AI-powered Nutrition Analyzer for Fitness Enthusiasts

Team Leader   **: S.Kaviya**
Team Member   : M.Maheshwari, A.Sandhiya,M.Inbanila
Faculty Mentor Name : S.V.Rajeshwari

Loading and pre-processing the data:

Data is gold as far as deep learning models are concerned. Your imageclassification model has a far better chance of performing well if you have a good amount of images in the training set. Also, the shape of the data varies according to the architecture/framework that we use.

Hence, the critical data pre-processing step (the eternally important step in anyproject). I highly recommend going through the "basics of image processing using Python

we use Keras' Image Data Generator class to perform data augmentation. i.e, we are using some kind of parameters to process our collected data. The word "augment" means to make something "greater" or "increase" something (in this case, data), the Keras Image Data Generator class actually works by:

- ✓ Accepting a batch of images used for training.
- ✓ Taking this batch and applying a series of random transformations to each image in the batch (including random rotation, resizing, shearing, etc.).
- ✓ Replacing the original batch with the new, randomly transformed batch.
- ✓ Training the CNN on this randomly transformed batch (i.e., the originaldata itself is not used for training).

Note: The Image Data Generator accepts the original data, randomly transformsit, and returns only the new, transformed data.
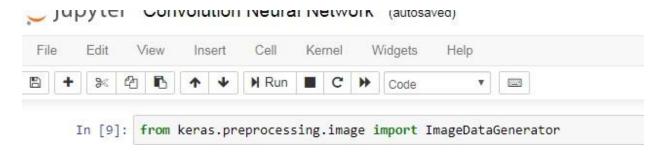
Input Image

Augmented Images

❖ Import the library

```
In [9]: from keras.preprocessing.image import ImageDataGenerator
```

❖ Define the parameters /arguments for Image Data Generator class

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

Note: The Image Data Generator transforms each image in the batch by a series of random translations, these translations are based on the arguments

❖ Applying Image Data Generator functionality to trainset and testset

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

x_train = train_datagen.flow_from_directory(r'E:\dataset\training_set',target_size=(64,64),batch_size=32,class_mode='categorical'
x_test = train_datagen.flow_from_directory(r'E:\dataset\test_set',target_size=(64,64),batch_size=32,class_mode='catagorical')
```

```
Found 8000 images belonging to 2 classes.
Found 2000 images belonging to 2 classes.
```