# Install TensorFlow !pip install tensorflow # Import libraries import tensorflow as tf # Load the MNIST dataset (X\_train, y\_train), (X\_test, y\_test) = tf.keras.datasets.mnist.load\_data() # Preprocess the data by reshaping and scaling X\_train = X\_train.reshape(-1, 28\*28) / 255.0 X\_test = X\_test.reshape(-1, 28\*28) / 255.0 # Convert the labels to one-hot encoding y\_train = tf.keras.utils.to\_categorical(y\_train) y\_test = tf.keras.utils.to\_categorical(y\_test)

In this code, we first install and import the TensorFlow library. Then, we use the load\_data function from the keras.datasets module to load the MNIST dataset. The dataset is returned as a tuple of training and test sets, which consist of images and labels. Next, we preprocess the data by reshaping the images from 2D arrays to 1D arrays and scaling them to the range [0, 1]. This is done using the reshape and /255.0 functions, respectively. Finally, we use the to\_categorical function to convert the labels to one-hot encoding, which is a common format for categorical data in machine learning. With the data preprocessed, we can now define and train an MLP model on the MNIST dataset.