ASSIGNMENT-3

Assignment Date	30 September 2022
Student Name	M.Vivek
Student Roll Number	962719106039
Maximum Marks	2 Marks

1. Download the Dataset

Link: https://drive.google.com/file/d/1zZ87e7GDpN90-Sa_AKbvMm3EEfQkEQ_R/view

2. Image Augmentation

Solution:

```python
pwd
from tensorflow.keras.preprocessing.image import ImageDataGenerator
train_datagen=ImageDataGenerator(rescale=1./255,zoom_range=0.2,horizontal_flip=True,vertical_flip=False)
test_datagen=ImageDataGenerator(rescale=1./255)
ls
pwd
x_train=train_datagen.flow_from_directory(r"/content/drive/MyDrive/flowers",target_size=(64,64),
                                           class_mode='categorical',batch_size=24)
x_test=test_datagen.flow_from_directory(r"/content/drive/MyDrive/flowers",target_size=(64,64),
                                        class_mode='categorical',batch_size=24)
x_train.class_indices
```

Image Augmentation

```python
[9] pwd
    '/content/drive/MyDrive'

[10] from tensorflow.keras.preprocessing.image import ImageDataGenerator

[11] train_datagen=ImageDataGenerator(rescale=1./255,zoom_range=0.2,horizontal_flip=True,vertical_flip=False)

[12] test_datagen=ImageDataGenerator(rescale=1./255)

[13] ls
     685imguf_NAD-student-registration-Process19.pdf   Flowers-Dataset.zip
     Classroom/                                        'Getting started.pdf'
    'Colab Notebooks'/                                 'Student Registration'
     flowers/                                          'Student Registration (1)'

[14] pwd
     '/content/drive/MyDrive'
```

```
[14] pwd

     '/content/drive/MyDrive'

[15] x_train=train_datagen.flow_from_directory(r"/content/drive/MyDrive/flowers",target_size=(64,64),
                                                class_mode='categorical',batch_size=24)

     Found 4317 images belonging to 6 classes.

[16] x_test=test_datagen.flow_from_directory(r"/content/drive/MyDrive/flowers",target_size=(64,64),
                                              class_mode='categorical',batch_size=24)

     Found 4317 images belonging to 6 classes.

[18] x_train.class_indices

     {'.ipynb_checkpoints': 0,
      'daisy': 1,
      'dandelion': 2,
      'rose': 3,
      'sunflower': 4,
      'tulip': 5}
```
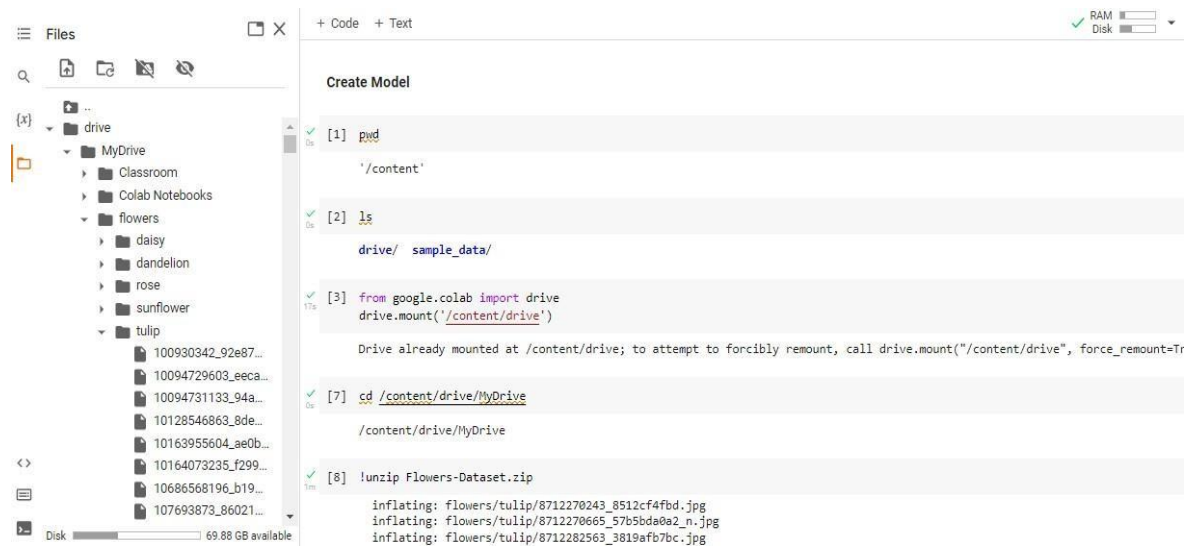
3. Create Model

Solution:

```
pwd
ls
from google.colab import drive
drive.mount('/content/drive')
cd /content/drive/MyDrive
!unzip Flowers-Dataset.zip
```
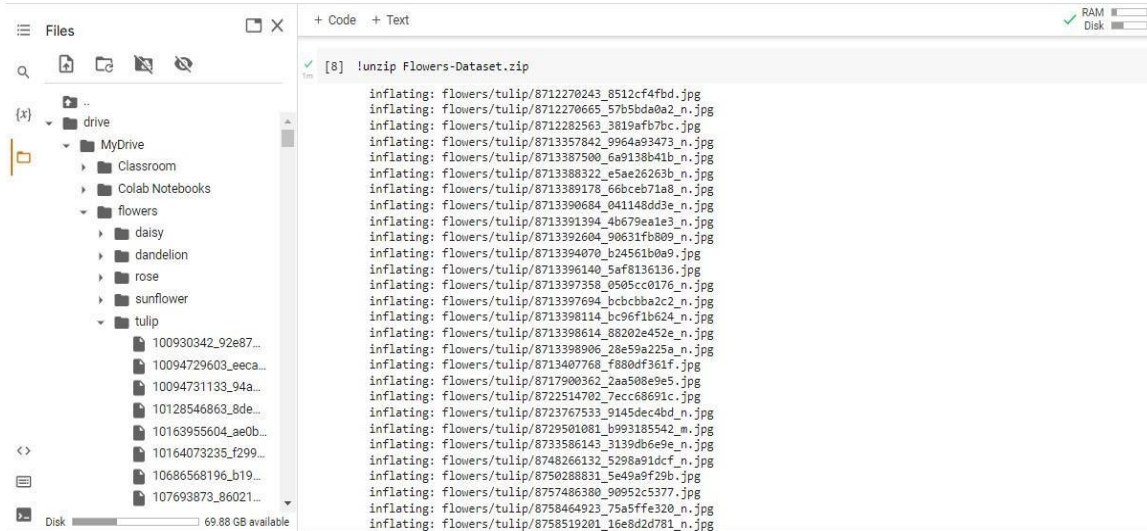
4. Add Layers(Convolution,Maxpooling,Flatten,Dense-(Hidden Layers),Output)

Solution:

```python
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense,Convolution2D,MaxPooling2D,Flatten
model=Sequential()
model.add(Convolution2D(32,(3,3),input_shape=(64,64,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())
model.summary()
32*(3*3*3+1)
```

Hidden layer

```python
model.add(Dense(300,activation='relu'))
model.add(Dense(150,activation='relu'))
```

Output layer

```python
model.add(Dense(4,activation='softmax'))
```

+ Code + Text

[24] model.summary()

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d (MaxPooling2D)	(None, 31, 31, 32)	0
flatten (Flatten)	(None, 30752)	0

Total params: 896
Trainable params: 896
Non-trainable params: 0

[25] 32*(3*3*3+1)

896

Hidden Layers

[26] model.add(Dense(300,activation='relu'))

Files

[25] 32*(3*3*3+1)

896

Hidden Layers

[26] model.add(Dense(300,activation='relu'))
     model.add(Dense(150,activation='relu'))

Output Layers

[27] model.add(Dense(4,activation='softmax'))

Compile the model

[28] model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])

[29] len(x_train)

180

---

5. Compile The Model

Solution:

```
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
len(x_train)
1238/24
```
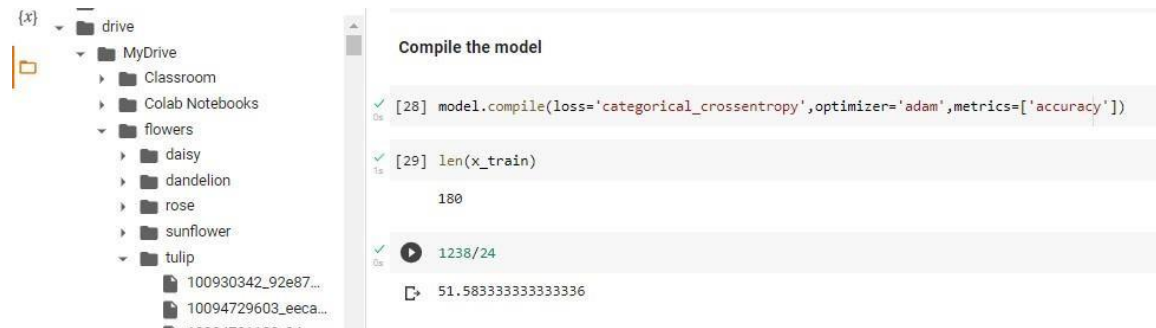
6. Fit The Model

Solution:

```python
import numpy as np
from sklearn.linear_model import LinearRegression
x=30*np.random.random((20,1))
y=0.5*x+1.0+np.random.normal(size=x.shape)
model=LinearRegression()
model.fit(x,y)
x_new=np.linspace(0,30,100)
y_new=model.predict(x_new[:,np.newaxis])
```
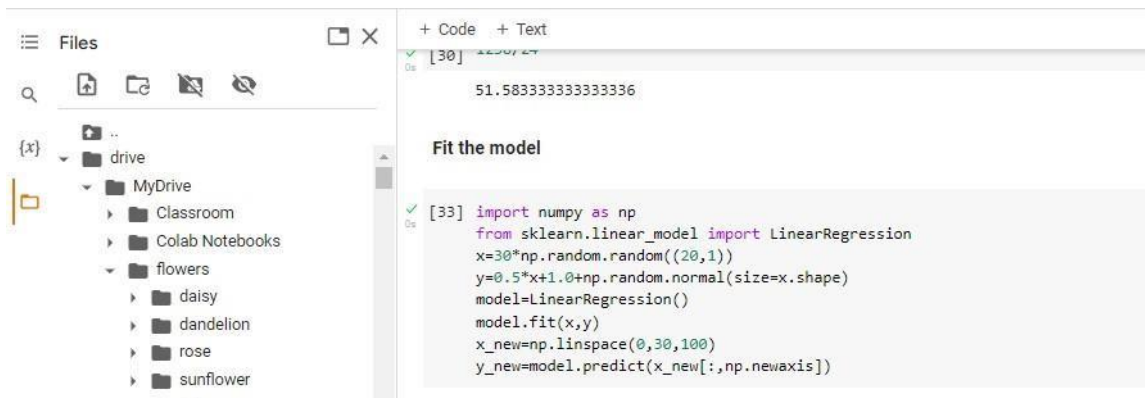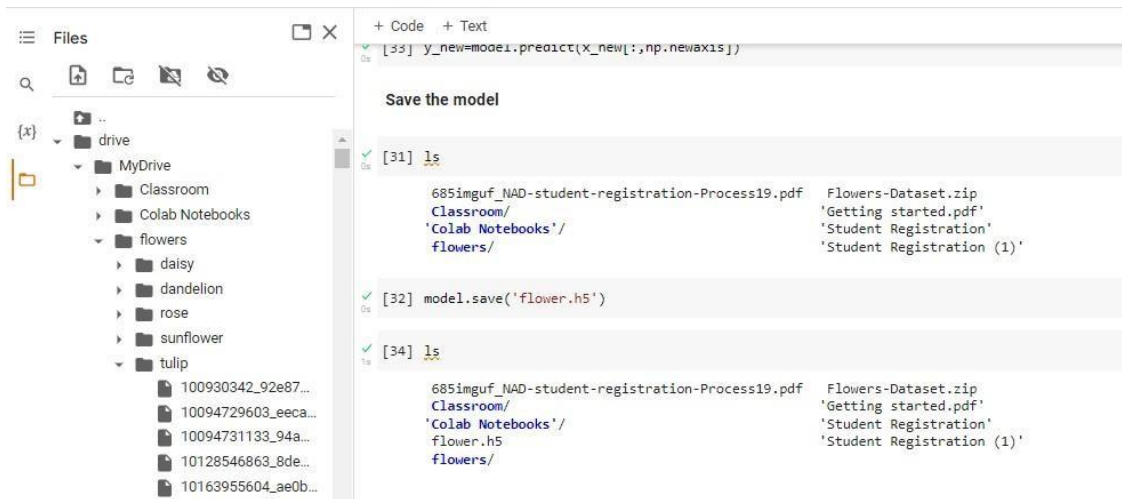

7. Save The Model

Solution:

```python
Ls
model.save('flower.h5')
ls
```
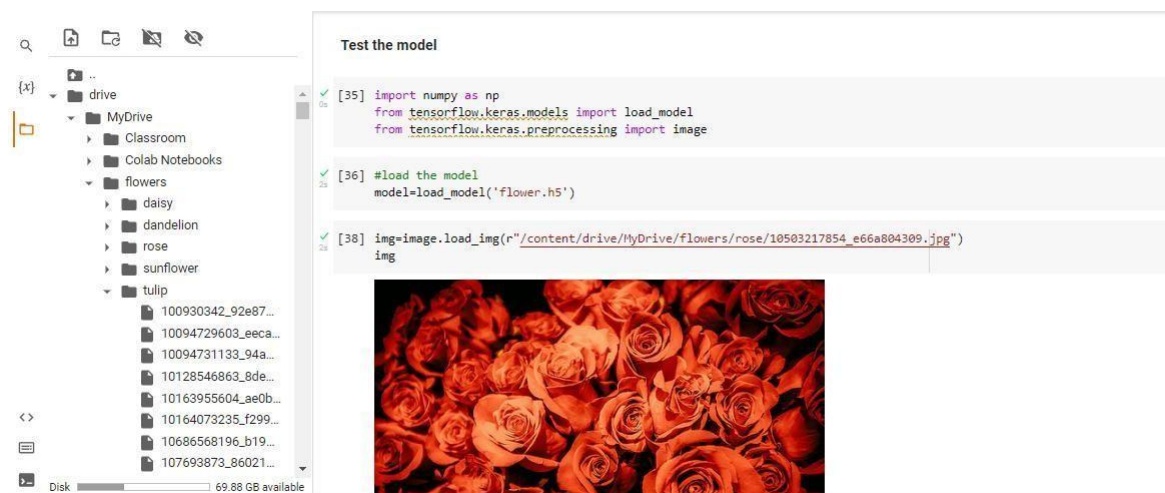
8. Test The Model

Solution:

```python
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
#load the model
model=load_model('flower.h5')
img=image.load_img(r"/content/drive/MyDrive/flowers/rose/10503217854_e66a804309.jpg")
img
img=image.load_img(r"/content/drive/MyDrive/flowers/rose/10503217854_e66a804309.jpg",target_size=(64,64))
img
x=image.img_to_array(img)
x
x=np.expand_dims(x,axis=0)
x
y=np.argmax(model.predict(x),axis=1)
y
x_train.class_indices
index=['daisy','dandelion','rose','sunflower','tulip']
index[y[0]]


img=image.load_img(r"/content/drive/MyDrive/flowers/daisy/100080576_f52e8ee070_n.jpg",target_size=(64,64))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
y=np.argmax(model.predict(x),axis=1)
index=['daisy','dandelion','rose','sunflower','tulip']
index[y[0]]


img=image.load_img(r"/content/drive/MyDrive/flowers/dandelion/10043234166_e6dd915111_n.jpg",target_size=(64,64))
```

```
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
y=np.argmax(model.predict(x),axis=1)
index=['daisy','dandelion','rose','sunflower','tulip']
index[y[0]]
```

```
img=image.load_img(r"/content/drive/MyDrive/flowers/rose/10090824183_d02c613f10_m.jpg"
,target_size=(64,64))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
y=np.argmax(model.predict(x),axis=1)
index=['daisy','dandelion','rose','sunflower','tulip']
index[y[0]]
```

```
img=image.load_img(r"/content/drive/MyDrive/flowers/sunflower/1008566138_6927679c8a.jp
g",target_size=(64,64))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
y=np.argmax(model.predict(x),axis=1)
index=['daisy','dandelion','rose','sunflower','tulip']
index[y[0]]
```

```
img=image.load_img(r"/content/drive/MyDrive/flowers/tulip/100930342_92e8746431_n.jpg",
target_size=(64,64))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
y=np.argmax(model.predict(x),axis=1)
index=['daisy','dandelion','rose','sunflower','tulip']
index[y[0]]
```

+ Code   + Text

```
[39] img=image.load_img(r"/content/drive/MyDrive/flowers/rose/10503217854_e66a804309.jpg",target_size=(64,64))
     img
```

```
[40] x=image.img_to_array(img)
     x

     array([[[  0.,   2.,   0.],
             [  0.,   2.,   0.],
             [  0.,   2.,   0.],
             ...,
             [ 92.,  14.,   0.],
             [ 61.,  13.,   9.],
             [ 17.,   7.,   5.]],

            [[  0.,   2.,   0.],
             [  0.,   2.,   0.],
             [  0.,   2.,   0.],
             ...,
             [150.,   3.,   0.],
             [ 85.,  10.,   7.],
             [119.,   4.,   1.]],

            [[  0.,   2.,   0.],
```

+ Code   + Text

```
[40]      [[  1.,   3.,   0.],
           [  0.,   2.,   0.],
           [  1.,   1.,   0.],
           ...,
           [ 29.,   5.,   1.],
           [ 41.,  13.,   0.],
           [  5.,   4.,   0.]]], dtype=float32)
```

```
[41] x=np.expand_dims(x,axis=0)
     x

     array([[[[  0.,   2.,   0.],
              [  0.,   2.,   0.],
              [  0.,   2.,   0.],
              ...,
              [ 92.,  14.,   0.],
              [ 61.,  13.,   9.],
              [ 17.,   7.,   5.]],

             [[  0.,   2.,   0.],
              [  0.,   2.,   0.],
              [  0.,   2.,   0.],
              ...,
              [150.,   3.,   0.],
              [ 85.,  10.,   7.],
              [119.,   4.,   1.]],

             [[  0.,   2.,   0.],
              [  0.,   2.,   0.],
              [  0.,   2.,   0.],
```

```
[42] y=np.argmax(model.predict(x),axis=1)
     y

     array([3])
```

```
[43] x_train.class_indices

     {'.ipynb_checkpoints': 0,
      'daisy': 1,
      'dandelion': 2,
      'rose': 3,
      'sunflower': 4,
      'tulip': 5}
```

```
[44] index=['daisy','dandelion','rose','sunflower','tulip']
```

```
[46] index[y[0]]

     'sunflower'
```

```
[50] img=image.load_img(r"/content/drive/MyDrive/flowers/daisy/100080576_f52e8ee070_n.jpg",target_size=(64,64))
     x=image.img_to_array(img)
```

Files

```
[50] img=image.load_img(r"/content/drive/MyDrive/flowers/daisy/100080576_f52e8ee070_n.jpg",target_size=(64,64))
     x=image.img_to_array(img)
     x=np.expand_dims(x,axis=0)
     y=np.argmax(model.predict(x),axis=1)
     index=['daisy','dandelion','rose','sunflower','tulip']
     index[y[0]]

     'sunflower'
```

```
[51] img=image.load_img(r"/content/drive/MyDrive/flowers/dandelion/10043234166_e6dd915111_n.jpg",target_size=(64,64))
     x=image.img_to_array(img)
     x=np.expand_dims(x,axis=0)
     y=np.argmax(model.predict(x),axis=1)
     index=['daisy','dandelion','rose','sunflower','tulip']
     index[y[0]]

     'rose'
```

```
[52] img=image.load_img(r"/content/drive/MyDrive/flowers/rose/10090824183_d02c613f10_m.jpg",target_size=(64,64))
     x=image.img_to_array(img)
     x=np.expand_dims(x,axis=0)
     y=np.argmax(model.predict(x),axis=1)
     index=['daisy','dandelion','rose','sunflower','tulip']
     index[y[0]]

     'sunflower'
```

Files

```
     img=image.load_img(r"/content/drive/MyDrive/flowers/sunflower/1008566158_692767979c8a.jpg",target_size=(64,64))
     x=image.img_to_array(img)
     x=np.expand_dims(x,axis=0)
     y=np.argmax(model.predict(x),axis=1)
     index=['daisy','dandelion','rose','sunflower','tulip']
     index[y[0]]

     'rose'
```

```
[54] img=image.load_img(r"/content/drive/MyDrive/flowers/tulip/100930342_92e8746431_n.jpg",target_size=(64,64))
     x=image.img_to_array(img)
     x=np.expand_dims(x,axis=0)
     y=np.argmax(model.predict(x),axis=1)
     index=['daisy','dandelion','rose','sunflower','tulip']
     index[y[0]]

     'dandelion'
```