UNZIP FILE

```
ls
```

```
 flowers/  'Flowers-Dataset (1).zip'
```

```
cd /content/drive/MyDrive/data
```

```
/content/drive/MyDrive/data
```

```
ls
```

```
 flowers/  'Flowers-Dataset (1).zip'
```

```
pwd
```

```
'/content/drive/MyDrive/data'
```

```
!unzip'/content/drive/MyDrive/data/Flowers-Dataset (1).zip
```

```
/bin/bash: unzip/content/drive/MyDrive/data/flowers: No such file or directory
```

IMAGE AUGMENTATION

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

```
train_data = ImageDataGenerator(rescale=1./255,zoom_range=0.2,horizontal_flip=True,vertica
```

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

```
X_train = train_data.flow_from_directory('/content/drive/MyDrive/data/flowers',target_size
```

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

```
X_test = test_data.flow_from_directory('/content/drive/MyDrive/data/flowers',target_size=(
```

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

CREATING CNN MODEL

```
import numpy as np
from tensorflow.keras.models import Sequential
```

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

ADDING LAYERS

```
model = Sequential()

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(300,activation='relu'))

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

COMPILING THE MODEL

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

```
108
```

FIT THE MODEL

```
model.fit(X_train,steps_per_epoch=len(X_train),epochs=5,validation_data=X_test,validation_
```

```
Epoch 1/5
108/108 [==============================] - 34s 299ms/step - loss: 1.6467 - accuracy:
Epoch 2/5
108/108 [==============================] - 32s 298ms/step - loss: 1.6194 - accuracy:
Epoch 3/5
108/108 [==============================] - 32s 296ms/step - loss: 1.6132 - accuracy:
Epoch 4/5
108/108 [==============================] - 32s 298ms/step - loss: 1.6143 - accuracy:
Epoch 5/5
108/108 [==============================] - 33s 305ms/step - loss: 1.6039 - accuracy:
<keras.callbacks.History at 0x7f096a183510>
```

SAVE THE MODEL

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

TESTING THE MODEL

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

```
model=load_model("/content/drive/MyDrive/data/flower.h5")
```

```
img=image.load_img("/content/drive/MyDrive/data/flowers/sunflower/10386522775_4f8c616999_n
```

```
img
```

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

```
'dandelion'
```