

```

{
  "nbformat": 4,
  "nbformat_minor": 0,
  "metadata": {
    "colab": {
      "provenance": [],
      "collapsed_sections": []
    },
    "kernelspec": {
      "name": "python3",
      "display_name": "Python 3"
    },
    "language_info": {
      "name": "python"
    }
  },
  "cells": [
    {
      "cell_type": "markdown",
      "source": [
        "IBM Project Id : PNT2022TMID21192\n",
        "\n",
        "Sprint1: Image Preprocessing\n",
        "\n",
        "Step1 : Import Image Generator Library and configure it\n",
        "\n",
        "Stpe2 : Apply Image Datagenerator Functionality to Train and Test
set\n",
        "\n",
        "\n"
      ],
      "metadata": {
        "id": "Pm53bJggtuSj"
      }
    },
    {
      "cell_type": "code",
      "execution_count": 1,
      "metadata": {
        "id": "IcJY0GJtkN20"
      },
      "outputs": [],
      "source": [
        "from tensorflow.keras.preprocessing.image import
ImageDataGenerator"
      ]
    },
    {
      "cell_type": "code",
      "source": [
        "# Training Datagen\n",
        "train_datagen =
ImageDataGenerator(rescale=1/255,zoom_range=0.2,horizontal_flip=True,vertic
al_flip=False)\n",
        "# Testing Datagen\n",
        "test_datagen = ImageDataGenerator(rescale=1/255)"
      ],
      "metadata": {
        "id": "8wRmBf_iktb3"
      },
      "execution_count": 2,
    }
  ]
}

```

```

    "outputs": []
  },
  {
    "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": "q2Zh8IyHkuVP"
    },
    "execution_count": 3,
    "outputs": []
  },
  {
    "cell_type": "code",
    "source": [
      "from google.colab import drive"
    ],
    "metadata": {
      "id": "5hH7XiHnkwZb"
    },
    "execution_count": 4,
    "outputs": []
  },
  {
    "cell_type": "code",
    "source": [
      "!unzip '/content/DataSet.zip'"
    ],
    "metadata": {
      "colab": {
        "base_uri": "https://localhost:8080/"
      },
      "id": "LEv7IFYnpzzR",
      "outputId": "9d93ea61-0b6e-4fe6-fddf-5241b1b60dc4"
    },
    "execution_count": 15,
    "outputs": [
      {
        "output_type": "stream",
        "name": "stdout",
        "text": [
          "\u001b[1;30;43mStreaming output truncated to the last 5000
lines.\u001b[0m\n",
          " extracting: DataSet/training_set/G/1225.png  \n",
          " extracting: DataSet/training_set/G/1226.png  \n",
          " extracting: DataSet/training_set/G/1227.png  \n",
          " extracting: DataSet/training_set/G/1228.png  \n",
          " extracting: DataSet/training_set/G/1229.png  \n",
          " inflating: DataSet/training_set/G/123.png  \n",

```

[illegible]



[illegible]









[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

```

" inflating: DataSet/training_set/I/951.png \n",
" inflating: DataSet/training_set/I/952.png \n",
" inflating: DataSet/training_set/I/953.png \n",
" inflating: DataSet/training_set/I/954.png \n",
" inflating: DataSet/training_set/I/955.png \n",
" inflating: DataSet/training_set/I/956.png \n",
" inflating: DataSet/training_set/I/957.png \n",
" inflating: DataSet/training_set/I/958.png \n",
" inflating: DataSet/training_set/I/959.png \n",
" inflating: DataSet/training_set/I/96.png \n",
" inflating: DataSet/training_set/I/960.png \n",
" inflating: DataSet/training_set/I/961.png \n",
" inflating: DataSet/training_set/I/962.png \n",
" inflating: DataSet/training_set/I/963.png \n",
" inflating: DataSet/training_set/I/964.png \n",
" inflating: DataSet/training_set/I/965.png \n",
" inflating: DataSet/training_set/I/966.png \n",
" inflating: DataSet/training_set/I/967.png \n",
" inflating: DataSet/training_set/I/968.png \n",
" inflating: DataSet/training_set/I/969.png \n",
" inflating: DataSet/training_set/I/97.png \n",
" inflating: DataSet/training_set/I/970.png \n",
" inflating: DataSet/training_set/I/971.png \n",
" inflating: DataSet/training_set/I/972.png \n",
" extracting: DataSet/training_set/I/973.png \n",
" inflating: DataSet/training_set/I/974.png \n",
" inflating: DataSet/training_set/I/975.png \n",
" inflating: DataSet/training_set/I/976.png \n",
" inflating: DataSet/training_set/I/977.png \n",
" inflating: DataSet/training_set/I/978.png \n",
" inflating: DataSet/training_set/I/979.png \n",
" inflating: DataSet/training_set/I/98.png \n",
" inflating: DataSet/training_set/I/980.png \n",
" inflating: DataSet/training_set/I/981.png \n",
" inflating: DataSet/training_set/I/982.png \n",
" extracting: DataSet/training_set/I/983.png \n",
" inflating: DataSet/training_set/I/984.png \n",
" inflating: DataSet/training_set/I/985.png \n",
" inflating: DataSet/training_set/I/986.png \n",
" inflating: DataSet/training_set/I/987.png \n",
" inflating: DataSet/training_set/I/988.png \n",
" inflating: DataSet/training_set/I/989.png \n",
" inflating: DataSet/training_set/I/99.png \n",
" inflating: DataSet/training_set/I/990.png \n",
" inflating: DataSet/training_set/I/991.png \n",
" inflating: DataSet/training_set/I/992.png \n",
" extracting: DataSet/training_set/I/993.png \n",
" inflating: DataSet/training_set/I/994.png \n",
" inflating: DataSet/training_set/I/995.png \n",
" extracting: DataSet/training_set/I/996.png \n",
" inflating: DataSet/training_set/I/997.png \n",
" inflating: DataSet/training_set/I/998.png \n",
" inflating: DataSet/training_set/I/999.png \n"
]
}
],
{
  "cell_type": "code",
  "source": [

```

```

        "from tensorflow.keras.preprocessing.image import
ImageDataGenerator\n",
        "print(\"This dataset has been created and uploaded by IBM-ProjetID
: PNT2022TMDID24263\")"
    ],
    "metadata": {
        "colab": {
            "base_uri": "https://localhost:8080/"
        },
        "id": "yDSzQm6orQcO",
        "outputId": "80b799f9-df9b-4c14-81b5-1fde05c109ec"
    },
    "execution_count": 17,
    "outputs": [
        {
            "output_type": "stream",
            "name": "stdout",
            "text": [
                "This dataset has been created and uploaded by IBM-ProjetID :
PNT2022TMDID24263\n"
            ]
        }
    ]
},
{
    "cell_type": "code",
    "source": [
        "train_datagen =
ImageDataGenerator(rescale=1./255, zoom_range=0.2, horizontal_flip=True,
vertical_flip=False)"
    ],
    "metadata": {
        "id": "vhib66fjrzGD"
    },
    "execution_count": 18,
    "outputs": []
},
{
    "cell_type": "code",
    "source": [
        "test_datagen= ImageDataGenerator(rescale=1./255)"
    ],
    "metadata": {
        "id": "N8yfEBDFrz2m"
    },
    "execution_count": 21,
    "outputs": []
},
{
    "cell_type": "code",
    "source": [
        "x_train =
train_datagen.flow_from_directory('/content/DataSet/training_set', target_si
ze=(64,64), batch_size=300,\n",
        "
class_mode='categorical', color_mode = \"grayscale\")"
    ],
    "metadata": {
        "colab": {
            "base_uri": "https://localhost:8080/"
        },

```

```

      "id": "e84w9dHNR9vu",
      "outputId": "1009e9af-8f0e-44db-fce8-8a71d05d0d40"
    },
    "execution_count": 26,
    "outputs": [
      {
        "output_type": "stream",
        "name": "stdout",
        "text": [
          "Found 15130 images belonging to 9 classes.\n"
        ]
      }
    ]
  },
  {
    "cell_type": "code",
    "source": [
      "x_test =
test_datagen.flow_from_directory('/content/DataSet/test_set',target_size=(6
4,64), batch_size=300,\n",
      "
class_mode='categorical', color_mode = \"grayscale\")"
    ],
    "metadata": {
      "colab": {
        "base_uri": "https://localhost:8080/"
      },
      "id": "V9pj52QteSL",
      "outputId": "175dd706-638a-42b7-ced6-68ae5e421ec1"
    },
    "execution_count": 27,
    "outputs": [
      {
        "output_type": "stream",
        "name": "stdout",
        "text": [
          "Found 2250 images belonging to 9 classes.\n"
        ]
      }
    ]
  },
  {
    "cell_type": "code",
    "source": [
      "x_train.class_indices"
    ],
    "metadata": {
      "colab": {
        "base_uri": "https://localhost:8080/"
      },
      "id": "tMoJLwUatkHp",
      "outputId": "b4f9c406-3223-4d6c-b2ba-2c40638c45a6"
    },
    "execution_count": 28,
    "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": 28
    }
  ]
},
{
  "cell_type": "code",
  "source": [
    "x_test.class_indices"
  ],
  "metadata": {
    "colab": {
      "base_uri": "https://localhost:8080/"
    },
    "id": "ernG1j-xtl0U",
    "outputId": "6a2f47ae-fc1d-4ae5-b1d1-9cc71e89a8f2"
  },
  "execution_count": 29,
  "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": 29
    }
  ]
}
]
}

```