```
{
                  "nbformat": 4,
                  "nbformat_minor": 0,
                  "metadata": {
                    "colab": {
                      "provenance": [],
                      "collapsed_sections": []
                    },
                    "kernelspec": {
                      "name": "python3",
                      "display_name": "Python 3"
                    },
                    "language_info": {
                      "name": "python"
                    }
                  },
                  "cells": [
                    {
                      "cell_type": "code",
                      "execution_count": null,
                      "metadata": {
                        "id": "Kmn87h7Qqva8"
                      },
                      "outputs": [],
                      "source": [
                        "from keras.preprocessing.image import ImageDataGenerator\n",
                        "train_datagen=ImageDataGenerator(rescale=1./255,shear_range=0.2,zoom_range=0.
                        "test_datagen=ImageDataGenerator(rescale=1)"
                      ]
                    },
                    {
                      "cell_type": "code",
                      "source": [
                        "x_train=train_datagen.flow_from_directory(r'/content/drive/MyDrive/DataSet/Da
                dataset/test',target_size=(128,128),batch_size=2,class_mode='categorical')\n",
                        "x_test=test_datagen.flow_from_directory(r'/content/drive/MyDrive/DataSet/Datas
                dataset/train',target_size=(128,128),batch_size=2,class_mode='categorical')"
                      ],
                      "metadata": {
                        "colab": {
                          "base_uri": "https://localhost:8080/"
                        },
    "id": "KwwIP59PrMHD",
    "outputId": "82eb5c7b-4e25-4d69-88ea-d8b1c8371987"
  },
  "execution_count": null,
  "outputs": [
    {
      "output_type": "stream",
      "name": "stdout",
      "text": [
        "Found 1686 images belonging to 6 classes.\n",
        "Found 5384 images belonging to 6 classes.\n"
      ]
    }
  ]
},
{
  "cell_type": "markdown",
  "source": [
    "1. import the libraries"
  ],
  "metadata": {
    "id": "H7sgmVjl_xeg"
  }
},
{
  "cell_type": "code",
  "source": [
    "from keras.models import Sequential\n",
    "from keras.layers import Dense\n",
    "from keras.layers import Convolution2D\n",
    "from keras.layers import MaxPooling2D\n",
    "from keras.layers import Flatten"
  ],
  "metadata": {
    "id": "-83sTKQ2rtqU"
  },
  "execution_count": null,
  "outputs": []
},
{
  "cell_type": "markdown",
  "source": [
        "2. initializing the model"
      ],
      "metadata": {
        "id": "C6h6Utk9AAhu"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "model=Sequential()"
      ],
      "metadata": {
        "id": "c6Gn5yy-rxi9"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "markdown",
      "source": [
        "3. Add CNN layers"
      ],
      "metadata": {
        "id": "uejNXnGCAyJ2"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "model.add(Convolution2D(32,(3,3),input_shape=(128,128,3),activation='relu'))"
      ],
      "metadata": {
        "id": "Llp2PuTTr2IS"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "model.add(MaxPooling2D(pool_size=(2,2)))"
      ],
    "metadata": {
      "id": "1HH4kT-Vr8gz"
    },
    "execution_count": null,
    "outputs": []
  },
  {
    "cell_type": "code",
    "source": [
      "model.add(Flatten())"
    ],
    "metadata": {
      "id": "RiF3j7y7sDlT"
    },
    "execution_count": null,
    "outputs": []
  },
  {
    "cell_type": "markdown",
    "source": [
      "4. Add dense layer"
    ],
    "metadata": {
      "id": "JQG8XDfcC9x2"
    }
  },
  {
    "cell_type": "code",
    "source": [
      "model.add(Dense(units=40,kernel_initializer='uniform',activation='relu'))"
    ],
    "metadata": {
      "id": "rzYOisl9sHl0"
    },
    "execution_count": null,
    "outputs": []
  },
  {
    "cell_type": "code",
    "source": [
      "model.add(Dense(units=20,kernel_initializer='random_uniform',activation='relu
    ],
      "metadata": {
        "id": "0ilxaf50sI_7"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "model.add(Dense(units=6,kernel_initializer='random_uniform',activation='softma
      ],
      "metadata": {
        "id": "RWDSMiy_sSkj"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "markdown",
      "source": [
        "5. Train and save the model"
      ],
      "metadata": {
        "id": "JiyVSqmPGiLA"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "model.compile(loss='categorical_crossentropy',optimizer=\"adam\",metrics=[\"ac
      ],
      "metadata": {
        "id": "M38faTM6sWyy"
      },
      "execution_count": null,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "model.fit(x_train,steps_per_epoch=89,epochs=20,validation_data=x_test,validati
      ],
"metadata": {
  "colab": {
    "base_uri": "https://localhost:8080/"
  },
  "id": "BWOC9xsSsam6",
  "outputId": "2429a4bb-2094-482f-aad4-e1adc2e820ef"
},
"execution_count": null,
"outputs": [
  {
    "output_type": "stream",
    "name": "stdout",
    "text": [
      "Epoch 1/20\n",
      "89/89 [==============================] - 146s 2s/step - loss: 1.6616 - acc
      "Epoch 2/20\n",
      "89/89 [==============================] - 129s 1s/step - loss: 1.7158 - acc
      "Epoch 3/20\n",
      "89/89 [==============================] - 125s 1s/step - loss: 1.6271 - acc
      "Epoch 4/20\n",
      "89/89 [==============================] - 112s 1s/step - loss: 1.3890 - acc
      "Epoch 5/20\n",
      "89/89 [==============================] - 112s 1s/step - loss: 0.9276 - acc
      "Epoch 6/20\n",
      "89/89 [==============================] - 105s 1s/step - loss: 0.7846 - acc
      "Epoch 7/20\n",
      "89/89 [==============================] - 99s 1s/step - loss: 0.7925 - accu
      "Epoch 8/20\n",
      "89/89 [==============================] - 98s 1s/step - loss: 0.7527 - accu
      "Epoch 9/20\n",
      "89/89 [==============================] - 92s 1s/step - loss: 0.7694 - accu
      "Epoch 10/20\n",
      "89/89 [==============================] - 95s 1s/step - loss: 0.6303 - accu
      "Epoch 11/20\n",
      "89/89 [==============================] - 88s 978ms/step - loss: 0.6382 - a
      "Epoch 12/20\n",
      "89/89 [==============================] - 87s 980ms/step - loss: 0.6182 - a
      "Epoch 13/20\n",
      "89/89 [==============================] - 84s 938ms/step - loss: 0.5206 - a
      "Epoch 14/20\n",
      "89/89 [==============================] - 86s 976ms/step - loss: 0.5636 - a
      "Epoch 15/20\n",
        "89/89 [==============================] - 83s 937ms/step - loss: 0.5015 - a
        "Epoch 16/20\n",
        "89/89 [==============================] - 84s 943ms/step - loss: 0.4755 - a
        "Epoch 17/20\n",
        "89/89 [==============================] - 85s 965ms/step - loss: 0.4559 - a
        "Epoch 18/20\n",
        "89/89 [==============================] - 82s 923ms/step - loss: 0.3686 - a
        "Epoch 19/20\n",
        "89/89 [==============================] - 80s 901ms/step - loss: 0.4244 - a
        "Epoch 20/20\n",
        "89/89 [==============================] - 80s 897ms/step - loss: 0.5965 - a
      ]
    },
    {
      "output_type": "execute_result",
      "data": {
        "text/plain": [
          "<keras.callbacks.History at 0x7f2f0fc41d90>"
        ]
      },
      "metadata": {},
      "execution_count": 19
    }
  ]
},
{
  "cell_type": "code",
  "source": [
    "model.save('fruit.h5')    "
  ],
  "metadata": {
    "id": "Gq1cVMV5x11C"
  },
  "execution_count": null,
  "outputs": []
},
{
  "cell_type": "code",
  "source": [
    "model.summary()"
  ],
  "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/"
        },
        "id": "M8fo2vjByGJ6",
        "outputId": "9009f9bc-8a26-4002-9bc4-201901c34521"
      },
      "execution_count": null,
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
            "Model: \"sequential\"\n",
            "_________________________________________________________________\n",
            " Layer (type)                Output Shape              Param #   \n",
            "=================================================================\n",
            " conv2d (Conv2D)             (None, 126, 126, 32)      896       \n",
            "                                                                 \n",
            " max_pooling2d (MaxPooling2D  (None, 63, 63, 32)       0         \n",
            " )                                                               \n",
            "                                                                 \n",
            " flatten (Flatten)           (None, 127008)            0         \n",
            "                                                                 \n",
            " dense (Dense)               (None, 300)               38102700  \n",
            "                                                                 \n",
            " dense_1 (Dense)             (None, 40)                12040     \n",
            "                                                                 \n",
            " dense_2 (Dense)             (None, 20)                820       \n",
            "                                                                 \n",
            " dense_3 (Dense)             (None, 6)                 126       \n",
            "                                                                 \n",
            "=================================================================\n",
            "Total params: 38,116,582\n",
            "Trainable params: 38,116,582\n",
            "Non-trainable params: 0\n",
            "_________________________________________________________________\n"
          ]
        }
      ]
    }
  ]
}
```