Exercise: GPU performance for fashion mnist dataset

This notebook is derived from a tensorflow tutorial here: https://www.tensorflow.org/tutorials/keras/classification So please refer to it before starting work on this exercise

You need to write code wherever you see `your code goes here` comment. You are going to do image classification for fashion mnist dataset and then you will benchmark the performance of GPU vs CPU for 1 hidden layer and then for 5 hidden layers. You will eventually fill out this table with your performance benchmark numbers

Hidden Layer	CPU	GPU
1	?	?
5	?	?

In [ ]:
```
# TensorFlow and tf.keras
import tensorflow as tf
from tensorflow import keras

# Helper libraries
import numpy as np
import matplotlib.pyplot as plt

print(tf.__version__)
```

In [ ]:
```
fashion_mnist = keras.datasets.fashion_mnist

(train_images, train_labels), (test_images, test_labels) = fashion_mnist
```

In [ ]:
```
class_names = ['T-shirt/top', 'Trouser', 'Pullover', 'Dress', 'Coat',
               'Sandal', 'Shirt', 'Sneaker', 'Bag', 'Ankle boot']
```

In [ ]:
```
train_images.shape
```

In [ ]:
```
plt.imshow(train_images[0])
```

In [ ]:
```
train_labels[0]
```

In [ ]:
```
class_names[train_labels[0]]
```

In [ ]:
```
plt.figure(figsize=(3,3))
for i in range(5):
    plt.imshow(train_images[i])
    plt.xlabel(class_names[train_labels[i]])
    plt.show()
```

In [ ]:
```
train_images_scaled = train_images / 255.0
test_images_scaled = test_images / 255.0
```

In [ ]:
```
def get_model(hidden_layers=1):
    layers = []
    # Your code goes here----------START
    # Create Flatten input layers
    # Create hidden layers that are equal to hidden_layers argument in t
    # Create output
    # Your code goes here----------END
    model = keras.Sequential(layers)

    model.compile(optimizer='adam',
                  loss='sparse_categorical_crossentropy',
                  metrics=['accuracy'])

    return model
```

In [ ]:
```
model = get_model(1)
model.fit(train_images_scaled, train_labels, epochs=5)
```

In [ ]:
```
model.predict(test_images_scaled)[2]
```

In [ ]:
```
test_labels[2]
```

In [ ]:
```
tf.config.experimental.list_physical_devices()
```

5 Epochs performance comparison for 1 hidden layer

In [ ]:
```
%%timeit -n1 -r1
with tf.device('/CPU:0'):
    # your code goes here
```

In [ ]:
```
%%timeit -n1 -r1
with tf.device('/GPU:0'):
    # your code goes here
```

5 Epocs performance comparison with 5 hidden layers

In [ ]:
```
%%timeit -n1 -r1
with tf.device('/CPU:0'):
    # your code here
```

In [ ]:
```
%%timeit -n1 -r1
with tf.device('/GPU:0'):
    # your code here
```