W4D1 Homework

No starter code for today's homework. For cookies you can see an example at: https://github.com/expressjs/express/blob/master/examples/cookies/index.js For sessions see: https://www.npmjs.com/package/express-session#example

Be sure to use the following commands when creating an express project that uses cookies (question1):

```
$ npm init
$ npm install express -save
$ npm install cookie-parser -save
```

And use the following commands when setting up a project with sessions (questions 2 through 4):

```
$ npm init
$ npm install express -save
$ npm install express-session -save
```

Assignments:

1. Make a page with an "Add Cookie" form that has an input field for a key and an input field for a value. When you POST the form, the server should get the input and add a cookie with the given key and value to the cookies.

   To see if it works also add a table with all the current cookies at the bottom of the page.

2. Copy your W3D5 question4 solution for this exercise. Update the code to make the Post/Redirect/Get work properly by putting the name and age in the session instead of on the redirect URL

   Notice that once you've put a key/value pair into the session they stay in the session. Because this might not be what you'd want there the concept of 'flash' session variables exists, these are values that are removed after they're used.

   Although it's not required for this course, you can read more about the flash middleware (which uses the session and automatically removes) at: https://www.npmjs.com/package/flash

   You've correctly solved this exercise as long as the values are no longer passed on the URL.

3. Today's q3_4 builds on q3_4 from W3D6. Using the product template from yesterday make 3 different product pages. The page should have a form that POSTs the name and price to /addToCart. In the handler for this request store the product in the 'cart' object in the session.

   If you add a product a second time you should update the quantity and price of the item already in the cart. At the end the controller should give a redirect to /cart so that the user can see the

updated shopping cart contents.

You are free to choose what the 3 products are. I recommend re-using the q3_4 code from the W3D6 for both this exercise and the next.

4. Update the shoppingcart page to retrieve its data from the session (data added by in the previous exercise). Feel free to built on the code from W3D6.

The main difference from W3D6 is the fact that the session has a cart object (with key/value pairs), while the cart in W3D6 had an array. Remember, you can iterate over all the keys in an object using **`for (key in obj) { … }`** .