PROGRAM:

Import numpy as np

Import tensorflow as tf

From tensorflow.keras.datasets import mnist

From tensorflow.keras.layers import Input,Dense

From tensorflow.keras.models import Sequential

From tensorflow.keras.callbacks import EarlyStopping #stop overtraining

Import matplotlib.pyplot as plt

(x\_train,y\_train),(x\_test,y\_test)=mnist.load\_data()

X\_train=x\_train.reshape(-1,784)

X\_test=x\_test.reshape(-1,784)

X\_train=x\_train.astype(np.float32)/255.0

X\_test=x\_test.astype(np.float32)/255.0

Y\_train=tf.keras.utils.to\_categorical(y\_train,num\_classes=10)

Y\_test=tf.keras.utils.to\_categorical(y\_test,num\_classes=10)

Model=Sequential([

Dense(50,input\_shape=(784,),activation=’sigmoid’),

Dense(10) #output layer

])

Model.compile(optimizer=tf.keras.optimizers.Adam(learning\_rate=0.001),

Loss=tf.keras.losses.CategoricalCrossentropy(from\_logits=True),

Metrics=[‘accuracy’])

Batch\_size=100

Max\_epochs=30

Early\_stopping=EarlyStopping(patience=2,restore\_best\_weights=True)

History = model.fit(x\_train, y\_train, batch\_size=batch\_size, epochs=max\_epochs, validation\_data=(x\_test, y\_test), callbacks=[early\_stopping])

Model.save(‘mnist\_model’)

Test\_loss, test\_accuracy = model.evaluate(x\_test, y\_test)

Test\_accuracy\_percentage=test\_accuracy \* 100

Print(test\_accuracy\_percentage)

Loaded\_model=tf.keras.models.load\_model(‘mnist\_model’)

#path

(\_, \_) ,(x\_test,y\_test)=mnist.load\_data()

Index=[3,6,9,2,4,5]

Index=input\_size=28\*28

Preprocessed\_image=x\_test[index].reshape(1,input\_size).astype(‘float32’)/255.0

Print(“preprocessed image shape”,preprocessed\_image.shape)

#load training model

Model = tf.keras.models.load\_model(‘mnist\_model’)

#make prediction

Predictions=model.predict(preprocessed\_image)

Predicted\_label=np.argmax(predictions)

Print(“predicted label:”,predicted\_label)

Import numpy as np

Import matplotlib.pyplot as plt

From tensorflow.keras.datasets import mnist

(\_,\_),(x\_test,y\_test)=mnist.load\_data()

Num\_samples=16

Plt.figure(figsize=(15,6))

For I in range(num\_samples):

Plt.subplot(1,num\_samples,i+1)

Plt.imshow(x\_test[i],cmap=’gray’)

Plt.title(f”Label:{y\_test[i]}”)

Plt.axis(‘off’)

Plt.show()

OUTPUT:

600/600 [==============================] – 18s 12ms/step – loss: 0.7134 – accuracy: 0.8462 – val\_loss: 0.3468 – val\_accuracy: 0.9112

Epoch 2/30

600/600 [==============================] – 6s 10ms/step – loss: 0.3029 – accuracy: 0.9186 – val\_loss: 0.2577 – val\_accuracy: 0.9268

Epoch 3/30

600/600 [==============================] – 7s 11ms/step – loss: 0.2398 – accuracy: 0.9333 – val\_loss: 0.2178 – val\_accuracy: 0.9352

Epoch 4/30

600/600 [==============================] – 5s 8ms/step – loss: 0.2051 – accuracy: 0.9427 – val\_loss: 0.1921 – val\_accuracy: 0.9431

Epoch 5/30

600/600 [==============================] – 4s 7ms/step – loss: 0.1813 – accuracy: 0.9491 – val\_loss: 0.1741 – val\_accuracy: 0.9494

Epoch 6/30

600/600 [==============================] – 5s 9ms/step – loss: 0.1630 – accuracy: 0.9535 – val\_loss: 0.1591 – val\_accuracy: 0.9524

Epoch 7/30

600/600 [==============================] – 7s 11ms/step – loss: 0.1481 – accuracy: 0.9573 – val\_loss: 0.1508 – val\_accuracy: 0.9553

Epoch 8/30

600/600 [==============================] – 12s 19ms/step – loss: 0.1359 – accuracy: 0.9610 – val\_loss: 0.1409 – val\_accuracy: 0.9583

Epoch 9/30

600/600 [==============================] – 8s 13ms/step – loss: 0.1254 – accuracy: 0.9643 – val\_loss: 0.1334 – val\_accuracy: 0.9603

Epoch 10/30

600/600 [==============================] – 8s 13ms/step – loss: 0.1162 – accuracy: 0.9670 – val\_loss: 0.1288 – val\_accuracy: 0.9614

Epoch 11/30

600/600 [==============================] – 8s 13ms/step – loss: 0.1086 – accuracy: 0.9690 – val\_loss: 0.1236 – val\_accuracy: 0.9628

Epoch 12/30

600/600 [==============================] – 8s 13ms/step – loss: 0.1018 – accuracy: 0.9714 – val\_loss: 0.1230 – val\_accuracy: 0.9632

Epoch 13/30

600/600 [==============================] – 9s 14ms/step – loss: 0.0959 – accuracy: 0.9728 – val\_loss: 0.1166 – val\_accuracy: 0.9659

Epoch 14/30

600/600 [==============================] – 9s 15ms/step – loss: 0.0899 – accuracy: 0.9749 – val\_loss: 0.1132 – val\_accuracy: 0.9660

Epoch 15/30

600/600 [==============================] – 8s 13ms/step – loss: 0.0852 – accuracy: 0.9760 – val\_loss: 0.1101 – val\_accuracy: 0.9659

Epoch 16/30

600/600 [==============================] – 10s 16ms/step – loss: 0.0802 – accuracy: 0.9779 – val\_loss: 0.1082 – val\_accuracy: 0.9682

Epoch 17/30

600/600 [==============================] – 8s 13ms/step – loss: 0.0762 – accuracy: 0.9789 – val\_loss: 0.1080 – val\_accuracy: 0.9672

Epoch 18/30

600/600 [==============================] – 7s 12ms/step – loss: 0.0724 – accuracy: 0.9798 – val\_loss: 0.1084 – val\_accuracy: 0.9677

Epoch 19/30

600/600 [==============================] – 8s 14ms/step – loss: 0.0686 – accuracy: 0.9812 – val\_loss: 0.1040 – val\_accuracy: 0.9681

Epoch 20/30

600/600 [==============================] – 8s 14ms/step – loss: 0.0655 – accuracy: 0.9823 – val\_loss: 0.1039 – val\_accuracy: 0.9681

Epoch 21/30

600/600 [==============================] – 9s 15ms/step – loss: 0.0623 – accuracy: 0.9830 – val\_loss: 0.1043 – val\_accuracy: 0.9679

Epoch 22/30

600/600 [==============================] – 8s 13ms/step – loss: 0.0594 – accuracy: 0.9840 – val\_loss: 0.1019 – val\_accuracy: 0.9684

Epoch 23/30

600/600 [==============================] – 9s 14ms/step – loss: 0.0568 – accuracy: 0.9848 – val\_loss: 0.1029 – val\_accuracy: 0.9697

Epoch 24/30

600/600 [==============================] – 8s 13ms/step – loss: 0.0542 – accuracy: 0.9856 – val\_loss: 0.0994 – val\_accuracy: 0.9698

Epoch 25/30

600/600 [==============================] – 8s 13ms/step – loss: 0.0519 – accuracy: 0.9864 – val\_loss: 0.1017 – val\_accuracy: 0.9686

Epoch 26/30

600/600 [==============================] – 8s 14ms/step – loss: 0.0491 – accuracy: 0.9874 – val\_loss: 0.0993 – val\_accuracy: 0.9698

Epoch 27/30

600/600 [==============================] – 8s 13ms/step – loss: 0.0469 – accuracy: 0.9881 – val\_loss: 0.0983 – val\_accuracy: 0.9702

Epoch 28/30

600/600 [==============================] – 8s 13ms/step – loss: 0.0451 – accuracy: 0.9889 – val\_loss: 0.0995 – val\_accuracy: 0.9708

Epoch 29/30

600/600 [==============================] – 8s 14ms/step – loss: 0.0432 – accuracy: 0.9890 – val\_loss: 0.0999 – val\_accuracy: 0.9695

313/313 [==============================] – 4s 9ms/step – loss: 0.0983 – accuracy: 0.9702

97.02000021934509

Preprocessed image shape (1, 784)

1/1 [==============================] – 0s 484ms/step

Predicted label: 4