

### ASSIGNMENT-3

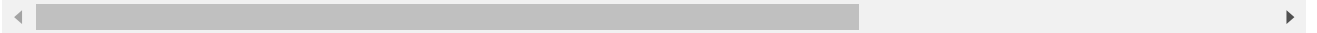
Assignment Date	6 October 2022
Student Name	Gokul Prasath.A
Student Roll Number	732119104020
Maximum Marks	2 Marks

## 1.Download the dataset:

### Import Dataset from Drive and Unzip it

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

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.m



```
!unzip '/content/drive/MyDrive/Flowers-Dataset.zip'
```

inflating: flowers/tulip/8521597402\_4b6169ba05.jpg inflating:  
flowers/tulip/8523133474\_d2c0845b54.jpg inflating:  
flowers/tulip/8554190977\_37ac747799\_m.jpg inflating:  
flowers/tulip/8555123165\_2fe57eff4f.jpg inflating:  
flowers/tulip/8562853756\_73778dac25\_n.jpg inflating:  
flowers/tulip/8572847041\_d0cc07861f\_n.jpg inflating:  
flowers/tulip/8585101979\_4398146bf1\_n.jpg inflating:  
flowers/tulip/8585102511\_fc452e6700\_n.jpg inflating:  
flowers/tulip/8585102913\_d80d3e2ff7\_n.jpg inflating:  
flowers/tulip/8585103457\_d64697c3cf\_n.jpg inflating:  
flowers/tulip/8586204750\_2891bd2ec9\_n.jpg inflating:  
flowers/tulip/8586205168\_8294e67195\_n.jpg inflating:  
flowers/tulip/8586205446\_8953a6c70e\_n.jpg inflating:  
flowers/tulip/8586205946\_cda045f3f8\_n.jpg inflating:  
flowers/tulip/8601596054\_33e40c2a7a.jpg inflating:  
flowers/tulip/8603340662\_0779bd87fd.jpg inflating:  
flowers/tulip/8605564823\_7a59d3d92a.jpg inflating:  
flowers/tulip/8614237582\_74417799f4\_m.jpg inflating:  
flowers/tulip/8619064872\_dea79a9eb9.jpg inflating:  
flowers/tulip/8622237974\_b362574785\_n.jpg inflating:  
flowers/tulip/8623170936\_83f4152431.jpg inflating:  
flowers/tulip/8623173256\_3f0eb4c506.jpg inflating:  
flowers/tulip/8628453641\_6f87755815\_m.jpg inflating:  
flowers/tulip/8659691170\_09db83d023.jpg inflating:  
flowers/tulip/8668973377\_c69527db42\_m.jpg inflating:  
flowers/tulip/8668974855\_8389ecbdca\_m.jpg inflating:  
flowers/tulip/8669794378\_97dda6036f\_n.jpg inflating:  
flowers/tulip/8673412732\_f8fd690ee4\_n.jpg inflating:  
flowers/tulip/8673416166\_620fc18e2f\_n.jpg inflating:  
flowers/tulip/8673416556\_639f5c88f1\_n.jpg inflating:  
flowers/tulip/8677713853\_1312f65e71.jpg inflating:  
flowers/tulip/8681825637\_837a63513a\_n.jpg inflating:  
flowers/tulip/8686013485\_3c4dfbfd1f\_n.jpg inflating:  
flowers/tulip/8686332852\_c6dcb2e86b.jpg inflating:  
flowers/tulip/8687675254\_c93f50d8b0\_m.jpg inflating:  
flowers/tulip/8688502760\_1c8d6de921\_m.jpg inflating:  
flowers/tulip/8689672277\_b289909f97\_n.jpg inflating:  
flowers/tulip/8690789564\_394eb04982\_n.jpg inflating:  
flowers/tulip/8690791226\_b1f015259f\_n.jpg inflating:  
flowers/tulip/8695367666\_0809529eaf\_n.jpg inflating:  
flowers/tulip/8695372372\_302135aeb2.jpg inflating:  
flowers/tulip/8697784345\_e75913d220.jpg

inflating: flowers/tulip/8702982836\_75222725d7.jpg inflating:  
flowers/tulip/8706523526\_a0f161b72b.jpg inflating:  
flowers/tulip/8708209606\_d3aede4801.jpg inflating:  
flowers/tulip/8708856019\_f3be2353a4\_n.jpg inflating:  
flowers/tulip/8710148289\_6fc196a0f8\_n.jpg inflating:  
flowers/tulip/8711277462\_b43df5454b\_m.jpg inflating:  
flowers/tulip/8712230357\_1298b8513b.jpg inflating:  
flowers/tulip/8712243901\_54d686319e\_m.jpg inflating:  
flowers/tulip/8712244311\_da8e90bf8e\_n.jpg inflating:  
flowers/tulip/8712260079\_c0ff42e0e2\_n.jpg inflating:  
flowers/tulip/8712263493\_3db76c5f82.jpg inflating:  
flowers/tulip/8712266605\_3787e346cd\_n.jpg inflating:  
flowers/tulip/8712267391\_c756f18ee7\_n.jpg inflating:  
flowers/tulip/8712267813\_f7a9be2ec5.jpg inflating:  
flowers/tulip/8712268519\_f4c2c39a06\_n.jpg

## 2.Data Augmentation:

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

```
train_gen = ImageDataGenerator(rescale=1./255,  
                                zoom_range=0.2,  
                                horizontal_flip=True) test_gen =  
ImageDataGenerator(rescale=1./255)
```

```
xtrain = train_gen.flow_from_directory('/content/flowers',  
                                        target_size=(64,64),  
                                        class_mode='categorical',  
                                        batch_size=100)
```

Found 4317 images belonging to 5 classes.

## 3.Create the model:

```
from tensorflow.keras.models import Sequential  
from tensorflow.keras.layers import Convolution2D, MaxPooling2D, Dense, Flatten  
from keras.callbacks import EarlyStopping, ReduceLROnPlateau
```

```
model = Sequential()
```

## 4.Add Layers(Convolution, MaxPooling, Flatten):

```
model.add(Convolution2D(32,(3,3),activation='relu',input_shape=(64,64,3))  
) model.add(MaxPooling2D((2,2))) model.add(Flatten())
```

```
model.add(Dense(400,activation='relu'))  
model.add(Dense(200,activation='relu'))  
model.add(Dense(100,activation='relu'))  
model.add(Dense(5,activation='softmax'))
```

## 5. Compile the model:

```
model.compile(optimizer='adam',loss='categorical_crossentropy',metrics=['accuracy'])  
early_stopping = EarlyStopping(monitor='accuracy',patience=3)  
reduce_lr = ReduceLROnPlateau(monitor='accuracy',patience=5,factor=0.5,min_lr=0.00001)  
callback = [reduce_lr,early_stopping]
```

## 6. Fit the model:

```
model.fit_generator(xtrain, steps_per_epoch = len(xtrain),
```

```
callbacks=callback, epochs=100)
```

```
Epoch 9/100
44/44 [=====] - 27s 616ms/step - loss: 0.7767 -
accuracy: Epoch 10/100
44/44 [=====] - 27s 616ms/step - loss: 0.7265 -
accuracy: Epoch 11/100
44/44 [=====] - 29s 662ms/step - loss: 0.7027 -
accuracy:
Epoch 12/100
44/44 [=====] - 27s 614ms/step - loss: 0.6503 -
accuracy: Epoch 13/100
44/44 [=====] - 27s 615ms/step - loss: 0.6306 -
accuracy: Epoch 14/100
44/44 [=====] - 27s 615ms/step - loss: 0.5722 -
accuracy:
Epoch 15/100
44/44 [=====] - 27s 614ms/step - loss: 0.5580 -
accuracy:
Epoch 16/100
44/44 [=====] - 29s 654ms/step - loss: 0.5490 -
accuracy: Epoch 17/100
44/44 [=====] - 27s 618ms/step - loss: 0.5037 -
accuracy: Epoch 18/100
44/44 [=====] - 27s 617ms/step - loss: 0.4643 -
accuracy:
Epoch 19/100
44/44 [=====] - 27s 614ms/step - loss: 0.4619 -
accuracy: Epoch 20/100
44/44 [=====] - 27s 618ms/step - loss: 0.4153 -
accuracy: Epoch 21/100
44/44 [=====] - 29s 658ms/step - loss: 0.3803 -
accuracy: Epoch 22/100
44/44 [=====] - 27s 613ms/step - loss: 0.3659 -
accuracy: Epoch 23/100
44/44 [=====] - 27s 614ms/step - loss: 0.3458 -
accuracy: Epoch 24/100
44/44 [=====] - 27s 615ms/step - loss: 0.3449 -
accuracy:
Epoch 25/100
44/44 [=====] - 27s 613ms/step - loss: 0.3288 -
accuracy:
Epoch 26/100
44/44 [=====] - 29s 659ms/step - loss: 0.3370 -
accuracy: Epoch 27/100
44/44 [=====] - 27s 614ms/step - loss: 0.2959 -
accuracy: Epoch 28/100
44/44 [=====] - 27s 613ms/step - loss: 0.2518 - accuracy:
Epoch 29/100
44/44 [=====] - 27s 611ms/step - loss: 0.2707 - accuracy:
Epoch 30/100
44/44 [=====] - 27s 614ms/step - loss: 0.2498 - accuracy:
Epoch 31/100
44/44 [ ] 27s 613ms/step loss: 0.2335 accuracy:
```

```

44/44 [=====] - 27s 613ms/step - loss: 0.2335 - accuracy:
Epoch 32/100
44/44 [=====] - 27s 612ms/step - loss: 0.2134 - accuracy:
Epoch 33/100
44/44 [=====] - 27s 613ms/step - loss: 0.1918 - accuracy:
Epoch 34/100
44/44 [=====] - 27s 613ms/step - loss: 0.2038 - accuracy:
Epoch 35/100
44/44 [=====] - 27s 610ms/step - loss: 0.1908 - accuracy:
Epoch 36/100
44/44 [=====] - 27s 614ms/step - loss: 0.2035 - accuracy:
<keras.callbacks.History at 0x7facb3ae6350>

```

## 7. Save the model:

```
model.save('flower.h5')
```

## 8. Test The Model:

```
import numpy as np from
tensorflow.keras.preprocessing
```

```
import image img =
image.load_img('/content/flowers/rose/100908241
83_d02c613f10_m.jpg',target_size=(64,6)
```

img



```
h = image.img_to_array(img)
h
```

```

array([[14., 22.,  7.],
 [11., 22.,  6.],
        [ 8., 19.,  3.],
        ...,
        [32., 47., 24.],
        [30., 48., 22.],
        [33., 49., 23.]],

      [[13., 20., 12.],
        [11., 21., 10.],
        [11., 22.,  8.],
        ...,
        [37., 51., 26.],
        [35., 49., 26.],
        [25., 45., 20.]],

```

```

[[19., 30., 16.],
 [19., 31., 17.],
 [16., 29., 12.],
 ...,
 [31., 47., 20.],
 [28., 49., 18.],
 [27., 43., 17.]],
 ...,
 [[15., 17., 6.],
 [ 2., 9., 2.],
 [ 2., 9., 1.],
 ...,
 [ 8., 21., 11.],
 [ 2., 12., 3.],
 [ 9., 16., 9.]],

[[12., 20., 9.],
 [ 1., 8., 1.],
 [ 5., 10., 3.],
 ...,
 [ 3., 8., 2.],
 [ 6., 16., 5.],
 [ 5., 7., 4.]],

[[24., 27., 18.],
 [11., 21., 13.],
 [ 8., 13., 6.],
 ...,
 [ 1., 6., 0.],
 [ 2., 9., 1.],
 [ 2., 9., 1.]]], dtype=float32)

```

```

h= np.expand_dims(h,axis= 0)
h

```

```

array([[[[14., 22., 7.],
 [11., 22., 6.],
 [ 8., 19., 3.],
 ...,
 [32., 47., 24.],
 [30., 48., 22.],
 [33., 49., 23.]],

[[13., 20., 12.],
 [11., 21., 10.],
 [11., 22., 8.],
 ...,
 [37., 51., 26.],
 [35., 49., 26.],
 [25., 45., 20.]],

[[19., 30., 16.],
 [19., 31., 17.],
 [16., 29., 12.],
 ...,
 [31., 47., 20.],
 [28., 49., 18.],
 [27., 43., 17.]],

```

...,

```
[[15., 17., 6.],  
 [ 2.,  9., 2.],  
 [ 2.,  9., 1.],
```