#Unzip the data

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

```
Archive:  /content/Flowers-Dataset.zip
warning [/content/Flowers-Dataset.zip]:  175894356 extra bytes at
beginning or within zipfile
  (attempting to process anyway)
file #1:  bad zipfile offset (local header sig):  175894356
  (attempting to re-compensate)
replace flowers/daisy/100080576_f52e8ee070_n.jpg? [y]es, [n]o, [A]ll,
[N]one, [r]ename:
```

#Image Augmentation

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

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

test_datagen=ImageDataGenerator(rescale=1./255)

xtrain=train_datagen.flow_from_directory('/content/flowers',
                                         target_size=(76,76),
                                         class_mode='categorical',
                                         batch_size=100)
```

Create Model

```
from tensorflow.keras.models import Sequential


from tensorflow.keras.layers import
Convolution2D,MaxPool2D,Flatten,Dense
```

Add Layers (Convolution,MaxPooling,Flatten,Dense-(Hidden Layers),Output)

```
model=Sequential()
model.add(Convolution2D(32,
(3,3),activation='relu',input_shape=(76,76,3)))
model.add(MaxPool2D(pool_size=(2,2)))
model.add(Flatten())
model.add(Dense(300,activation='relu'))
model.add(Dense(150,activation='relu'))
model.add(Dense(4,activation='softmax'))
```

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))
```

Save The Model

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

Test The Model

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

img=image.load_img('/content/flowers/daisy/
10140303196_b88d3d6cec.jpg',target_size=(76,76))
img

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

img=image.load_img('/content/flowers/rose/
10503217854_e66a804309.jpg',target_size=(76,76))
img

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

x=image.img_to_array(img)
x
x=np.expand_dims(x,axis=0)
pred=np.argmax(model.predict(x))
```

```
pred
op=['daisy','dandelion','rose','sunflower','tulip']
op[pred]

img=image.load_img('/content/flowers/tulip/
2087981909_fd468de5c4_n.jpg',target_size=(76,76))
img

img=image.load_img('/content/flowers/dandelion/
3383422012_6c9d83671f_n.jpg',target_size=(76,76))
img
```