CSc 337

LECTURE 15: WRITING YOUR OWN WEB SERVICE

Basic web service

```
// CSC 337 hello world server

const express = require("express");
const app = express();

app.use(express.static('public'));

app.get('/', function (req, res) {
     res.header("Access-Control-Allow-Origin", "*");
    res.send('Hello World!');
})

app.listen(3000);
```

Get Query Parameters in Express

Query parameters are saved in **`req.query`**

```
app.get('/', function (req, res) {
    res.header("Access-Control-Allow-Origin", "*");
   const queryParams = req.query;
   console.log(queryParams);
   const name = req.query.name;
   res.send('Hello' + name);
})
```

Exercise

Write a web service that takes an exponent and base as parameters and outputs the based raised to the exponent

Generating JSON

Create a JSON object:

- `var data = {};`

Add any data you want in your JSON to this:

- `data["name"] = "Merlin";`

Once you have put together the data you want:

- `var to_send = JSON.stringify(data);`

Exercise

Build a web service that takes two numbers as parameters and outputs JSON. For example, if the service were passed `2` for `num1` and `3` for `num2`:

```
{
   "plus" : 5,
   "minus": 1,
   "times": 6,
   "divide": 1.5
}
```

Reading from a file

```
let file = fs.readFileSync(file_name, 'utf8');
```

You can read from a file with the above code.

Returns the contents of the file as a string.

To use you will need to install the fs module

- `npm install fs`

Exercise

Read data from a file called `books.txt`

Make sure `books.txt` is in the same folder as your web service

Output JSON that that contains a list of all books in the file. Each list item should contain the book's name, author, category, year and price as individual items.

Exercise - part 2

Add a parameter to your service so that when the user supplies the parameter `books` with a value of a category name your service will only output books in that category. If the user doesn't supply a parameter your service should produce all books.

```
http://localhost:3000?books=computer
```

Exercise - part 3

If there are no books that are in the category that the user supplies have your service return a 410 status and a message about the category not being found.

Set the status with the following code:

```
res.status(410);
```

Reading all files in a directory

```
let files = fs.readDirSync(directory_name);
```

You can read the names of all of the files in a directory with the above code.

Returns the names of the files in a directory as a list of strings.

Pass in `"."` to get all of the files in the current directory.

To use you will need to install the fs module

- `npm install fs`