Upload Image Dataset and Prepare Training Data

```
from google.colab import drive
drive.mount('/content/drive')

!cp /content/drive/MyDrive/Colab_Notebooks/images.zip /content
```

```
Mounted at /content/drive
```

Split images into train, validation, and test folders

```
!mkdir /content/images
!unzip -q images.zip -d /content/images/all
!mkdir /content/images/train; mkdir /content/images/validation; mkdir /content/images/te
```

Next, we'll split the images into train, validation, and test sets. Here's what each set is used for:

1. **Train:** These are the actual images used to train the model. In each step of training, a batch of images from the "train" set is passed into the neural network. The network predicts classes and locations of objects in the images. The training algorithm calculates the loss (i.e. how "wrong" the predictions were) and adjusts the network weights through backpropagation.
2. **Validation:** Images from the "validation" set can be used by the training algorithm to check the progress of training and adjust hyperparameters (like learning rate). Unlike "train" images, these images are only used periodically during training (i.e. once every certain number of training steps).
3. **Test:** These images are never seen by the neural network during training. They are intended to be used by a human to perform final testing of the model to check how accurate the model is. I wrote a Python script to randomly move 80% of the images to the "train" folder, 10% to the "validation" folder, and 10% to the "test" folder. Click play on the following block to download the script and execute it.

```
!wget https://raw.githubusercontent.com/EdjeElectronics/TensorFlow-Lite-Object-Detection
!python train_val_test_split.py
```

```
--2024-09-05 18:23:55--  https://raw.githubusercontent.com/EdjeElectronics/TensorFl
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.108.133,
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.108.133
HTTP request sent, awaiting response... 200 OK
Length: 3100 (3.0K) [text/plain]
Saving to: ‘train_val_test_split.py’

train_val_test_spli 100%[===================>]   3.03K  --.-KB/s    in 0s
```