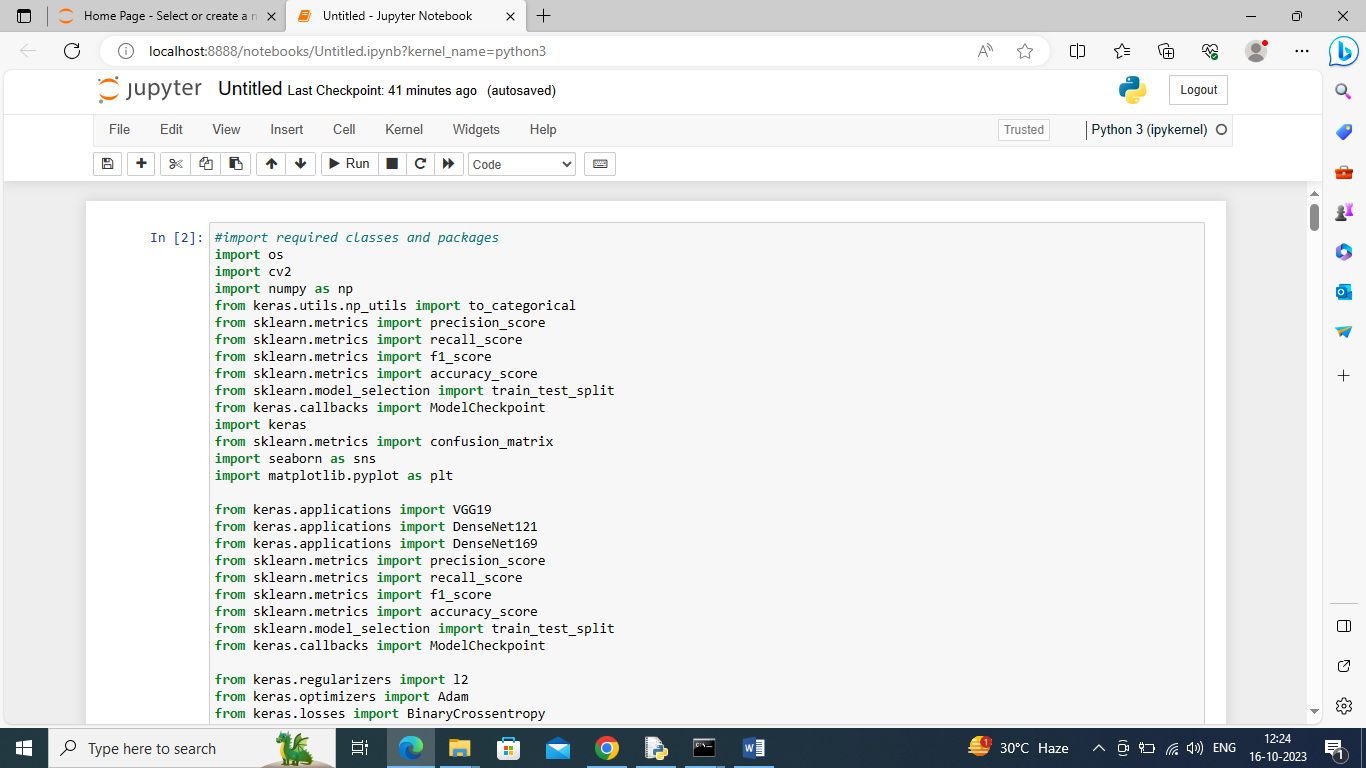
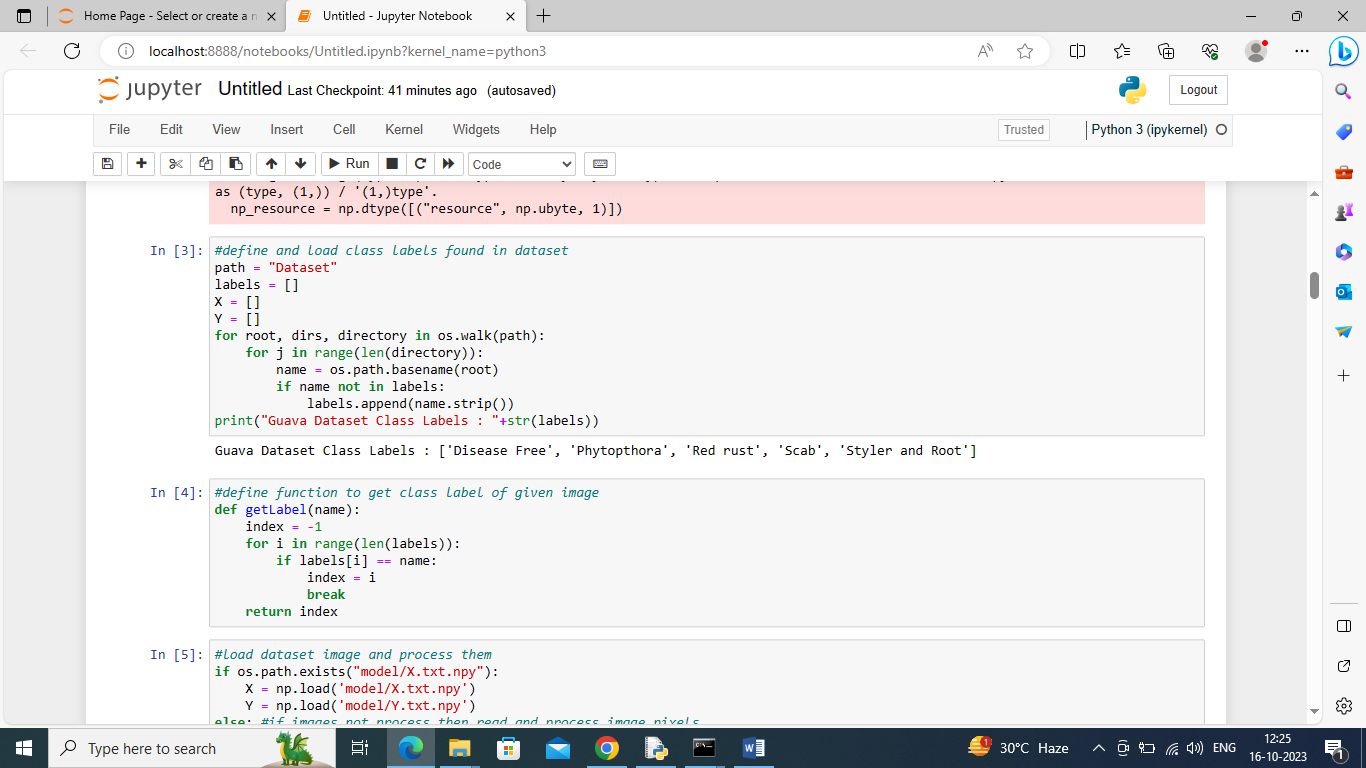
Guava Disease Detection using Convolution Neural Networks

In this project you ask us to implement 3 different pre-trained algorithms like DenseNet121, Dense169 and VGG19 and get same accuracy as per base paper but different algorithms are having different capability of prediction so after training and all optimizations we are getting DenseNet121 accuracy as 65%, DenseNet169 as 68% and VGG19 as 95.28%. So among all algorithms VGG is getting similar accuracy as per base paper.

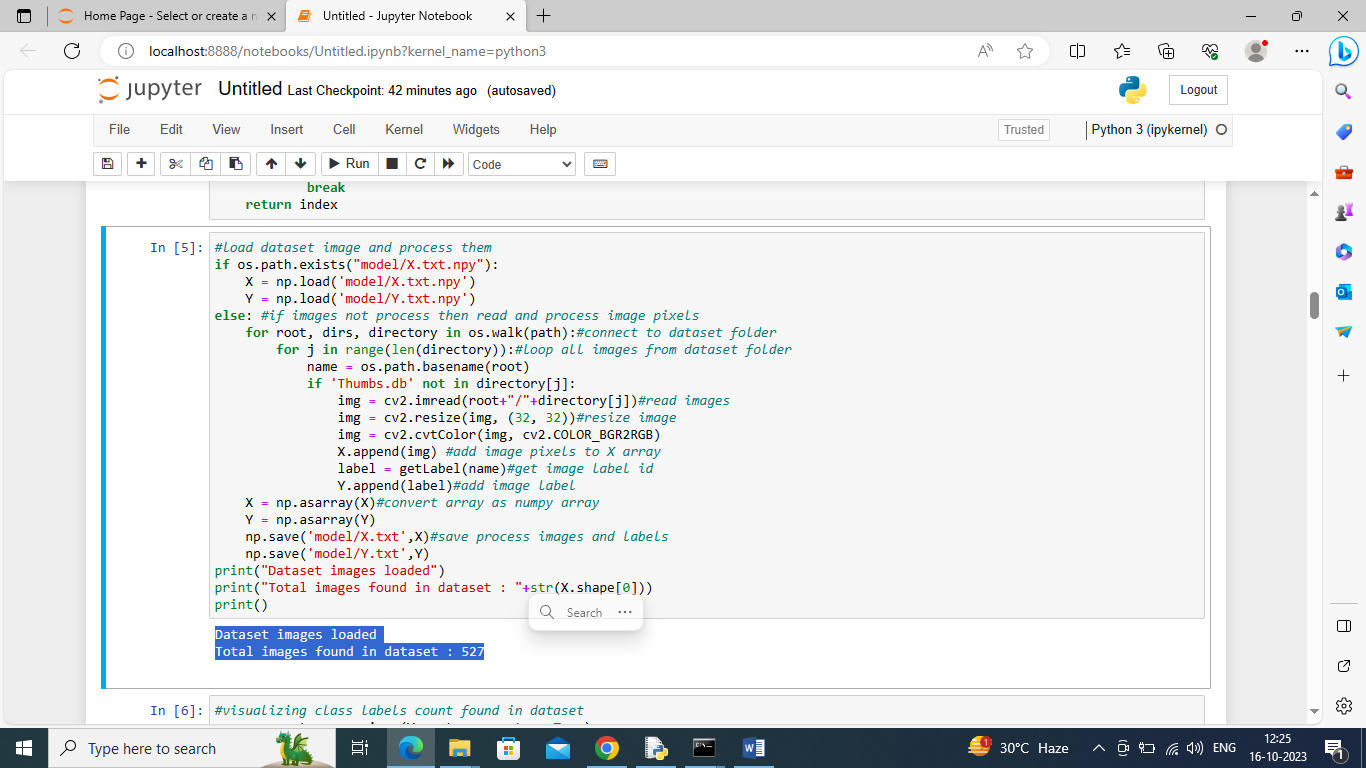
To trained and test all algorithms we have used same KAGGLE dataset given by you and we have coded all algorithms using JUPYTER notebook and below are the code and output screens with blue color comments



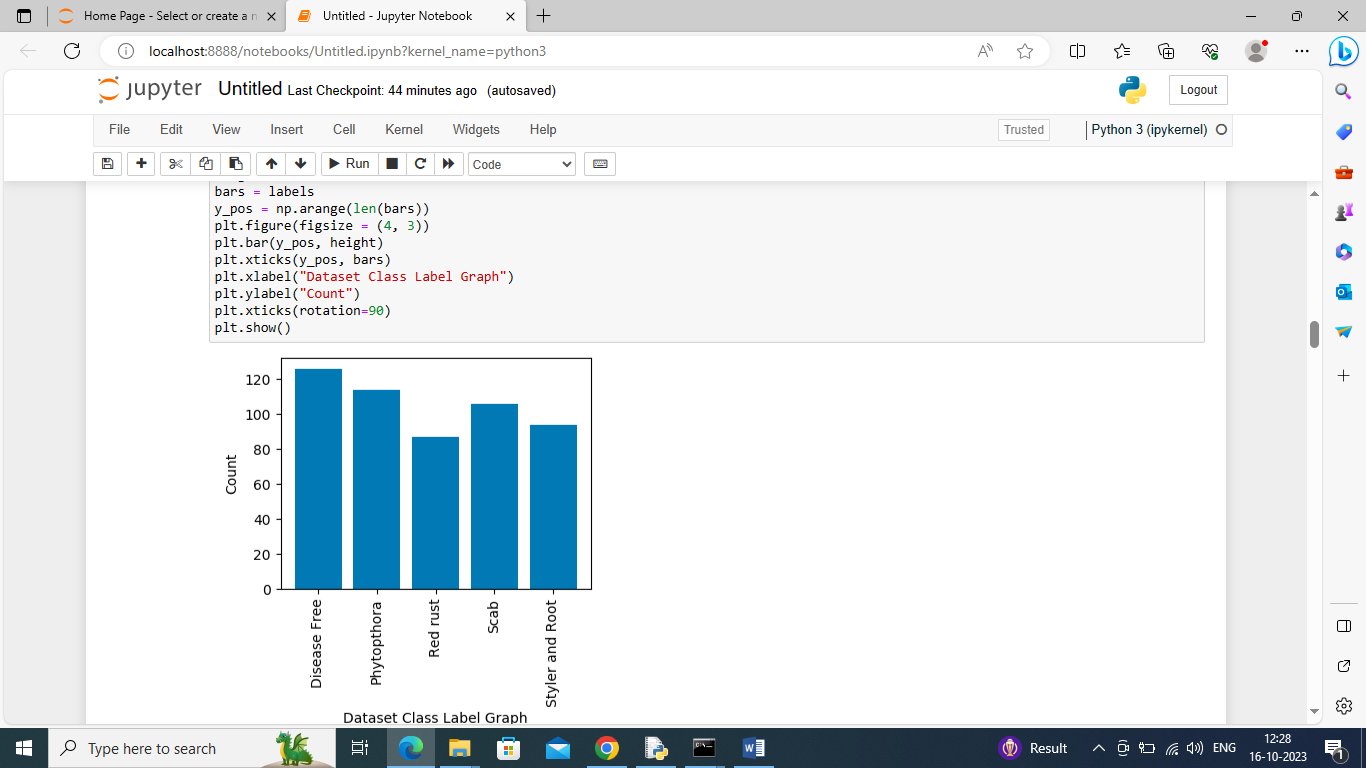
In above screen importing required packages and classes



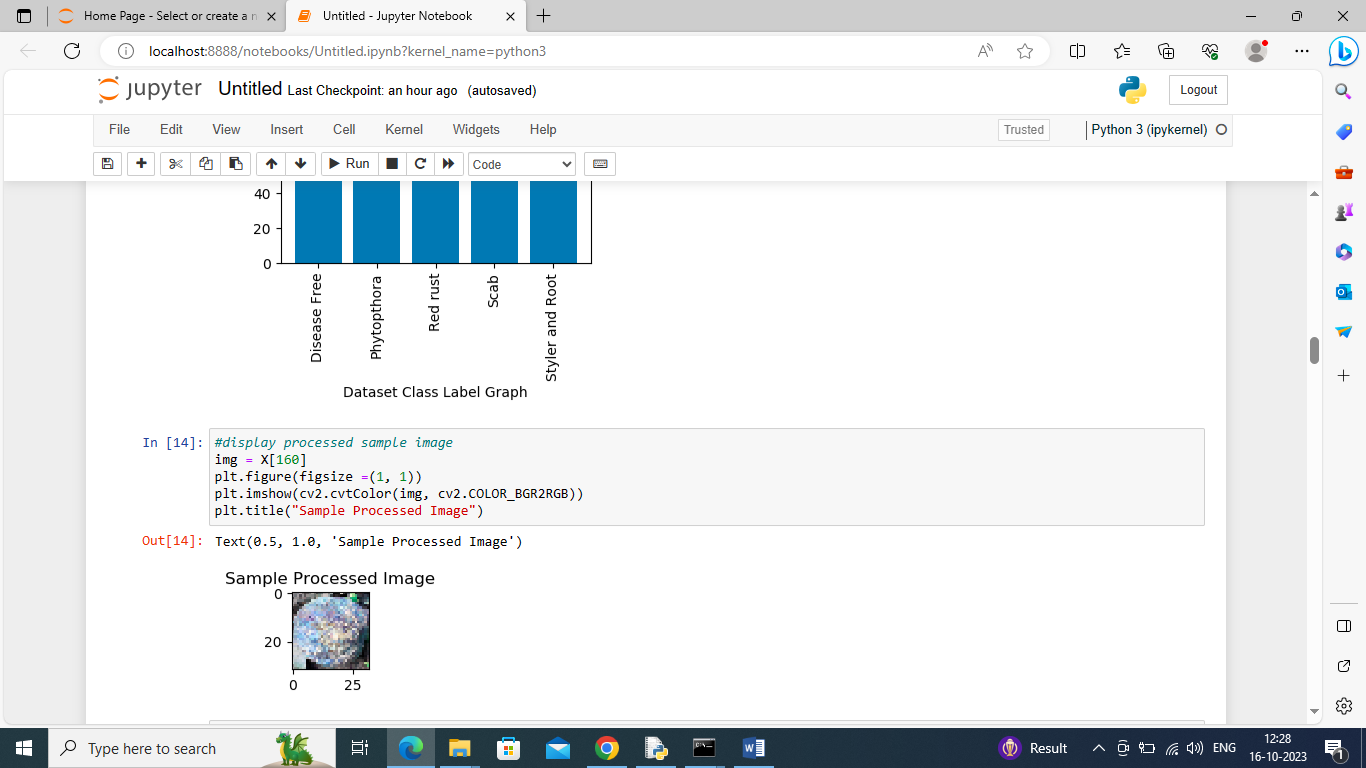
In above screen defining function to get class label names and integer ID from the dataset



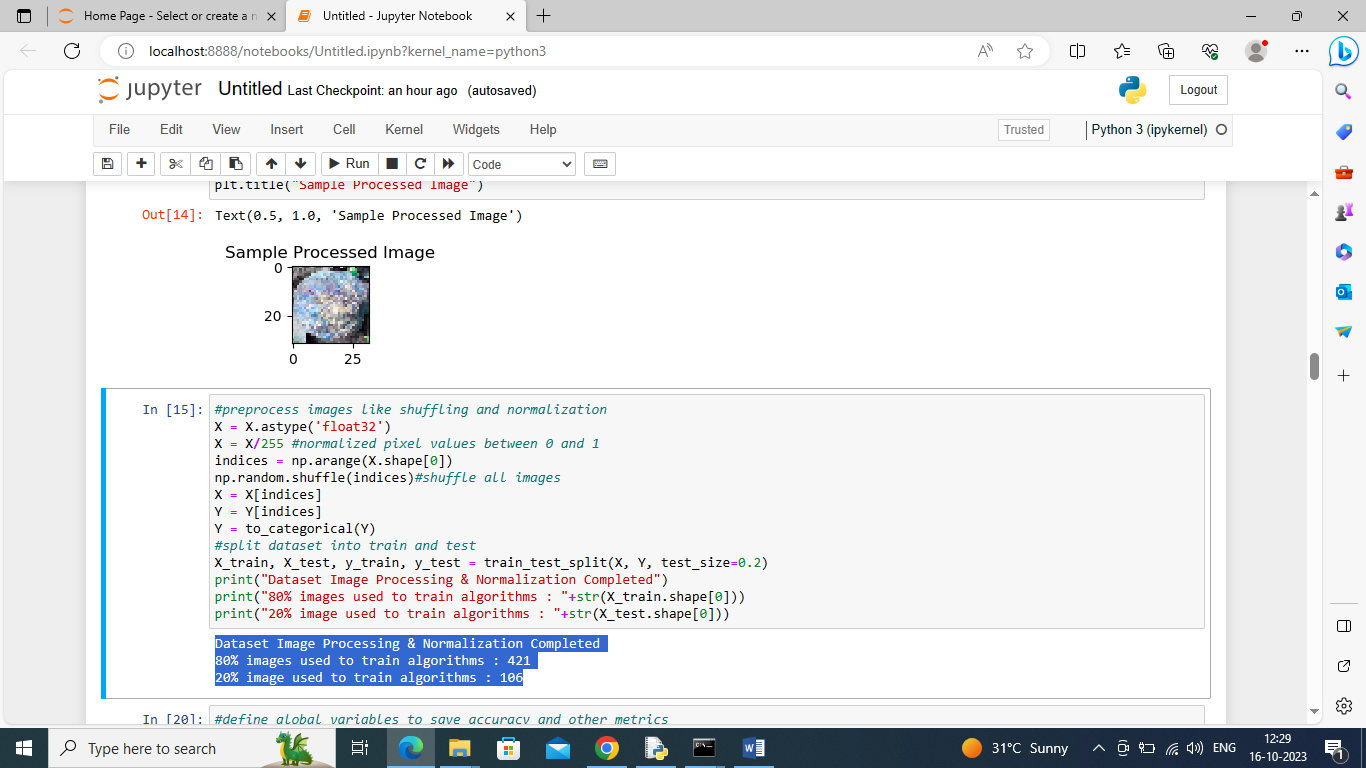
In above screen connecting to dataset folder and then reading each image, resizing, converting to RGB format and then extracting class labels and then adding to X and Y training array and then printing total images loaded



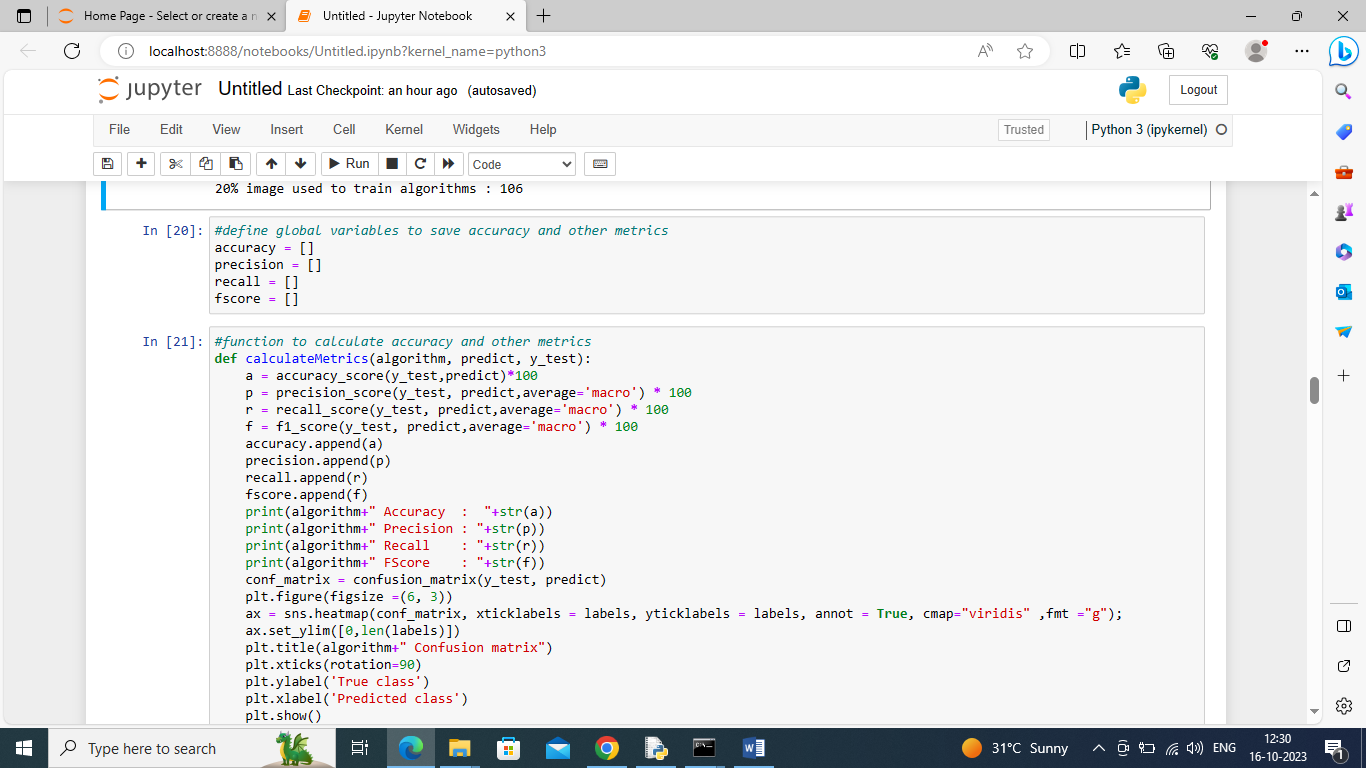
In above graph displaying number of images loaded from each class and in graph x-axis represents class label names and y-axis represents count



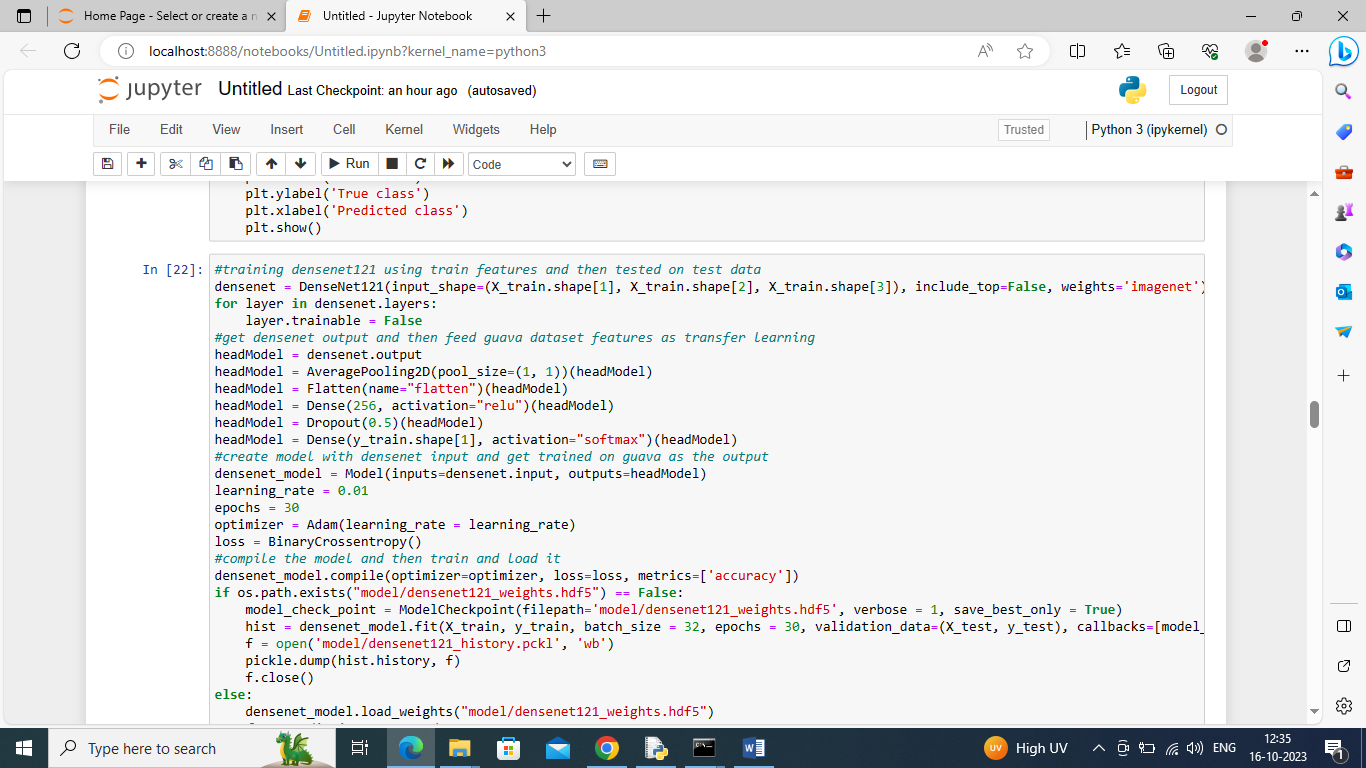
In above screen displaying sample processed image



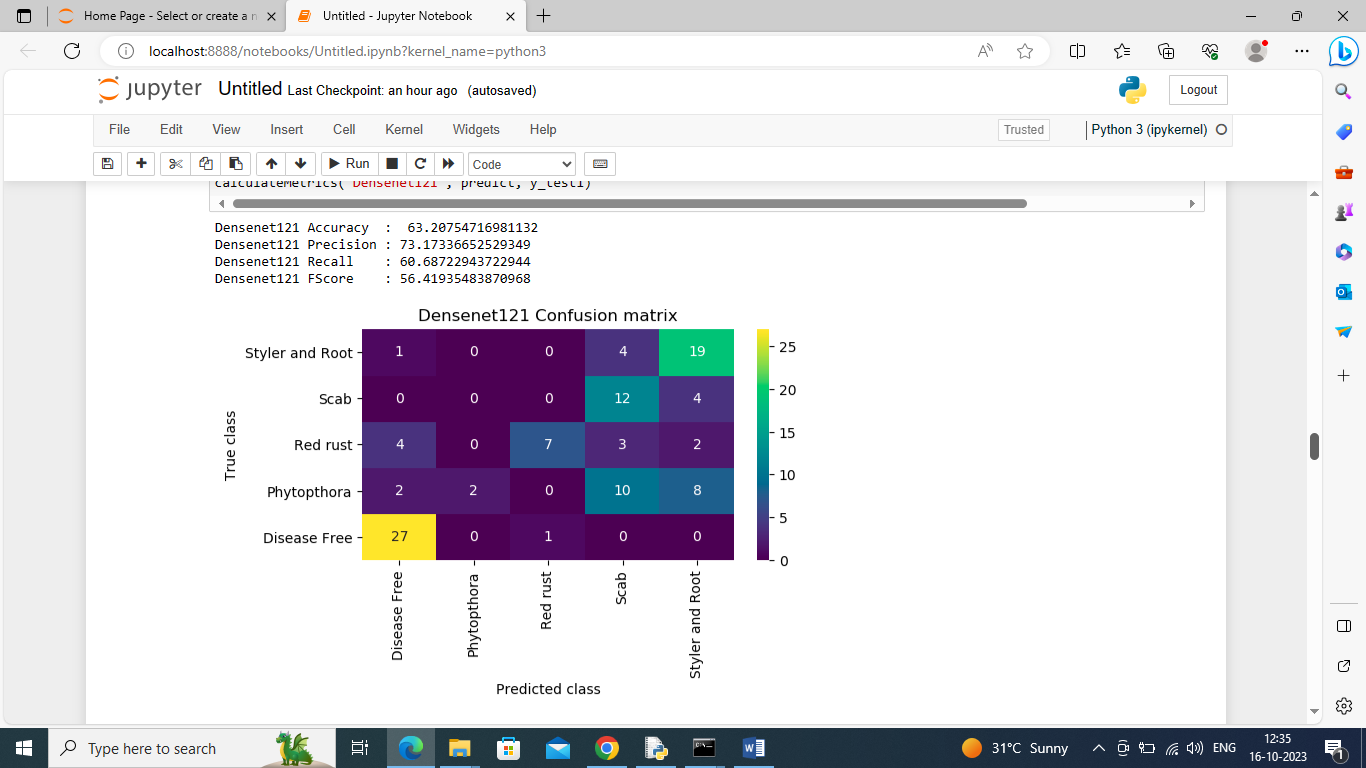
In above screen we are processing images such as shuffling, normalizing and splitting into train and test where application using 80% dataset for training and 20% for testing



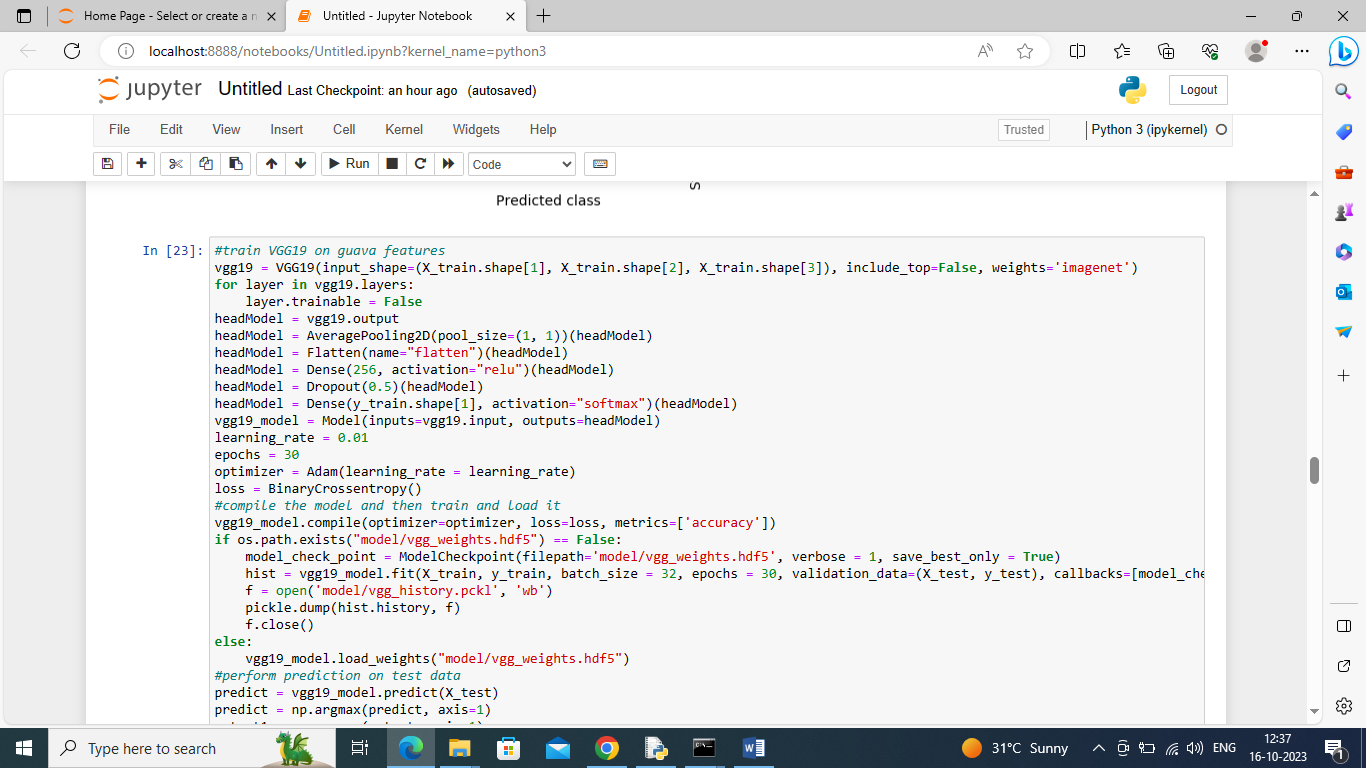
In above screen defining function to calculate accuracy and other metrics



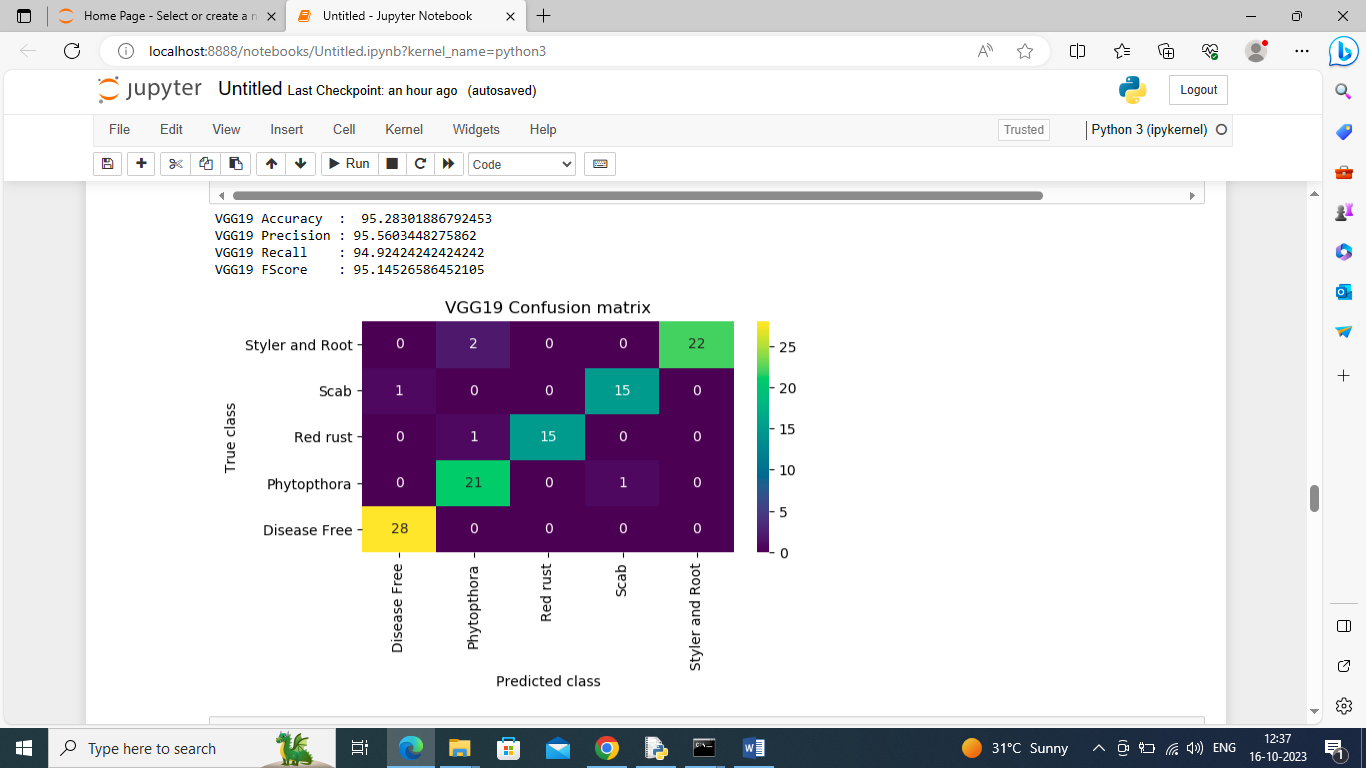
In above screen training DenseNet121 algorithm and after executing above block will get below output



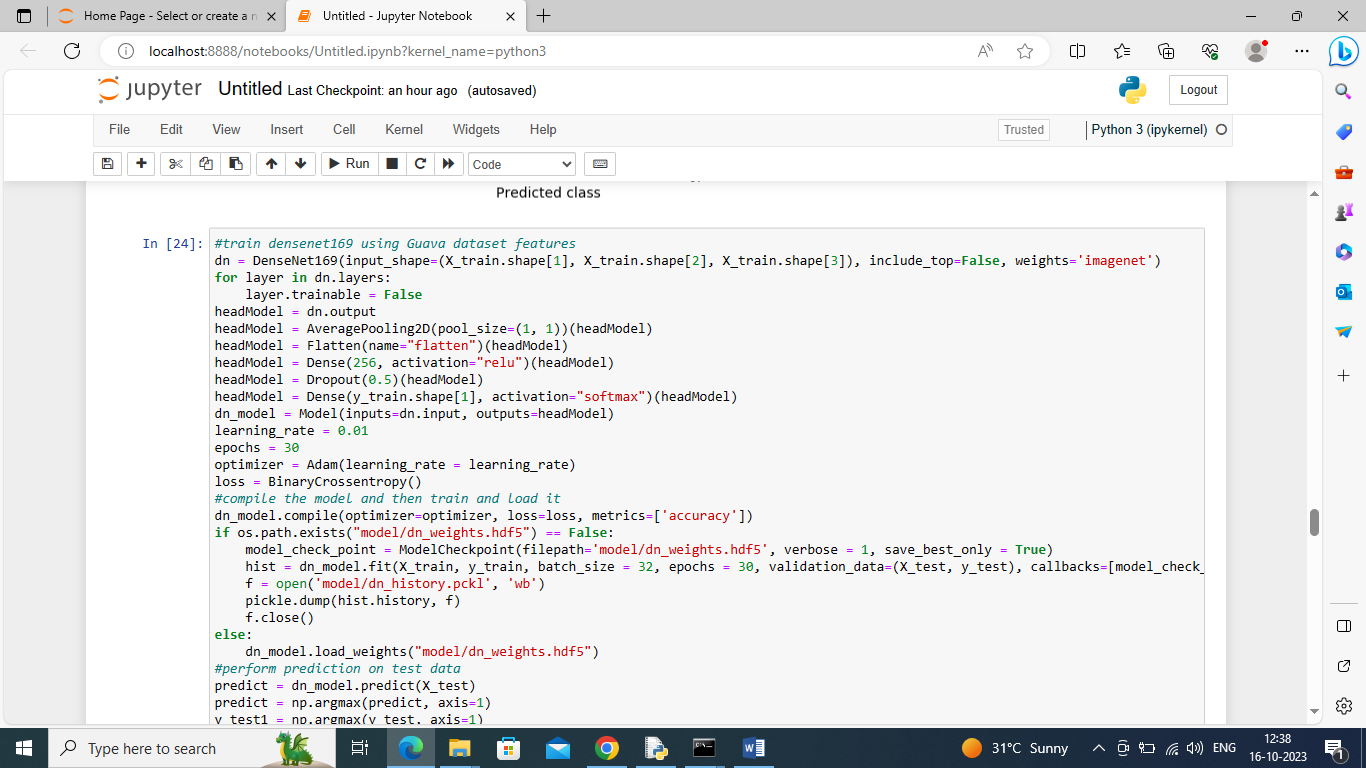
In above screen denseNet121 got 63% accuracy and can see other metrics also and in confusion matrix graph x-axis represents predicted labels and y-axis represents True Labels and all different color boxes in diagnol represents correct prediction count and remaining blue boxes represents incorrect prediction count



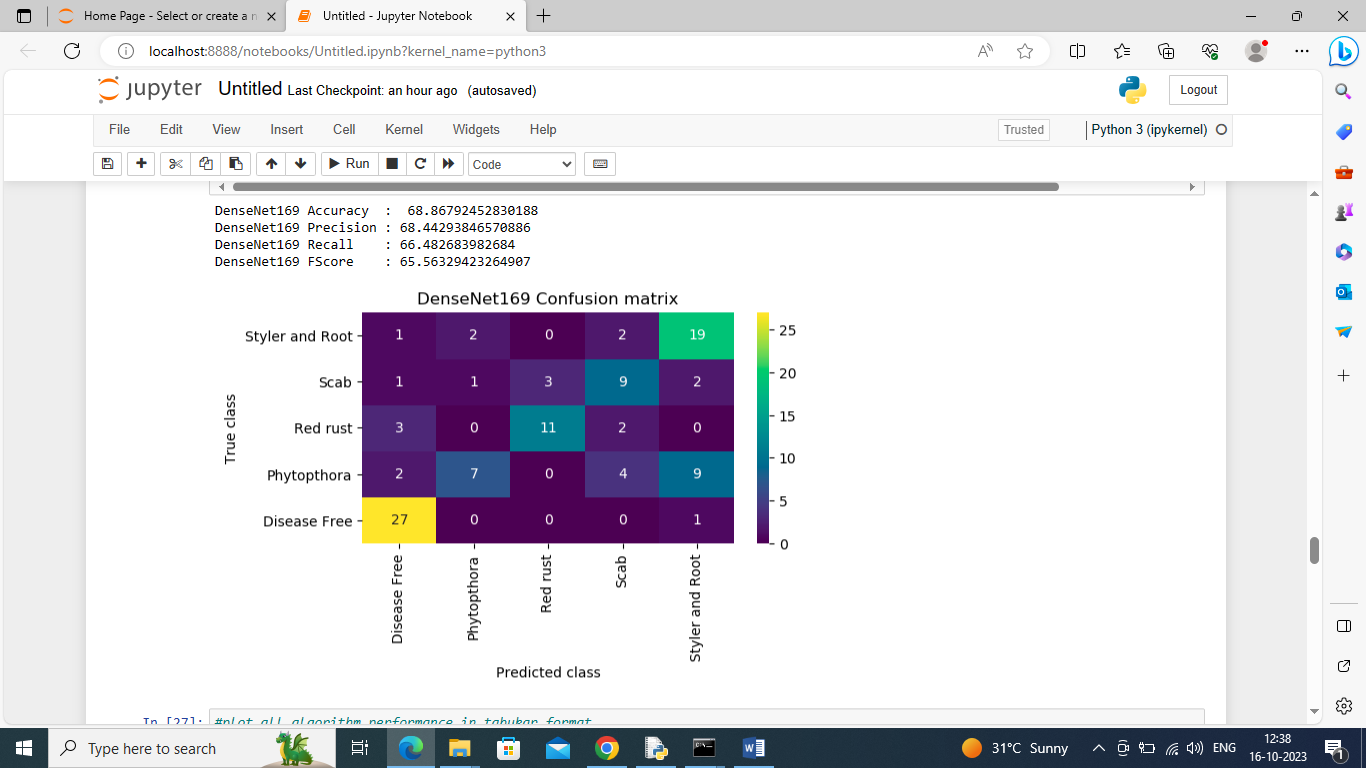
In above screen training VGG19 and below is the output



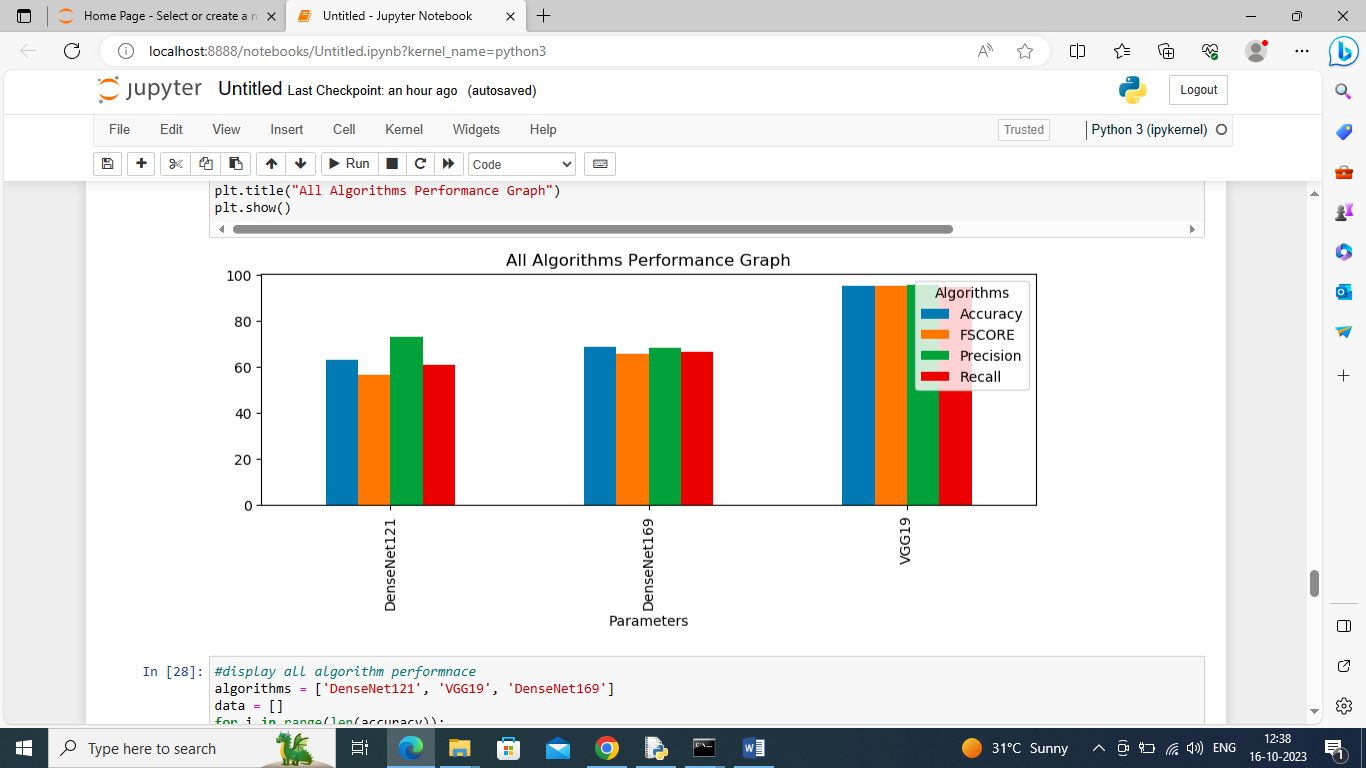
In above screen VGG19 got 95.28% accuracy and can see other metrics also



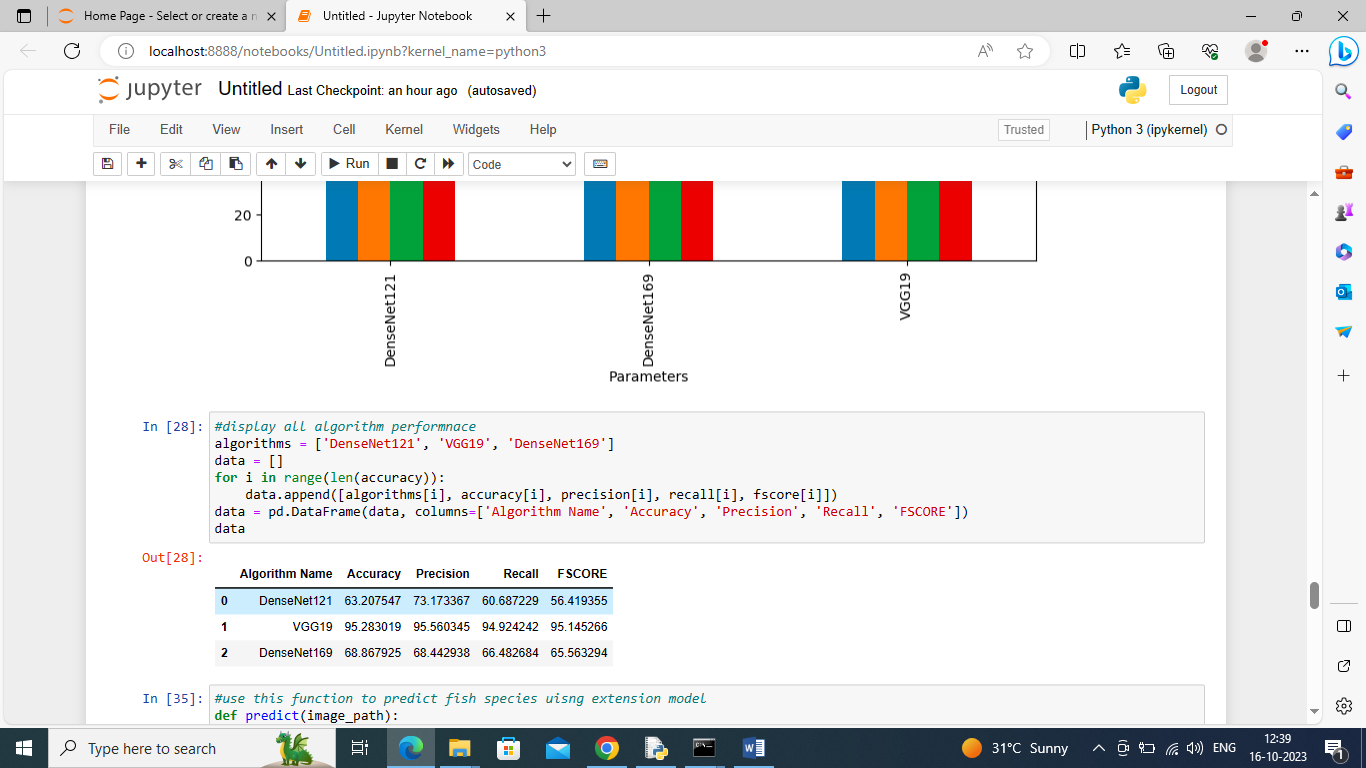
In above screen training DenseNet169 algorithm and below is the output



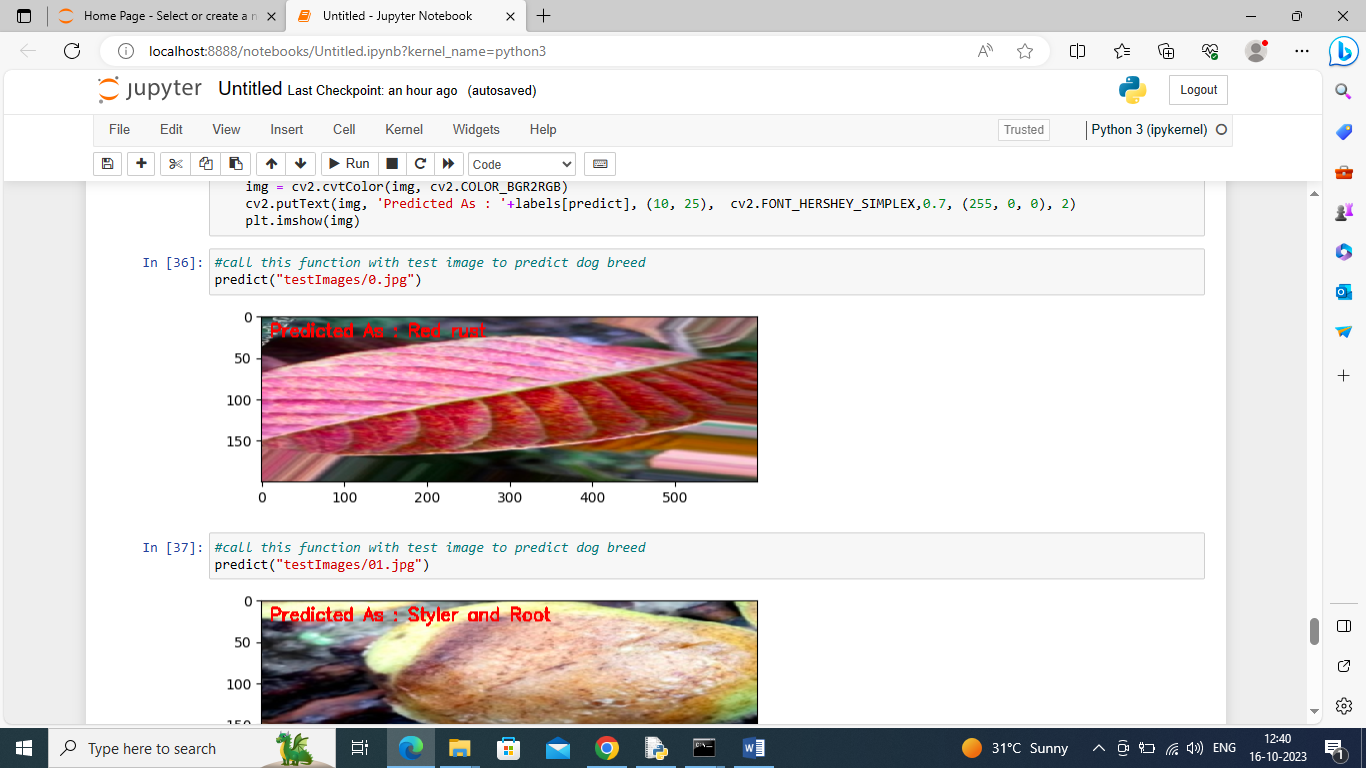
In above screen DenseNet169 got 68% accuracy



In above screen displaying all algorithms performance in graph format where x-axis represents algorithm names and y-axis represents accuracy and other metrics in different color bars



In above screen displaying all algorithms performance in tabular format



In above screen performing prediction on test image and then in red color text we can see predicted disease

