

```
In [1]: import numpy as np  
import pandas as pd
```

```
In [2]: dataset=pd.read_csv(r"D:\ML_Course\Works_on_python\Linear_Regression\data.csv")
```

```
In [3]: dataset.head()
```

Out[3]:

	sno	Temperature	Pressure
0	1	0	0.0002
1	2	20	0.0012
2	3	40	0.0060
3	4	60	0.0300
4	5	80	0.0900

```
In [4]: dataset.isnull().any()
```

Out[4]:

sno	False
Temperature	False
Pressure	False
dtype:	bool

```
In [5]: x=dataset.iloc[:,1:2].values  
y=dataset.iloc[:,2:3].values
```

```
In [6]: from sklearn.model_selection import train_test_split  
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=0)
```

```
In [7]: from sklearn.preprocessing import PolynomialFeatures  
poly_reg=PolynomialFeatures(degree=4)  
x_poly=poly_reg.fit_transform(x)
```

```
In [8]: x_poly
```

```
Out[8]: array([[1.000e+00, 0.000e+00, 0.000e+00, 0.000e+00, 0.000e+00],
 [1.000e+00, 2.000e+01, 4.000e+02, 8.000e+03, 1.600e+05],
 [1.000e+00, 4.000e+01, 1.600e+03, 6.400e+04, 2.560e+06],
 [1.000e+00, 6.000e+01, 3.600e+03, 2.160e+05, 1.296e+07],
 [1.000e+00, 8.000e+01, 6.400e+03, 5.120e+05, 4.096e+07],
 [1.000e+00, 1.000e+02, 1.000e+04, 1.000e+06, 1.000e+08]])
```

```
In [9]: poly_reg.fit(x_poly,y)
```

```
Out[9]: PolynomialFeatures(degree=4, include_bias=True, interaction_only=False)
```

```
In [12]: from sklearn.linear_model import LinearRegression
lin_reg=LinearRegression()
lin_reg.fit(x_poly,y)
```

```
Out[12]: LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None,
 normalize=False)
```

```
In [16]: poly=lin_reg.predict(x_poly)
```

```
In [17]: y
```

```
Out[17]: array([[2.0e-04],
 [1.2e-03],
 [6.0e-03],
 [3.0e-02],
 [9.0e-02],
 [2.7e-01]])
```

```
In [16]: poly
```

```
Out[16]: array([[ 4.6111118e-04],
 [-1.05555553e-04],
 [ 8.6111111e-03],
 [ 2.73888889e-02],
 [ 9.13055556e-02],
 [ 2.69738889e-01]])
```

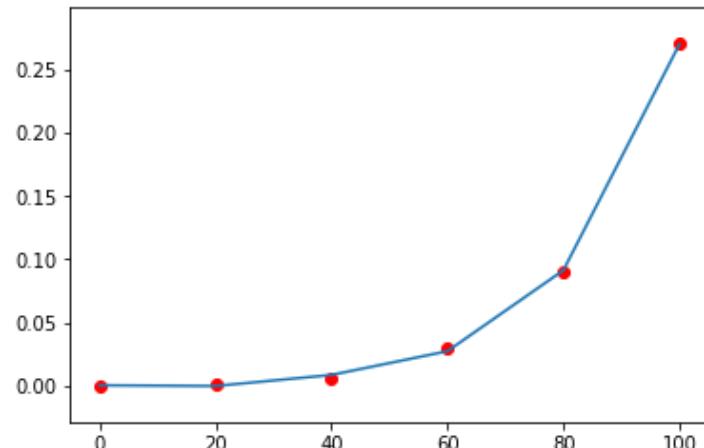
```
In [22]: from sklearn.metrics import r2_score  
accuracy=r2_score(y,poly)
```

```
In [23]: accuracy
```

```
Out[23]: 0.9996910780718014
```

```
In [18]: import matplotlib.pyplot as plt  
plt.scatter(x,y,color="red")  
plt.plot(x,poly)
```

```
Out[18]: [<matplotlib.lines.Line2D at 0x26496f374a8>]
```



```
In [ ]:
```