```
# 1) DOWNLOAD DATA SET AND UNZIP
!unzip '/content/Flowers-Datasets.zip'
```

```
  inflating: flowers/tulip/8712270665_57b5bda0a2_n.jpg
  inflating: flowers/tulip/8712282563_3819afb7bc.jpg
  inflating: flowers/tulip/8713357842_9964a93473_n.jpg
  inflating: flowers/tulip/8713387500_6a9138b41b_n.jpg
  inflating: flowers/tulip/8713388322_e5ae26263b_n.jpg
  inflating: flowers/tulip/8713389178_66bceb71a8_n.jpg
  inflating: flowers/tulip/8713390684_041148dd3e_n.jpg
  inflating: flowers/tulip/8713391394_4b679ea1e3_n.jpg
  inflating: flowers/tulip/8713392604_90631fb809_n.jpg
  inflating: flowers/tulip/8713394070_b24561b0a9.jpg
  inflating: flowers/tulip/8713396140_5af8136136.jpg
  inflating: flowers/tulip/8713397358_0505cc0176_n.jpg
  inflating: flowers/tulip/8713397694_bcbcbba2c2_n.jpg
  inflating: flowers/tulip/8713398114_bc96f1b624_n.jpg
  inflating: flowers/tulip/8713398614_88202e452e_n.jpg
  inflating: flowers/tulip/8713398906_28e59a225a_n.jpg
  inflating: flowers/tulip/8713407768_f880df361f.jpg
  inflating: flowers/tulip/8717900362_2aa508e9e5.jpg
  inflating: flowers/tulip/8722514702_7ecc68691c.jpg
  inflating: flowers/tulip/8723767533_9145dec4bd_n.jpg
  inflating: flowers/tulip/8729501081_b993185542_m.jpg
  inflating: flowers/tulip/8733586143_3139db6e9e_n.jpg
  inflating: flowers/tulip/8748266132_5298a91dcf_n.jpg
  inflating: flowers/tulip/8750288831_5e49a9f29b.jpg
  inflating: flowers/tulip/8757486380_90952c5377.jpg
  inflating: flowers/tulip/8758464923_75a5ffe320_n.jpg
  inflating: flowers/tulip/8758519201_16e8d2d781_n.jpg
  inflating: flowers/tulip/8759594528_2534c0ec65_n.jpg
  inflating: flowers/tulip/8759597778_7fca5d434b_n.jpg
  inflating: flowers/tulip/8759601388_36e2a50d98_n.jpg
  inflating: flowers/tulip/8759606166_8e475013fa_n.jpg
  inflating: flowers/tulip/8759618746_f5e39fdbf8_n.jpg
  inflating: flowers/tulip/8762189906_8223cef62f.jpg
  inflating: flowers/tulip/8762193202_0fbf2f6a81.jpg
  inflating: flowers/tulip/8768645961_8f1e097170_n.jpg
  inflating: flowers/tulip/8817622133_a42bb90e38_n.jpg
  inflating: flowers/tulip/8838347159_746d14e6c1_m.jpg
  inflating: flowers/tulip/8838354855_c474fc66a3_m.jpg
  inflating: flowers/tulip/8838914676_8ef4db7f50_n.jpg
  inflating: flowers/tulip/8838975946_f54194894e_m.jpg
  inflating: flowers/tulip/8838983024_5c1a767878_n.jpg
  inflating: flowers/tulip/8892851067_79242a7362_n.jpg
  inflating: flowers/tulip/8904780994_8867d64155_n.jpg
  inflating: flowers/tulip/8908062479_449200a1b4.jpg
  inflating: flowers/tulip/8908097235_c3e746d36e_n.jpg
```

```
inflating: flowers/tulip/9019694597_2d3bbedb17.jpg
inflating: flowers/tulip/9030467406_05e93ff171_n.jpg
inflating: flowers/tulip/9048307967_40a164a459_m.jpg
inflating: flowers/tulip/924782410_94ed7913ca_m.jpg
inflating: flowers/tulip/9378657435_89fabf13c9_n.jpg
inflating: flowers/tulip/9444202147_405290415b_n.jpg
inflating: flowers/tulip/9446982168_06c4d71da3_n.jpg
inflating: flowers/tulip/9831362123_5aac525a99_n.jpg
inflating: flowers/tulip/9870557734_88eb3b9e3b_n.jpg
inflating: flowers/tulip/9947374414_fdf1d0861c_n.jpg
inflating: flowers/tulip/9947385346_3a8cacea02_n.jpg
inflating: flowers/tulip/9976515506_d496c5e72c.jpg
```

```
# 2) IMAGE AUGMENTATION
from tensorflow.keras.preprocessing.image import ImageDataGenerator

train_datagen = ImageDataGenerator(rescale=1./255, zoom_range=0.2, horizon
test_datagen = ImageDataGenerator(rescale=1./255)

xtrain = train_datagen.flow_from_directory('/content/flowers', target_size
```

```
Found 4317 images belonging to 5 classes.
```

3) CREATE MODEL

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Convolution2D, MaxPooling2D, Flatten,

# Initializing the model
model = Sequential()
```

4)ADD LAYERS

```
model.add(Convolution2D(32,(3,3),activation='relu',input_shape=(64,64,3)))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())
model.add(Dense(300,activation='relu'))
model.add(Dense(150,activation='relu'))
model.add(Dense(5,activation='softmax'))
```

5)COMPLETE MODEL

```
model.compile(optimizer='adam',loss='categorical_crossentropy',metrics=['a
```

6)FIT THE MODEL

```
model.fit_generator(xtrain,
                    steps_per_epoch=len(xtrain),
                    epochs=10)
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:3: UserW
  This is separate from the ipykernel package so we can avoid doing i
Epoch 1/10
44/44 [==============================] - 35s 763ms/step - loss: 1.661
Epoch 2/10
44/44 [==============================] - 33s 762ms/step - loss: 1.128
Epoch 3/10
44/44 [==============================] - 33s 734ms/step - loss: 1.041
Epoch 4/10
44/44 [==============================] - 32s 722ms/step - loss: 0.964
Epoch 5/10
44/44 [==============================] - 33s 750ms/step - loss: 0.928
Epoch 6/10
44/44 [==============================] - 32s 740ms/step - loss: 0.876
Epoch 7/10
44/44 [==============================] - 33s 733ms/step - loss: 0.832
Epoch 8/10
44/44 [==============================] - 33s 733ms/step - loss: 0.777
Epoch 9/10
44/44 [==============================] - 33s 738ms/step - loss: 0.769
Epoch 10/10
44/44 [==============================] - 33s 736ms/step - loss: 0.723
<keras.callbacks.History at 0x7fbce9274190>
```

7)SAVE THE MODEL

```
model.save('fLOWERS@IBM.h5')
```

8)TEST THE MODEL

```
import numpy as np
from tensorflow.keras.preprocessing import image
op = ['daisy', 'dandelion','rose', 'sunflower', 'tulip']


img = image.load_img('/content/rose1.jpg',target_size=(64,64))
x = image.img_to_array(img)
x = np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x))
print(pred, model.predict(x))
print(op[pred])
```

```
1/1 [==============================] - 0s 83ms/step
1/1 [==============================] - 0s 33ms/step
2 [[0. 0. 1. 0. 0.]]
rose
```

```
img
```

```
img = image.load_img('/content/tulip1.webp',target_size=(64,64))
x = image.img_to_array(img)
x = np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x))
print(pred, model.predict(x))
print(op[pred])
```

```
1/1 [==============================] - 0s 25ms/step
1/1 [==============================] - 0s 28ms/step
4 [[0. 0. 0. 0. 1.]]
tulip
```

```
img
```

```
img = image.load_img('/content/DANDI.jpg',target_size=(64,64))
x = image.img_to_array(img)
x = np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x))
print(pred, model.predict(x))
print(op[pred])
```

```
1/1 [==============================] - 0s 30ms/step
1/1 [==============================] - 0s 29ms/step
0 [[1.0000000e+00 0.0000000e+00 2.5551024e-13 0.0000000e+00 0.000000(
daisy
```

```
img
```

Colab paid products - Cancel contracts here

13m 16s completed at 5:05 PM