Exercise 3

In the videos you looked at how you would improve Fashion MNIST using Convolutions. For your exercise see if you can improve MNIST to 99.8% accuracy or more using only a single convolutional layer and a single MaxPooling 2D. You should stop training once the accuracy goes above this amount. It should happen in less than 20 epochs, so it's ok to hard code the number of epochs for training, but your training must end once it hits the above metric. If it doesn't, then you'll need to redesign your layers.

I've started the code for you -- you need to finish it!

When 99.8% accuracy has been hit, you should print out the string "Reached 99.8% accuracy so cancelling training!"

```python
import tensorflow as tf

class myCallback(tf.keras.callbacks.Callback):

  def on_epoch_end(self, epoch, logs={}):
    if(logs.get('acc') > 0.998):
      print("\nReached 99.8% accuracy so cancelling training now")
      self.model.stop_training = True


mnist = tf.keras.datasets.mnist
(training_images, training_labels), (test_images, test_labels) = mnist.load_data()

# need to reshape the images to one tensor so 4D
training_images = training_images.reshape(60000, 28, 28, 1)

test_images = test_images.reshape(10000, 28, 28, 1)

# normalize to reduce variance caused by large differences between values

training_images = training_images/255.0

test_images = test_images/255.0

callbacks=myCallback()


model = tf.keras.models.Sequential([
    tf.keras.layers.Conv2D(64, (3, 3), activation='relu', input_shape=(28,28,1)),
    tf.keras.layers.MaxPooling2D(2, 2), # quarter the image to reduce dimensionality
    tf.layers.Flatten(),
    tf.layers.Dense(512, activation=tf.nn.relu),
    tf.layers.Dense(10, activation=tf.nn.softmax)
])

model.compile(optimizer="adam", loss="sparse_categorical_crossentropy", metrics=["ac

model.fit(training_images, training_labels, epochs=20, callbacks=[callbacks])

results = model.evaluate(test_images, test_labels)

print("The accuracy of convolutional neural network on 10K test images is %.2f perce
Epoch 1/20
60000/60000 [==============================] - 219s 4ms/sample - loss: 0.1169
Epoch 2/20
60000/60000 [==============================] - 220s 4ms/sample - loss: 0.0390
Epoch 3/20
60000/60000 [==============================] - 220s 4ms/sample - loss: 0.0218
Epoch 4/20
60000/60000 [==============================] - 218s 4ms/sample - loss: 0.0135
Epoch 5/20
60000/60000 [==============================] - 220s 4ms/sample - loss: 0.0094
Epoch 6/20
60000/60000 [==============================] - 221s 4ms/sample - loss: 0.0076
Epoch 7/20
59968/60000 [============================>.] - ETA: 0s - loss: 0.0068 - acc: 0
Reached 99.8% accuracy so cancelling training now
60000/60000 [==============================] - 222s 4ms/sample - loss: 0.0068
10000/10000 [==============================] - 6s 605us/sample - loss: 0.0496
The accuracy of convolutional neural network on 10K test images is 98.75 perce
```