Image Classification:

The pictures of houses are classified into : Bathroom, Bedroom, Kitchen, Others.

Keep in mind:

- The location from where Data5.csv is being should be correct. It is currently my local PC's address.
- The aws access key, may not work after my account is deactivated. It needs to be updated accordingly.
- In the last part we can check the prediction for any image in AWS S3 by specifying the Bucket and name of the image. If the image is inside a folder in the bucket, the name will be 'folder/image.jpg'
- We also print the prediction probability of the prediction in form [bathroom,bedroom,kitchen,others]. So if the probability is [0,0,0.9,0.1] the code is 90% sure that it's a kitchen, and 10% sure that it is 'others'. The result will be Kitchen finally.

Part 1 : DataSet Creation (Data.ipynb)

- The DataSet is combination of Images from MLS listings and Google, with about 400-500 images of Bathroom, Bedroom and Kitchen each and about 600-700 other MLS listings image. Total being 2171 images.
- Earlier we went with using all the labels the pictures returned that had confidence >80 to create the Dataset. But lots of unnecessary labels were being returned. Some labels like bird, attic, folding door were very few in number and irrelevant. Labels like flooring, Indoors, room were almost true most of the time and hence irrelevant.
- While selecting the labels, what we can keep in mind is what pointers do we look for while classifying a room. For example, to decide if it's a kitchen, we'll look for the cooktop, oven, sink, etc. So we need to keep only those labels for DataSet creation. These labels are kept separately in 'Imp labels.csv' file.
- In case the label is returned for the type of room itself, with decent confidence, give it a higher value than the rest in order to increase it's weightage. I have assigned the label bathroom a value of 2 in case it is being returned. Similarly, for other 2 kind of rooms.
- Often setting the confidence>80, we won't get enough relevant labels to make accurate prediction. So in that case we lower the confidence threshold till we at least get 3 relevant labels.

Part 2 : Training and testing the data (Classifier.ipynb)

- The dataset we created, we divide it into sets of 80% and 20% randomly. We use 80% to train the classifier, and 20% to test it.
- Here, we got best results from Random Forest Classifier.
- For testing other Images, we use the same logic to generate labels as we did while creating the dataset.
- From input the labels we generate an array, which we use to predict they type of room it is.