

```
{
  "nbformat": 4,
  "nbformat_minor": 0,
  "metadata": {
    "colab": {
      "provenance": []
    },
    "kernelspec": {
      "name": "python3",
      "display_name": "Python 3"
    },
    "language_info": {
      "name": "python"
    }
  },
  "cells": [
    {
      "cell_type": "markdown",
      "source": [
        "Import Image Data Generator from keras"
      ],
      "metadata": {
        "id": "QFQuAs3zewfb"
      }
    },
    {
      "cell_type": "code",
      "source": [
        "from keras.preprocessing.image import ImageDataGenerator"
      ],
      "metadata": {
```

```

    "id": "rgB5fpFye0-p"
  },
  "execution_count": null,
  "outputs": []
},
{
  "cell_type": "markdown",
  "source": [
    "Image Data Augmentation"
  ],
  "metadata": {
    "id": "f4GGlaMye_pN"
  }
},
{
  "cell_type": "code",
  "source": [
    "#setting parameter for Image Data augmentation to the training data\n",
    "train_datagen = ImageDataGenerator\n",
    "(rescale=1./255,shear_range=0.2,zoom_range=0.2,horizontal_flip=True)\n",
    "#Image Data augmentation to the testing data\n",
    "test_datagen=ImageDataGenerator(rescale=1./255)"
  ],
  "metadata": {
    "id": "UJIQ6nl1fBut"
  },
  "execution_count": null,
  "outputs": []
},
{
  "cell_type": "markdown",

```

```

"source": [
  "Loading our data and performing data augmentation\n"
],
"metadata": {
  "id": "PWSh6UDLfM3z"
}
},
{
  "cell_type": "code",
  "source": [
    "x_train = train_datagen.flow_from_directory\n",
    "(' ../data/train_set',target_size=(64,64),batch_size=5,color_mode='rgh',\n",
    "class_mode='categorical')\n",
    "x_test = test_datagen.flow_from_directory\n",
    "(' ../data/test_set',target_size=(64,64),batch_size=5,color_mode='rgh',\n",
    "class_mode='categorical')\n"
  ],
  "metadata": {
    "id": "_ZOkvY1rfOiv"
  },
  "execution_count": null,
  "outputs": []
},
{
  "cell_type": "markdown",
  "source": [
    "Found 742 images belonging to 4 classes.\n",
    "Found 198 images belonging to 4 classes."
  ],
  "metadata": {
    "id": "m-6HNeMlgHr_"
  }
}

```

```
}  
,  
{  
  "cell_type": "markdown",  
  "source": [  
    "Importing the Required Libraries"  
  ],  
  "metadata": {  
    "id": "jAjckwoVgcSI"  
  }  
},  
{  
  "cell_type": "code",  
  "source": [  
    "import numpy as np\n",  
    "import tensorflow\n",  
    "from tensorflow.keras.models import Sequential\n",  
    "from tensorflow.keras import layers\n",  
    "from tensorflow.keras.layers import Dense, Flatten\n",  
    "from tensorflow.keras.layers import Conv2D\n",  
    "from keras.optimizers import Adam\n",  
  ],  
  "metadata": {  
    "id": "EdDzPs3rgfRZ"  
  },  
  "execution_count": null,  
  "outputs": []  
},  
{  
  "cell_type": "code",  
  "source": [  

```

[illegible]

"[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 3, 18, 18, 18, 126, 136,\n",
"175, 26, 166, 255, 247, 127, 0, 0, 0, 0],\n",
"[0, 0, 0, 0, 0, 0, 0, 0, 30, 36, 94, 154, 170, 253, 253, 253,\n",
"253, 253, 225, 172, 253, 242, 195, 64, 0, 0, 0, 0],\n",
"[0, 0, 0, 0, 0, 0, 0, 49, 238, 253, 253, 253, 253, 253, 253,\n",
"253, 253, 251, 93, 82, 82, 56, 39, 0, 0, 0, 0, 0],\n",
"[0, 0, 0, 0, 0, 0, 0, 18, 219, 253, 253, 253, 253, 253, 198,\n",
"182, 247, 241, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],\n",
"[0, 0, 0, 0, 0, 0, 0, 80, 156, 107, 253, 253, 205, 11, 0,\n",
"43, 154, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],\n",
"[0, 0, 0, 0, 0, 0, 0, 0, 14, 1, 154, 253, 90, 0, 0, 0, 0, 0,\n",
"0, 0, 0, 0, 0, 0, 0, 0, 0],\n",
"[0,0,0,0,0,0,0,0,0,0,0,139,253, 190,2,0,0,0,0,0,0,0,0,0,0,\n",
"0,0],\n",
"[0,0,0,0,0,0,0,0,0,0,11,190, 253,70,0,0,0,0,0,0,0,0,0,0,\n",
"0,0],\n",
"[0,0,0,0,0,0,0,0,0,0,0,35, 241,225,160,108,1,0,0,0,0,0,0,0,\n",
"0,0],\n",
"[0,0,0,0,0,0,0,0,0,0,0, 81,240,253,253,119,25,0,0,0,0,0,0,\n",
"0,0],\n",
"[0,0,0,0,0,0,0,0,0,0,0, 0,45,186,253,253,150,27,0,0,0,0,0,\n",
"0,0],\n",
"[0,0,0,0,0,0,0,0,0,0,0, 0,0,16,93,252,253,187,0,0,0,0,0,\n",
"0,0],\n",
"[0,0,0,0,0,0,0,0,0,0,0, 0,0,0,0,249,253,249,64,0,0,0,0,\n",
"0,0],\n",
"[0,0,0,0,0,0,0,0,0,0,0, 0,46,130,183,253,253,207,2,0,0,0,0,\n",
"0,0],\n",
"[0,0,0,0,0,0,0,0,0,0,0,39,\n",
"148,229,253,253,253,250,182,0,0,0,0,0, 0,0],\n",
"[0,0,0,0,0,0,0,0,0,0,24,114,221,\n"]

```

"253,253,253,253,201,78,0,0,0,0,0,0,0,0,0,0,\n",
"[0,0,0,0,0,0,0,0,23,66,213,253,253,\n",
"253,253,198,81,2,0,0,0,0,0,0,0,0,0,0,0,\n",
"[0,0,0,0,0,0,18,171,219,253,253,253,253,\n",
"195,80,9,0,0,0,0,0,0,0,0,0,0,0,0,0,\n",
"[0,0,0,0,55,172,226,253,253,253,253,244,133,\n",
"11,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,\n",
"[0,0,0,0,136,253,253,253,212,135,132,16,0,\n",
"0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,\n",
"[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,\n",
"[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,\n",
"[0,0,0,0,0,0,0,0,0,0,0,0,0,0,\n",
"0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]], dtype=uint8)\n",
"\n",
"y_train[0]\n",
"\n",
"5"
],
"metadata": {
  "id": "w1w3mz8NgxgY"
},
"execution_count": null,
"outputs": []
},
{
  "cell_type": "markdown",
  "source": [
    "Reshaping The Data"
  ],
  "metadata": {
    "id": "Kru2YI-ohiwq"
  }
}

```

```

    }
  },
  {
    "cell_type": "code",
    "source": [
      "x_train=x_train.shape.reshape (60000,28,28,1).astype ('float32')\n",
      "\n",
      "x_test=x_test.shape.reshape (10000,28,28,1).astype ('float32')\n",
    ],
    "metadata": {
      "id": "azAOZ7rFhl0o"
    },
    "execution_count": null,
    "outputs": []
  },
  {
    "cell_type": "markdown",
    "source": [
      "Applying One Hot Encoding"
    ],
    "metadata": {
      "id": "gU-RMeH2h1BC"
    }
  },
  {
    "cell_type": "code",
    "source": [
      "number_of_classes = 10\n",
      "\n",
      "\n",
      "y_train=np.utils.to_categorical(y_train, number_of_classes)\n",

```



```
"y_test=np.utils.to_categorical(y_test, number_of_classes)\n",  
"\n",  
"y_train[0]\n",  
"array([0 . , 0 . , 0 . , 0 . , 0 . , 1 . , 0 . , 0 . , 0 .], dtype=float32"  
],  
"metadata": {  
  "id": "tqRQ2q3Ph2j9"  
},  
"execution_count": null,  
"outputs": []  
}  
]  
}
```