PRM - Polynomial Regression Model Algorithm

In [2]:
```
#importing libraries

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

In [3]:
```
#importing  the dataset

dataset = pd.read_csv(r"C:\Users\Jan Saida\Downloads\emp_sal.csv")
dataset
```

Out[3]:

	Position	Level	Salary
0	Jr Software Engineer	1	45000
1	Sr Software Engineer	2	50000
2	Team Lead	3	60000
3	Manager	4	80000
4	Sr manager	5	110000
5	Region Manager	6	150000
6	AVP	7	200000
7	VP	8	300000
8	CTO	9	500000
9	CEO	10	1000000

In [4]:
```
x=dataset.iloc[:, 1:2].values  #independent varianble
y=dataset.iloc[:,2].values     #dependent varianble
```

In [5]:
```
x
```

Out[5]:
```
array([[ 1],
       [ 2],
       [ 3],
       [ 4],
       [ 5],
       [ 6],
       [ 7],
       [ 8],
       [ 9],
       [10]], dtype=int64)
```

In [6]:
```
y
```

Out[6]:
```
array([  45000,   50000,   60000,   80000,  110000,  150000,  200000,
        300000,  500000, 1000000], dtype=int64)
```

In [7]:
```
#polynomial  regression model (bydefault degree - 2)

from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import PolynomialFeatures
poly_reg=PolynomialFeatures(degree=6)
X_poly=poly_reg.fit_transform(x)
```

In [8]:
```
poly_reg
```

Out[8]:
```
PolynomialFeatures(degree=6)
```

In [9]:
```
X_poly
```

Out[9]:
```
array([[1.00000e+00, 1.00000e+00, 1.00000e+00, 1.00000e+00, 1.00000e+00,
        1.00000e+00, 1.00000e+00],
       [1.00000e+00, 2.00000e+00, 4.00000e+00, 8.00000e+00, 1.60000e+01,
        3.20000e+01, 6.40000e+01],
       [1.00000e+00, 3.00000e+00, 9.00000e+00, 2.70000e+01, 8.10000e+01,
        2.43000e+02, 7.29000e+02],
       [1.00000e+00, 4.00000e+00, 1.60000e+01, 6.40000e+01, 2.56000e+02,
        1.02400e+03, 4.09600e+03],
       [1.00000e+00, 5.00000e+00, 2.50000e+01, 1.25000e+02, 6.25000e+02,
        3.12500e+03, 1.56250e+04],
       [1.00000e+00, 6.00000e+00, 3.60000e+01, 2.16000e+02, 1.29600e+03,
        7.77600e+03, 4.66560e+04],
       [1.00000e+00, 7.00000e+00, 4.90000e+01, 3.43000e+02, 2.40100e+03,
        1.68070e+04, 1.17649e+05],
       [1.00000e+00, 8.00000e+00, 6.40000e+01, 5.12000e+02, 4.09600e+03,
        3.27680e+04, 2.62144e+05],
       [1.00000e+00, 9.00000e+00, 8.10000e+01, 7.29000e+02, 6.56100e+03,
        5.90490e+04, 5.31441e+05],
       [1.00000e+00, 1.00000e+01, 1.00000e+02, 1.00000e+03, 1.00000e+04,
        1.00000e+05, 1.00000e+06]])
```

In [10]:
```
poly_reg.fit(X_poly,y)
```

Out[10]:
```
PolynomialFeatures(degree=6)
```

In [11]:
```
lin_reg_2=LinearRegression()
```

In [12]:
```
lin_reg_2.fit(X_poly,y)
```

Out[12]:
```
LinearRegression()
```

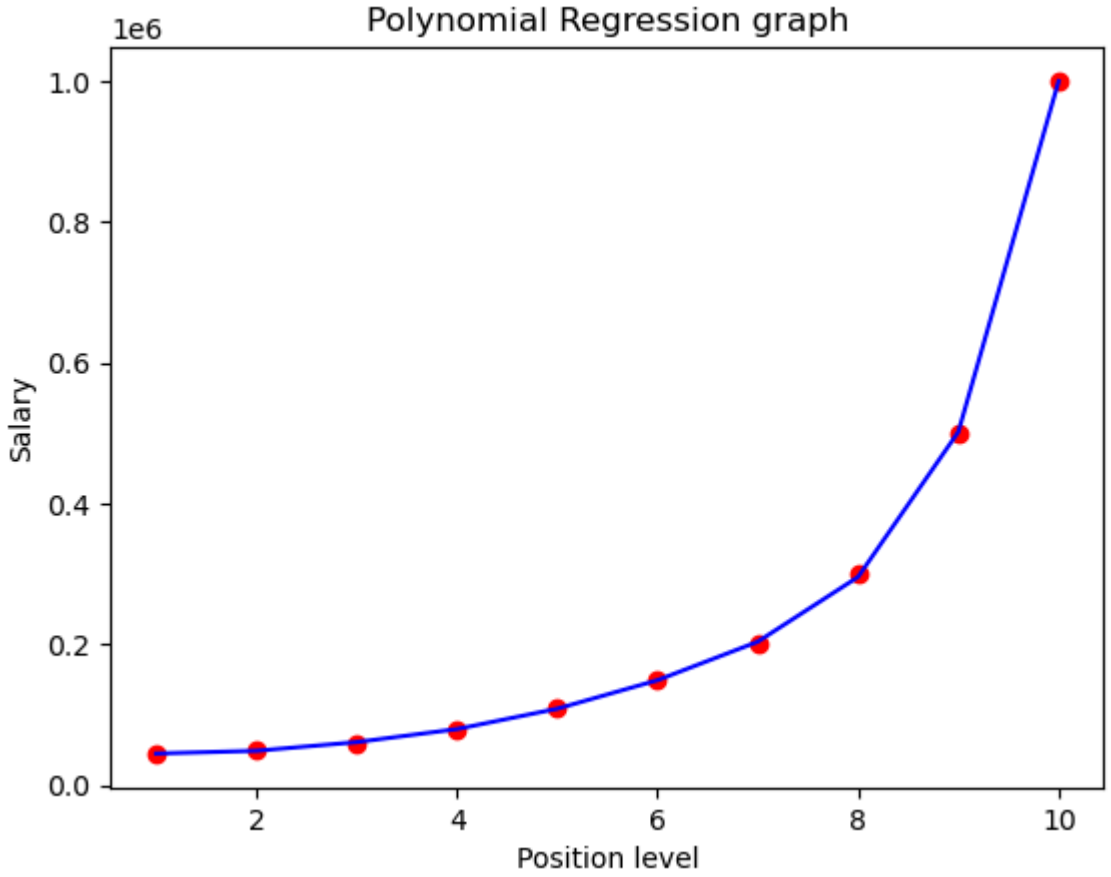
In [13]:
```
#Polynomial regression Predictions
```

```
poly_model_pred=lin_reg_2.predict(poly_reg.fit_transform([[6.5]]))
poly_model_pred
```

Out[13]:
```
array([174192.81930661])
```

In [14]:
```
#polynomial visualisation

plt.scatter(x,y,color='red')
plt.plot(x,lin_reg_2.predict(poly_reg.fit_transform(x)),color='blue')
plt.title('Polynomial Regression graph')
plt.xlabel('Position level')
plt.ylabel('Salary')
plt.show()
```


Polynomial Regression graph
1e6
1.0
0.8
0.6
0.4
0.2
0.0
Salary
2
4
6
8
10
Position level

In [ ]: