```python
import numpy as np
import pandas as pd
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import models, datasets, layers
import matplotlib.pyplot as plt
```

```python
(train_images,train_labels),(test_images,test_labels) = datasets.mnist.load_data()
```

```
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/mnist.npz
11493376/11490434 [==============================] - 0s 0us/step
11501568/11490434 [==============================] - 0s 0us/step
```

```python
print(train_images.shape)
print(train_labels.shape)
print(test_images.shape)
print(test_labels.shape)
```

```
(60000, 28, 28)
(60000,)
(10000, 28, 28)
(10000,)
```

```python
pd.DataFrame(train_images[0])
```

	0	1	2	3	4	5	6	7	8	9	...	18	19	20	21	22	23	24	25	26	27
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	...	175	26	166	255	247	127	0	0	0	0
6	0	0	0	0	0	0	0	0	30	36	...	225	172	253	242	195	64	0	0	0	0
7	0	0	0	0	0	0	0	49	238	253	...	93	82	82	56	39	0	0	0	0	0
8	0	0	0	0	0	0	0	18	219	253	...	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	80	156	...	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	14	...	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0

```python
train_images[0].shape
```

```
(28, 28)
```

| 15 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 150 | 27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

```python
train_images = train_images/255
test_images = test_images/255
```

| 17 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 253 | 249 | 64 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

```python
model = models.Sequential()
model.add(layers.Conv2D(28,kernel_size=(3,3),activation='relu',input_shape=(28,28,1)))
model.add(layers.MaxPooling2D(pool_size=(2,2)))
model.add(layers.Conv2D(28,kernel_size=(3,3),activation='relu'))
model.add(layers.MaxPooling2D(pool_size=(2,2)))
model.add(layers.Conv2D(28,kernel_size=(3,3),activation='relu'))
model.add(layers.MaxPooling2D(pool_size=(2,2)))
model.add(layers.Flatten())
model.add(layers.Dense(128,activation='relu'))
model.add(layers.Dense(64,activation='relu'))
model.add(layers.Dense(32,activation='relu'))
model.add(layers.Dense(10,activation='softmax'))


model.compile(optimizer='adam',loss='sparse_categorical_crossentropy',metrics=['accuracy'])
```

| 27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

```python
history = model.fit(train_images,train_labels,epochs=10,validation_data=(test_images,test_labels))
```

```
Epoch 1/10
1875/1875 [==============================] - 56s 29ms/step - loss: 0.3067 - accuracy: 0.9020 - va:
Epoch 2/10
1875/1875 [==============================] - 52s 28ms/step - loss: 0.1110 - accuracy: 0.9662 - va:
Epoch 3/10
1875/1875 [==============================] - 50s 26ms/step - loss: 0.0847 - accuracy: 0.9748 - va:
```

```
Epoch 4/10
1875/1875 [==============================] - 49s 26ms/step - loss: 0.0681 - accuracy: 0.9793 - va
Epoch 5/10
1875/1875 [==============================] - 49s 26ms/step - loss: 0.0581 - accuracy: 0.9818 - va
Epoch 6/10
1875/1875 [==============================] - 52s 28ms/step - loss: 0.0511 - accuracy: 0.9847 - va
Epoch 7/10
1875/1875 [==============================] - 50s 27ms/step - loss: 0.0439 - accuracy: 0.9862 - va
Epoch 8/10
1875/1875 [==============================] - 50s 27ms/step - loss: 0.0391 - accuracy: 0.9878 - va
Epoch 9/10
1875/1875 [==============================] - 50s 27ms/step - loss: 0.0354 - accuracy: 0.9892 - va
Epoch 10/10
1875/1875 [==============================] - 49s 26ms/step - loss: 0.0318 - accuracy: 0.9899 - va
```

model.summary()

```
Model: "sequential"
_____
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 26, 26, 28)        280

 max_pooling2d (MaxPooling2D  (None, 13, 13, 28)        0
 )

 conv2d_1 (Conv2D)           (None, 11, 11, 28)        7084

 max_pooling2d_1 (MaxPooling  (None, 5, 5, 28)          0
 2D)

 conv2d_2 (Conv2D)           (None, 3, 3, 28)          7084

 max_pooling2d_2 (MaxPooling  (None, 1, 1, 28)          0
 2D)

 flatten (Flatten)           (None, 28)                0

 dense (Dense)               (None, 128)               3712

 dense_1 (Dense)             (None, 64)                8256

 dense_2 (Dense)             (None, 32)                2080

 dense_3 (Dense)             (None, 10)                330

=================================================================
Total params: 28,826
Trainable params: 28,826
Non-trainable params: 0
_____
```

result = model.evaluate(train_images,train_labels)

```
1875/1875 [==============================] - 18s 10ms/step - loss: 0.0198 - accuracy: 0.9939
```

✓ 18s   completed at 8:21 AM   ● ✕