```
"nbformat": 4,
"nbformat minor": 0.
"metadata": {
  "colab": {
    "provenance": []
  "kernelspec": {
    "name": "python3",
    "display_name": "Python 3"
  "language_info": {
    "name": "python"
},
cells": [
    "cell_type": "code",
    "execution_count": 11,
    "metadata": {
      "id": "7CsLvQ7EzJdw"
    "outputs": [],
    "source": [
      "import numpy as np\n",
       "import pandas as pd \n",
      "import matplotlib.pyplot as plt\n",
      "from matplotlib import style \n",
      "import seaborn as sns\n",
      "from sklearn.model_selection import train_test_split\n",
      "from sklearn.metrics import accuracy_score,confusion_matrix\n",
      "from sklearn.preprocessing import LabelEncoder\n",
      "from keras.preprocessing.image import ImageDataGenerator\n",
      "import cv2\n",
       "from tgdm import tgdm \n",
       "import os\n".
      "from random import shuffle\n",
      "from PIL import Image\n",
      "from zipfile import ZipFile\n",
      "import tensorflow as tf\n",
      "import random as rn"
    ]
  },
    "cell_type": "code",
    "source": [
      "x=[]\n",
       "y=[]\n",
      "IMG_SIZE=200\n",
      "FLOWER_DAISY_DIR=\"/content/Flowers-Dataset.zip\"\n",
      "FLOWER_DANDELION_DIR=\"/content/Flowers-Dataset.zip\"\n",
      "FLOWER_ROSE_DIR=\"/content/Flowers-Dataset.zip\"\n",
      "FLOWER_SUNFLOWER_DIR=\"/content/Flowers-Dataset.zip\"\n",
      "FLOWER_TULIP_DIR=\"/content/Flowers-Dataset.zip\"\n"
    "metadata": {
```

```
"id": "baG3zjbizKZ-"
  "execution_count": 33,
  "outputs": []
  "cell_type": "code",
  "source": [
    "def assign_label(img,flower_type):\n",
     " return flower_type"
  "metadata": {
     "id": "Yvb6z824zKpA"
  "execution_count": 34,
  "outputs": []
  "cell_type": "code",
  "source": [
     "def make_train_data(flower_type,DIR):\n",
       for img in tqdm(os.listdir(DIR)):\n",
          label=assign_label(img,flower_type)\n",
          path=os.path.join(DIR,img)\n",
          img=cv2.imread(path,cv2.IMREAD_COLOR)\n",
          img=cv2.resize(img,(IMG_SIZE,IMG_SIZE))\n",
          x.append(np.array(imq))\n",
          y.append(str(label))"
  "metadata": {
     "id": "05_ZOelGzK31"
  "execution_count": 35,
  "outputs": []
},
  "cell_type": "code",
  "source": [
    "make_train_data('Daisy',FLOWER_DAISY_DIR)\n",
     "print(len(x))"
  "metadata": {
     "colab": {
       "base_uri": "https://localhost:8080/",
       "height": 304
    },
"id": "6A-xbAWXz0zA",
     "outputId": "a23f4b35-c421-4df5-b8c0-fb6158e895f0"
  execution_count": 37,
  "outputs": [
       "output_type": "error",
       "ename": "NotADirectoryError",
       "evalue": "ignored",
       "traceback": [
```

```
-----\u001b[0m",
                                      "\u001b[0;31m-----
                                      "\u001b[0;31mNotADirectoryError\u001b[0m
                                                                                                                                                                                                                                                Traceback
(most recent call last)".
                                      "\u001b[0;32m<ipython-input-37-b58ce26c683f>\u001b[0m
                                                                                                                                                                                                                                                                        in
\u001b[0;36m<module>\u001b[0;34m\u001b[0m\n\u001b[0;32m---->
                                                                                                                                                                                                                                  1\u001bl0:31m
\u001bl0mmake train data\u001bl0m\u001bl0:34m(\u001bl0m\u001bl0:34m'Daisv'\u001bl0m\
u001b[0;34m,\u001b[0m\u001b[0mFLOWER_DAISY_DIR\u001b[0m\u001b[0;34m)\u001b[0m\u0
01b[0;34m\u001b[0m\u001b[0;34m\u001b[0m\u001b[0m\n\u001b[0m\n\u001b[0m\u001b[1;32m
2\u001b[0m
\u001b[0mprint\u001b[0m\u001b[0;34m(\u001b[0m\u001b[0m]u001b[0m\u001b[0m\u001b[0;34m(\u001b[0m\u001b[0m]u001b[0m]u001b[0m\u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b[0m]u001b
1b[0m\u001b[0mx\u001b[0m\u001b[0;34m)\u001b[0m\u001b[0;34m)\u001b[0;34m)\u001b[0m\u001b[0;34m\
u001b[0m\u001b[0;34m\u001b[0m\u001b[0m\n",
                                       \u001b[0;32m<ipython-input-35-eb07650082aa>\u001b[0m
                                                                                                                                                                                                                                                                        in
\u001b[0;36mmake_train_data\u001b[0;34m(flower_type,
                                                                                                                                                                                     DIR)\u001b[0m\n\u001b[1;32m
                                                                                                                                                                                               \u001bl0:32mdef\u001bl0m
1\u001b[0m
\u001b[0mmake_train_data\u001b[0m\u001b[0;34m(\u001b[0m\u001b[0mflower_type\u001b[0
m\u001b[0;34m,\u001b[0m\u001b[0mDIR\u001b[0m\u001b[0;34m)\u001b[0m\u001b[0;34m:\u0
01b[0m\u001b[0;34m\u001b[0m\u001b[0;34m\u001b[0m\u001b[0m\n\u001b[0;32m---->
2\u001bl0:31m
                                                                                                     \u001b[0;32mfor\u001b[0m
                                                                                                                                                                                                     \u001b[0mimg\u001b[0m
\u001b[0;32min\u001b[0m
\u001b[0mtqdm\u001b[0m\u001b[0;34m(\u001b[0m\u001b[0mos\u001b[0m\u001b[0m\u001b]
1b[0m\u001b[0mlistdir\u001b[0m\u001b[0;34m(\u001b[0m\u001b[0mDIR\u001b[0m\u001b[0;3
4m)\u001b[0m\u001b[0;34m)\u001b[0m\u001b[0;34m:\u001b[0m\u001b[0;34m\u001b[0m\u00
1b[0;34m\u001b[0m\u001b[0m\n\u001b[0m\u001b[1;32m
                                                                                                                                                                                                                                           3\u001b[0m
\u001bl0mlabel\u001bl0m\u001bl0:34m=\u001bl0m\u001bl0massign_label\u001bl0m\u001bl0:
34m(\u001b[0m\u001b[0mimg\u001b[0m\u001b[0;34m,\u001b[0m\u001b[0mflower_type\u001
b[0m\u001b[0;34m)\u001b[0m\u001b[0;34m\u001b[0m\u001b]0;34m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001b[0m\u001b]0m\u001
001b[1:32m
                                                                                                                                                                                                                                           4\u001bl0m
\u001b[0mpath\u001b[0m\u001b[0;34m=\u001b[0m\u001b[0mos\u001b[0m\u001b[0m\u001b]
1b[0m\u001b[0mpath\u001b[0m\u001b[0;34m.\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b]
m(\u001b[0m\u001b[0mDIR\u001b[0m\u001b[0;34m,\u001b[0m\u001b[0mimq\u001b[0m\u001
b[0;34m)\u001b[0m\u001b[0;34m\u001b[0m\u001b[0;34m\u001b[0m\u001b[0m\u001b[0;34m\u001b[0]]
                                                                                                                                                                                                                                           5\u001b[0m
\u001b[0mimq\u001b[0m\u001b[0;34m=\u001b[0m\u001b[0mcv2\u001b[0m\u001b[0;34m.\u00]]])
1b[0m\u001b[0mimread\u001b[0m\u001b[0;34m(\u001b[0m\u001b[0mpath\u001b[0m\u001b[0;
34m.\u001bl0m\u001bl0mcv2\u001bl0m\u001bl0:34m.\u001bl0m\u001bl0m\u001bl0mIMREAD COLOR\u
001b[0m\u001b[0;34m)\u001b[0m\u001b[0;34m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[0m\u001b[
n",
                                     "\u001bl0:31mNotADirectorvError\u001bl0m:
                                                                                                                                                                                                   201
                                                                                                                                                                          [Errno
                                                                                                                                                                                                                    Not
                                                                                                                                                                                                                                                   directory:
'/content/Flowers-Dataset.zip'"
                        }
                  ]
            },
                   "cell_type": "code",
                   "source": [
                         "make train data('Dendelion'.FLOWER DENDALION DIR)\n".
                         "print(len(x))"
                    "metadata": {
                          "id": "BUk0i_tq29-k"
                    "execution_count": null,
                    "outputs": []
            },
```

```
"cell_type": "code",
  "source": [
     "make_train_data('Rose',FLOWER_ROSE_DIR)\n",
     "print(len(x))"
  "metadata": {
     "id": "9LAw_six2-LH"
  "execution_count": null,
  "outputs": []
  "cell_type": "code",
  "source": [
     "make_train_data('Sunflower',FLOWER_SUNFLOWER_DIR)\n",
     "print(len(x))"
  "metadata": {
     "id": "eaBggBgR2-fY"
  "execution_count": null,
  "outputs": []
},
  "cell_type": "code",
  "source": [
     "make_train_data('Tulip',FLOWER_TULIP_DIR)\n",
     "print(len(x))"
  "metadata": {
     "id": "XXPx-rXlz1Ko"
  "execution_count": null,
  "outputs": []
  "cell_type": "code",
  "source": [
     "fig,ax=plt.subplots(5,2)\n",
     "fig.set_size_inches(15,15)\n",
     "for i in range(5):\n",
       for j in range(2):\n",
          l=rn.randint(0,len(y))\n",
          ax[i,i].imshow(x[l])\n",
          ax[i,j].set_title('Flowers:'+y[l])\n",
          plt.tight_layouts()"
   "metadata": {
     "id": "-dCtc2p95z9L"
  "execution_count": null,
  "outputs": []
  "cell_type": "code",
```

```
"source": [
         "x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=0)"
       "metadata": {
         "id": "fVicakYa50Ny"
       "execution_count": null,
       "outputs": []
    },
       "cell_type": "code",
       "source": [
"model.compile(optimizer=Adam(lr=0.001),loss='categorical_crossen_tropy',metrics=['accuracy'])"
       "metadata": {
         "id": "HN4vupE850gW"
       "execution_count": null,
       "outputs": []
       "cell_type": "code",
       "source": [
         "model.summary()"
       "metadata": {
         "id": "8dq-gmKP75ti"
       "execution_count": null,
       "outputs": []
       "cell_type": "code",
       "source": [
         "plt.plot(History.history['loss']) \n",
         "plt.plot(History.history['val_loss']) \n",
         "plt.title('Model Loss')\n",
         "plt.ylabel('Loss')\n",
         "plt.xlabel('Epochs')\n",
         "plt.legend(['train','test'])\n",
         "plt.show()"
       "metadata": {
         "id": "rf0yRrx_76Bn"
       "execution_count": null,
       "outputs": []
       "cell_type": "code",
       "source": [],
       "metadata": {
         "id": "K3ePO5kG760m"
       "execution_count": null,
```

```
"outputs": []
}
]
}
```