EXP NO 2

```
import tensorflow as tf
import tensorflow_datasets
import matplotlib.pyplot as plt
import numpy as np
```

```
(X_train,y_train),(X_test,y_test)=tf.keras.datasets.mnist.load_data()
```

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

```
model = tf.keras.models.Sequential()
model.add(tf.keras.layers.Flatten(input_shape=(28,28)))
model.add(tf.keras.layers.Dense(128,activation="relu"))
model.add(tf.keras.layers.Dense(10,activation="softmax"))
model.compile(optimizer="adam",loss="sparse_categorical_crossentropy",metrics=['accuracy'])
model.fit(X_train,y_train, epochs=5)
```

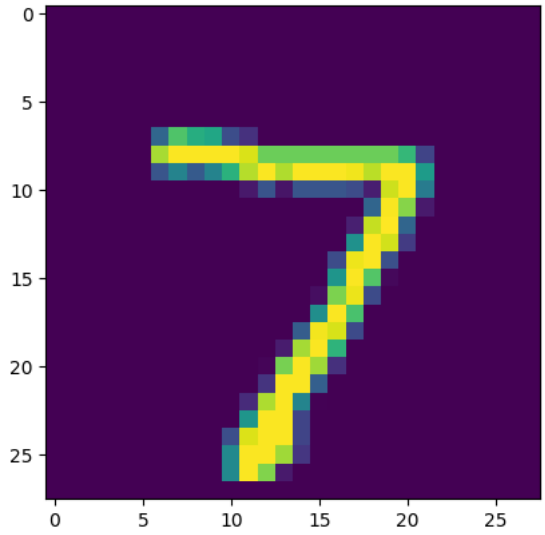
```
Epoch 1/5
1875/1875 [==============================] - 11s 5ms/step - loss: 2.4112 - accuracy: 0.8497
Epoch 2/5
1875/1875 [==============================] - 9s 5ms/step - loss: 0.3664 - accuracy: 0.9107
Epoch 3/5
1875/1875 [==============================] - 9s 5ms/step - loss: 0.2758 - accuracy: 0.9295
Epoch 4/5
1875/1875 [==============================] - 9s 5ms/step - loss: 0.2473 - accuracy: 0.9364
Epoch 5/5
1875/1875 [==============================] - 10s 6ms/step - loss: 0.2217 - accuracy: 0.9432
<keras.callbacks.History at 0x7bb7aead2740>
```

```
pred=model.predict(X_test)
```

```
313/313 [==============================] - 1s 2ms/step
```

```
plt.imshow(X_test[0])
```

```
<matplotlib.image.AxesImage at 0x7bb70e0736d0>
```




```
print(np.argmax(pred[0]))
```

```
7
```

✓ 0s completed at 3:54 PM