

```

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib import style
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, confusion_matrix
from sklearn.preprocessing import LabelEncoder
from keras.preprocessing.image import ImageDataGenerator
import cv2
from tqdm import tqdm
import os
from random import shuffle
from PIL import Image
from zipfile import ZipFile
import tensorflow as tf
import random as rn

```

```

In [33]:
x=[]
y=[]
IMG_SIZE=200
FLOWER_DAISY_DIR="/content/Flowers-Dataset.zip"
FLOWER_DANDELION_DIR="/content/Flowers-Dataset.zip"
FLOWER_ROSE_DIR="/content/Flowers-Dataset.zip"
FLOWER_SUNFLOWER_DIR="/content/Flowers-Dataset.zip"
FLOWER_TULIP_DIR="/content/Flowers-Dataset.zip"

```

```

In [34]:
def assign_label(img, flower_type):
    return flower_type

```

```

In [35]:
def make_train_data(flower_type, DIR):
    for img in tqdm(os.listdir(DIR)):
        label=assign_label(img, flower_type)
        path=os.path.join(DIR, img)
        img=cv2.imread(path, cv2.IMREAD_COLOR)
        img=cv2.resize(img, (IMG_SIZE, IMG_SIZE))
        x.append(np.array(img))
        y.append(str(label))

```

```

In [37]:
make_train_data('Daisy', FLOWER_DAISY_DIR)
print(len(x))

```

```

-----
NotADirectoryError                                Traceback (most recent call last)
<ipython-input-37-b58ce26c683f> in <module>
----> 1 make_train_data('Daisy', FLOWER_DAISY_DIR)
      2 print(len(x))

<ipython-input-35-eb07650082aa> in make_train_data(flower_type, DIR)
      1 def make_train_data(flower_type, DIR):

```

```

----> 2   for img in tqdm(os.listdir(DIR)):
        3       label=assign_label(img,flower_type)
        4       path=os.path.join(DIR,img)
        5       img=cv2.imread(path,cv2.IMREAD_COLOR)

```

**NotADirectoryError**: [Errno 20] Not a directory: '/content/Flowers-Dataset.zip'

```

In [ ]: make_train_data('Dandelion',FLOWER_DENDALION_DIR)
        print(len(x))

```

```

In [ ]: make_train_data('Rose',FLOWER_ROSE_DIR)
        print(len(x))

```

```

In [ ]: make_train_data('Sunflower',FLOWER_SUNFLOWER_DIR)
        print(len(x))

```

```

In [ ]: make_train_data('Tulip',FLOWER_TULIP_DIR)
        print(len(x))

```

```

In [ ]: fig,ax=plt.subplots(5,2)
        fig.set_size_inches(15,15)
        for i in range(5):
            for j in range(2):
                l=mn.randint(0,len(y))
                ax[i,j].imshow(x[l])
                ax[i,j].set_title('Flowers:'+y[l])
                plt.tight_layouts()

```

```

In [ ]: x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=0)

```

```

In [ ]: model.compile(optimizer=Adam(lr=0.001),loss='categorical_crossentropy',metrics=['
        accuracy'])

```

```

In [ ]: model.summary()

```

```

In [ ]: plt.plot(History.history['loss'])
        plt.plot(History.history['val_loss'])
        plt.title('Model Loss')
        plt.ylabel('Loss')
        plt.xlabel('Epochs')
        plt.legend(['train','test'])
        plt.show()

```