

```
import tensorflow as tf
mnist = tf.keras.datasets.mnist
```

```
(x_train, y_train), (x_test, y_test) = mnist.load_data()
x_train, x_test = x_train / 255.0, x_test / 255.0
```

```
x_train.shape
```

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

```
x_test.shape
```

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

```
model = tf.keras.models.Sequential([
    tf.keras.layers.Flatten(input_shape=(28, 28)),
    tf.keras.layers.Dense(256, activation=tf.nn.relu),
    tf.keras.layers.Dense(256, activation=tf.nn.relu),
    tf.keras.layers.Dense(10, activation=tf.nn.softmax)
])
model.compile(optimizer='adam',
              loss='sparse_categorical_crossentropy',
              metrics=['accuracy'])
```

```
model.summary()
```

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

Layer (type)	Output Shape	Param #
flatten_2 (Flatten)	(None, 784)	0
dense_5 (Dense)	(None, 256)	200960
dense_6 (Dense)	(None, 256)	65792
dense_7 (Dense)	(None, 10)	2570
Total params: 269,322		
Trainable params: 269,322		
Non-trainable params: 0		

```
784 * 512 + 512 * 512 + 512 * 10 + 10
```

```
407050
```

```
model.fit(x_train, y_train, epochs=5)
```

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

```

Epoch 1/5
60000/60000 [=====] - 12s 195us/sample - loss: 0.2012 - acc: 0.93
Epoch 2/5
60000/60000 [=====] - 12s 196us/sample - loss: 0.0853 - acc: 0.97
Epoch 3/5
60000/60000 [=====] - 11s 190us/sample - loss: 0.0569 - acc: 0.98
Epoch 4/5
60000/60000 [=====] - 12s 197us/sample - loss: 0.0449 - acc: 0.98
Epoch 5/5

```

#### ▼ test accuracy

```
model.evaluate(x_test, y_test)
```

```

↳ 10000/10000 [=====] - 1s 57us/sample - loss: 0.0781 - acc: 0.9787
[0.07812852754267806, 0.9787]

```

```
model.evaluate(x_test[:2], y_test[:2])
```

```

↳ 2/2 [=====] - 0s 936us/sample - loss: 5.2452e-06 - acc: 1.0000
[5.245192824077094e-06, 1.0]

```

더블클릭 또는 Enter 키를 눌러 수정

#### ▼ train accuracy

```
model.evaluate(x_train, y_train)
```

```

↳ 60000/60000 [=====] - 3s 53us/sample - loss: 0.0247 - acc: 0.9924
[0.024703782076669935, 0.9924333]

```

test a

#### ▼ *Real World Challenge: Large difference between training and testing set accuracy*

