Downloading the data

#Importing Libraries

#Locating and loading datasets

import pathlib

from pathlib import Path

import os, gc, glob, random

from PIL import Image

#DataManagement and matrix calculations

import pandas as pd

import numpy as np

#Model Building

import tensorflow as tf

import keras

import keras.backend as K

from keras.optimizers import SGD, Adam, Adagrad, RMSprop

from keras.applications import \*

from keras.preprocessing import \*

from keras.preprocessing.image import ImageDataGenerator

from keras.callbacks import EarlyStopping, ModelCheckpoint

from keras.models import Sequential

from keras.layers import Dense, Conv2D, MaxPool2D, Flatten, Activation,

BatchNormalization,Dropout

from keras.models import Model

from keras.utils.np\_utils import to\_categorical

from sklearn.model\_selection import train\_test\_split

# Data Visualization

import matplotlib.pyplot as plt

#Loading and testing models

from keras.models import load\_model

from keras.models import model\_from\_json

# Directory operations

import os

from os import listdir

#

=====================================================================================

================================================= #

#

=====================================================================================

================================================= #

# ===========================================DEFINING THE REQUIRED

FUNCTIONS============================================================ #

#

=====================================================================================

================================================= #

#

=====================================================================================

================================================= #

def generateListofFiles(dirName):

&quot;&quot;&quot;This function returns a list with exact paths of files inside the given

directory &quot;&quot;&quot;

listOfFile = os.listdir(dirName)

allFiles = list()

for fol\_name in listOfFile:

fullPath = os.path.join(dirName, fol\_name)

allFiles.append(fullPath)

return allFiles

def Configure\_CNN\_Model(output\_size):

&quot;&quot;&quot;This function defines the cnn model structure and configures the layers&quot;&quot;&quot;

K.clear\_session()

model = Sequential()

model.add(Dropout(0.4,input\_shape=(224, 224, 3)))

model.add(Conv2D(256, (5, 5),input\_shape=(224, 224, 3),activation=&#39;relu&#39;))

model.add(MaxPool2D(pool\_size=(2, 2)))

#model.add(BatchNormalization())

model.add(Conv2D(128, (3, 3),