# EMERGING METHODS FOR EARLY DETECTION OF FOREST FIRE

**MODEL BUILDING TRAINING THE MODEL**

|  |  |
| --- | --- |
| Team ID | PNT2022TMID34010 |
| Project Name | Project- Emerging Methods For Early Detection of Forest Fire |

# TRAINING THE MODEL:

At this point, we have training data and a fully configured neural network to train. All that is left is to pass the data to the model for the training process to commence, a process that is completed by iterating on the training data. Training begins by calling the fit ( ) method.

The arguments are the batch size as you are using “adam” (bath gradient descent and epochs: no: of times the model should get trained).

steps\_per\_epoch:

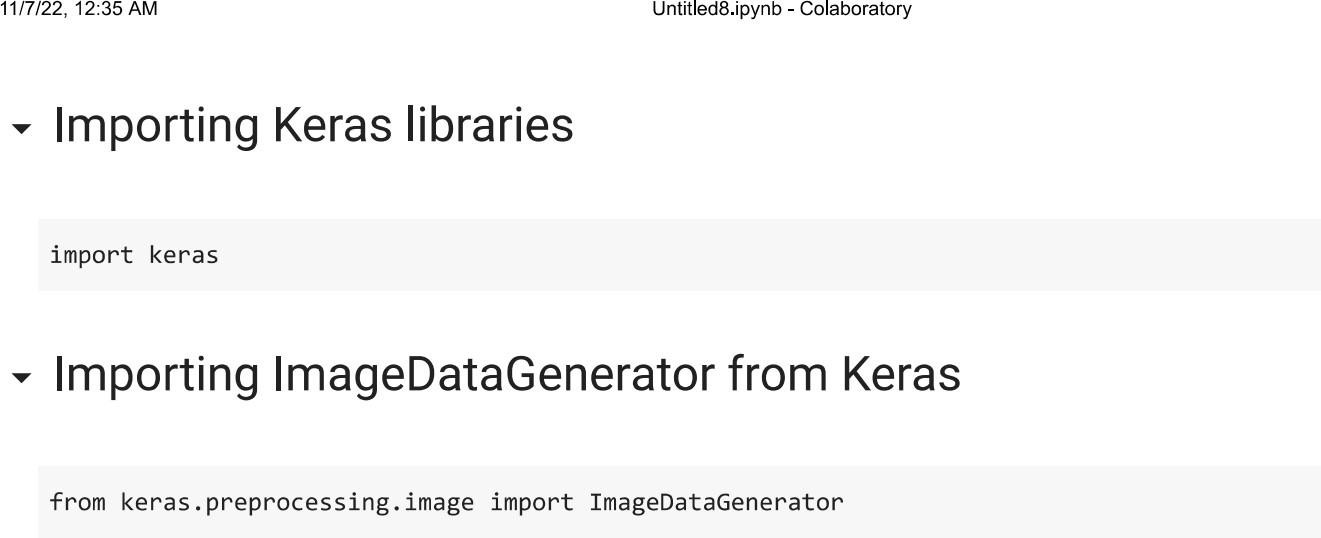
* It specifies the total number of steps taken from the generator as soon as one epoch is finished and the next epoch has started. We can calculate the value of steps\_per\_epoch as the total number of samples in your training folder divided by the batch size.
* Epochs:an integer and number of epochs we want to train our model for.
* Validation\_datacan be either input and targets list generator inputs, targets, and sample\_weights list which can be used to evaluate.

The loss and metrics for any model after any epoch has ended.

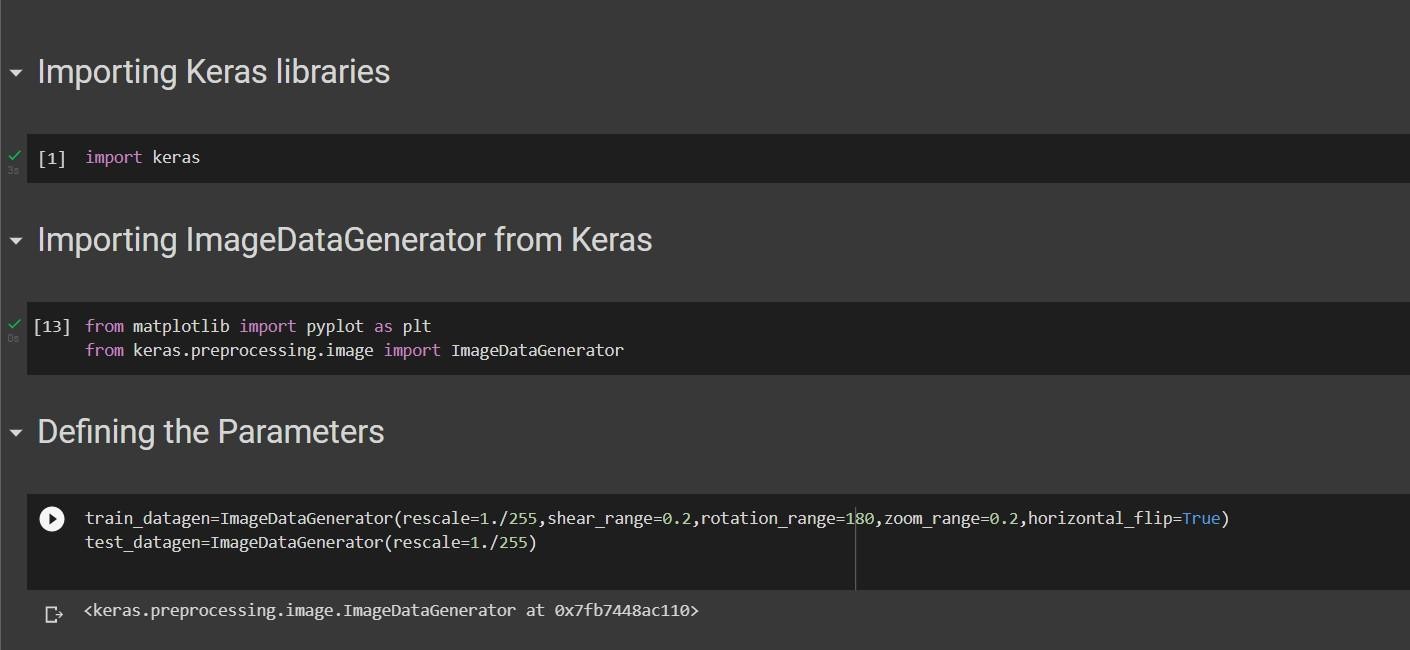
* Validation\_steps:

Only if the validation\_data is a generator then only this argument can be used. It specifies the total number of steps taken from the generator before it is stopped at every epoch and its value is calculated as the total number of validation data points in your dataset divided by the validation batch size.

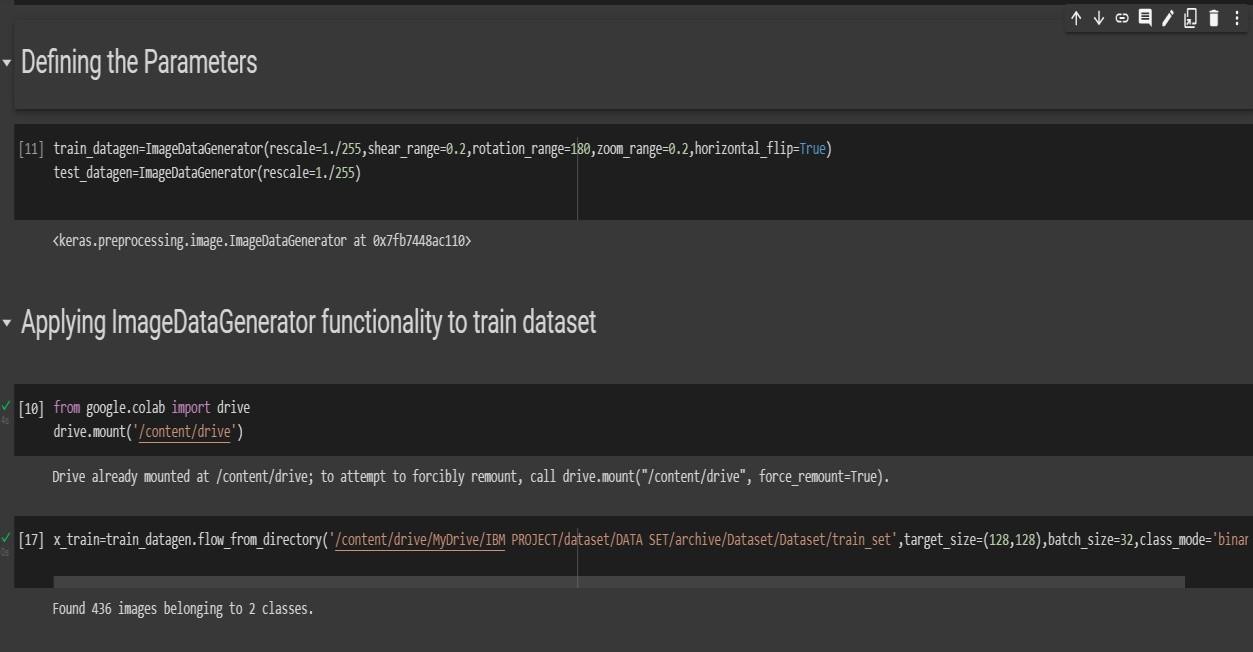
**IMPORT LIBRARIES:**



**IMPORT ImageDataGenerator FROM KERAS:**

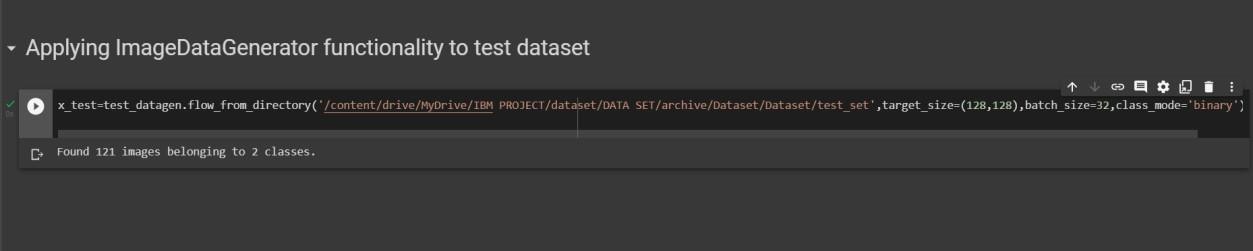


# APPLYING ImageDataGenerator to train dataset:

ply**flow\_from\_directory ( )**methodfor Train folder.

# APPLYING ImageDataGenerator to test dataset:

Applying the **flow\_from\_directory ( )** methodfortest folder.



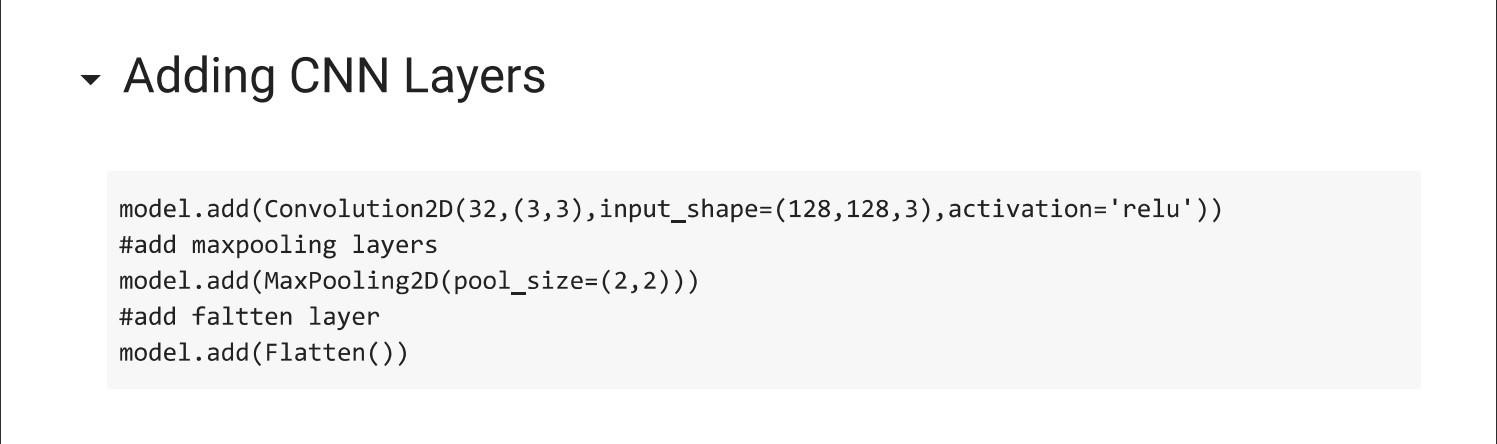
# IMPORTING MODEL BUILDING LIBRARIES:



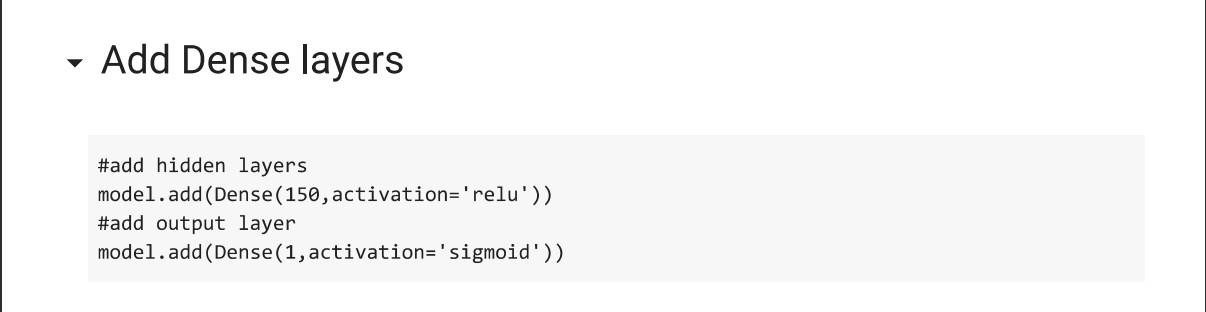
**INITIALIZING THE MODEL:**



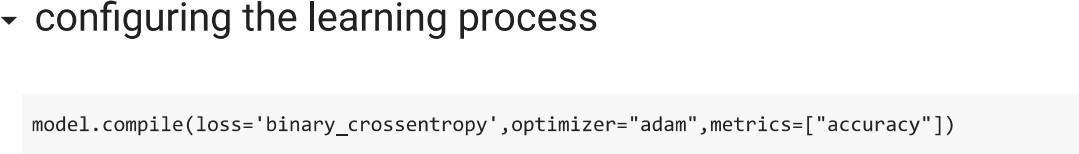
**ADDING CNN LAYERS:**



**ADDING DENSE LAYERS:**



**CONFIGURING THE LEARNING PROCESS:**



**TRAINING THE MODEL:**

