* At first, we have imported the libraries
* Then declared some variables
* Then we accessed the train.csv file using pandas
* This train.csv file contain the level of Diabetic retinopathy of each image
* We have defined check\_data function here, which give us the total number of label data, id\_code which indicates the image file name, diagnosis which indicates the level of Diabetic retinopathy
* This describe\_data function gives the detail about csv file
* Image\_category function returns the total number of images in each DR category
* Plot graph function was defined to plot the graph, we can see that No DR have the highest number of images while proliferative DR have the lowest number of images. So as a data analyst we can say it is class imbalanced. It could affect in our model training, so we need to fix this, and we will do this later
* This function gives the total number of healthy and unhealthy eye retinal images
* As we can see in the pie chart, data contain unhealthier eye
* And here we wrote this code to check images
* This image\_processing function gives us the dimension of images
* As we can see here dimension of image is 4D
* And this array\_image function gives us the images in array format
* At first, we have imported the libraries
* Then declared this variable
* Then accessed this csv file
* De defined this function to display the sample images
* Then we have defined this function to add more brightness to the images
* Again, here we have imported these libraries
* Then declared some variables
* It gives us the base directory name
* Then we have accessed the CSV file using pandas
* The data which we have taken from Kaggle have five Diabetic Retinopathy categories
* Where 0 level indicates = No DR, 1 label indicate = Mild, 2 indicates = Moderate, 3 indicates = Severe, and 4 indicates = proliferative DR
* So here we have merged the data and made three categories

No DR, Moderate and Severs

* 2 level is replaced by 1, 3 and 4 is replaced by 2
* So, now
* 0 indicates No DR
* 1 and 2 label will indicated Moderate DR
* And 3 & 4 label indicated Severe DR
* As we can see here now, we have only three labels of images
* We have defined this image\_category function to check total number of images in these three categories.

Plot graph function was defined to plot the bar graph

As we can see No DR have hihges number of retinal images

And Severe have the lowest number of retinal images

* As a data analyst we can say this is class imbalanced, so we need to fix this
* At first, we have created this separate\_img folder
* Where we have copied all the label 2 images
* Then created augment\_img folder
* Then we have increased the label 2 images by using augmentation method
* This is our original image
* This is how our image look liked after data augmentation
* Then we have created separate CSV file for augmented images
* After that we have merged our augmented CSV file with original CSV file
* Now I have copied augmented images into main folder
* This image\_Category function gives us the total number of images in each category after data augmentation
* Now as we see in this graph there is not a huge gap between NO DR, Moderate and severe
* The images that taken from Kaggle dataset cannot be used directly for model training, so we must process that before feed for model training
* At first, we have resized the images to 224\* 224 pixel to main the consistency
* Since there were black border in the images wo we have added more light to the images
* You can see here
* This is the image before increase brightness
* This is the same images after applying some color brightness
* Then we have normalized the data
* Then converted the processed data into numpy
* We can see that our array format image data is in 4D dimension with 224\*224 pixel
* Then we have saved the processed data