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

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

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

```
  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
  inflating: flowers/tulip/8712269349_2b933da2b8_n.jpg
  inflating: flowers/tulip/8712270243_8512cf4fbd.jpg
  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
```

Image augmentation

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

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

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

```
test_data_aug = ImageDataGenerator(rescale=1./255)
```

```
xtrain = train_data_aug.flow_from_directory('/content/flowers',target_size=(64,64),class_m
```

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

```
xtest = test_data_aug.flow_from_directory('/content/flowers',target_size=(64,64),class_mod
```

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

Create Model

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

Add Layers (Convolution, Max-Pooling, Flatten, Dense-(Hidden Layers), Output)

```
model = Sequential()
model.add(Convolution2D(32,(3,3),activation='relu',input_shape=(64,64,3))) # Convolution l
model.add(MaxPooling2D(pool_size=(2,2))) # Max pooling layer
model.add(Flatten()) # Flatten layer
# Fully connected layers (ANN)
model.add(Dense(300,activation='relu')) # Hidden layer 1
model.add(Dense(150,activation='relu')) # Hidden layer 2
```

```
model.add(Dense(5,activation='softmax')) # Output layer
```

Compile the Model

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

Fit the Model

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

```
Epoch 1/10
44/44 [==============================] - 48s 1s/step - loss: 1.5197 - accuracy: 0.40
Epoch 2/10
44/44 [==============================] - 45s 1s/step - loss: 1.1072 - accuracy: 0.54
Epoch 3/10
44/44 [==============================] - 45s 1s/step - loss: 1.0195 - accuracy: 0.60
Epoch 4/10
44/44 [==============================] - 47s 1s/step - loss: 0.9714 - accuracy: 0.62
Epoch 5/10
44/44 [==============================] - 45s 1s/step - loss: 0.9062 - accuracy: 0.65
Epoch 6/10
44/44 [==============================] - 45s 1s/step - loss: 0.8899 - accuracy: 0.65
Epoch 7/10
44/44 [==============================] - 45s 1s/step - loss: 0.8113 - accuracy: 0.69
Epoch 8/10
44/44 [==============================] - 45s 1s/step - loss: 0.7739 - accuracy: 0.70
Epoch 9/10
44/44 [==============================] - 45s 1s/step - loss: 0.7225 - accuracy: 0.72
Epoch 10/10
44/44 [==============================] - 45s 1s/step - loss: 0.7019 - accuracy: 0.73
<keras.callbacks.History at 0x7fa7110c9090>
```

Double-click (or enter) to edit

Save the Model

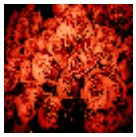
```
model.save('/content/flowers')
```

Test the Model

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

```
img = image.load_img('/content/flowers/rose/10503217854_e66a804309.jpg',target_size=(64,64
```

```
img
```



```
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.],
        [  0.,   2.,   0.],
        [  0.,   2.,   0.],
        ...,
        [ 88.,   9.,   0.],
        [207.,   7.,  10.],
        [152.,   0.,   0.]],

       ...,

       [[  0.,   4.,   0.],
        [  1.,   3.,   0.],
        [  0.,   2.,   0.],
        ...,
        [  2.,   2.,   4.],
        [  0.,   2.,   5.],
        [ 51.,  10.,   6.]],

       [[  0.,   2.,   0.],
        [  1.,   3.,   0.],
        [  1.,   3.,   0.],
        ...,
        [  0.,   3.,   1.],
        [  0.,   3.,   4.],
        [  0.,   5.,   3.]],

       [[  1.,   3.,   0.],
```

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

```
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.],
              ...,
              [ 88.,   9.,   0.],
              [207.,   7.,  10.],
              [152.,   0.,   0.]],

             ...,

             [[  0.,   4.,   0.],
              [  1.,   3.,   0.],
              [  0.,   2.,   0.],
              ...,
              [  2.,   2.,   4.],
              [  0.,   2.,   5.],
              [ 51.,  10.,   6.]],

             [[  0.,   2.,   0.],
              [  1.,   3.,   0.],
              [  1.,   3.,   0.],
              ...,
              [  0.,   3.,   1.],
              [  0.,   3.,   4.],
              [  0.,   5.,   3.]],

             [[  1.,   3.,   0.],
              [  0.,   2.,   0.],
              [  1.,   1.,   0.],
              ...,
              [ 29.,   5.,   1.],
```

```
                [ 41.,  13.,   0.],
                [  5.,   4.,   0.]]]], dtype=float32)
```

```
model.predict(x)
```

```
     1/1 [==============================] - 0s 121ms/step
     array([[0., 0., 1., 0., 0.]], dtype=float32)
```

```
xtrain.class_indices
```

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

```
op = ['daisy','dandelion','rose','sunflower','tulip']
pred = np.argmax(model.predict(x))
op[pred]
```

```
     1/1 [==============================] - 0s 26ms/step
     'rose'
```

```
img = image.load_img('/content/flowers/dandelion/10043234166_e6dd915111_n.jpg',target_size
x = image.img_to_array(img)
x = np.expand_dims(x,axis=0)
pred = np.argmax(model.predict(x))
op[pred]
```

```
     1/1 [==============================] - 0s 27ms/step
     'daisy'
```