

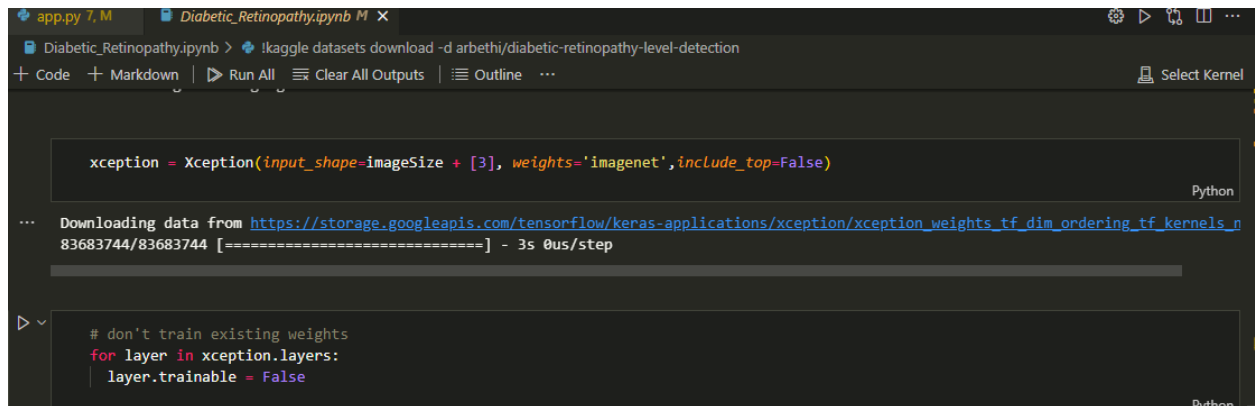
Pre-Trained CNN Model As A Feature Extractor

For one of the models, we will use it as a simple feature extractor by freezing all the five convolution blocks to make sure their weights don't get updated after each epoch as we train our own model.

Here, we have considered images of dimension (229,229,3).

Also, we have assigned `include_top = False` because we are using convolution layer for features extraction and wants to train fully connected layer for our images classification(since it is not the part of Imagenet dataset)

Flatten layer flattens the input. Does not affect the batch size.



```
app.py 7, M | Diabetic_Retinopathy.ipynb M X
Diabetic_Retinopathy.ipynb > | Kaggle datasets download -d arbethi/diabetic-retinopathy-level-detection
+ Code + Markdown | Run All | Clear All Outputs | Outline ... | Select Kernel

xception = Xception(input_shape=imageSize + [3], weights='imagenet',include_top=False)
Python

... Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/xception/xception_weights_tf_dim_ordering_tf_kernels_n
83683744/83683744 [=====] - 3s 0us/step

# don't train existing weights
for layer in xception.layers:
    layer.trainable = False
Python
```