Introduction to Tensorflow Image Processing

FASHION MNIST DATASET

```
In [1]: # import all libraries
        import numpy as np
        import tensorflow as tf
        from tensorflow import keras
        import matplotlib.pyplot as plt
```

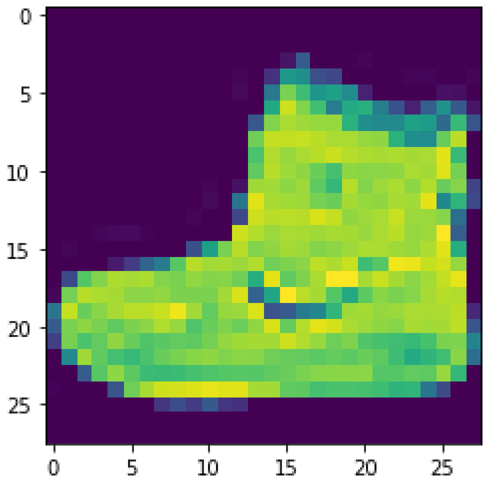
Loading and exploring the Dataset

```
In [2]: # load the dataset
        fashion_mnist = keras.datasets.fashion_mnist
```

```
In [4]: # split train-test data
        # train:test = 6:1 of the 70k labeled images in the dataset
        (train_images, train_labels), (test_images, test_labels) = fashion_mnist.lo
```

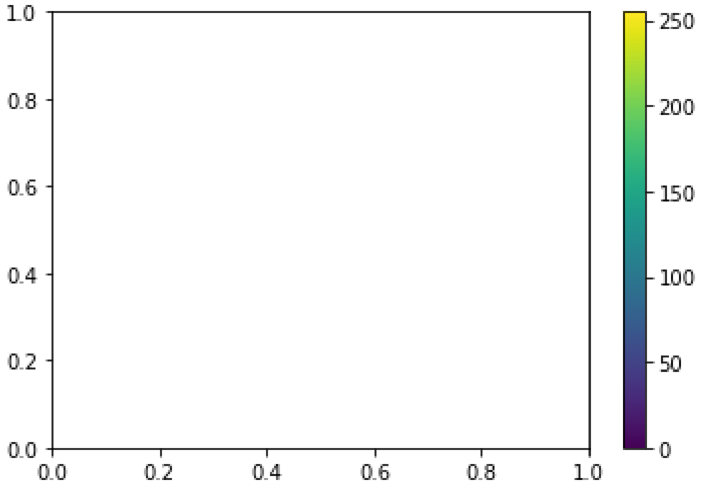
```
In [5]: plt.figure()

Out[5]: <Figure size 432x288 with 0 Axes>

        <Figure size 432x288 with 0 Axes>
```

```
In [12]: mappable = plt.imshow(train_images[0])
```

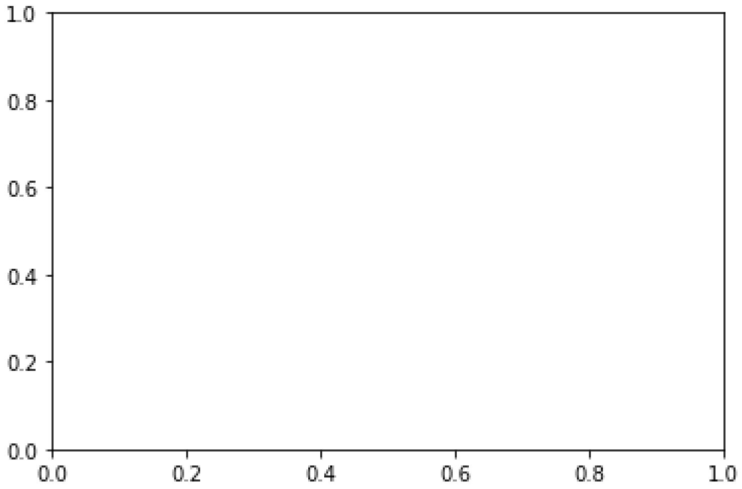


In [13]:
```
plt.colorbar(mappable=mappable)
```

Out[13]:
```
<matplotlib.colorbar.Colorbar at 0x7fab36846760>
```



In [14]:
```
plt.grid(False)
```



In [15]:
```
plt.show()
```

Pre-processing the Dataset

In [17]:
```
# normalizing values of the pixel in the greyscale images to 0-1 from 0-255
train_images = train_images / 255.0
test_images = test_images / 255.0
```

Build the Model

```
In [20]: model = keras.Sequential([keras.layers.Flatten(input_shape=(28,28)),
                                   keras.layers.Dense(128, activation=tf.nn.relu),
                                   keras.layers.Dense(10, activation=tf.nn.softmax)]
```

Train the Model

```
In [21]: model.compile( optimizer = 'adam', loss = 'sparse_categorical_crossentropy'
```

```
In [22]: model.fit(train_images, train_labels, epochs=5)

         Epoch 1/5
         1875/1875 [==============================] - 3s 1ms/step - loss: 0.4983 -
         accuracy: 0.8269: 1s - loss: 0.5722 - accuracy: 0.80
         Epoch 2/5
         1875/1875 [==============================] - 2s 1ms/step - loss: 0.3759 -
         accuracy: 0.8641
         Epoch 3/5
         1875/1875 [==============================] - 2s 1ms/step - loss: 0.3348 -
         accuracy: 0.8770
         Epoch 4/5
         1875/1875 [==============================] - 2s 1ms/step - loss: 0.3111 -
         accuracy: 0.8853
         Epoch 5/5
         1875/1875 [==============================] - 2s 1ms/step - loss: 0.2949 -
         accuracy: 0.8911: 1s - los - ETA: 0s - loss: 0.2961 - ac

Out[22]: <tensorflow.python.keras.callbacks.History at 0x7faaf5784550>
```

Evaluating the Model

```
In [23]: test_loss, test_acc = model.evaluate(test_images, test_labels)

         313/313 [==============================] - 0s 851us/step - loss: 0.3403 -
         accuracy: 0.8800
```

```
In [24]: predictions = model.predict(test_images)
```

```
In [25]: predictions[0]

Out[25]: array([5.1621549e-07, 1.7003784e-09, 1.2960587e-06, 8.0824298e-09,
                6.9123791e-07, 4.8180288e-04, 3.5897237e-06, 2.9780578e-02,
                2.1086351e-05, 9.6971053e-01], dtype=float32)
```

```
In [27]: np.argmax(predictions[0])

Out[27]: 9
```

```
In [28]: test_labels[0]

Out[28]: 9
```