1. **Training the binary classifier based on the custom data (Part A)**

   Step 1: Importing the necessary libraries.

   Step 2: Getting the data directories.

   Step 3: Data Processing which includes, Mapping images into data frame.

   Step 4: Turning images path into image.

   Step 5: Data Augmentation

   Step 6: Compiling the model and using early stopping.

   Step 7: Performance evaluation: Printing the Accuracy, Loss plot and confusion matrix

2. **Building a Super Resolution Generative Adversarial Network (SRGAN) for the binary classification for part B**

   Step 1: Importing the necessary libraries.

   Step 2: Getting the data directories.

   Step 3: we are resizing them to 128×128 that will be used as HR images and down sampling images to 32×32 that will be used as LR images.

   Step 4: Creating the data directory for the HR and LR for both DME and DRUSEN classes.

   Step 5: Displaying the resized images.

   Step 6: Defining the generator and discriminator model.

   Step 7: Loading the VGG19 pretrained model.

   Step 8: Combing the generator and discriminator model.

   Step 9: Separating the HR and LR classes and portraying the generated image.

   Step 10: Applying the data augmentation.

   Step 11: Splitting the dataset into train and test (70:30).

   Step 12: Building the GAN model.

   Step 13: Defining the two losses: adversarial loss and content (VGG) loss.

   Strep 14: Creating a list of images for LR and HR in batches from which a batch of images.

   Step 15: Creating a fake image.

Step 16: Training the discriminator on fake and real HR images and the generator by fixing discriminator as non-trainable.

Step 17: Average the discriminator loss.

Step 18: Training the generator via GAN.

Step 19: Finding the generator and discriminator losses.

Step 20: Testing the performance on SRGAN.

Step 21: Predicting the super resolution image from LR images.