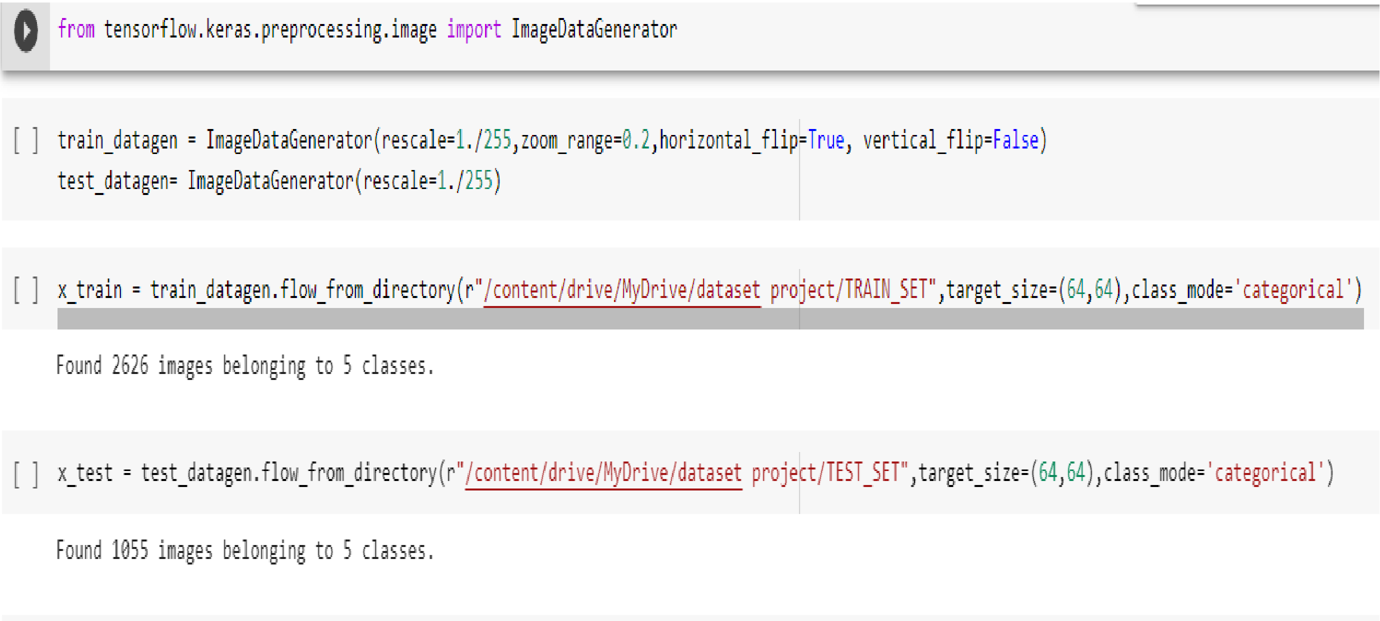
Apply Image DataGenerator Functionality To Trainset And Testset

Let us apply ImageDataGenerator functionality to Trainset and Testset by using the following code
For Training set using flow_from_directory function.

This function will return batches of images from the subdirectories'apples', 'banana', 'orange', 'pineapple', 'watermelon' together with labels 0 to 4{'apples': 0, 'banana': 1, 'orange': 2, 'pineapple': 3, 'watermelon': 4}

Arguments:

- directory: Directory where the data is located. If labels are "inferred", it should contain subdirectories, each containing images for a class. Otherwise, the directory structure is ignored.
- batch_size: Size of the batches of data. Default: 32.
- target_size: Size to resize images after they are read from disk.
- class_mode:
  - ‘int': means that the labels are encoded as integers (e.g. for sparse_categorical_crossentropy loss).
  - 'categorical' means that the labels are encoded as a categorical vector (e.g. for categorical_crossentropy loss).
  - 'binary' means that the labels (there can be only 2) are encoded as float32 scalars with values 0 or 1 (e.g. for binary_crossentropy).
  - None (no labels).



```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

```
train_datagen = ImageDataGenerator(rescale=1./255,zoom_range=0.2,horizontal_flip=True, vertical_flip=False)
test_datagen= ImageDataGenerator(rescale=1./255)
```

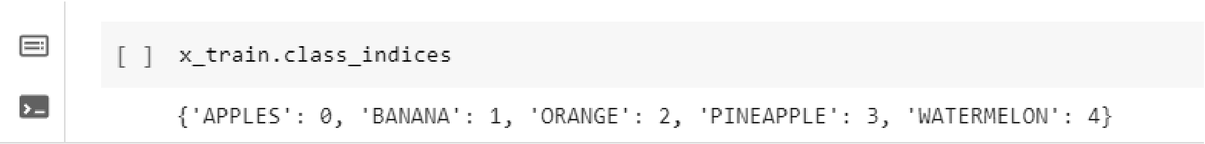
```
x_train = train_datagen.flow_from_directory(r"/content/drive/MyDrive/dataset project/TRAIN_SET",target_size=(64,64),class_mode='categorical')
```

```
Found 2626 images belonging to 5 classes.
```

```
x_test = test_datagen.flow_from_directory(r"/content/drive/MyDrive/dataset project/TEST_SET",target_size=(64,64),class_mode='categorical')
```

```
Found 1055 images belonging to 5 classes.
```

We notice that n images are belonging to 5 classes for training and m images belong to 5 classes for testing purposes.



```
[ ]  x_train.class_indices

     {'APPLES': 0, 'BANANA': 1, 'ORANGE': 2, 'PINEAPPLE': 3, 'WATERMELON': 4}
```

Here we are checking the number of classes in train and test data and counting the number of images in each class of train set data by using the counter function.