

## Q1)

```
import pandas as pd  
import matplotlib.pyplot as plt  
from sklearn.linear_model import LinearRegression  
from sklearn.preprocessing import PolynomialFeatures  
from sklearn.metrics import r2_score  
  
df = pd.read_csv('Position_Salaries .csv')  
df  
  
X = df[['Level']].values  
y = df['Salary'].values  
  
lin_reg = LinearRegression()  
lin_reg.fit(X, y)  
  
poly_reg = PolynomialFeatures(degree=4)  
X_poly = poly_reg.fit_transform(X)  
lin_reg_poly = LinearRegression()  
lin_reg_poly.fit(X_poly, y)  
  
level_11_salary_linear = lin_reg.predict([[11]])[0]  
level_12_salary_linear = lin_reg.predict([[12]])[0]  
level_11_salary_poly = lin_reg_poly.predict(poly_reg.transform([[11]]))[0]  
level_12_salary_poly = lin_reg_poly.predict(poly_reg.transform([[12]]))[0]
```

```
print("\nPredicted Salaries:")
print("level_11_salary_linear:",level_11_salary_linear)
print("level_12_salary_linear:",level_12_salary_linear)
print("level_11_salary_poly:",level_11_salary_poly)
print("level_12_salary_poly:",level_12_salary_poly)

y_pred_linear = lin_reg.predict(X)
y_pred_poly = lin_reg_poly.predict(X_poly)

print("Linear Regression R2 Score:", r2_score(y, y_pred_linear))
print("Polynomial Regression R2 Score:", r2_score(y, y_pred_poly))

plt.scatter(X, y, color='red', label='Original Data')
plt.plot(X, y_pred_linear, color='blue', label='Linear Regression')
plt.plot(X,y_pred_poly, color='green', label='Polynomial Regression')
plt.title('Position Level vs Salary')
plt.xlabel('Position Level')
plt.ylabel('Salary')
plt.show()
```

## Q2)

```
import pandas as pd

data = {

    'Name': ['Amit', 'Priya', 'Rahul', 'Sneha', 'Vikas'],
    'Age': [25, None, 30, 28, None],
    'City': ['Pune', 'Mumbai', None, 'Delhi', 'Chennai'],
    'Salary': [35000, 40000, None, 38000, 42000]
}
```

```
df = pd.DataFrame(data)
```

```
print("Original Dataset:")
print(df)
```

```
print("\nNull Values in Each Column:")
print(df.isnull().sum())
```

```
df_cleaned = df.dropna()
```

```
print("\nDataset After Removing Null Values:")
print(df_cleaned)
```