**Brain Tumor Analysis and Early Detection Using MRI Images and Working With Explainable AI**

**Abstract**

Brain “tumours are a critical health concern, frequently requiring early detection and precise examination for successful treatment arranging. Magnetic Resonance Imaging (MRI) is a generally involved indicative instrument for brain tumour detection because of its predominant delicate tissue contrast and painless nature. In any case, the manual examination of MRI images is tedious and inclined to human blunder. This paper proposes a system for early detection and investigation of brain tumours utilizing MRI images, expanded with Explainable Man-made consciousness (XAI) methods. The system aims to work on indicative accuracy, reduce responsibility for radiologists, and give experiences into the dynamic cycle.

**Algorithm Details** To detect disease we have experimented with multiple Machine and deep learning algorithms such as SVM, KNN, Random Forest, and in Deep Learning we have employed Convolution 2D Neural Networks (CNN2D) and pre-trained VGG16. Each algorithm performance is evaluated in terms of accuracy, precision, recall, FSCORE and Confusion Matrix. Among all algorithms CNN2D is giving highest accuracy of 98%.

The ongoing manual examination of MRI brain images for tumour detection is tedious, emotional, and inclined to analyse. This prompt postpones treatment and the absence of straightforwardness in the dynamic cycle. Moreover, the intricacy of MRI translation obstructs cooperation among clinical experts. There is a basic requirement for a mechanised framework that improves symptomatic precision, decreases responsibility for radiologists, and gives straightforwardness through Explainable AI methods.

**Techniques used for Data Processing**

Before train algorithms we have applied various features processing techniques like Resizing, shuffling, normalizations and then splitting dataset into train and test where application using 80% dataset for training and 20% dataset for testing.

For better performance we have tuned each algorithm with multiple hyper parameters.

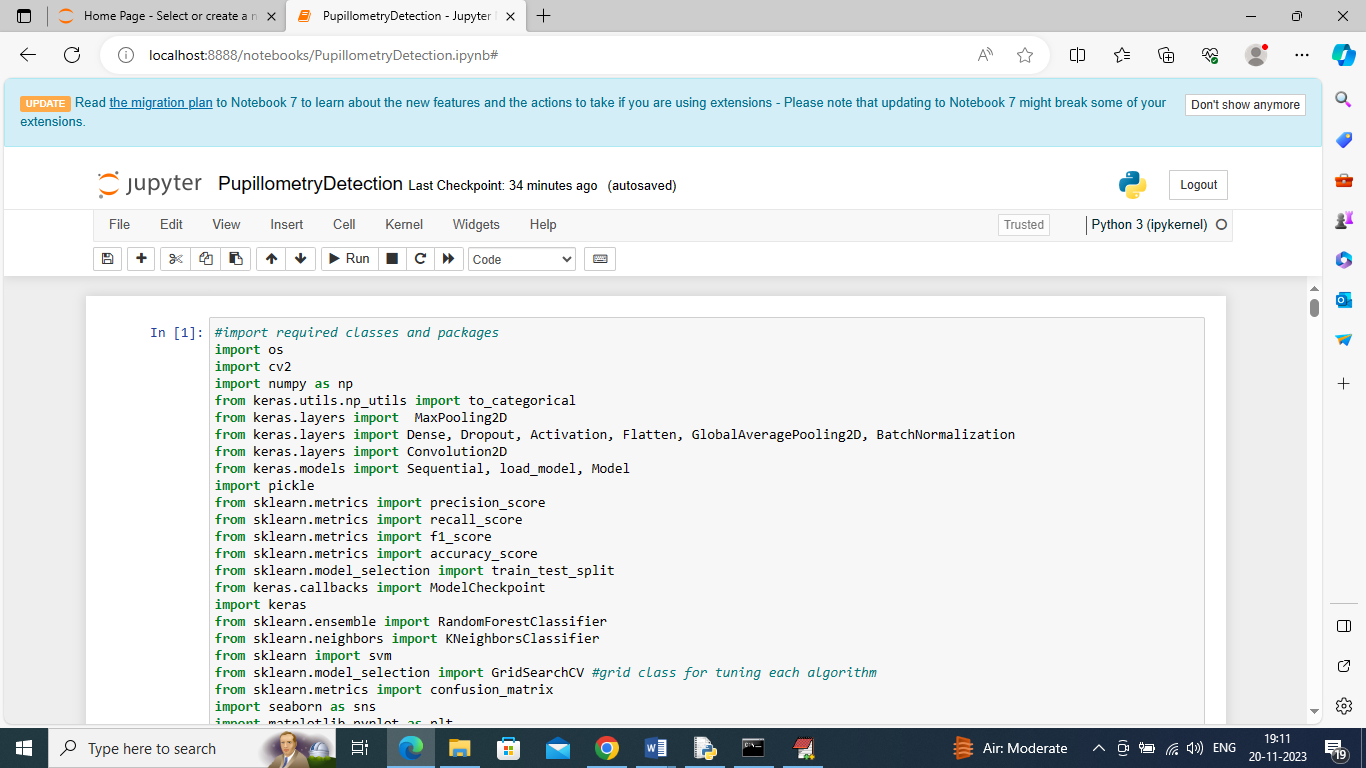
**Dataset Description**

For experiment we have utilized same dataset given by you and we have downloaded this dataset from below URL

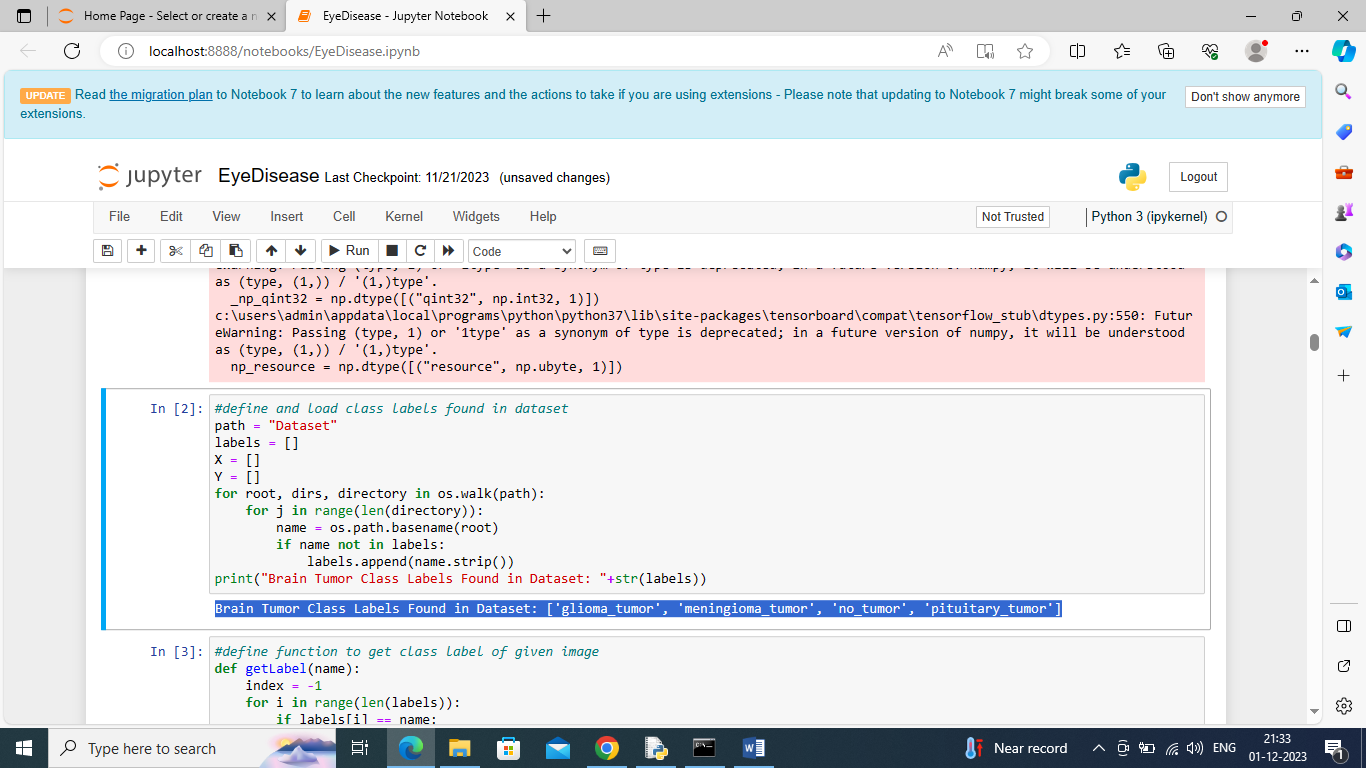
<https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset>

Above dataset consists of 4 different class labels such as ['glioma\_tumor', 'meningioma\_tumor', 'no\_tumor', 'pituitary\_tumor']

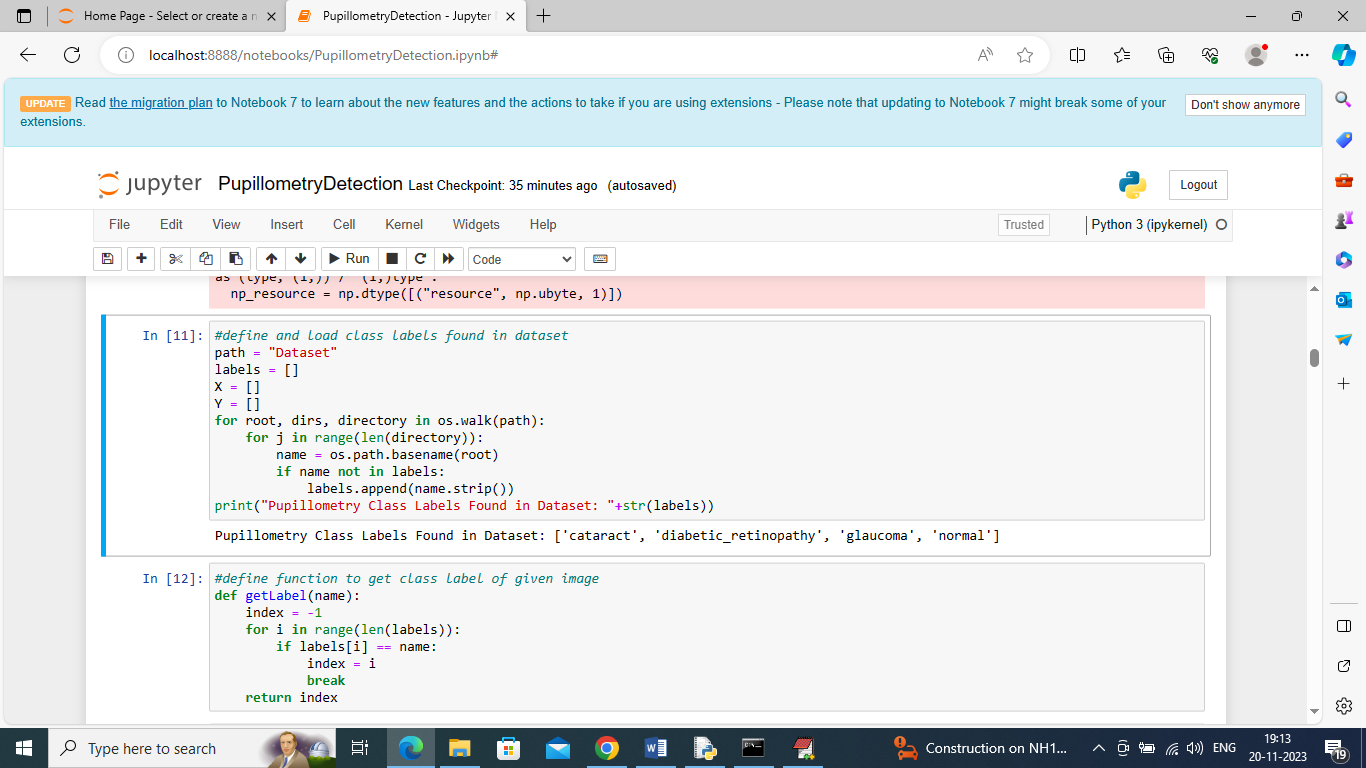
We have coded this project using JUPYTER notebook and below are the code and output screens with blue colour comments



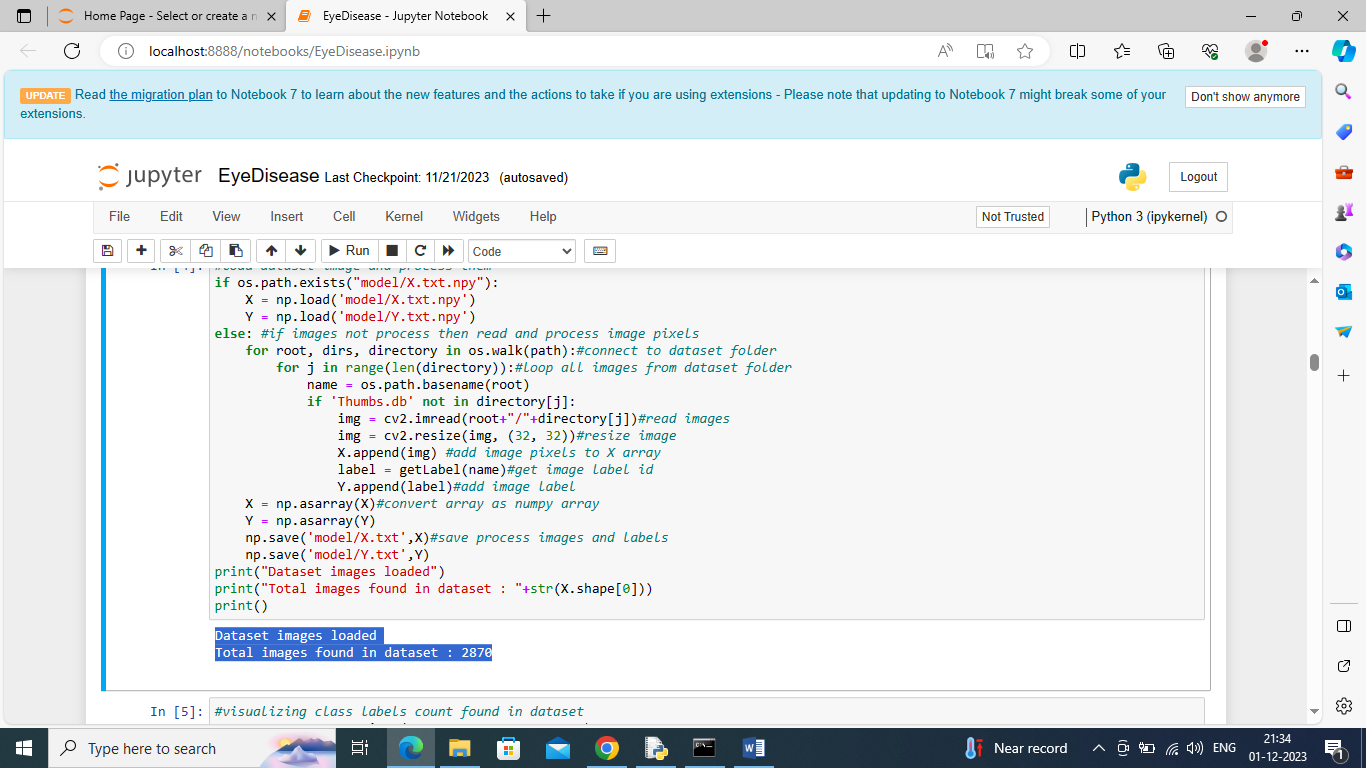
In above screen importing required python classes and packages



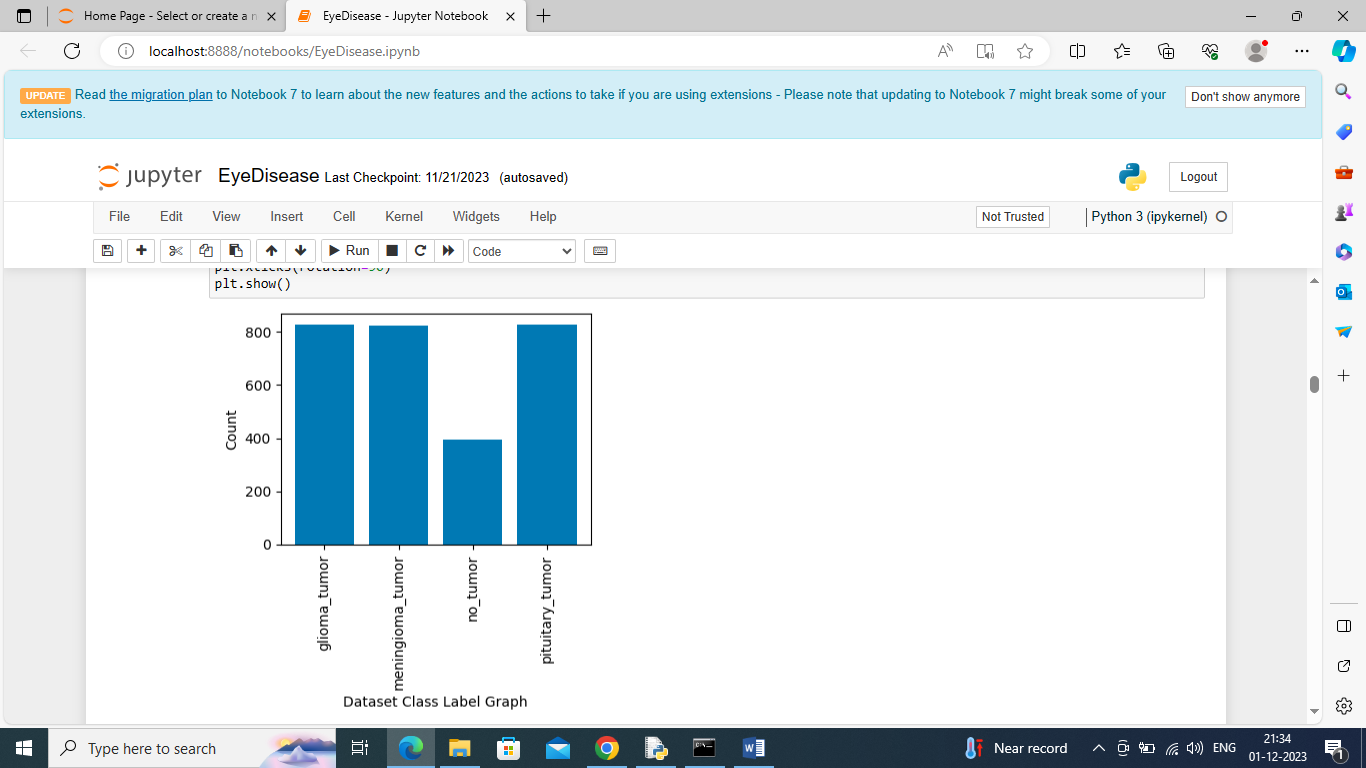
In above screen defining function to loop and display all class labels found in dataset



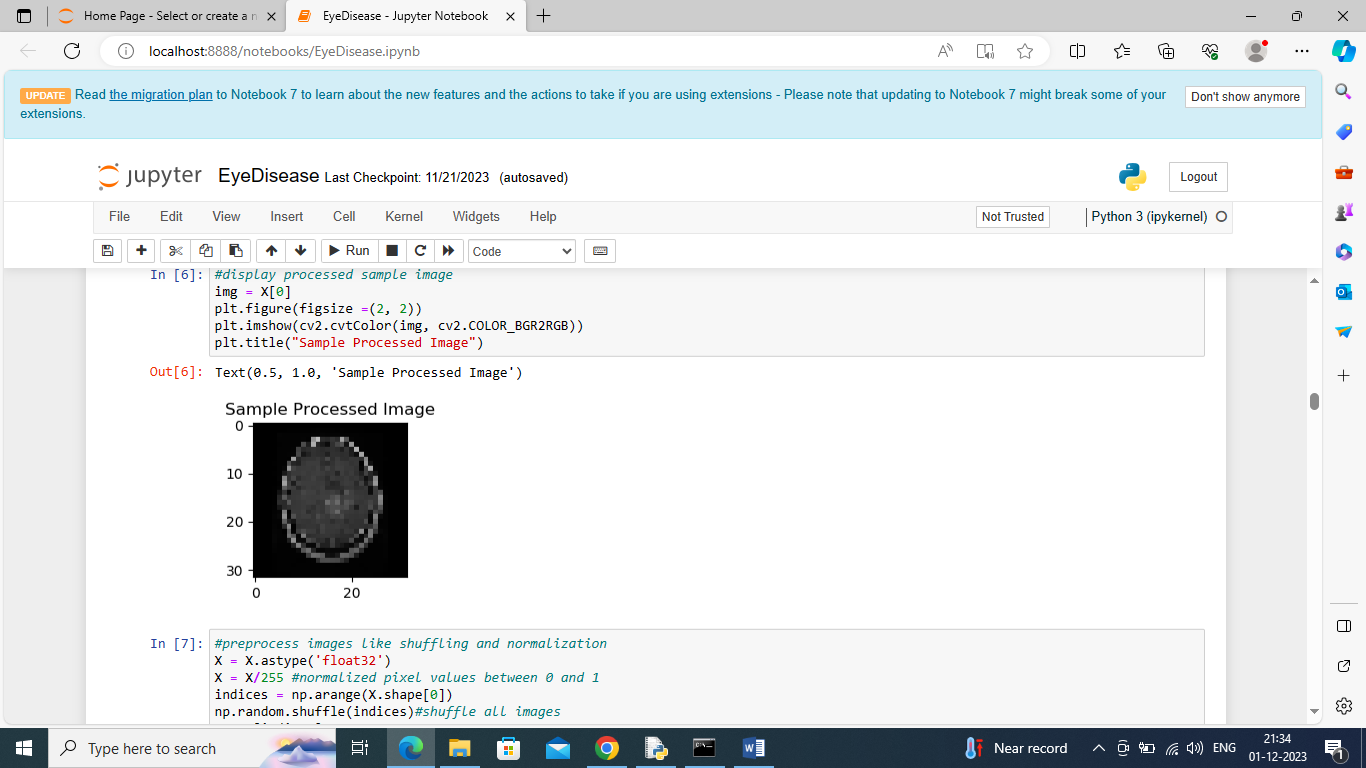
In above screen defining function to get integer class label from given image name



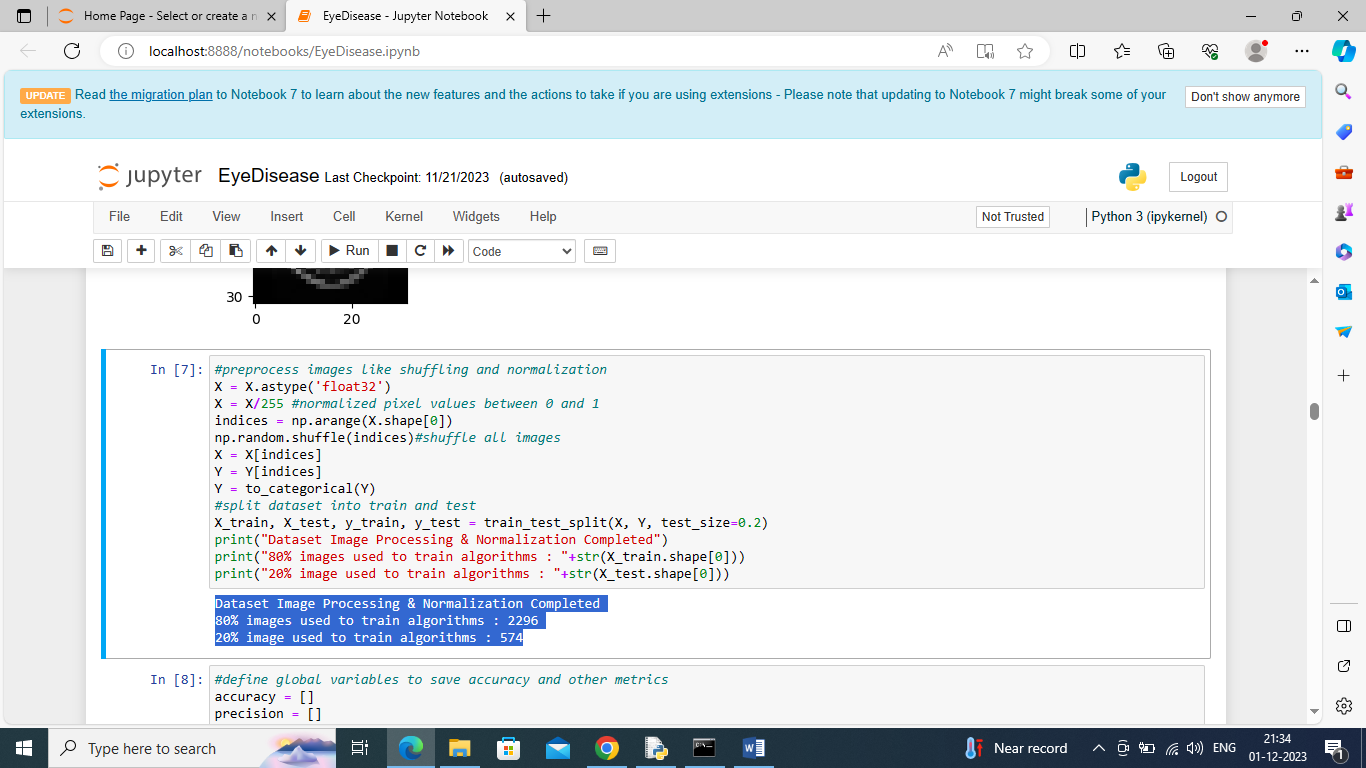
In above screen reading all images from given dataset folder and then in blue colour text displaying total images loaded



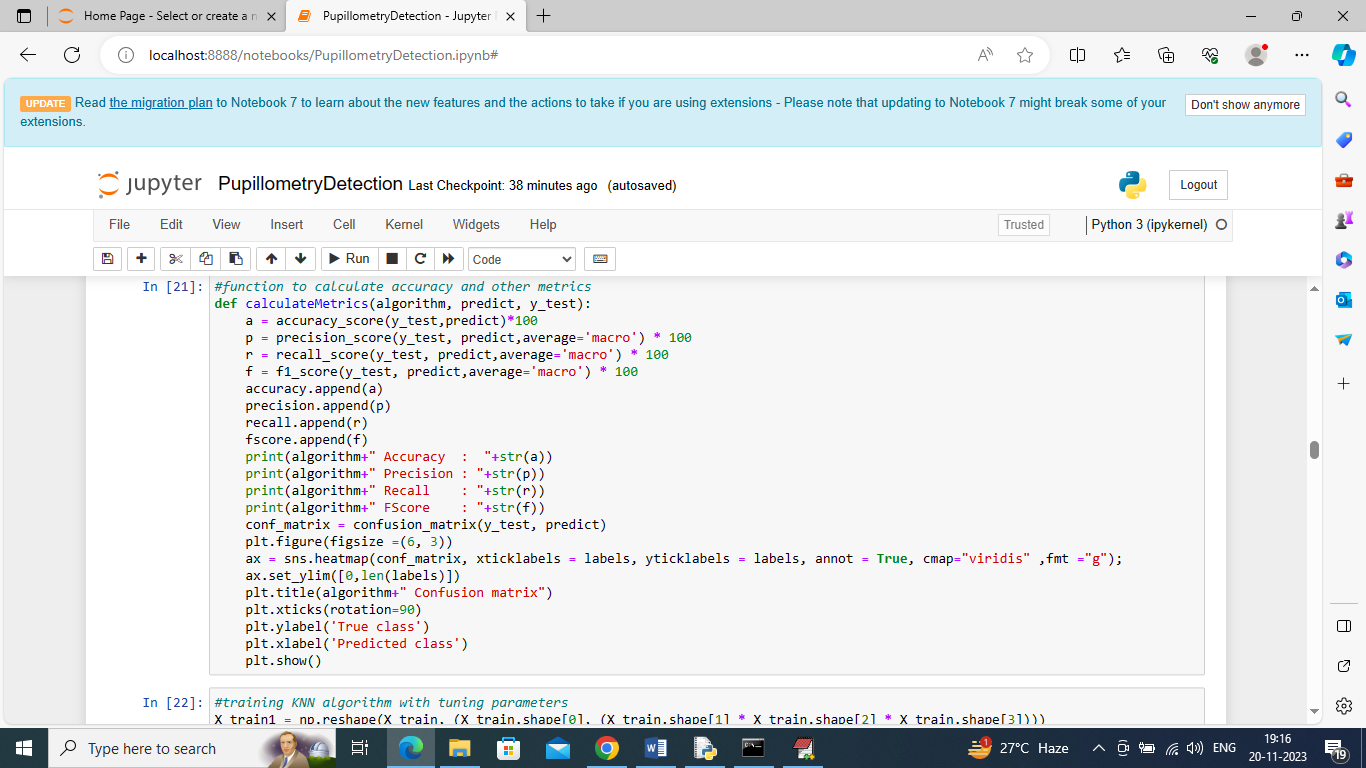
In above graph x-axis represents pupillometry disease class labels and y-axis represents of count of those class labels found in dataset



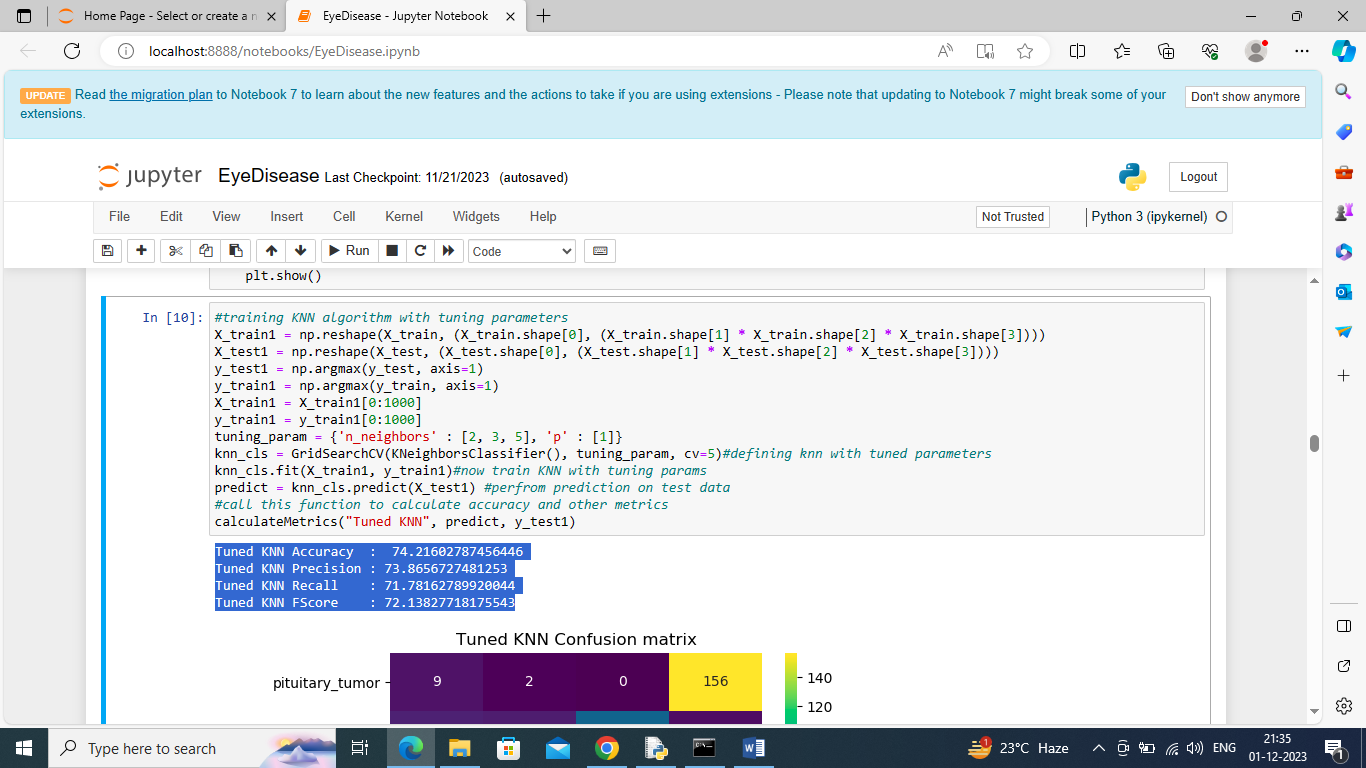
In above screen displaying processed sample image



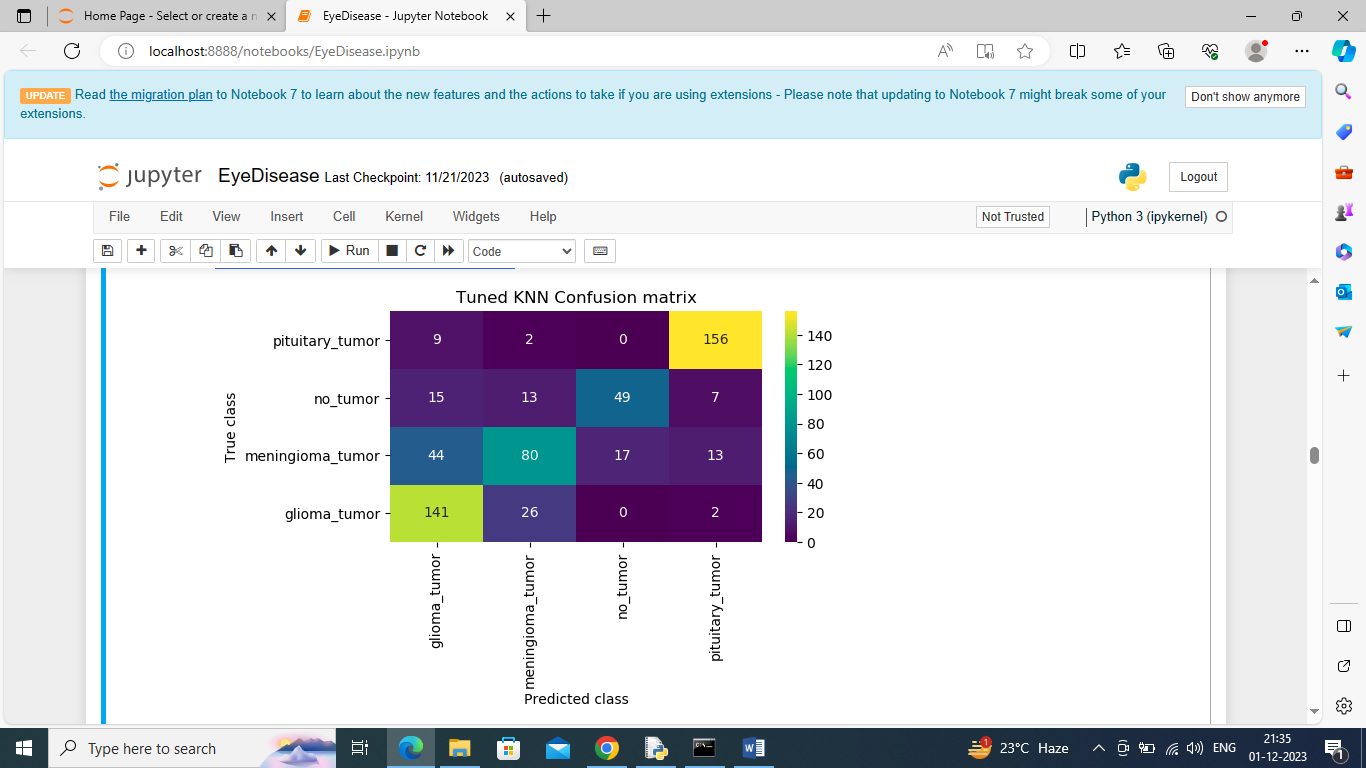
In above screen applying processing techniques such as shuffling, normalization and then splitting dataset into train and test and then in blue colour text displaying train and test data size



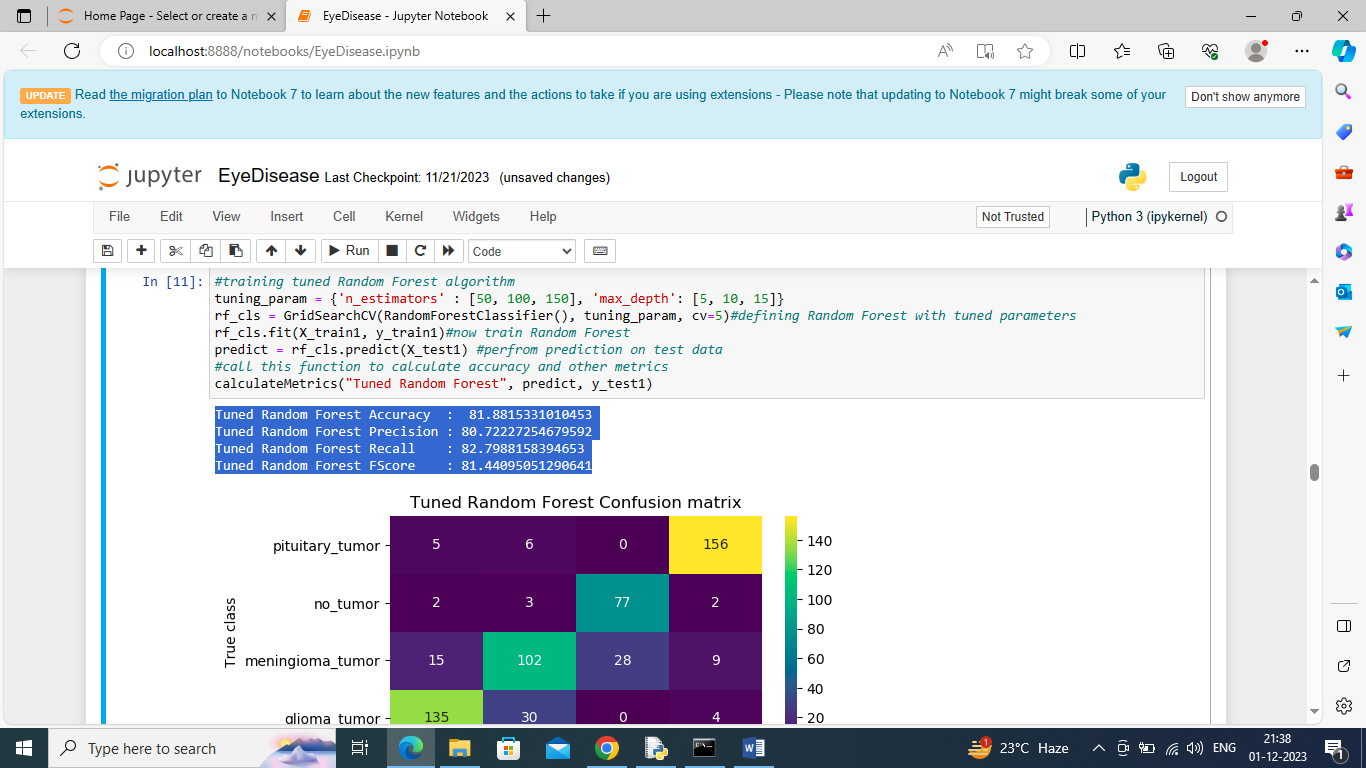
In above screen defining function to calculate accuracy and other metrics



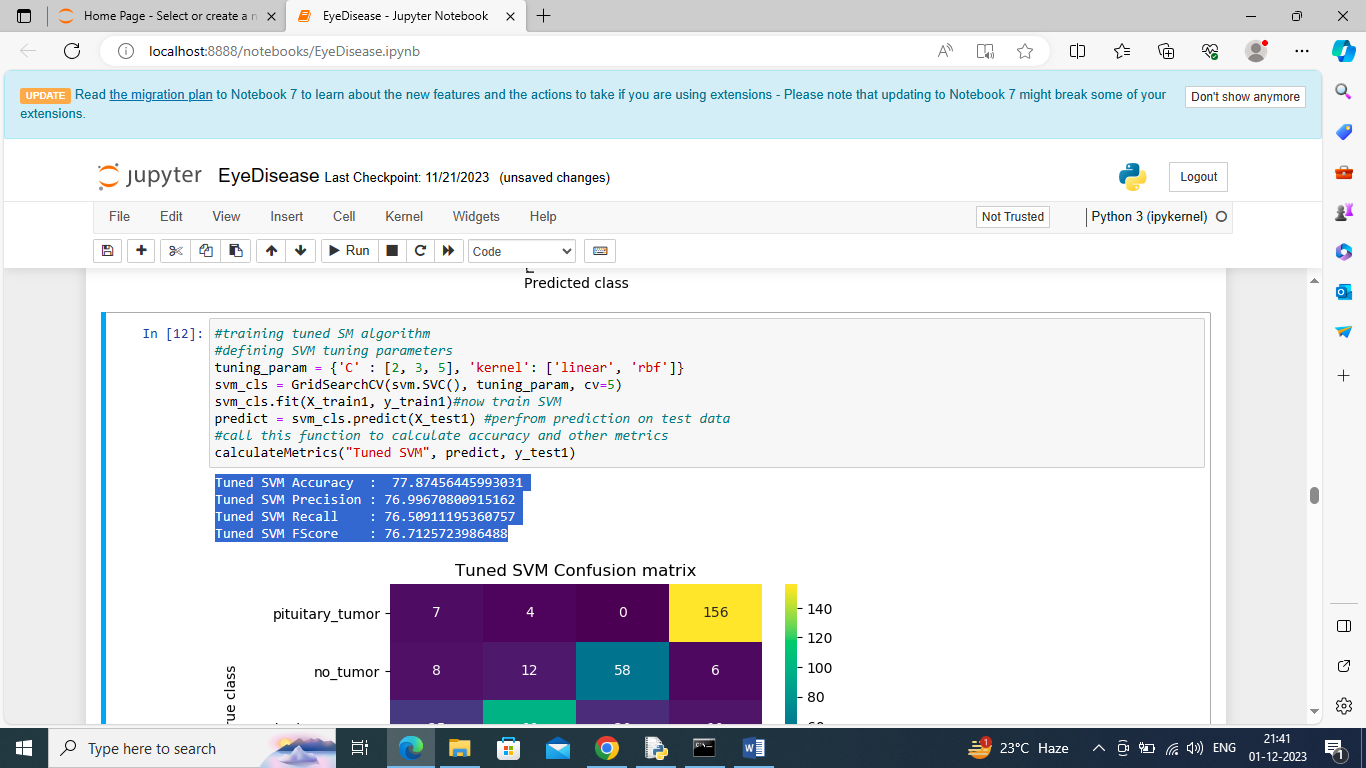
In above screen training KNN with tuned parameters and after training KNN got 74% accuracy and can see other metrics also and below is the KNN prediction confusion matrix graph



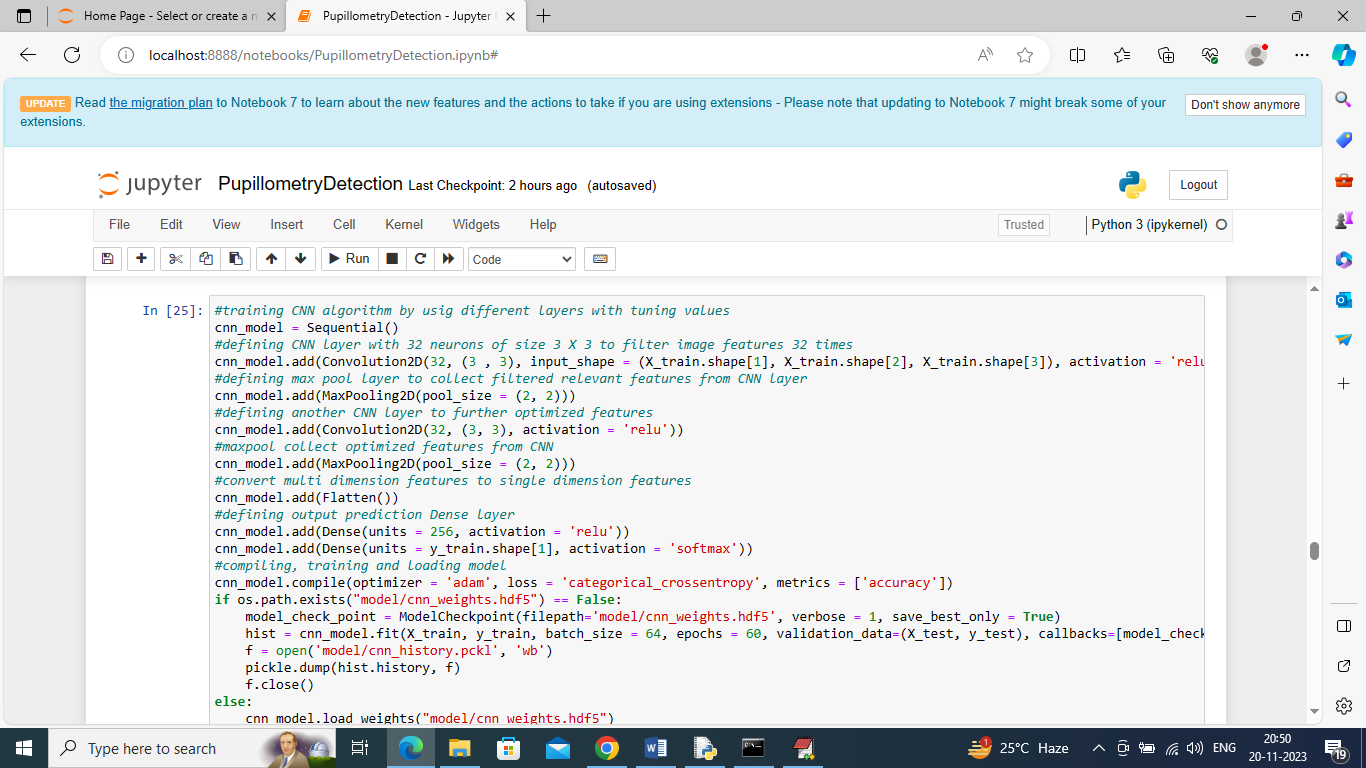
In above KNN confusion matrix graph x-axis represents Predicted Labels and y-axis represents True Labels and all boxes in diagnol represents correct prediction count and remaining boxes represents incorrect prediction count



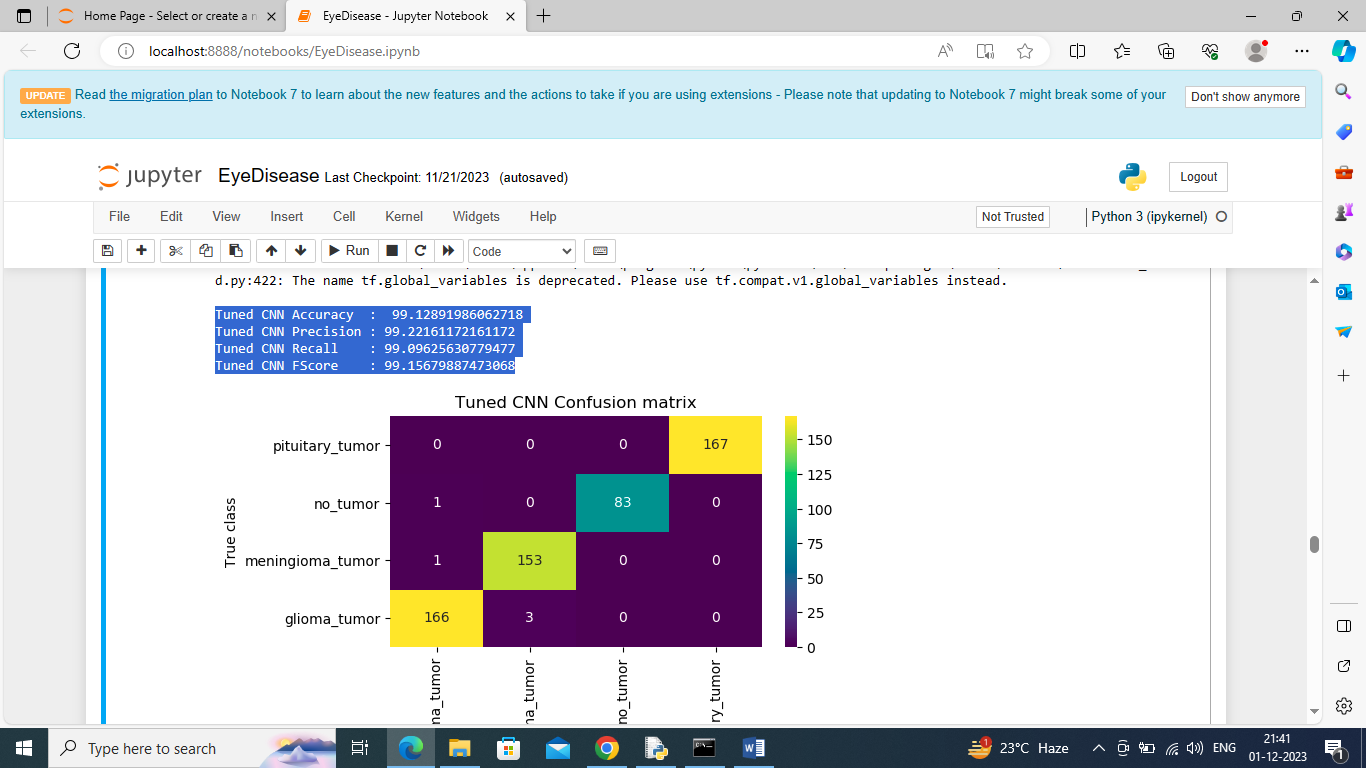
In above screen training Random Forest with tuned hyper parameters and then performing prediction on test data and then Random Forest got 81% accuracy and can see other metrics also



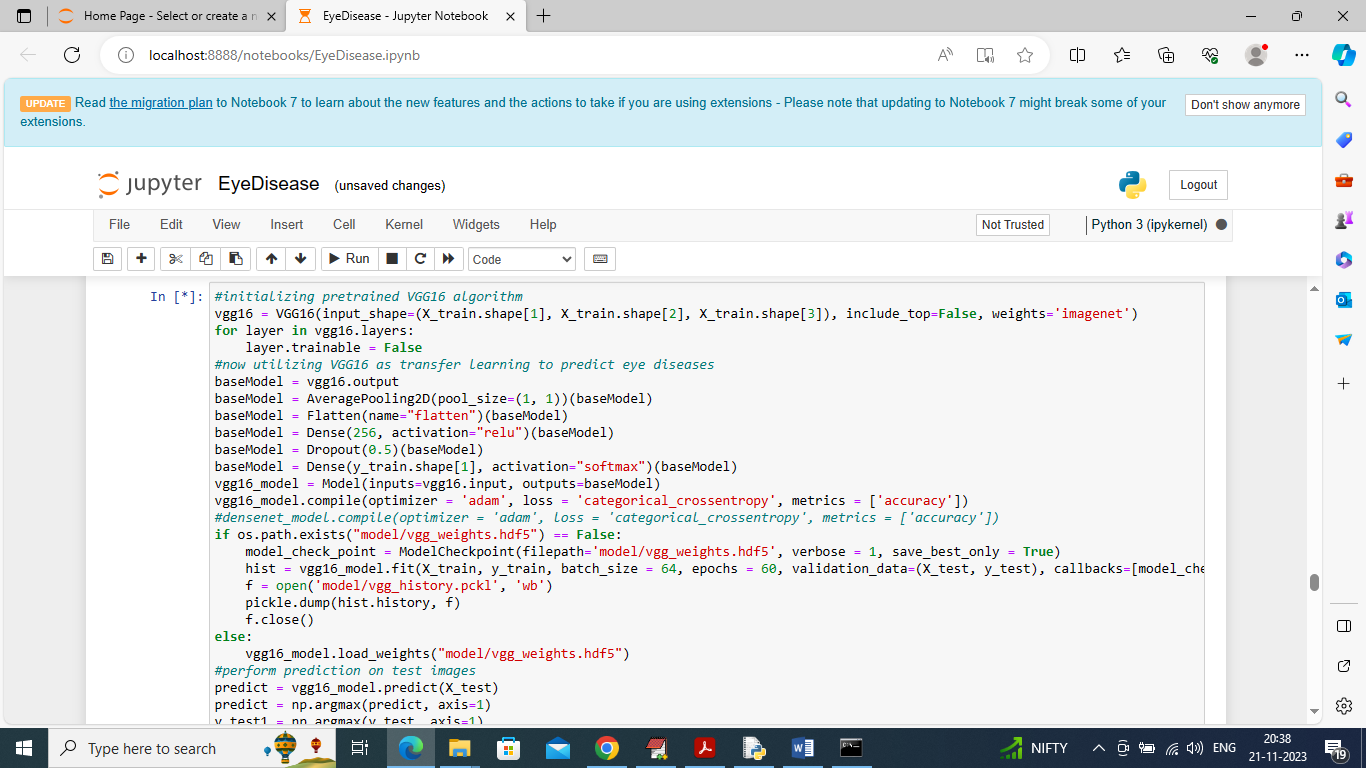
In above screen training SVM algorithm and after prediction on test data SVM got 77% accuracy



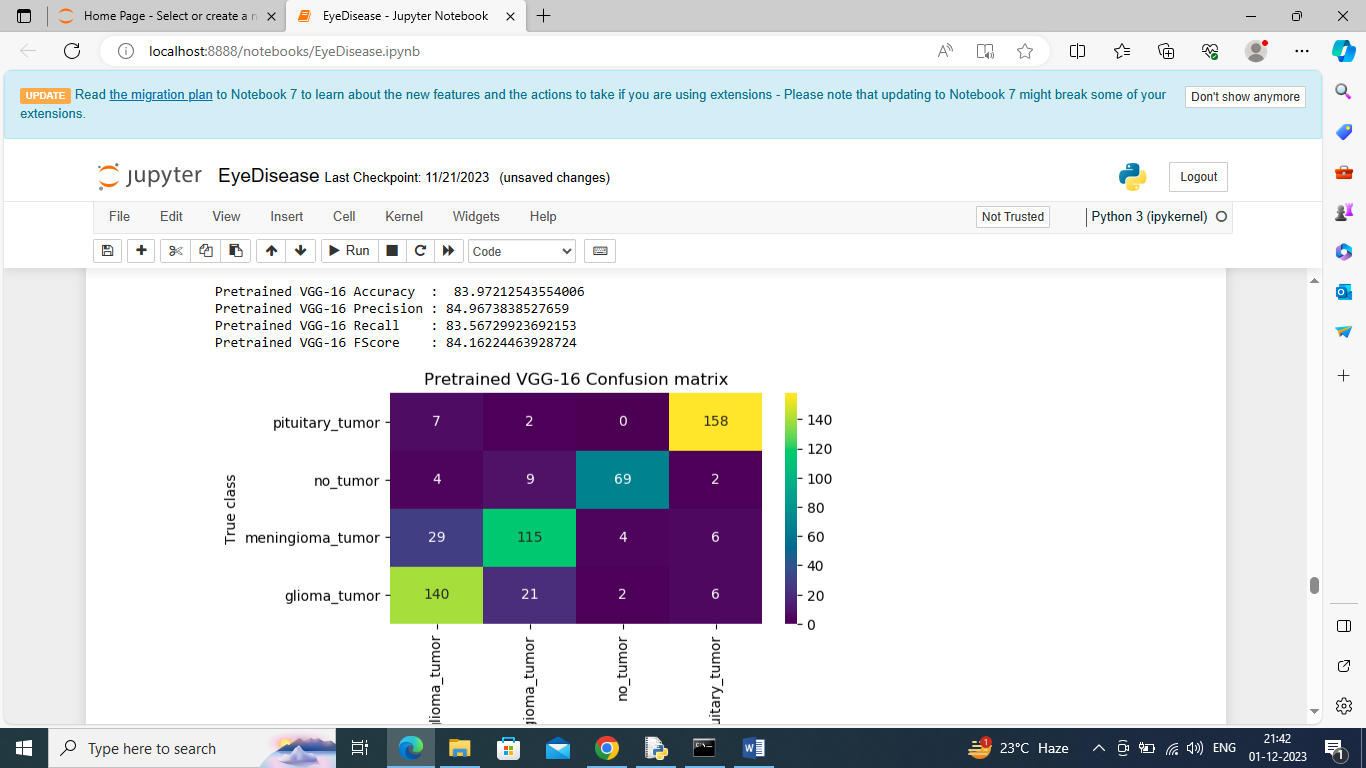
In above screen defining CNN2D neural network and after execution of this block will get below output



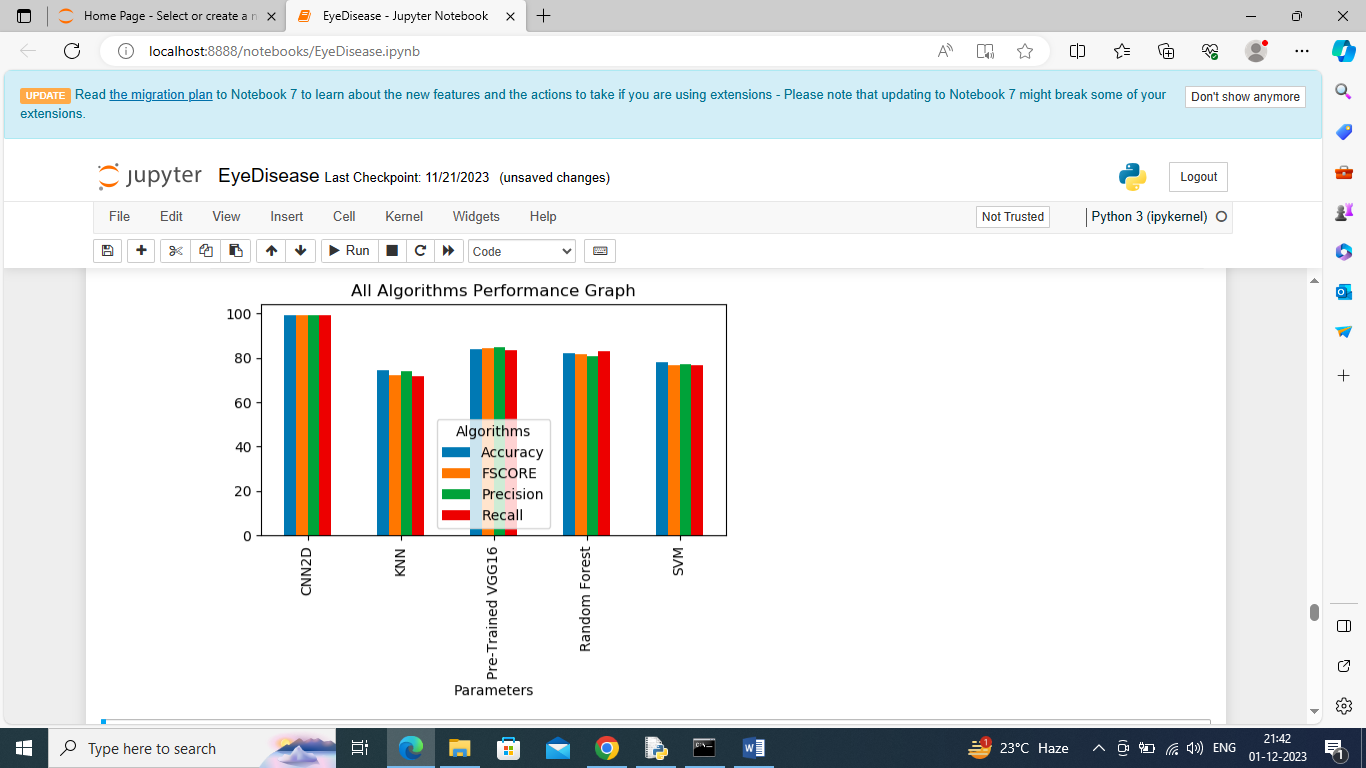
In above screen CNN got 99% accuracy and in confusion matrix graph yellow boxes contains correct prediction count and all blue boxed contains incorrect prediction count which are very few.



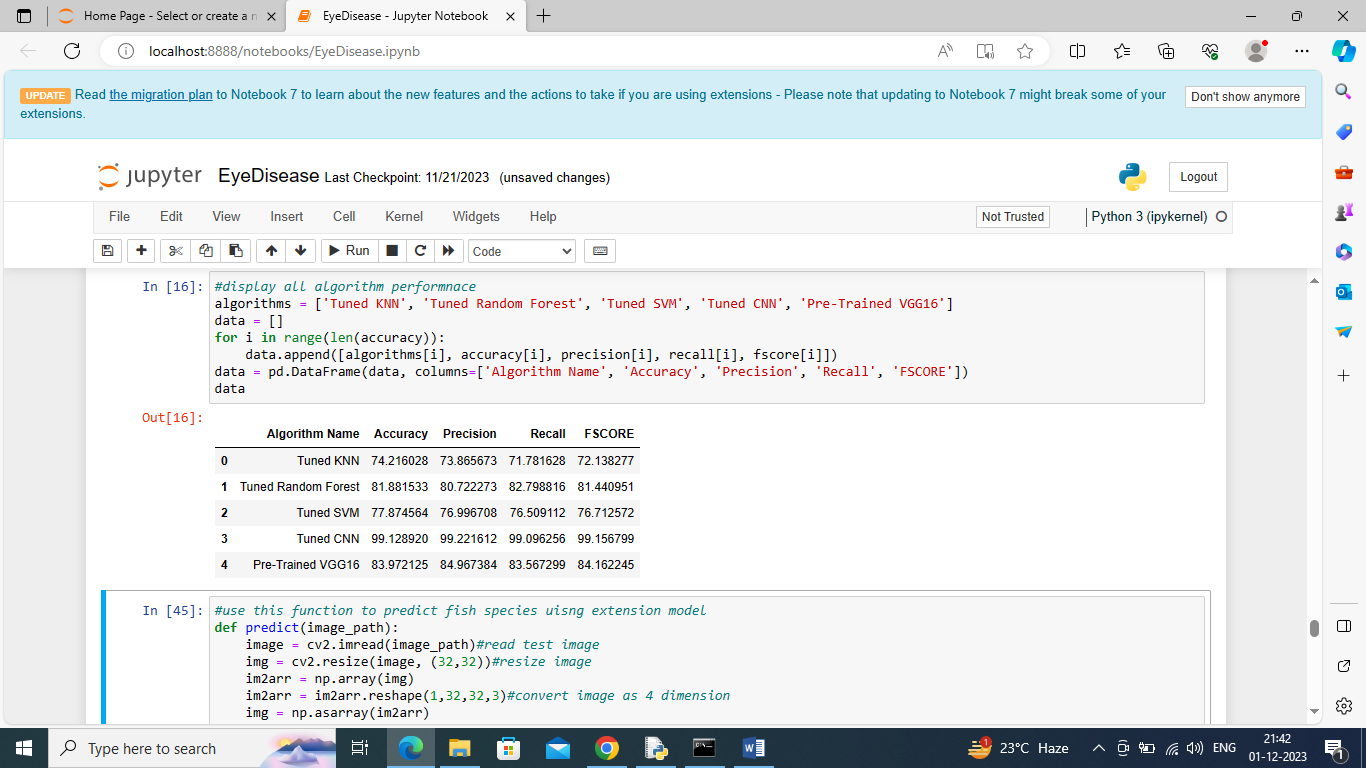
In above screen training pre-trained VGG16 model and after executing above model will get below output



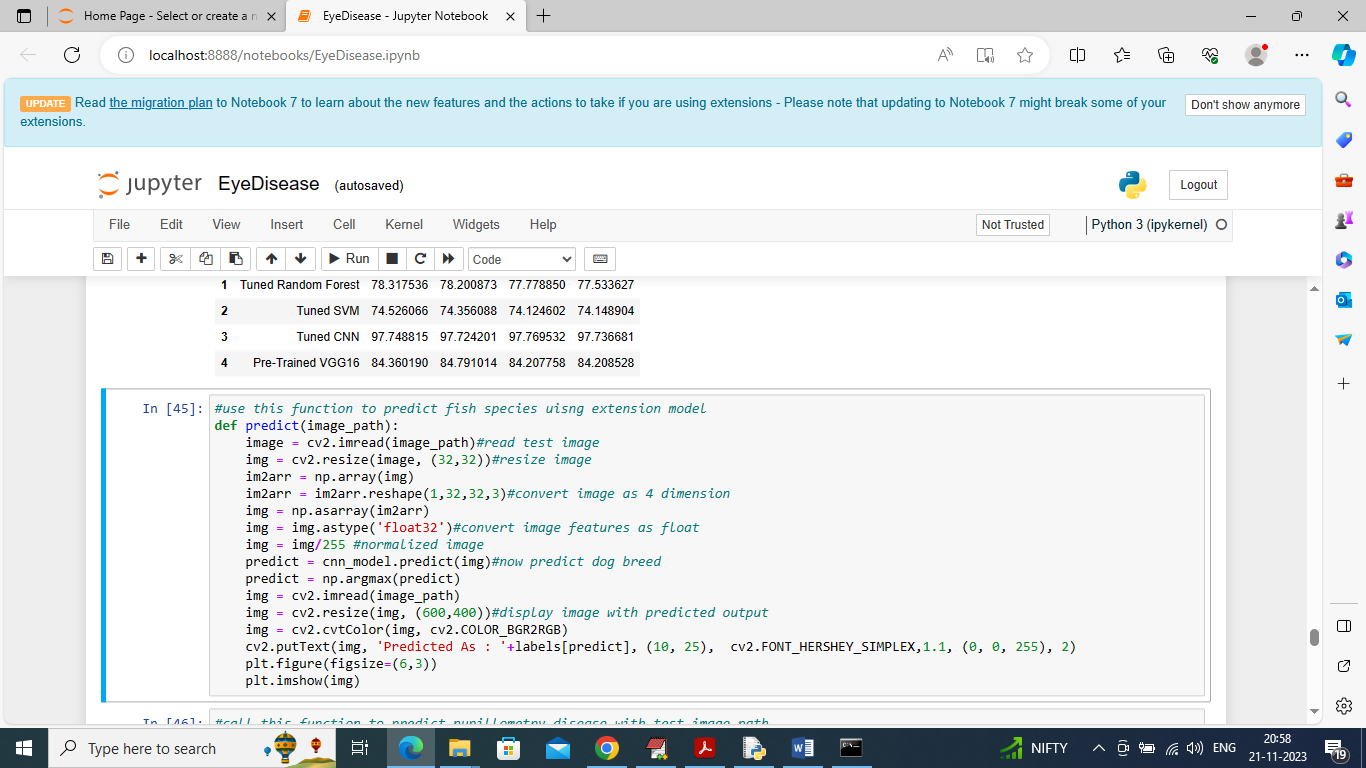
In above screen VGG16 got 83% accuracy and this model is the second highest in accuracy



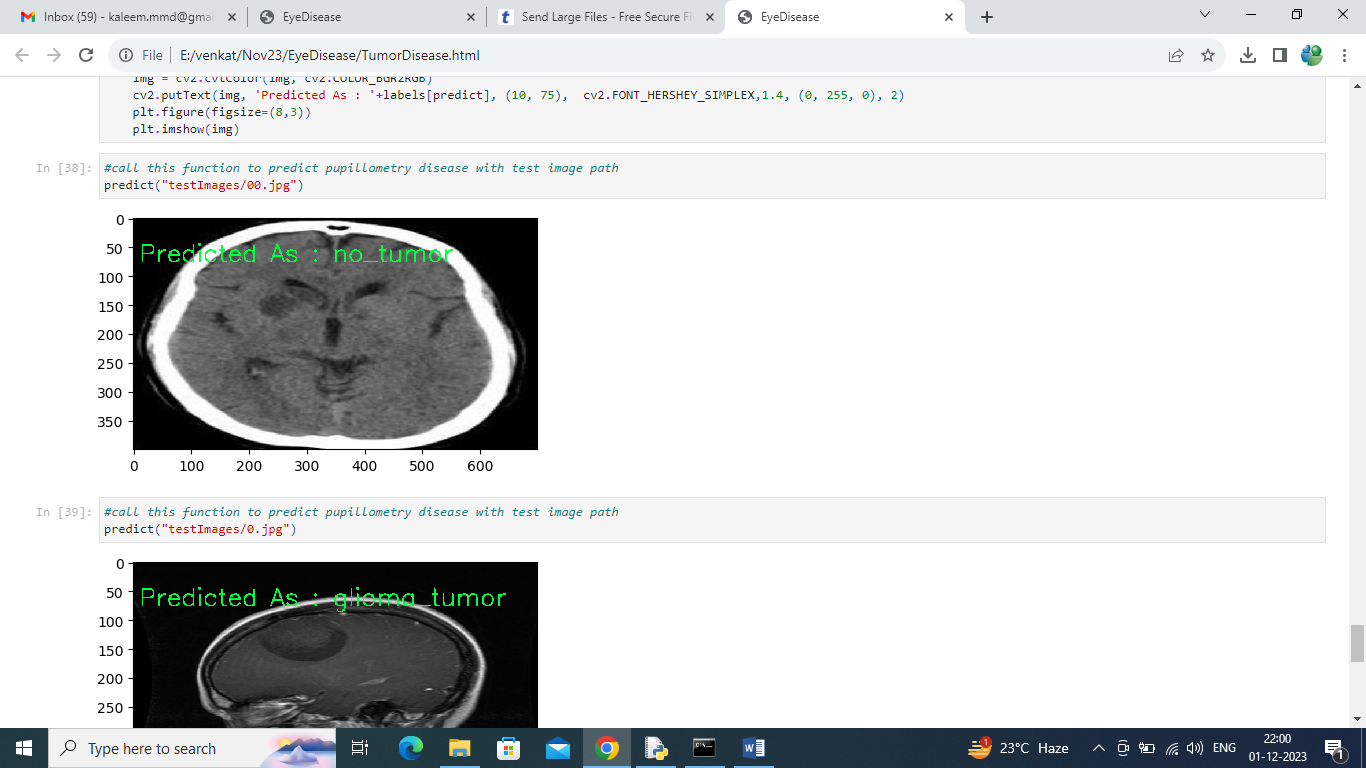
In above graph displaying performance of all algorithms where x-axis represents algorithm names and y-axis represents accuracy and other metrics and in all algorithms CNN got high accuracy



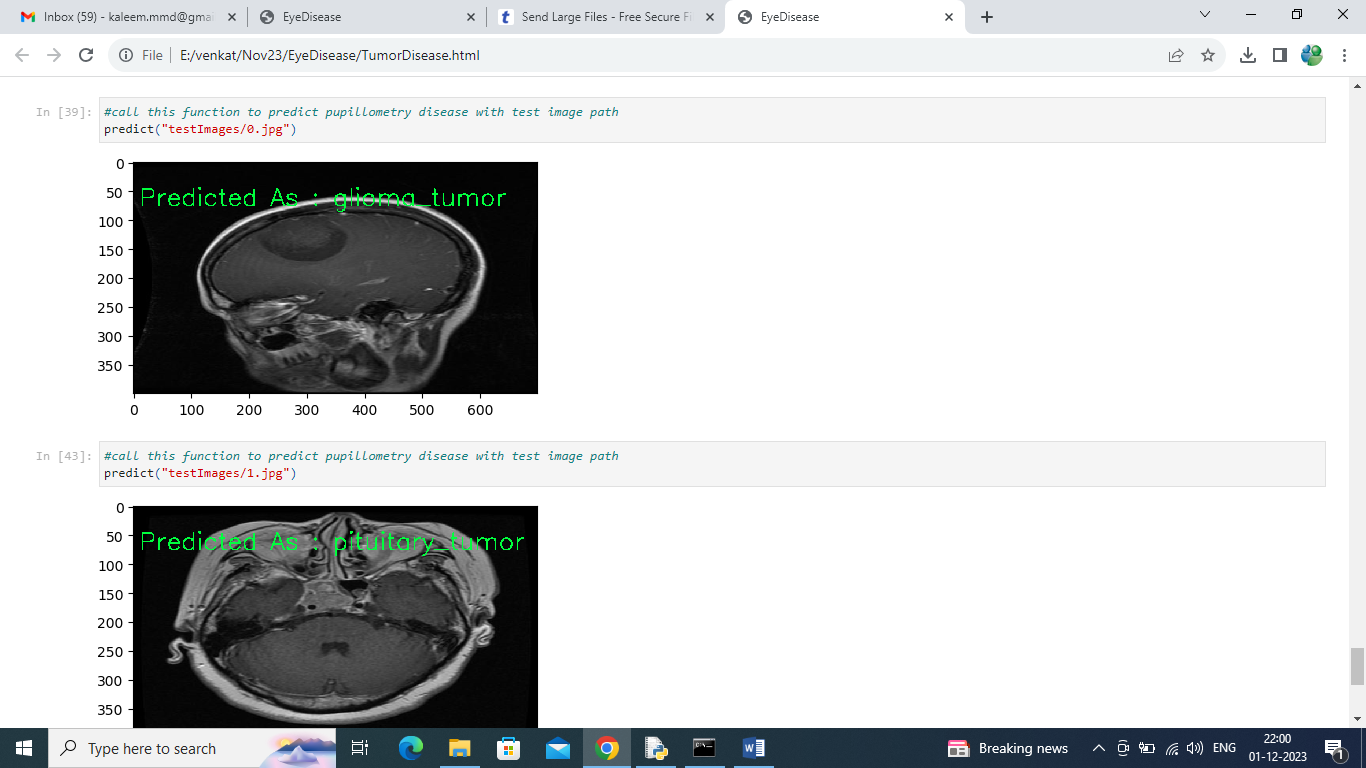
In above screen displaying all algorithm performance in tabular format



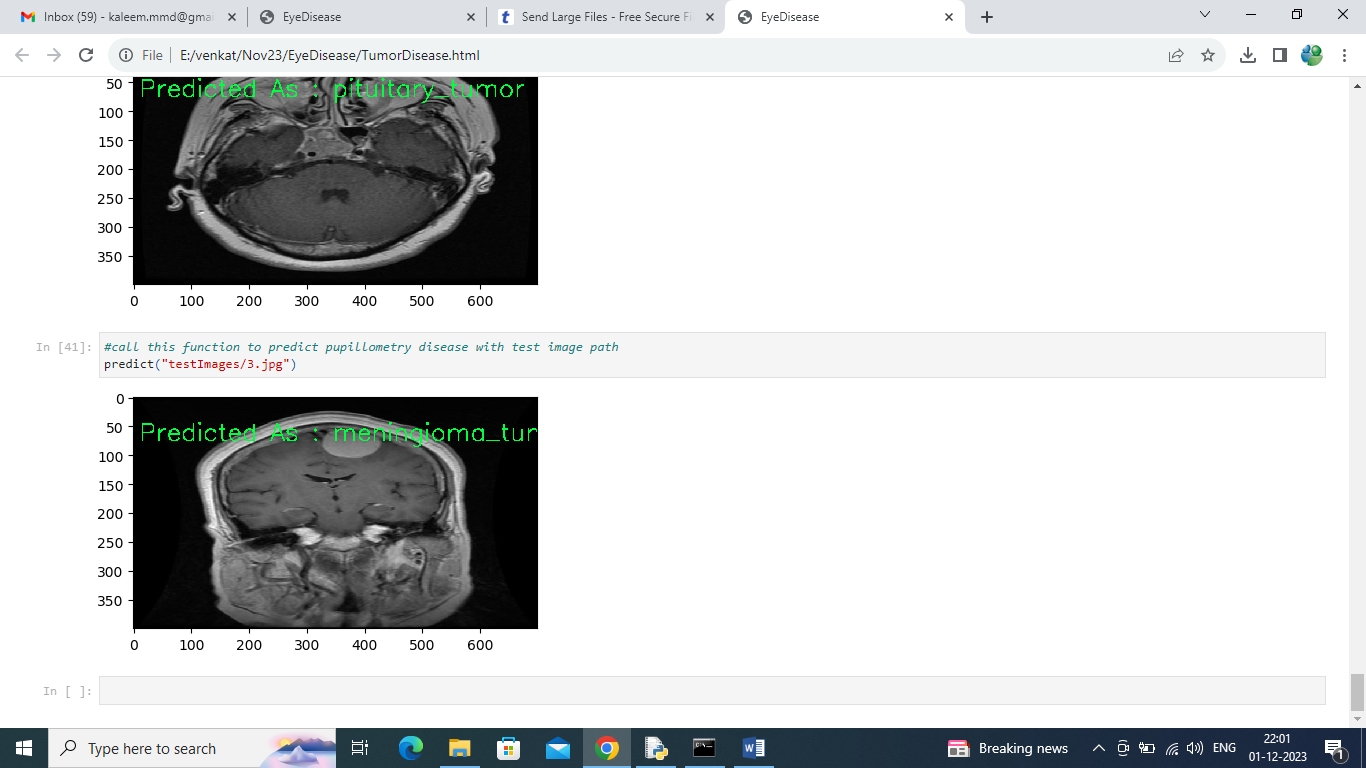
In above screen defining predict function which will take input image path and then predict disease type



In above screen calling predict function with test image path and then in green colour text we can see predicted label



In above screen can see other image predicted disease type



Above screen showing prediction of another image. Similarly by giving test image path we can predict pupillometry disease