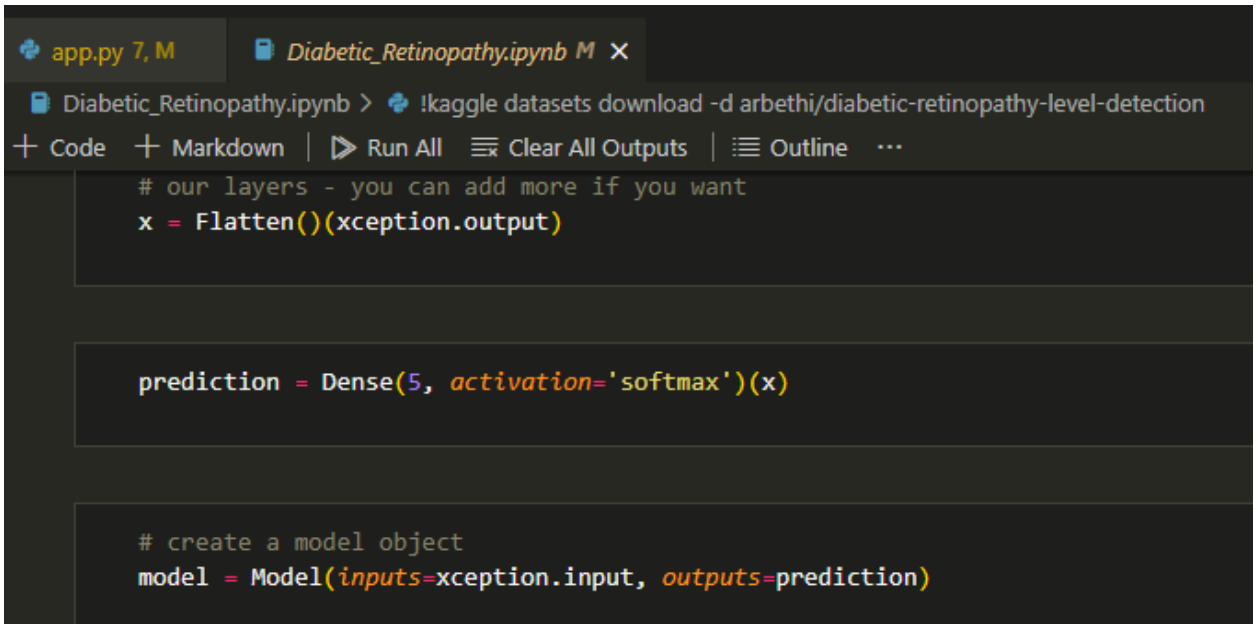
Adding Dense Layers

A dense layer is a deeply connected neural network layer. It is the most common and frequently used layer.

Let us create a model object named model with inputs as xception.input and output as dense layer.



The number of neurons in the Dense layer is the same as the number of classes in the training set. The neurons in the last Dense layer, use softmax activation to convert their outputs into respective probabilities.

Understanding the model is a very important phase to properly use it for training and prediction purposes. Keras provides a simple method, summary to get the full information about the model and its layers.

app.py 7, M | *Diabetic_Retinopathy.ipynb M* ×

Diabetic_Retinopathy.ipynb > # view the structure of the model

+ Code + Markdown | Run All | Clear All Outputs | Outline ···

```
# view the structure of the model
model.summary()
```

```
Model: "model"
_________________________________________________________________________________________________
 Layer (type)                Output Shape              Param #     Connected to
=================================================================================================
 input_1 (InputLayer)        [(None, 299, 299, 3       0           []
                             )]

 block1_conv1 (Conv2D)       (None, 149, 149, 32       864         ['input_1[0][0]']
                             )

 block1_conv1_bn (BatchNormaliz  (None, 149, 149, 32   128         ['block1_conv1[0][0]']
 ation)                      )

 block1_conv1_act (Activation)  (None, 149, 149, 32    0           ['block1_conv1_bn[0][0]']
                             )

 block1_conv2 (Conv2D)       (None, 147, 147, 64       18432       ['block1_conv1_act[0][0]']
                             )

 block1_conv2_bn (BatchNormaliz  (None, 147, 147, 64   256         ['block1_conv2[0][0]']
 ation)                      )

 block1_conv2_act (Activation)  (None, 147, 147, 64    0           ['block1_conv2_bn[0][0]']
                             )

...
Total params: 21,885,485
Trainable params: 1,024,005
Non-trainable params: 20,861,480
_________________________________________________________________________________________________
```

Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...