PREPROCESSING

1. Importing Libraries

In [1]:
```
import pandas as pd
import numpy as np
from sklearn.impute import SimpleImputer
```

2. Loading the Data

In [2]:
```
dataset=pd.read_csv("crop_yield.csv")
dataset
```

Out[2]:

	Region	Soil_Type	Crop	Rainfall_mm	Temperature_Celsius	Fertilizer_Used	Irrigation_Used	Weather_Condition	Days_to_Harvest	Yield_tons_per_hectare
0	West	Sandy	Cotton	897.077239	27.676966	False	True	Cloudy	122	6.555816
1	South	Clay	Rice	992.673282	18.026142	True	True	Rainy	140	8.527341
2	North	Loam	Barley	147.998025	29.794042	False	False	Sunny	106	1.127443
3	North	Sandy	Soybean	986.866331	16.644190	False	True	Rainy	146	6.517573
4	South	Silt	Wheat	730.379174	31.620687	True	True	Cloudy	110	7.248251
...	...	...	...	...	...	...	...	...	...	...
999995	West	Silt	Rice	302.805345	27.987428	False	False	Sunny	76	1.347586
999996	South	Chalky	Barley	932.991383	39.661039	True	False	Rainy	93	7.311594
999997	North	Peaty	Cotton	867.362046	24.370042	True	False	Cloudy	108	5.763182
999998	West	Silt	Wheat	492.812857	33.045505	False	False	Sunny	102	2.070159
999999	West	Sandy	Maize	180.936180	27.298847	True	False	Sunny	76	2.937243

1000000 rows × 10 columns

3. Checking for Missing Values

In [3]:
```
dataset.isnull().sum()
```

Out[3]:
```
Region                  4
Soil_Type               2
Crop                    3
Rainfall_mm             0
Temperature_Celsius     0
Fertilizer_Used         0
Irrigation_Used         0
Weather_Condition       3
Days_to_Harvest         0
Yield_tons_per_hectare  0
dtype: int64
```

4. Separating Quantitative and Qualitative Columns

In [4]:
```
def quanQual(dataset):
    quan=[]
    qual=[]
    for columnName in dataset.columns:
        if(dataset[columnName].dtype=='O'):
            qual.append(columnName)
        else:
            quan.append(columnName)
    return quan,qual
quan,qual=quanQual(dataset)
```

In [5]:
```
dataset[quan]
```

Out[5]:

	Rainfall_mm	Temperature_Celsius	Fertilizer_Used	Irrigation_Used	Days_to_Harvest	Yield_tons_per_hectare
0	897.077239	27.676966	False	True	122	6.555816
1	992.673282	18.026142	True	True	140	8.527341
2	147.998025	29.794042	False	False	106	1.127443
3	986.866331	16.644190	False	True	146	6.517573
4	730.379174	31.620687	True	True	110	7.248251
...	...	...	...	...	...	...
999995	302.805345	27.987428	False	False	76	1.347586
999996	932.991383	39.661039	True	False	93	7.311594
999997	867.362046	24.370042	True	False	108	5.763182
999998	492.812857	33.045505	False	False	102	2.070159
999999	180.936180	27.298847	True	False	76	2.937243

1000000 rows × 6 columns

5. Imputing Missing Values in Quantitative Columns

In [6]:
```
import numpy as np
from sklearn.impute import SimpleImputer
imp = SimpleImputer(missing_values=np.nan, strategy='mean')
print(quan)
new_dataset1=imp.fit(dataset[quan])
new_dataset1=imp.transform(dataset[quan])
```

```
['Rainfall_mm', 'Temperature_Celsius', 'Fertilizer_Used', 'Irrigation_Used', 'Days_to_Harvest', 'Yield_tons_per_hectare']
```

In [7]:
```
new_dataset1=pd.DataFrame(new_dataset1,columns=quan)
new_dataset1
```

Out[7]:

	Rainfall_mm	Temperature_Celsius	Fertilizer_Used	Irrigation_Used	Days_to_Harvest	Yield_tons_per_hectare
0	897.077239	27.676966	0.0	1.0	122.0	6.555816
1	992.673282	18.026142	1.0	1.0	140.0	8.527341
2	147.998025	29.794042	0.0	0.0	106.0	1.127443
3	986.866331	16.644190	0.0	1.0	146.0	6.517573
4	730.379174	31.620687	1.0	1.0	110.0	7.248251
...	...	...	...	...	...	...
999995	302.805345	27.987428	0.0	0.0	76.0	1.347586
999996	932.991383	39.661039	1.0	0.0	93.0	7.311594
999997	867.362046	24.370042	1.0	0.0	108.0	5.763182
999998	492.812857	33.045505	0.0	0.0	102.0	2.070159
999999	180.936180	27.298847	1.0	0.0	76.0	2.937243

1000000 rows × 6 columns

6. Imputing Missing Values in Qualitative Columns

In [8]:
```
imp2=SimpleImputer(missing_values=np.nan, strategy='most_frequent')
new_dataset2=imp2.fit(dataset[qual])
new_dataset2=imp2.transform(dataset[qual])
```

In [9]:
```
new_dataset2=pd.DataFrame(new_dataset2,columns=qual)
new_dataset2
```

Out[9]:

	Region	Soil_Type	Crop	Weather_Condition
0	West	Sandy	Cotton	Cloudy
1	South	Clay	Rice	Rainy
2	North	Loam	Barley	Sunny
3	North	Sandy	Soybean	Rainy
4	South	Silt	Wheat	Cloudy
...	...	...	...	...
999995	West	Silt	Rice	Sunny
999996	South	Chalky	Barley	Rainy
999997	North	Peaty	Cotton	Cloudy
999998	West	Silt	Wheat	Sunny
999999	West	Sandy	Maize	Sunny

1000000 rows × 4 columns

7. Combining the Imputed DataFrames

In [10]:
```
combined_data=pd.concat([new_dataset1,new_dataset2],axis=1)
combined_data
```

Out[10]:

	Rainfall_mm	Temperature_Celsius	Fertilizer_Used	Irrigation_Used	Days_to_Harvest	Yield_tons_per_hectare	Region	Soil_Type	Crop	Weather_Condition
0	897.077239	27.676966	0.0	1.0	122.0	6.555816	West	Sandy	Cotton	Cloudy
1	992.673282	18.026142	1.0	1.0	140.0	8.527341	South	Clay	Rice	Rainy
2	147.998025	29.794042	0.0	0.0	106.0	1.127443	North	Loam	Barley	Sunny
3	986.866331	16.644190	0.0	1.0	146.0	6.517573	North	Sandy	Soybean	Rainy
4	730.379174	31.620687	1.0	1.0	110.0	7.248251	South	Silt	Wheat	Cloudy
...	...	...	...	...	...	...	...	...	...	...
999995	302.805345	27.987428	0.0	0.0	76.0	1.347586	West	Silt	Rice	Sunny
999996	932.991383	39.661039	1.0	0.0	93.0	7.311594	South	Chalky	Barley	Rainy
999997	867.362046	24.370042	1.0	0.0	108.0	5.763182	North	Peaty	Cotton	Cloudy
999998	492.812857	33.045505	0.0	0.0	102.0	2.070159	West	Silt	Wheat	Sunny
999999	180.936180	27.298847	1.0	0.0	76.0	2.937243	West	Sandy	Maize	Sunny

1000000 rows × 10 columns

8. Attempting to Fill Missing Values

In [11]:
```
dataset.fillna(combined_data,inplace=True)
```

In [12]:
```
dataset.isnull().sum()
```

Out[12]:
```
Region                  0
Soil_Type               0
Crop                    0
Rainfall_mm             0
Temperature_Celsius     0
Fertilizer_Used         0
Irrigation_Used         0
Weather_Condition       0
Days_to_Harvest         0
```

```
Yield_tons_per_hectare    0
dtype: int64
```

In [ ]: