✅ **Congratulations! You passed!**

Grade	Latest Submission	To pass 80% or
received 100%	Grade 100%	higher

Go to next item

---

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

   ☑ Sparkfun Edge

   ✓ Correct

   ☑ Raspberry Pi

   ✓ Correct

   ☐ RISC

   ☑ Coral

   ✓ Correct

---

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

   ⦿ Inference and Training

   ○ Inference Only

   ○ Training Only

   ○ It doesn't work on Pi

   ✓ 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**

   ○ 100

   ⦿ 1000

   ○ 800

   ○ 500

   ✓ Correct

---

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

   ○ tf.lite.load(saved_model)

   ○ tf.lite.load(lite_model)

   ○ tf.lite.Interpreter(directory_of_saved_model)

   ⦿ tf.lite.Interpreter(directory_of_lite_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 initializing the interpreter

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

✓ **Correct**

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

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

✓ **Correct**

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

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

✓ **Correct**