Congratulations! You passed!

Grade received 100% **To pass** 80% or higher

Go to next item

Week 4 Quiz

Latest Submission Grade 100%

1. Which Devices support TensorFlow Lite for Inference? (Check all that apply) **1 / 1 point**

- [x] Coral
  - Correct
- [x] Sparkfun Edge
  - Correct
- [ ] RISC
- [x] Raspberry Pi
  - Correct

2. With a Raspberry Pi, how can you use TensorFlow? **1 / 1 point**

- ( ) Inference Only
- ( ) Training Only
- ( ) It doesn't work on Pi
- (•) Inference and Training
  - Correct

3. If you only want to do inference on a Pi, what's the best way? **1 / 1 point**

- (•) Install the standalone interpreter using pip
- ( ) Do nothing, the Pi base image has TensorFlow in it
- ( ) Install the full TensorFlow with Pip install
- ( ) Compile all of TensorFlow from Source and run it
  - Correct

4. When using ImageNet on a Raspberry Pi for Image Classification, how many classes are supported? **1 / 1 point**

- ( ) 500
- ( ) 800
- ( ) 100
- (•) 1000
  - Correct

5. How do you initialize the standalone interpreter in Python? **1 / 1 point**

- tf.lite.Interpreter(directory_of_saved_model)
- (●) tf.lite.Interpreter(directory_of_lite_Model)
- tf.lite.load(lite_model)
- tf.lite.load(saved_model)

Correct

6. How do you get the input tensors for a model with the standalone interpreter? **1 / 1 point**

- (●) Call get_input_details() after calling allocate_tensors() on the interpreter
- Call get_input_tensors() after calling allocate_tensors() on the interpreter
- Call get_input_tensors() after initializing the interpreter
- Call get_input_details() after initializing the interpreter

Correct

7. How do you perform inference using the interpreter? **1 / 1 point**

- (●) Set the Input tensor with the set_tensor command and then call invoke()
- Call invoke(), and pass it both the input and output tensors
- Just call invoke(), TensorFlow can do the rest
- Call invoke(), and pass it the input tensor

Correct

8. How do you read the results of inference using the interpreter? **1 / 1 point**

- Call invoke(), and the the output will be rendered automatically
- Call invoke(), pass it the input tensor, read the results
- (●) Call invoke(), and then call get_tensor() on the interpreter to read the output
- Call invoke(), pass it the input and output tensors, and then read the output tensor

Correct