

```
In [31]: import tensorflow as tf
        from tensorflow import keras
```

```
In [33]: (x_train,y_train),(x_test,y_test)=keras.datasets.mnist.load_data()
```

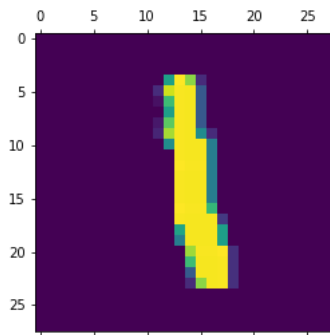
```
In [35]: x_train[0]
```

[illegible]

```
In [36]: import matplotlib.pyplot as plt
```

```
In [48]: plt.matshow(x_train[6])
```

```
Out[48]: <matplotlib.image.AxesImage at 0x1ef7c3419d0>
```



```
In [43]: y_train[6]
```

```
Out[43]: 1
```

```
In [45]: x_train=x_train/255  
x_test=x_test/255
```

```
In [46]: y_train[0]
```

```
Out[46]: 5
```

```
In [52]: x_train_flattened=x_train.reshape(len(x_train),28*28)  
x_test_flattened=x_test.reshape(len(x_test),28*28)
```

```
In [53]: x_train_flattened.shape
```

```
Out[53]: (60000, 784)
```

```
In [54]: x_test_flattened.shape
```

```
Out[54]: (10000, 784)
```

```
In [64]: model=keras.Sequential([  
    keras.layers.Dense(10,input_shape=(784,),activation='sigmoid')  
])
```

```
model.compile(optimizer='adam',  
              loss='sparse_categorical_crossentropy',  
              metrics=['accuracy'])  
model.fit(x_train_flattened,y_train,epochs=5)
```

```
Epoch 1/5  
1875/1875 [=====] - 3s 1ms/step - loss: 0.4711 - accuracy: 0.8773  
Epoch 2/5  
1875/1875 [=====] - 3s 2ms/step - loss: 0.3041 - accuracy: 0.9151  
Epoch 3/5  
1875/1875 [=====] - 3s 1ms/step - loss: 0.2835 - accuracy: 0.9208  
Epoch 4/5  
1875/1875 [=====] - 3s 1ms/step - loss: 0.2731 - accuracy: 0.9234  
Epoch 5/5  
1875/1875 [=====] - 2s 1ms/step - loss: 0.2662 - accuracy: 0.9258
```

```
Out[64]: <keras.callbacks.History at 0x1ef7f981ac0>
```

```
In [65]: model.evaluate(x_test_flattened,y_test)
```

```
313/313 [=====] - 0s 1ms/step - loss: 0.2693 - accuracy: 0.9257
```

```
Out[65]: [0.26925358176231384, 0.925700088691711]
```

```
In [66]: y_predicted=model.predict(x_test_flattened)
```

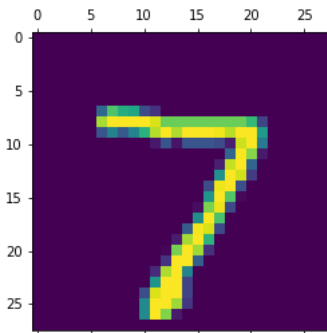
```
313/313 [=====] - 0s 934us/step
```

```
In [68]: y_predicted[0]
```

```
Out[68]: array([2.1681301e-02, 2.9891120e-07, 5.7053618e-02, 9.4716448e-01,  
                2.3006762e-03, 9.4873942e-02, 1.5762353e-06, 9.9981618e-01,  
                1.1235481e-01, 7.3142552e-01], dtype=float32)
```

```
In [69]: plt.matshow(x_test[0])
```

```
Out[69]: <matplotlib.image.AxesImage at 0x1ef008f75b0>
```



```
In [70]: y_test[0]
```

```
Out[70]: 7
```

```
In [ ]:
```