Normization and Standarization Scaler

```
# 1- import library

import numpy as np # Arrary

import matplotlib.pyplot as plt

import pandas as pd

# 2- import dataset

dataset = pd.read_csv(r"D:\Samsom - All Data\Naresh IT Institute\Data.csv")

# 3- split the data into x & y

x = dataset.iloc[:, :-1].values

y = dataset.iloc[:,3].values

# 4- transformer to fill missing value number, category, transform cate - number

from sklearn.impute import SimpleImputer # SPYDER 4

imputer = SimpleImputer()

imputer = imputer.fit(x[:,1:3])

x[:, 1:3] = imputer.transform(x[:,1:3])
```

```
# How to encode categorical data & create a dummpy variable
from sklearn.preprocessing import LabelEncoder

labelencoder_x = LabelEncoder()

labelencoder_x.fit_transform(x[:,0])

x[:,0] = labelencoder_x.fit_transform(x[:,0])

# transformer using for dv
labelencoder_y = LabelEncoder()

y = labelencoder_y.fit_transform(y)

#-----------------------------------------------------------------------------

# Spliting the dataset in training set & testing set
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)

# if you remove random_stat then your model not behave as accurate

#Feature scaling

from sklearn.preprocessing import Normalizer
```

```
sc_x = Normalizer()

x_train = sc_x.fit_transform(x_train)

x_test = sc_x.transform(x_test)

# Feature Standardscaler
from sklearn.preprocessing import StandardScaler

sc_x = StandardScaler()

x_train = sc_x.fit_transform(x_train)

x_test = sc_x.transform(x_test)
```