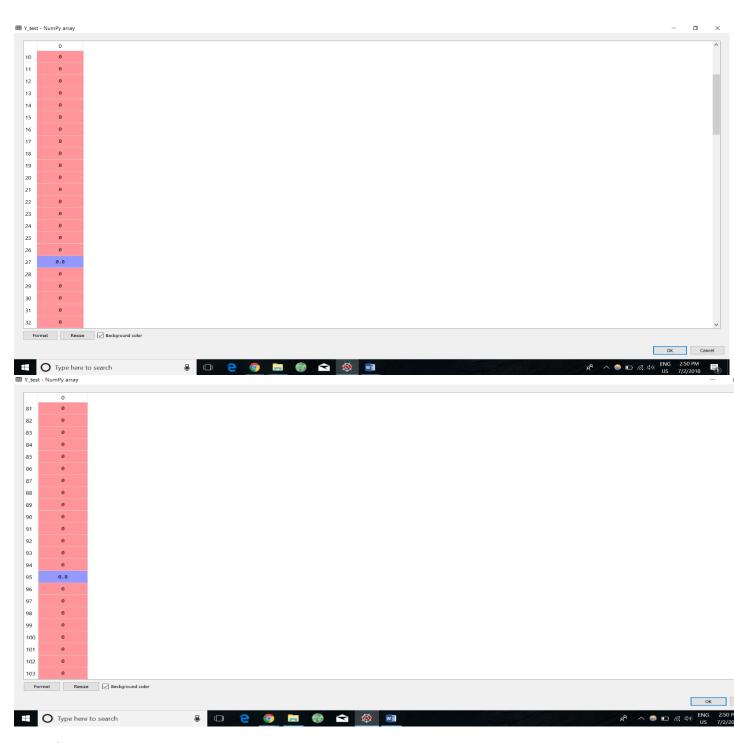
```
# -*- coding: utf-8 -*-
Created on Thu Jun 28 21:57:49 2018
@author: shubham b thorat
Rain prediction
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
dataset=pd.read_csv('rain_prediction.csv')
X = dataset.iloc[:,:-1].values
Y = dataset.iloc[:,10].values
"""from sklearn.preprocessing import Imputer
imputer = Imputer(missing values="NaN", strategy="mean",axis=0)
imputer = imputer.fit(X[:,2:11], y=None)
X[:,2:11] = imputer.transform(X[:,2:11])"""
from sklearn.preprocessing import LabelEncoder
label_encoder_X = LabelEncoder()
label_encoder_Y = LabelEncoder()
X[:,0] = label_encoder_X.fit_transform(X[:,0])
X[:,1] = label encoder X.fit transform(X[:,1])
```

```
"""onehotencoder = OneHotEncoder(categorical_features= [0])
X = onehotencoder.fit_transform(X).toarray()
onehotencoder = OneHotEncoder(categorical features= [1])
X = onehotencoder.fit transform(X).toarray()"""
from sklearn.model selection import train test split
X_train, X_test, Y_train, Y_test = train_test_split(X,Y, test_size = .2, random_state = 0)
# feature scaling
#from sklearn.preprocessing import StandardScaler
#sc_X = StandardScaler()
#X train = sc X.fit transform(X train)
#X_test = sc_X.fit_transform(X_test)
from sklearn.linear model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train,Y_train)
#predicting the test set result
y_pred = regressor.predict(X_test)
y_test =
```



Y\_pred =

