

Mario Franco-Munoz
CS290-400
Activity: Practice

This week's material was quite the challenge when applying this material to my CS340 project as well as while trying to get some of the example pieces to work.

Problem #1: could not get static filepath to work.

To begin, for some reason I ran into issues with the declaration of setting the static file path via express. When trying to emulate example code I just could not get it to work until after a few unsuccessful google searches I found a way to make it work for me:

```
express.static('http://localhost:3000/');  
app.use(express.static('public'));
```

Problem #2: Multiple post handlers for one page causing problems.

Perhaps for an experienced web developer this one is probably trivial, but when messing around with the POST handler, I was stuck for a bit in trying to figure out how to have multiple POST handlers on one page that had different ajax calls. Adding to just one list as in the example is fine, but what about if we have multiple lists? Well I figured this one out before too long by assigning different POST "addresses" in the form's action call.

So one action would be:

```
router.post('/thing1', function(req, res) etc etc
```

And another would be:

```
router.post('/thing2', function(req, res) etc etc
```

Problem #3: Updating the context in the get/ request via a post submission form.

Say I wanted to add something to context via a submitting a post form submission. Because I already had a context in the get/ when reloading the page I would get an error saying something along the lines of "Error('Can't set headers after they are sent.');" So take the to-do list for example, say I wanted to update {{name}} via a post submission. Nope, you get that error. I tried to "recreate" the context fields inside the POST handler, and this would get the page to load with the correct information but then would crash immediately after the last element was loaded in and give me the same error. I never figured this one out. Only way I was able to work around this problem was to load the updated context in a new page via a POST handler.

Summary:

All in all I now feel better about making requests that get and display data (such as in the "Express and HTTP" activity) now what I need to improve upon is updating data via POST request, and displaying that data on the same page without triggering a reload of the page. (For example a dynamic drop down). Needless to say I do not have all of this stuff memorized yet, and I certainly do need to refer to my notes when making requests.