1. What HTML5 tag is used to show the contents of a webcam? 0 / 1 point

-
- <webcam>
- <video>
- <pre>

! **Incorrect**

2. If I initialize a webcam object like this:

1 / 1 point

```
1  const webcam = new Webcam(document.getElementById('wc'));
```

Which code will then start the webcam feed to render in the page?

- ```
  1  async function init(){await webcam.go();}async function init(){await webcam.go
     ();}
  ```

- ```
  1  async function init(){await webcam.start();}
  ```

- ```
  1  async function init(){await webcam.setup();}
  ```

- ```
  1  async function init(){await webcam.initialize();}
  ```

✓ **Correct**

3. If I want to create a model that uses transfer learning, with everything in mobilenet up to layer 'foo', and my layers afterwards, how do I do it? Assume this code was used to find layer 'foo'

1 / 1 point

```
1  const layer = mobilenet.getLayer('foo');
```

- ```
  1  return tf.model({inputs: mobilenet, outputs: layer});
  ```
- ```
  1  return tf.model({inputs: mobilenet.inputs, outputs: layer.outputs});
  ```
- ```
  1  return tf.model({inputs: mobilenet.input, outputs: layer.outputs});
  ```
- ```
  1  return tf.model({inputs: mobilenet.inputs, outputs: layer.output});
  ```

✓ **Correct**

4. If I am transfer learning from a mobilenet, and I want to use my own dense layers after the mobilenet ones, what is the correct syntax to use at <INSERT CODE HERE>

1 / 1 point

```
model = tf.sequential({
    layers: [
      tf.layers.flatten(<INSERT CODE HERE>),
      tf.layers.dense({ units: 100, activation: 'relu'}),
      tf.layers.dense({ units: 3, activation: 'softmax'})
    ]
  });
```

- (●) `{inputShape: mobilenet.outputs[0].shape.slice(1)}`
- ( ) `{inputShape: mobilenet.outputs[0].slice(1)}`
- ( ) `{inputShape: mobilenet.outputs[1].slice(0)}`
- ( ) `{inputShape: mobilenet.outputs[1].shape.slice(0)}`

✓ **Correct**

5. If I am using a mobilenet with my own DNN for transfer learning in TensorFLow.js, how do I get a prediction for an image? 1 / 1 point

- ○ Get a set of prediction embeddings from your model and pass them to mobilenet
- ◉ Get a set of prediction embeddings from mobilenet and pass them to your model
- ○ Just pass the prediction to mobilenet, because you've already added your layers to it
- ○ Just pass the prediction to your own model, it already includes the mobilenet layers

✓ **Correct**

6. If you have a set of predictions returned from model.predict(something) and you want to take the one with the largest probability, how do you do it? 1 / 1 point

- ○ predictions[0] contains the one with the largest probability
- ◉ predictions.as1D().argMax(), then look at the 0th element
- ○ predictions.argMax() then look at the 0th element
- ○ predictions.sort() then look at the 0th element

✓ **Correct**

7. If you already have a function called predict() in a class called 'foo' which captures a frame from the webcam and predicts it, what's the best way to call it, particularly if you plan to do continuous predictions?

1 / 1 point

- ○
```
1 foo.predict(); tf.tidy();
```
- ○
```
1 foo.predict(tf.tidy());
```
- ○
```
1 tf.tidy(foo.predict());
```
- ◉
```
1 tf.tidy(() => foo.predict());
```

✓ **Correct**

8. Why is transfer learning a huge advantage, particularly when training in the browser?

1 / 1 point

- ◉ It allows you to use already-learned convolutions for distinguishing features, saving training time
- ○ It allows you to use already-learned convolutions for distinguishing features, saving space
- ○ It lets you skip training altogether
- ○ It gives you a smaller model

✓ **Correct**