Day 2-1

Assignment:

Lab 02, Due Today

Homework 03, Due Tuesday (once written)

Today's Topics:

Analog In

PWM

Reading a Switch from a web page

- JSFIDDLE

03-2 IoT

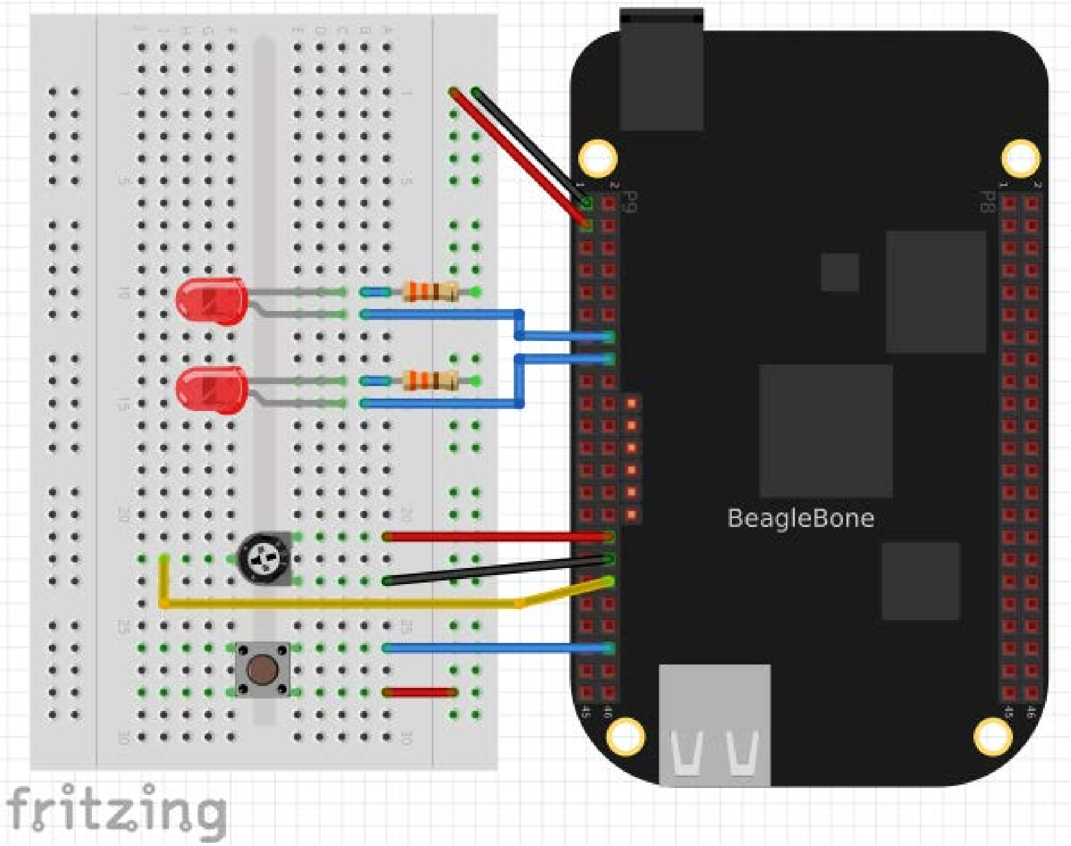
CONNECTING TO THE WEB

Connecting physical to the web

Now that

- The Bone is on the network and
- You know how to read switches and blink LEDs

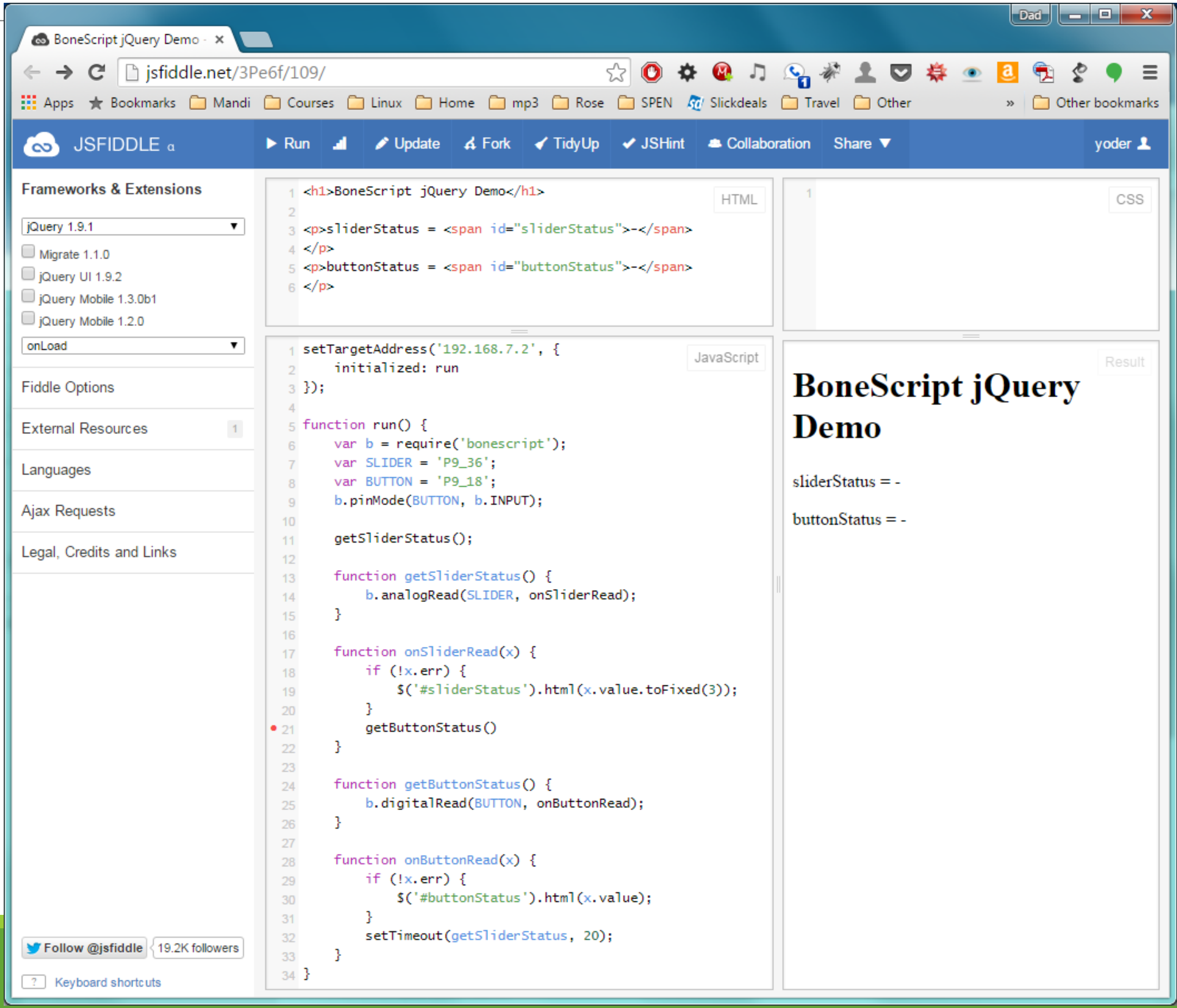
It’s easy to have a web page read your Bone



jsfiddle

Jsfiddle is a site for fiddling with your JavaScript

For example: http://jsfiddle.net/3Pe6f/109/



jQuery 1.9.1

□ Migrate 1.1.0
□ jQuery UI 1.9.2
□ jQuery Mobile 1.3.0b1
□ jQuery Mobile 1.2.0

onLoad

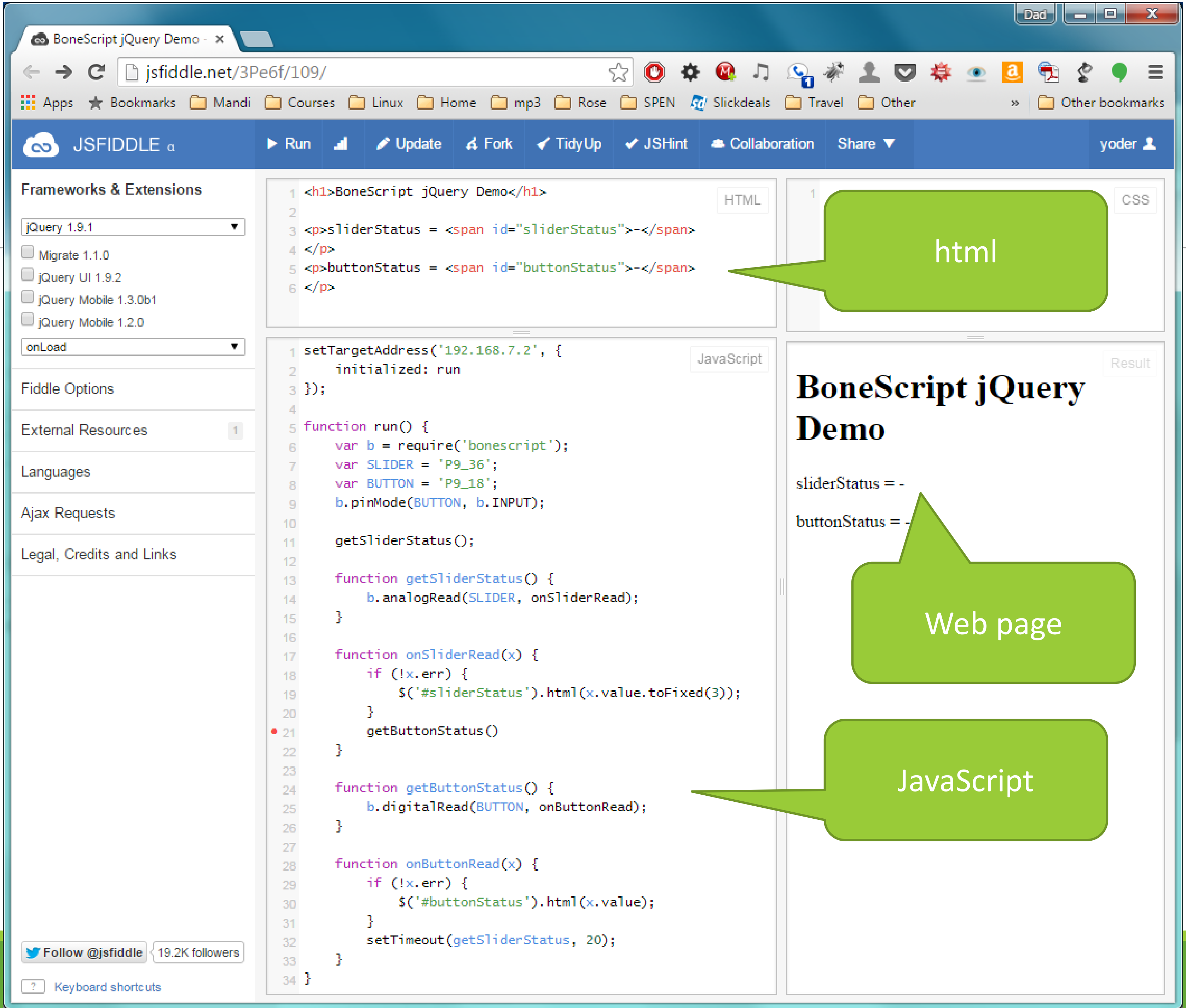
```html
<h1>BoneScript jQuery Demo</h1>

<p>sliderStatus = <span id="sliderStatus">-</span>
</p>
<p>buttonStatus = <span id="buttonStatus">-</span>
</p>
```

```javascript
setTargetAddress('192.168.7.2', {
    initialized: run
});

function run() {
    var b = require('bonescript');
    var SLIDER = 'P9_36';
    var BUTTON = 'P9_18';
    b.pinMode(BUTTON, b.INPUT);

    getSliderStatus();

    function getSliderStatus() {
        b.analogRead(SLIDER, onSliderRead);
    }

    function onSliderRead(x) {
        if (!x.err) {
            $('#sliderStatus').html(x.value.toFixed(3));
        }
        getButtonStatus()
    }

    function getButtonStatus() {
        b.digitalRead(BUTTON, onButtonRead);
    }

    function onButtonRead(x) {
        if (!x.err) {
            $('#buttonStatus').html(x.value);
        }
        setTimeout(getSliderStatus, 20);
    }
}
```

BoneScript jQuery Demo

sliderStatus = -

buttonStatus = -



Frameworks & Extensions

jQuery 1.9.1

□ Migrate 1.1.0
□ jQuery UI 1.9.2
□ jQuery Mobile 1.3.0b1
□ jQuery Mobile 1.2.0

onLoad

```html
<h1>BoneScript jQuery Demo</h1>

<p>sliderStatus = <span id="sliderStatus">-</span>
</p>
<p>buttonStatus = <span id="buttonStatus">-</span>
</p>
```

```javascript
setTargetAddress('192.168.7.2', {
    initialized: run
});

function run() {
    var b = require('bonescript');
    var SLIDER = 'P9_36';
    var BUTTON = 'P9_18';
    b.pinMode(BUTTON, b.INPUT);

    getSliderStatus();

    function getSliderStatus() {
        b.analogRead(SLIDER, onSliderRead);
    }

    function onSliderRead(x) {
        if (!x.err) {
            $('#sliderStatus').html(x.value.toFixed(3));
        }
        getButtonStatus()
    }

    function getButtonStatus() {
        b.digitalRead(BUTTON, onButtonRead);
    }

    function onButtonRead(x) {
        if (!x.err) {
            $('#buttonStatus').html(x.value);
        }
        setTimeout(getSliderStatus, 20);
    }
}
```

BoneScript jQuery Demo

sliderStatus = -

buttonStatus = -

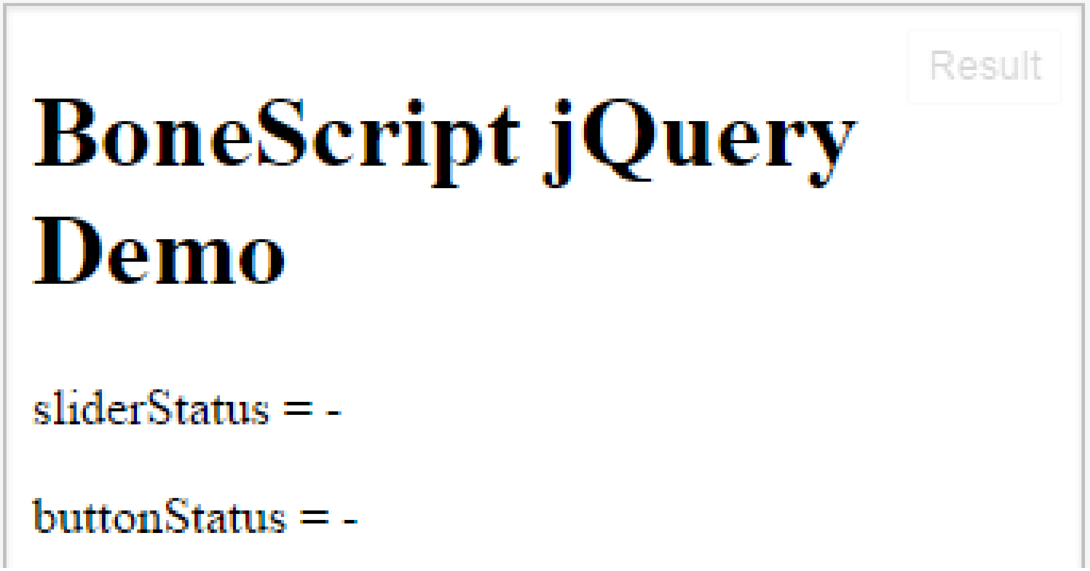
html

Web page

JavaScript

html

```
<h1>BoneScript jQuery Demo</h1>
<p>
    sliderStatus = <span id="sliderStatus">-</span>
</p>
<p>
    buttonStatus = <span id="buttonStatus">-</span>
</p>
```



BoneScript jQuery Demo

sliderStatus = -

buttonStatus = -

js

```
setTargetAddress('192.168.7.2', {
    initialized: run
});
```

```
function run() {
 var b = require('bonescript');
 var SLIDER = 'P9_36';
 var BUTTON = 'P9_18';
 b.pinMode(BUTTON, b.INPUT);
```

js

```
sliderStatus = <span id="sliderStatus">-</span>
buttonStatus = <span id="buttonStatus">-</span>
```

```
getSliderStatus();

function getSliderStatus() {
   b.analogRead(SLIDER, onSliderRead);
}
function onSliderRead(x) {
   if (!x.err) {
       $('#sliderStatus').html(x.value);
   }
   getButtonStatus()
}
```

```
 function getButtonStatus() {
    b.digitalRead(BUTTON, onButtonRead);
 }

 function onButtonRead(x) {
    if (!x.err) {
        $('#buttonStatus').html(x.value);
    }
    setTimeout(getSliderStatus, 200);
 }
} // End of run
```

Hit Run

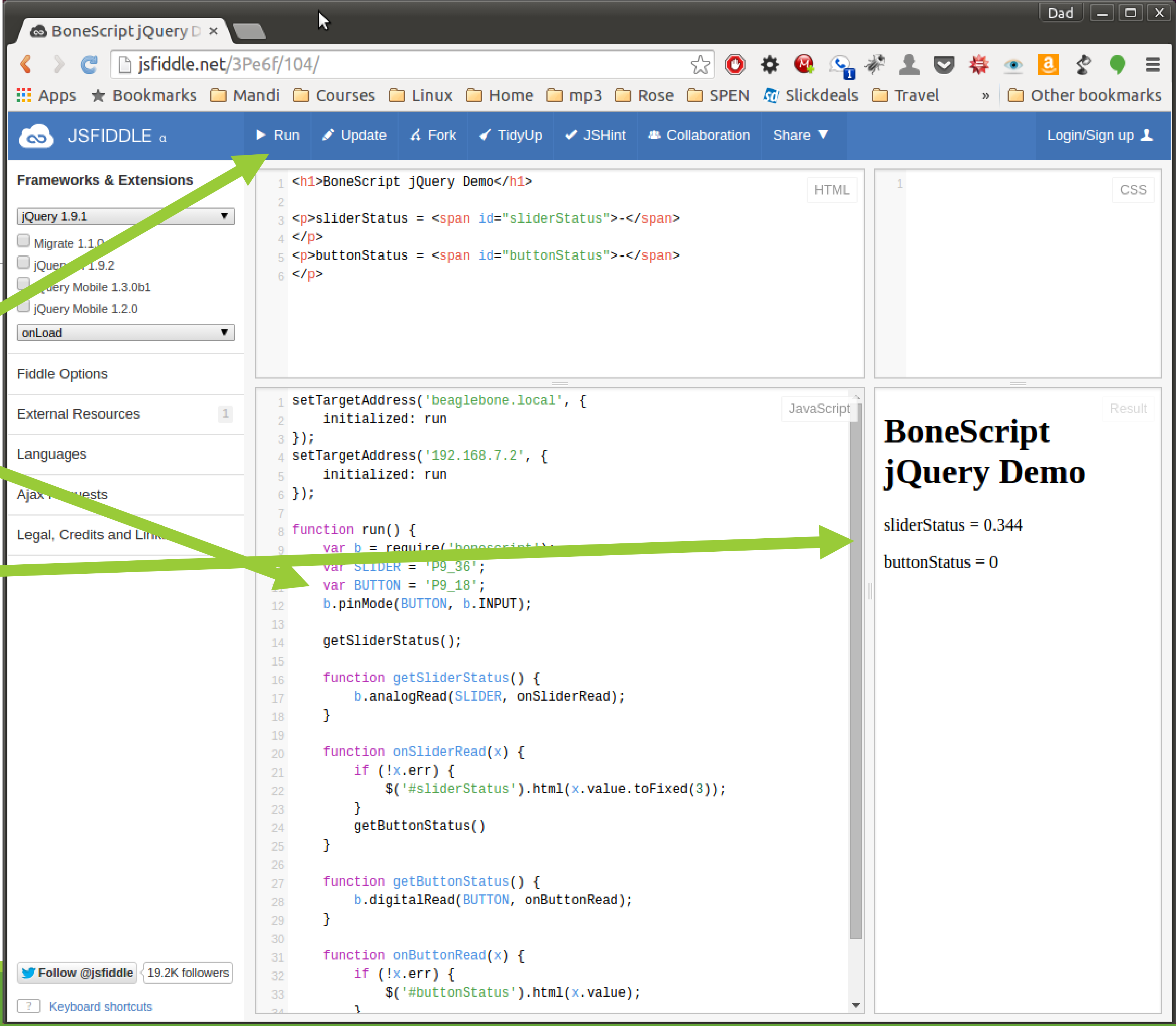
Change

- SLIDER and
- BUTTON

Then hit Run

- sliderStatus and
- buttonStatus

…will update



jQuery 1.9.1 ▼

☐ Migrate 1.1.0

☐ jQuery UI 1.9.2

☐ jQuery Mobile 1.3.0b1

☐ jQuery Mobile 1.2.0

onLoad ▼

```html
<h1>BoneScript jQuery Demo</h1>

<p>sliderStatus = <span id="sliderStatus">-</span>
</p>
<p>buttonStatus = <span id="buttonStatus">-</span>
</p>
```

```javascript
setTargetAddress('beaglebone.local', {
    initialized: run
});
setTargetAddress('192.168.7.2', {
    initialized: run
});

function run() {
    var b = require('bonescript');
    var SLIDER = 'P9_36';
    var BUTTON = 'P9_18';
    b.pinMode(BUTTON, b.INPUT);

    getSliderStatus();

    function getSliderStatus() {
        b.analogRead(SLIDER, onSliderRead);
    }

    function onSliderRead(x) {
        if (!x.err) {
            $('#sliderStatus').html(x.value.toFixed(3));
        }
        getButtonStatus()
    }

    function getButtonStatus() {
        b.digitalRead(BUTTON, onButtonRead);
    }

    function onButtonRead(x) {
        if (!x.err) {
            $('#buttonStatus').html(x.value);
        }
```

BoneScript jQuery Demo

sliderStatus = 0.344

buttonStatus = 0

Move to Bone

You have to have an Internet connection to use JSFIDDLE

You can move your code to the Bone so the connection isn’t needed.

Add the following to your html:

```
<h1>BoneScript jQuery Demo</h1>
<p>
    sliderStatus = <span id="sliderStatus">-</span>
</p>
<p>
    buttonStatus = <span id="buttonStatus">-</span>
</p>
```

Put in jsfiddleDemo.html

```
<html>
  <head>
    <title>BoneScript jsfiddle Demo</title>
    <script src="/static/jquery.js"></script>
    <script src="/static/bonescript.js"></script>
    <script src="jsfiddleDemo.js"></script>
  </head>
  <body>
   <h1>BoneScript jQuery Demo</h1>
   <p>sliderStatus = <span id="sliderStatus">-</span></p>
   <p>buttonStatus = <span id="buttonStatus">-</span></p>
  </body>
</html>
```

These files are already on the Bone

Put js in jsfiddleDemo.js

Copy to Bone

Put **`jsfiddleDemo.js`** and **`jsfiddleDemo.html`** in **`/var/lib/cloud9`**

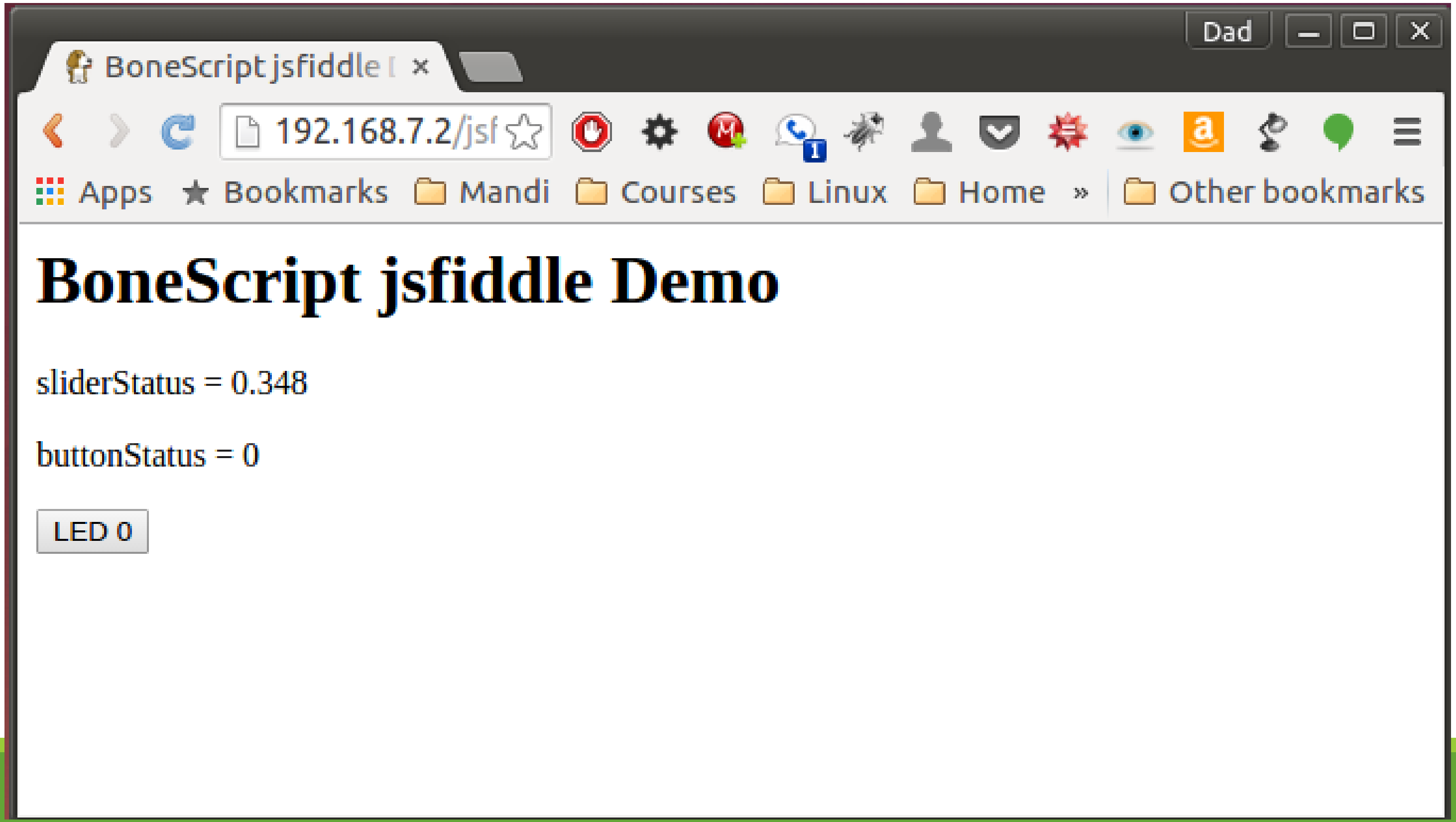
Browse to

http://192.168.7.2/jsfiddleDemo.html

Add a button

Optional: Add a button to the web page that controls an LED

```
<button id="led0" onClick='led(0)'/>LED 0</button>
```



BoneScript jsfiddle Demo

sliderStatus = 0.348

buttonStatus = 0