```
{
 "cells": [
  {
   "cell_type": "markdown",
   "source": [
    "TEST THE MODEL"
   ],
   "metadata": {
    "id": "0O-C-oAgbvLa"
   }
  },
  {
   "cell_type": "markdown",
   "source": [
    "Import datagenerator to train and test"
   ],
   "metadata": {
    "id": "bBLztdw8B4_e"
   }
  },
  {
   "cell_type": "code",
   "source": [
    "from tensorflow.keras.preprocessing.image import ImageDataGenerator"
   ],
   "metadata": {
    "id": "wDERJFpbbwcn"
   },
   "execution_count": 106,
   "outputs": []
  },
  {
   "cell_type": "code",
   "execution_count": 107,
   "metadata": {
    "id": "O_2ihY5d0Baf"
   },
   "outputs": [],
   "source": [
    "train_datagen = ImageDataGenerator(rescale = 1./255,shear_range=0.2,zoom_range= 0.2,horizontal_flip=True,vertical_flip=False)"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 108,
   "metadata": {
    "id": "woIttxeo0H6l"
   },
   "outputs": [],
   "source": [
    "test_datagen = ImageDataGenerator(rescale = 1./255)"
   ]
  },
  {
   "cell_type": "code",
   "source": [
    "import tensorflow as tf\n",
    "import os\n",
    "from tensorflow.keras.models import Sequential\n",
    "from tensorflow.keras.layers import Dense, Conv2D, Flatten, Dropout, MaxPooling2D\n",
    "from tensorflow.keras.preprocessing.image import ImageDataGenerator\n",
    "import numpy as np\n",
    "import matplotlib.pyplot as plt\n",
    "import IPython.display as display\n",
    "from PIL import Image\n",
    "import pathlib"
   ],
   "metadata": {
    "id": "zkRahElCas_0"
   },
   "execution_count": 105,
   "outputs": []
  },
  {
   "cell_type": "markdown",
      "source": [
        "Apply ImageDataGenerator Functionality To Train And Test set"
      ],
      "metadata": {
        "id": "nRcKkInLazLB"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "from google.colab import drive"
      ],
      "metadata": {
        "id": "EegzORgRa4Mo"
      },
      "execution_count": 109,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "from tensorflow.keras.preprocessing.image import ImageDataGenerator\n",
        "print(\"This dataset has been created and uploaded by IBM-TeamID-IBM-Project-11008-1659252006\")"
      ],
      "metadata": {
```

```
    "colab": {
     "base_uri": "https://localhost:8080/"
    },
    "id": "FLPYWmzwbmiF",
    "outputId": "f1fc9003-d1d1-4645-a483-fbc3bc301550"
   },
   "execution_count": 110,
   "outputs": [
    {
     "output_type": "stream",
     "name": "stdout",
     "text": [
      "This dataset has been created and uploaded by IBM-TeamID-IBM-Project-11008-1659252006\n"
     ]
    }
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 111,
   "metadata": {
    "colab": {
     "base_uri": "https://localhost:8080/"
    },
    "id": "tb_QlkBR0NkI",
    "outputId": "be4d0136-04b3-4e4e-9a8e-6e32193e944d"
   },
   "outputs": [
    {
     "output_type": "stream",
     "name": "stdout",
     "text": [
      "Found 10324 images belonging to 9 classes.\n"
     ]
    }
   ],
   "source": [
    "x_train= train_datagen.flow_from_directory(r\"/content/drive/MyDrive/dataset/dataset/training_set\",target_size=(64,64),class_mode=\"categorical\",batch_size=48)"
   ]
  },
  {
   "cell_type": "code",
   "execution_count": 112,
   "metadata": {
    "colab": {
     "base_uri": "https://localhost:8080/"
    },
    "id": "oMAXWl6nK_7i",
    "outputId": "26ff69ff-3f64-453b-a2b2-b7f778d958e7"
   },
   "outputs": [
    {
     "output_type": "stream",
     "name": "stdout",
     "text": [
      "Found 2280 images belonging to 9 classes.\n"
     ]
    }
   ],
   "source": [
    "x_test = test_datagen.flow_from_directory(r\"/content/drive/MyDrive/dataset/dataset/test_set\",target_size=(64,64),class_mode= \"categorical\",batch_size=48)\n"
   ]
  },
  {
   "cell_type": "code",
   "source": [
    "x_train.class_indices"
   ],
   "metadata": {
    "colab": {
     "base_uri": "https://localhost:8080/"
    },
    "id": "Pa6eD91IbWLV",
   "outputId": "fb6b7c2d-74e2-4280-e806-a3c1e7aa018e"
  },
  "execution_count": 113,
  "outputs": [
   {
    "output_type": "execute_result",
    "data": {
     "text/plain": [
      "{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}"
     ]
    },
    "metadata": {},
    "execution_count": 113
   }
  ]
 },
 {
  "cell_type": "code",
  "source": [
   "x_test.class_indices"
  ],
  "metadata": {
   "colab": {
    "base_uri": "https://localhost:8080/"
   },
   "id": "eX5O23vkbZYK",
   "outputId": "90b4cc12-0e8e-441f-ebb5-b9caa090fbd8"
  },
  "execution_count": 114,
  "outputs": [
   {
    "output_type": "execute_result",
    "data": {
     "text/plain": [
      "{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}"
     ]
    },
    "metadata": {},
    "execution_count": 114
   }
  ]
 },
 {
  "cell_type": "markdown",
  "source": [
   "\n",
   "MODEL BUILDING"
  ],
  "metadata": {
   "id": "HVYtoy_wgAjK"
   }
  },
  {
   "cell_type": "code",
   "source": [
    "from keras.models import Sequential\n",
    "from keras.layers import Dense\n",
    "from keras.layers import Convolution2D\n",
    "from tensorflow.keras.layers import Conv2D, MaxPooling2D\n",
    "from keras.layers import Dropout\n",
    "from keras.layers import Flatten"
   ],
   "metadata": {
    "id": "M-TTBCY1gcZv"
   },
   "execution_count": 115,
   "outputs": []
  },
  {
   "cell_type": "code",
   "source": [
    "model=Sequential()"
   ],
   "metadata": {
    "id": "T8SyKLh_ghOs"
   },
   "execution_count": 118,
   "outputs": []
  },
  {
   "cell_type": "code",
   "source": [
    "model.add(Convolution2D(32,(3,3), input_shape=(64,64,1), activation = 'relu'))"
   ],
   "metadata": {
    "id": "8_GLqje1gkuc"
   },
   "execution_count": 117,
   "outputs": []
  },
  {
   "cell_type": "code",
   "source": [
    "model.add(MaxPooling2D(pool_size=(2,2)))"
   ],
   "metadata": {
    "id": "VCjJWGK5gnCT"
   },
   "execution_count": 119,
   "outputs": []
```

```
},
{
 "cell_type": "code",
 "source": [
  "model.add(Flatten())"
 ],
 "metadata": {
  "id": "gmg-kQhTgpLN"
 },
 "execution_count": 120,
 "outputs": []
},
{
 "cell_type": "code",
 "source": [
  "model.add(Dense( units=512, activation='relu'))"
 ],
 "metadata": {
  "id": "JjA9nfThgrm6"
 },
 "execution_count": 121,
 "outputs": []
},
{
 "cell_type": "code",
    "source": [
     "model.add(Dense(units=9, activation='softmax'))"
    ],
    "metadata": {
     "id": "KtVrrVubgtyi"
    },
    "execution_count": 122,
    "outputs": []
   },
   {
    "cell_type": "code",
    "source": [
     "model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])"
    ],
    "metadata": {
     "id": "_awYZ1vZgwN2"
    },
    "execution_count": null,
    "outputs": []
   },
   {
    "cell_type": "code",
    "source": [
"model.fit(x_train,steps_per_epoch=len(x_train),epochs=5,validation_data=x_test,validation_steps=len(x_test))"
```

```
   ],
   "metadata": {
    "colab": {
     "base_uri": "https://localhost:8080/"
    },
    "id": "pDLjYtPr0PaU",
    "outputId": "6aedc5a3-8f5e-4cca-f4a4-eeef9695689a"
   },
   "execution_count": 126,
   "outputs": [
    {
     "output_type": "stream",
     "name": "stdout",
     "text": [
      "Epoch 1/5\n",
      "216/216 [==============================] - 162s 749ms/step - loss: 0.1131 - accuracy: 0.9633 - val_loss: 7.3499 - val_accuracy: 0.6456\n",
      "Epoch 2/5\n",
      "216/216 [==============================] - 39s 181ms/step - loss: 0.0356 - accuracy: 0.9900 - val_loss: 7.9273 - val_accuracy: 0.6461\n",
      "Epoch 3/5\n",
      "216/216 [==============================] - 42s 197ms/step - loss: 0.0294 - accuracy: 0.9923 - val_loss: 7.7494 - val_accuracy: 0.6469\n",
      "Epoch 4/5\n",
      "216/216 [==============================] - 41s 190ms/step - loss: 0.0173 - accuracy: 0.9951 - val_loss: 8.0277 - val_accuracy: 0.6461\n",
      "Epoch 5/5\n",
      "216/216 [==============================] - 42s 194ms/step - loss: 0.0072 - accuracy: 
0.9984 - val_loss: 8.4261 - val_accuracy: 0.6465\n"
     ]
    },
    {
     "output_type": "execute_result",
     "data": {
      "text/plain": [
       "<keras.callbacks.History at 0x7fa51f3652d0>"
      ]
     },
     "metadata": {},
     "execution_count": 126
    }
   ]
  },
  {
   "cell_type": "code",
   "source": [
    "model.save('Realtime.h5')"
   ],
   "metadata": {
    "id": "TSx4xTYnh4_0"
   },
   "execution_count": 133,
   "outputs": []
```

```
},
{
 "cell_type": "code",
 "source": [
  "a=len(x_train)\n",
  "b=len(x_test)"
 ],
 "metadata": {
  "id": "eqG-AvdRzpcr"
 },
 "execution_count": 128,
 "outputs": []
},
{
 "cell_type": "markdown",
 "source": [
  "Length of training and testing data"
 ],
 "metadata": {
  "id": "4VWpiBr-zyu-"
 }
},
{
 "cell_type": "code",
 "source": [
    "print(a)\n",
    "print(b)"
   ],
   "metadata": {
    "colab": {
     "base_uri": "https://localhost:8080/"
    },
    "id": "kCTyh470zupo",
    "outputId": "f366d812-3a13-4a35-91d8-7463fa0588bb"
   },
   "execution_count": 129,
   "outputs": [
    {
     "output_type": "stream",
     "name": "stdout",
     "text": [
      "216\n",
      "48\n"
     ]
    }
   ]
  },
  {
   "cell_type": "markdown",
   "source": [
        "TEST THE MODEL"
      ],
      "metadata": {
        "id": "DbSXYt2Y-pPW"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "from tensorflow.keras.models import load_model\n",
        "from tensorflow.keras.preprocessing import image\n",
        "import numpy as np\n",
        "import cv2"
      ],
      "metadata": {
        "id": "0jZMGOeA-rE2"
      },
      "execution_count": 141,
      "outputs": []
    },
    {
      "cell_type": "code",
      "source": [
        "img = image.load_img('/content/drive/MyDrive/dataset/dataset/test_set/F/107.png',target_size = (500,500))\n",
        "img"
    ],
    "metadata": {
     "id": "7_k-MbRF-5Qf",
     "outputId": "f4db9e20-b1ee-43f9-9afb-bec067674266",
     "colab": {
      "base_uri": "https://localhost:8080/",
      "height": 517
     }
    },
    "execution_count": 149,
    "outputs": [
     {
      "output_type": "execute_result",
      "data": {
       "text/plain": [
        "<PIL.Image.Image image mode=RGB size=500x500 at 0x7FA525504A90>"
       ],
       "image/png":
"iVBORw0KGgoAAAANSUhEUgAAAfQAAAH0CAIAAABEtEjdAAALgUlEQVR4nO3cvYvcVRuA4R03hlkstHNjI
VgYt7SMleAHaHqLlDaijXUKSyGxFC0Ee/UvUMEUdmolEWRFbCVrYWG1K9kwb/NCkP1FjZmdM3PvdZUHm
Xmybm6e4uRsbQEAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAABAzGz0AcU8//fTk+c8//3x
fn/Pcc8+dPPz222//y0xwBjw0egAAlk/cAYLEHSBI3AGCxB0gSNwBgsQdlMg9d07X119/PXn+/PPPP/iHX7hw
YfL84ODwgwT8cNprNHSBI3AGCxB0gSNwBgsQdIEjcAYLEHSDIPXeWY7FYjB7hrtlsjX6x33///cnzt99++yJUr
VybPP/vss/8yE2eAzR0gSNwBgsQdIEjcAYLEHSBI3AGCxB0g6NzoAWD5tre3J8/v3Llzel96qjf9qjf9P/3008nzxx57
bPL8o48+Or1h2Ag2d4AgcQIEneAIHEHCBJ3gCBxBwgSd4CgNXr2mo22Vu+5P/zww8fHD/7hTzzxxOT5z8fHD/7hTzzxxOT
5r7/++uAfvi9aI9Q9jcAYLEHSBI3AGCxB0gSNwBgsQdIMiTvwSdOzf9i72Uq5BPPvnkg38nDabO0CQuAMEi
TtAkLgDBIk7QJC4AwSJO0CQe+4E3b59/WwX788cfRI6OjzO/74YPQIrCmbO0CQuAMEiTtAkLgD
BIk7QJC4AwSJO0CQe+6wwX788cfRI7CmbO4AQeIOECTuAEHiDhAk7gBBB4g4QJO4AQbPRAxCxWCxGj3D
HaKv9hr9Se9l1P9CbARbO4AQeIOECTuAEHiDhAk7gBB4g4Q5Mlf2GBvzZ3gCBxBwgSdwAgsQdIEjcAYLEHSBI3AGCxB0gSNwBgsQdIEjc
EneAIHEHCBJ3gCD33Amaz+eT50888eT50dHRW+R8f0FRQ5J6O10Js+G/eT50dHRyuehNP28ssvT56/
AYLcc+f+nDvnd4bTtbe3d/Jwf39/KR/+8ccfT56/8cYbS/n89WFzBwgSd4AgcQcIEneAIHEHCBJ3gCBxBwiajR6
ADXPt2rXJ86tXr654kr+xs7Mzeb6U99wXi8WDf8iyzGbBv8JDfsK9n6TNHSBI3AGCxB0gSNwBgsQdIEjcAYLEH
SDI29wEXbx4cfL8hx9+uK/POX/+/DLGYdru7u7oEcps7gBB4g4QJO4AQeIOECTuAEHiDhDkKiRBN2/enDy/32
dd//zzz2WMw7Rbt26t/kt/++231X/pEDZ3gCBxBwgSd4AgcQcIEneAIHEHCBJ3gCD33IEz5PHHHx89worY3A
GCxB0gSNwBgsQdIEjcAYLEHSBI3AGC3HMHTtd8Ph89wl2//PLL6BFWxOYOECTuAEHiDhAk7gBB4g4QJO4A
QeIOEOSeO3CG3LhxY/QIK2JzBwgSd4AgcQcIEneAIHEHCBJ3gCBxBwhyz32DXb16dfL84sWLk+evv/765Pl33
303eX7p0qX/NtjGeeSRR0aPUHZ4eDh6hLveeuut0SOsiM0dIEjcAYLEHSBI3AGCxB0gSNwBgmajB+Avrl27Nn
l+r1uP3JfZbPoXfnd3d/L81q1bpznOEtzrT7RWFovF6r90I34yp8rmDhAk7gBB4g4QJO4AQeIOECTuAEHiDhD
kyd8x5vP55Ln77Ey6fv366BH+2Vo97fvSSy9nt+4cWPFk4xicwcIEneAIHEHCBJ3gCBxBwgSd4AgcQcIOutPH
o8y5IVr7vXG9/r/73j22Wcnz2/evLnaQf7O8fHx5Pn29vaKJ9nynrvNHSBJ3AGCxB0gSNwBgsQdIEjcAYLEHSDI
e+6wAZ555pnJ8yH33D/55JPJ8yH32b/88svVf+lGsLkDBIk7QJC4AwSJO0CQuAMEiTtAkLgDBLnnDhvgqaeeG
j3CXVeuXBk9wl2vvvrq6BHWlM0dIEjcAYLEHSBI3AGCxB0gSNwBgsQdIMg9d1gv169gv169fP3n43nvvrX6Sra2t/f3
9Id970pCX6zeazR0gSNwBgsQdIEjcAYLEHSBI3AGCXIUE7mlvb2/0CP9vjh49ejR1gSNwBgsQdIEjcAYLEH
NwBgsQdIEjcAYLcc+cMWSwWo0dYU2v1k/npp59HOv7+rn2Sj2dwBgsQdIEjcAYLEHSBI3AGCxB0gSNwB
gmajBzij1upaMetvNlvOX9Uvvvhi8vyVV15ZyucvxbL+sGeczR0gSNwBgsQdIEjcAYLEHSBI3AGCxB0gyHvusF4
+/PDDk4e7u7uT//HBwcF9nVNcAYLEHSBI3AGCxB0gSNwBgsQdIEjcAYK8mzyG99zBu+2nyuYOEC
TuAEHiDhAk7gBB4g4QJO4AQZ78BU7X0dHR6BHOHOIps7QJC4AwSJO0CQuAMEiTtAkLgDBIk7QJB77mPs7Ox
Mnh8eHq54EiDJ5g4QJO4AQeIOECTuAEHiDhAk7gBB4g4Q5J77GK+99troEWDJ7vWvN7znPoTNHSBI3AGCx
B0gSNwBgsQdIEjcAYLEHSBoNnqAM2o+n0ee8+dzeWe+1qxuQMEiTtAkLgDBIk7QJC4AwSJO0CQuAMEiTt
AkLgDBIk7QJC4AwSJO0CQuAMEiTtAkLgDBIk7QJC4AwSJO0CQuAMEiTtAkLgDBIk7QJC4AwSJO0CQuAMEiT
tAkLgDBIk7QJC4AwSJO0DQbPQA/MVisRg9AizZbYzA9jcAYLEHSBI3AGCxB0gSNwBgsQdIEjcAYLEHSBI3AG
CxB0gSNwBgsQdIEjcAYLEHSBI3AGCzo0eAIjY2dkZPQJ32dwBgsQdIEjcAYLEHSBI3AGCxB0gSNwBgsQdIEjcA
YLEHSBI3AGCxB0gSNwBgsQdIEjcAYLEHSBI3AGCxB0gSNwBgsQdIEjcAYLEHSBoNnqAM2o+n0+eHx4erngSO
G2zmc4MYHMHCBJ3gCBxBwgSd4AgcQcIEneAIHEHCHL/dL0sFovRI8CSuec+hM0dIEjcAYLEHSBI3AGCxB0gS
NwBgsQdIEjcAYLEHSBI3AGCxB0gSNwBgsQdIEjcAYLEHSDIO8vrZXd3d/L81q1bpznOEtzrT7RWFovF6r90I3AGCxB0gSN
wBgsQdIEjcAYLEHSDIFaXNsFgsRo8A/5GrkOEk4mDhHbkcPY3AGCxB0gSNwBgsQdIEjcAYLEHSBI3AGCxB0gSNwBgsQdIEjc
AYLEHSBI3AGCxB0gSNwBgs6NHoB/5fPPPz95eGXVVzT6P0Jx93ePny5dvx595dPPz4+97ePny5dVPAmwEmztAkLgDBIk7QJC4AwSJO0CQuAMEiTtA0G
z0APwre3t7w/ZWRWBQk7QJC4AwSJO0CQuAMEiTtAkLgDBIk7QJC4AwSJO0CQuAMEiTtAkLgDBIk7QJC4AwSJO0CQuAMEiTtA0G
z0APwre3t7w/ZWRWBQk7QJC4AwSJO0CQuAMEiTtAkLgDBIk7QJC4AwSJO0CQuAMEiTtAkLgDBIk7QJC4AwSJO0CQuAMEiTtA0G
C73HNfKzZ3gCBxBwgSd4AgcQcIEneAIHEHCHIVcoO5Cslamc30ZI3Y3AGCxB0gSNwBgsQdIEjcAYLEHSBI3AG
C3EvdYJcuXZo8/+abb1Y8CWy5575mbO4AQeIOECTuAEHiDhAk7gBB4g4QJO4AQe6lBnnnnSHcc18rNneAI
HEHCBJ3gCBxBwgSd4AgcQcIEneAoHOjB2D53nnnncnzd999999d8WTkPTCCy+MHoF/ZnMHCBJ3gCBxBwgSd4
AgcQcIEneAIHEHCPL+8hninXeW4vbt25PmPn58+fX/Ek/A2bO0CQuAMEiTtAkLgDBIk7QJC4AwSJO0CQ99zPkE
cffXTy/PLlyyue5CcffXTy/PLlyyue5CcffXTy/PLlyyue5CcffXTy/PLlyyue5CcffXTy/PLlyyue5CcffXTy/PLlyyue5Cc
L+hs0dIEjcAYLEHSBI3AGCxB0gSNwBgsQdIMg9d7a2t7cnz4+Pj1c8CWy5575mbO4AQeIOECTuAEHiDhAk7gBB4g4QJO4AQe
HSBI3AGCxB0gSNwBgv4H9q0WvRzP5NQAAAAASUVORK5CYII=\n"

        },

        "metadata": {},

        "execution_count": 149

      }
```

```
  ]
 },
 {
  "cell_type": "code",
  "source": [
   "from skimage.transform import resize\n",
   "def detect(frame):\n",
   "    img=image.img_to_array(frame)\n",
   "    img = resize(img,(64,64,1))\n",
   "    img = np.expand_dims(img,axis=0)\n",
   "    pred=np.argmax(model.predict(img))\n",
   "    op=['A','B','C','D','E','F','G','H','I']\n",
   "    print(\"THE PREDICTED LETTER IS \",op[pred])"
  ],
  "metadata": {
   "id": "iM2_tL1I_Dp0"
  },
  "execution_count": 150,
  "outputs": []
 },
 {
  "cell_type": "code",
  "source": [
   "from skimage.transform import resize\n",
   "def detect(frame):\n",
    "  img=resize(frame,(64,64,1))\n",
    "  img=np.expand_dims(img,axis=0)\n",
    "  if(np.max(img)>1):\n",
    "    prediction=model.predict(img)\n",
    "    print(prediction)\n",
    "    prediction=model.predict_classes(img)\n",
    "    print(prediction)"
   ],
   "metadata": {
    "id": "vWsj7yZSBYu9"
   },
   "execution_count": 153,
   "outputs": []
  },
  {
   "cell_type": "code",
   "source": [
    "arr= image.img_to_array(img)"
   ],
   "metadata": {
    "id": "gr_dYsltBmUW"
   },
   "execution_count": 157,
   "outputs": []
  },
{
 "cell_type": "code",
 "source": [
  "frame=cv2.imread('/content/drive/MyDrive/dataset/dataset/test_set/F/107.png')\n",
  "data=detect(frame)\n",
  "from google.colab.patches import cv2_imshow\n",
  "cv2_imshow(frame)\n",
  "cv2.waitKey(0)\n",
  "cv2.destroyAllWindows()"
 ],
 "metadata": {
  "id": "pR8Z3Sk5BpwH",
  "outputId": "31621d20-dad9-4b81-d297-ddb2f28a3f8c",
  "colab": {
   "base_uri": "https://localhost:8080/",
   "height": 81
  }
 },
 "execution_count": 159,
 "outputs": [
  {
   "output_type": "display_data",
   "data": {
    "text/plain": [
     "<PIL.Image.Image image mode=RGB size=64x64 at 0x7FA50E6BA110>"
```

```
          ],
          "image/png":
"iVBORw0KGgoAAAANSUhEUgAAAEAAAABACAIAAAAlC+aJAAACv0lEQVR4nO2aLc/qMBTHCyNkCwLJQOC
AOSxYXgyehI8AnwAEfprMIZB8BCAEgUSCICDwBIEeZCO9YrkLWaFb39hu7n6KZz09O//ntD3rVgBiYqJEqVSC
EEIIa7Va2LFQsd1u4V9UVQ07HEIgAmozmUycpl6v9/sIfUAFSJKEMej3+2GF+hlUQCqVclsLhUKQFJGSZHeB4
V1AsVgUcQuxAizLEuofiBbwer2E+geiBciyjGk1DIP9FmIF4Dkej+xOEuwuXNBVJZFIYFo9BnSEmQEudSDMOe
BLu932teEm4H3JZ0fTNAjher2GEE6nU46ev6LrOlpo3zNAVIaDW4odQuVy2fmRTqeD9yJ6jBUr4HA4OD+ez
2fwXtfr9f3P2+2GMQ5zFQpILpfDtEZOALpwXS4XjH3kBKBsNhtMK72A0Wg0m80ghLvdDm+ZyWSCuzVN03
NlMBgQB4fn44r5DQCAqqrfmlC473i8yLIcPHqMAF3XUeemaXrMWq0WZwFE0TsCPl6vVquoc9u20e54Qpv
ElUrFc2U+n3teAqxWK/435pWB4XDo65l/9LwEoBPgdDp5bPb7fZB4olIHNE3zXGk2m0E6ChcQZCSgNufz+X6
/iwqIC67D5XKJaY2oAMMw3GdmvDxfiPfUYv89AADCnX5UJrHL4/EgsicWoCgKaRehEO/Eu92uiDgcFEUhzQ
DxHJBlGX3i5QWFgMjNAVJiAWFD825VaCkgfd37z2cgWgIoiky0BFDw/wlgfOWPh6JExqtQ2NAIyOfz3OOghv
IjobhRFA+hYCwWC75x/BrnKyJ3KNZo+g/lIqbBL3ZkLiIEUJw8oJ/E9Xqdui9HmM5acE/CTzMAABiPxyzdPTQa
DYperKddOCbBsiyiD/oOrIUsm80yenDodDqCTgX6w6sI0J305XBiS5Ik27bZ/SSTSYoB+QclASxgQI0JigAAAABJ
RU5ErkJggg==\n"
        },
        "metadata": {}
      }
    ]
  }
 ],
 "metadata": {
  "colab": {
   "collapsed_sections": [],
   "provenance": []
  },
  "kernelspec": {
   "display_name": "Python 3",
   "name": "python3"
  },
  "language_info": {
   "name": "python"
  }
 },
 "nbformat": 4,
 "nbformat_minor": 0
}
```