
```



In [23]:

```
model2=LinearRegression()
```

In [24]:

```
model2.fit(X,y)
```

Out[24]:

```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=Fal
se)
```

In [26]:

```
model2.score(X,y)
```

Out[26]:

```
0.6690412331929895
```

In [27]:

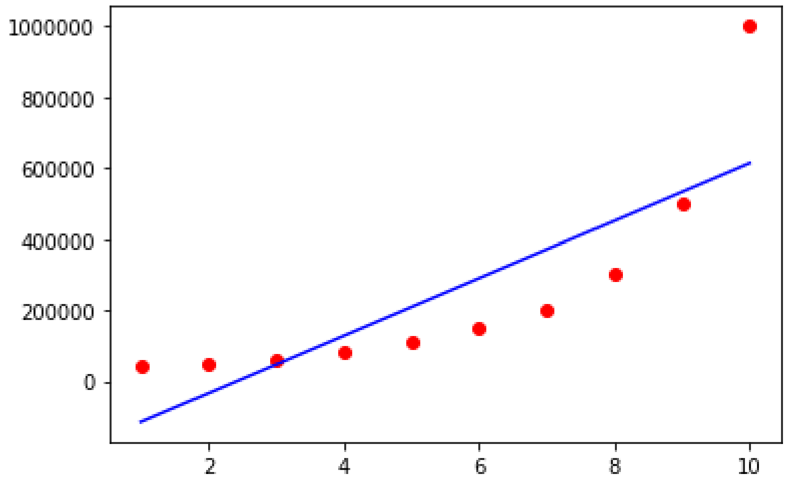
```
pre=model2.predict(X)
```

In [28]:

```
plt.scatter(X,y,c='red')
plt.plot(X,pre,c='b')
```

Out[28]:

```
[<matplotlib.lines.Line2D at 0x1786f826888>]
```



In [29]:

```
from sklearn.preprocessing import PolynomialFeatures
```

In [34]:

```
poly=PolynomialFeatures(degree=3)
```

In [36]:

```
X_poly=poly.fit_transform(X)
```

In [37]:

```
model3=LinearRegression()
```

In [38]:

```
model3.fit(X_poly,y)
```

Out[38]:

```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=Fal
se)
```

In [39]:

```
model3.score(X_poly,y)
```

Out[39]:

```
0.9812097727913366
```

In [41]:

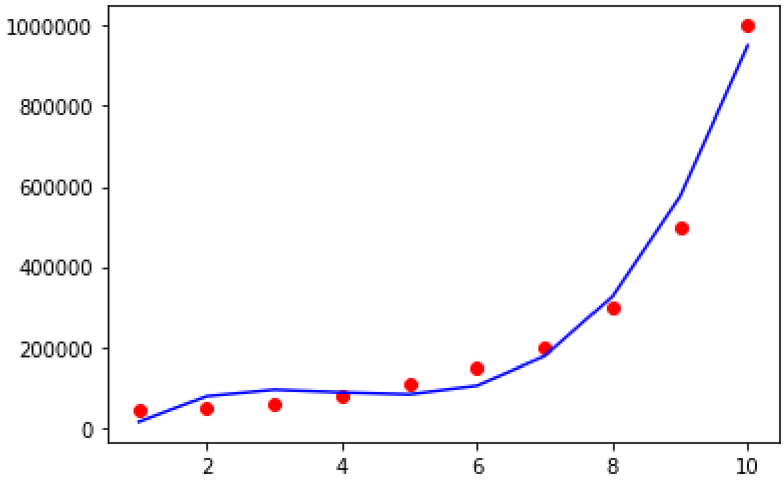
```
pre3=model3.predict(X_poly)
```

In [42]:

```
plt.scatter(X,y,c='red')
plt.plot(X,pre3,c='b')
```

Out[42]:

```
[<matplotlib.lines.Line2D at 0x17870198308>]
```



In [44]:

```
model3.coef_
```

Out[44]:

```
array([     0.        , 180664.33566437, -48548.95104896,   4120.04662005])
```

In [45]:

```
model3.intercept_
```

Out[45]:

```
-121333.33333338285
```

In [46]:

```
data=pd.read_csv('USA_Housing.csv')
```

In [47]:

```
data
```

Out[47]:

	Avg. Area Income	Avg. Area House Age	Avg. Area Number of Rooms	Avg. Area Number of Bedrooms	Area Population	Price	A
0	79545.458574	5.682861	7.009188	4.09	23086.800503	1.059034e+06	208 Michael Fe 674\nLaurab
1	79248.642455	6.002900	6.730821	3.09	40173.072174	1.505891e+06	188 Johnso Suite 07! Kathlee
2	61287.067179	5.865890	8.512727	5.13	36882.159400	1.058988e+06	9127 E Stravenue\nDan WI (
3	63345.240046	7.188236	5.586729	3.26	34310.242831	1.260617e+06	USS Barnett\nF
4	59982.197226	5.040555	7.839388	4.23	26354.109472	6.309435e+05	USNS Raymon AE
...	...	...	...	...	...	...	
4995	60567.944140	7.830362	6.137356	3.46	22837.361035	1.060194e+06	USNS William AP 301!
4996	78491.275435	6.999135	6.576763	4.02	25616.115489	1.482618e+06	PSC 92 8489\nAPO AA
4997	63390.686886	7.250591	4.805081	2.13	33266.145490	1.030730e+06	4215 Tracy Suite 076\nJosh
4998	68001.331235	5.534388	7.130144	5.44	42625.620156	1.198657e+06	USS Wallace\nF
4999	65510.581804	5.992305	6.792336	4.07	46501.283803	1.298950e+06	37778 George Apt. 509\nEa

5000 rows × 7 columns

In [48]:

```
x=data.drop(['Address','Price'],axis=1)
```

In [49]:

```
y=data.Price
```

In [51]:

```
from sklearn.model_selection import train_test_split
```

In [78]:

```
x_train,x_test,y_train,y_test=train_test_split(x,y)
```

In [79]:

```
MLR=LinearRegression()
```

In [80]:

```
MLR.fit(x_train,y_train)
```

Out[80]:

```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=Fal
se)
```

In [81]:

```
MLR.score(x_test,y_test)
```

Out[81]:

```
0.9176528918114587
```

In [82]:

```
data.corr()
```

Out[82]:

	Avg. Area Income	Avg. Area House Age	Avg. Area Number of Rooms	Avg. Area Number of Bedrooms	Area Population	Price
Avg. Area Income	1.000000	-0.002007	-0.011032	0.019788	-0.016234	0.639734
Avg. Area House Age	-0.002007	1.000000	-0.009428	0.006149	-0.018743	0.452543
Avg. Area Number of Rooms	-0.011032	-0.009428	1.000000	0.462695	0.002040	0.335664
Avg. Area Number of Bedrooms	0.019788	0.006149	0.462695	1.000000	-0.022168	0.171071
Area Population	-0.016234	-0.018743	0.002040	-0.022168	1.000000	0.408556
Price	0.639734	0.452543	0.335664	0.171071	0.408556	1.000000

In [ ]:

```
```