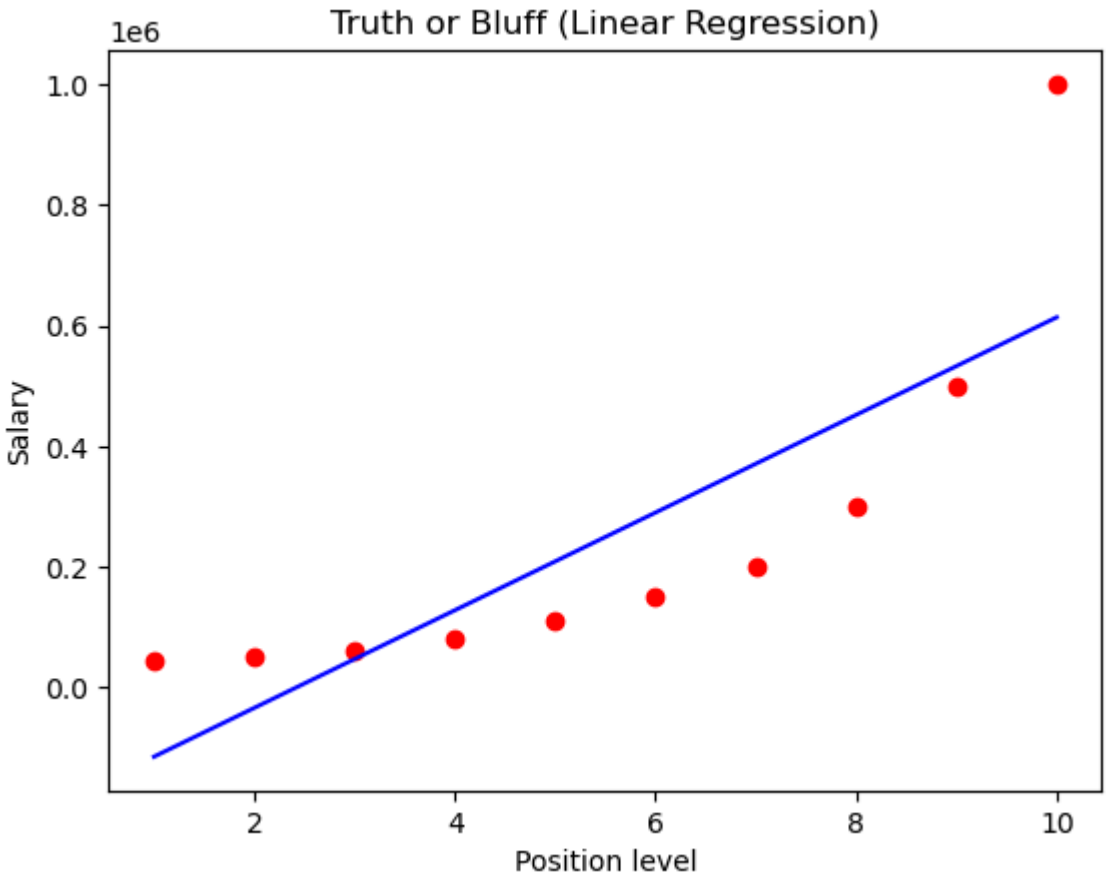
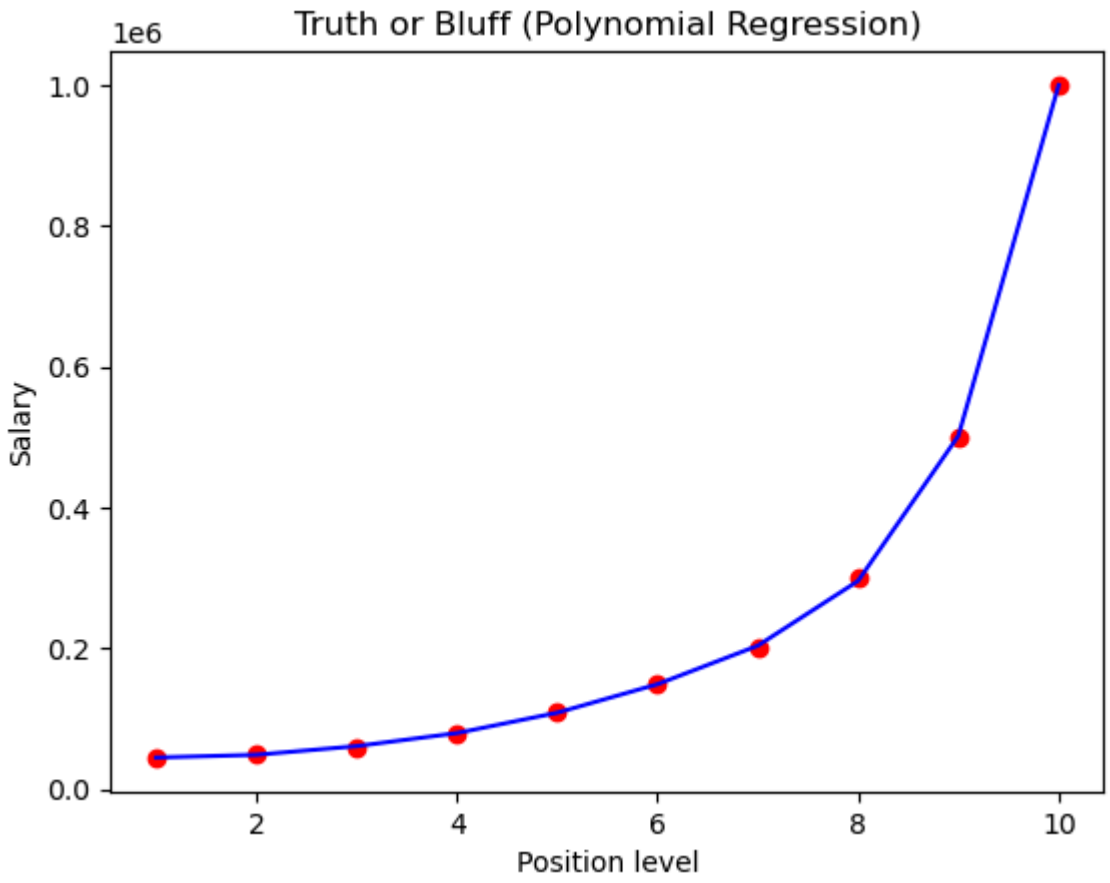
In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

dataset = pd.read_csv(r"C:\Users\soham\OneDrive\Desktop\(OCT 02)\1.POLYNOMIAL RE

X = dataset.iloc[:, 1:2].values
y = dataset.iloc[:, 2].values


# svm model
from sklearn.svm import SVR
svr_regressor = SVR(kernel='poly',degree = 5,gamma = 'scale' )
svr_regressor.fit(X,y)

svr_model_pred = svr_regressor.predict([[6.5]])
print(svr_model_pred)

# knn model
from sklearn.neighbors import KNeighborsRegressor
knn_reg_model = KNeighborsRegressor(n_neighbors=5, weights='distance', p=2)
knn_reg_model.fit(X,y)

knn_reg_pred = knn_reg_model.predict([[6.5]])
print(knn_reg_pred)
```

```
[164079.01344549]
[175348.8372093]
```

In [3]:
```
import numpy as np

import matplotlib.pyplot as plt

import pandas as pd


dataset = pd.read_csv(r"C:\Users\soham\OneDrive\Desktop\(OCT 02)\1.POLYNOMIAL RE


X = dataset.iloc[:, 1:2].values

y = dataset.iloc[:, 2].values

# linear model  -- linear algor ( degree - 1)
from sklearn.linear_model import LinearRegression
lin_reg = LinearRegression()
lin_reg.fit(X, y)

# polynomial model  ( bydefeaut degree - 2)

from sklearn.preprocessing import PolynomialFeatures
poly_reg = PolynomialFeatures(degree=6)
X_poly = poly_reg.fit_transform(X)

poly_reg.fit(X_poly, y)

lin_reg_2 = LinearRegression()

lin_reg_2.fit(X_poly, y)
```

```
# linear regression visualizaton
plt.scatter(X, y, color = 'red')
plt.plot(X, lin_reg.predict(X), color = 'blue')
plt.title('Truth or Bluff (Linear Regression)')
plt.xlabel('Position level')
plt.ylabel('Salary')
plt.show()


# poly nomial visualization

plt.scatter(X, y, color = 'red')
plt.plot(X, lin_reg_2.predict(poly_reg.fit_transform(X)), color = 'blue')
plt.title('Truth or Bluff (Polynomial Regression)')
plt.xlabel('Position level')
plt.ylabel('Salary')
plt.show()

# predicton

lin_model_pred = lin_reg.predict([[6.5]])
lin_model_pred

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





Out[3]: array([174192.81930584])

In [ ]: