W3D5 Homework

There are 4 assignments, I recommend making a w3d5 directory, and making subdirectories called q1, q2, q3, and q4. The code below is for a 'Hello Person' Express application. You can run it by, copying the code into an **index.js** file and placing it into your **q1** directory. Next execute the following commands inside your q1 directory, and then open http://localhost:3000 to see the result.

```
$ npm init
$ npm install express -save
$ node index.js
```

```
const express = require('express');
const app = express();

app.get('/', (req, res) => {
  let name = req.query.name;
  if (!name) {
    name = "person";
  }
  res.send(`Welcome ${name}`);
});

app.listen(3000);
```

Assignments:

1. Modify the provided code so that it looks for both a name and an age query parameter, and sends those back as part of the response.
2. In your q2 directory be sure to start with npm init and install express. Your index.js application for this exercise should return a form similar to what is shown below for GET requests to '/'.

The form should submit with method POST to "/result", and then output a simple string similar to the output of question1. Hint: you'll need to use the express.urlencoded middleware so that the post parameters are parsed on req.body.

Do not use POST/Redirect/GET in this exercise). Try and see what happens when you bookmark the result page (and then use the bookmark).

3. In your q3 directory npm init and install express. Update the form from question 2 to be fully correct HTML5 (include all the tags). Next add a dynamically generated html tag, the page should have a link tag to **day.css** between 6am and 6pm, and to **night.css** between 6pm and 6am. You can get the current hour with:

```
const date = new Date();
const hour = date_ob.getHours();
```

The day.css file should specify a lightblue background and black text, while night.css should specify a black background with white text. These files should be served from the file-system as a static resource. Create a directory '**css**' inside your q2 that contains the two files and then:

```
app.use('/css', express.static(path.join(__dirname, 'css')));
```

4. Remember to npm init and install express in your q4 directory. Copy index.js from q3, and update the code in the route for POST to /result to redirect to /output.

   You'll create a new route for GET /output that contains the same logic used in Q1 to get the name and age from the URL.

   Notice that if you run it now the name and age are unknown. This is because they are not present on the URL (where we are picking them up). Update the redirect to put them on the URL, in essence it becomes a redirect to: /output?name=${name}&age=${age}

   Run it again to check that everything is working.

   Please note that all of this is a bit silly, under normal circumstances the name and age would either be stored in the session (which we'll cover W4D1) or in a database on the POST request, and then after the redirect the GET would retrieve them from the session / database.