NAME : PAREENITA A.SHIRSATH PRN : 221101062 B.E.A.I.&.D.S. ROLL.NO : 57

**DEEP LEARNING EXPERIMENT NO : 3**

CODE:

import tensorflow as tf  
from tensorflow.keras.models import Sequential  
from tensorflow.keras.layers import Dense  
  
*# Load MNIST data (automatically downloaded if needed)*(x\_train, y\_train), (x\_test, y\_test) = tf.keras.datasets.mnist.load\_data()  
  
*# Preprocess: flatten 28x28 images to vectors of length 784, normalize pixel values to [0,1]*x\_train = x\_train.reshape(-1, 28\*28) / 255.0  
x\_test = x\_test.reshape(-1, 28\*28) / 255.0  
  
*# Build the DNN model with 2 hidden layers*model = Sequential([  
 Dense(128, activation='relu', input\_shape=(784,)), *# Hidden layer 1 with 128 neurons* Dense(64, activation='relu'), *# Hidden layer 2 with 64 neurons* Dense(10, activation='softmax') *# Output layer for 10 classes*])  
  
*# Compile the model with Adam optimizer and appropriate loss for multi-class classification*model.compile(  
 optimizer='adam',  
 loss='sparse\_categorical\_crossentropy',  
 metrics=['accuracy']  
)  
  
*# Train the model for 5 epochs with batch size 32 and 10% validation split*model.fit(  
 x\_train, y\_train,  
 epochs=5,  
 batch\_size=32,  
 validation\_split=0.1  
)  
  
*# Evaluate the model on test data*test\_loss, test\_acc = model.evaluate(x\_test, y\_test)  
print(f"\nTest accuracy: {test\_acc:.4f}")

OUTPUT:

