CS 465P/565 – Full Stack Web Development
Homework #3

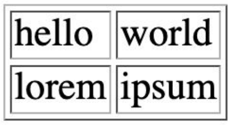
Topics for Assignment #3 - Node, Express, and Frameworks

- ❏ In this assignment, you will practice more JavaScript, this time using Node, Express, and other frameworks.
  - ❏ Use Node.js only for exercises 01 and 02
  - ❏ Use Express.js for exercises 03 and 04
  - ❏ Use the Pug Templating Engine for exercise 05
- ❏ Make sure you pull the latest changes from the webdev-exercises repository: https://github.com/caterinasworld/webdev-exercises
- ❏ **Due Date: Sunday, Nov. 15 at 11:59pm**
- ❏ **Submission Guidelines:** There is nothing to submit. You should be submitting all your code to the same private repository you used for Homework #0 - Setup.
  - ❏ Make sure you include a **Readme file** with directions for how to run all exercises and/or tutorials.
  - ❏ You should also include a **package.json**, **package-lock.json**, and a **.gitignore**. You will generate many **node_modules/** that should never be pushed to your remote repo.
  - ❏ Note: there is a **.gitignore** file included with the starter files.

- ❏ **Exercise 01 - Node Routes Exercise**
  - ❏ Use file **01-node-routes.js** and Node.js only in this exercise
  - ❏ Create each of the routes specified in the file:
    - ❏ / → status code 200 with a 'welcome' message
    - ❏ /redirect → redirect the request to '/redirected' by using 302 as the status code
    - ❏ /cache → return 'this resource was cached' in plain text and set the cache max age to a day

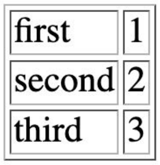
        - ❑ /cookie → return 'cookies… yummm' in plain text and set 'hello=world' as a cookie
        - ❑ /check-cookies → should return 'yes' / 'no' in plain text depending on whether the browser has the 'hello' cookie
        - ❑ 404 for all other routes

- ❑ **Exercise 02 - Node URL Exercise**
    - ❑ Use file **02-node-url.js** and Node.js only in this exercise
    - ❑ In this exercise, you will parse the URL for query parameters and return a table with the information given
        - ❑ http://localhost:8080/attributes?hello=world&lorem=ipsum



hello	world
lorem	ipsum

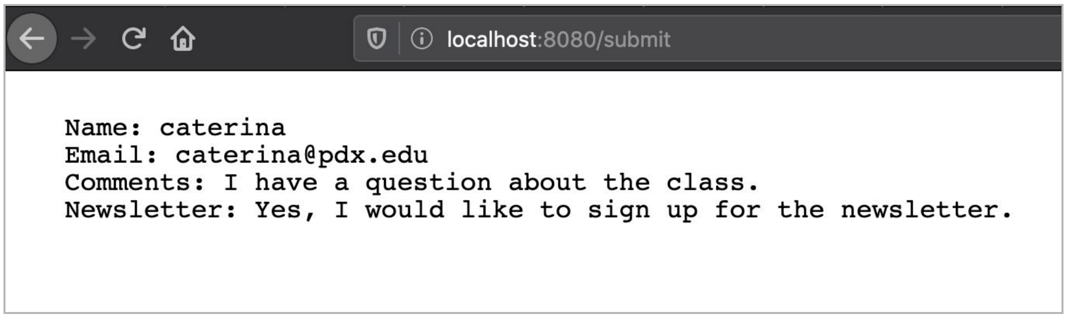
        - ❑ http://localhost:8080/attributes?first=1&second=2&third=3



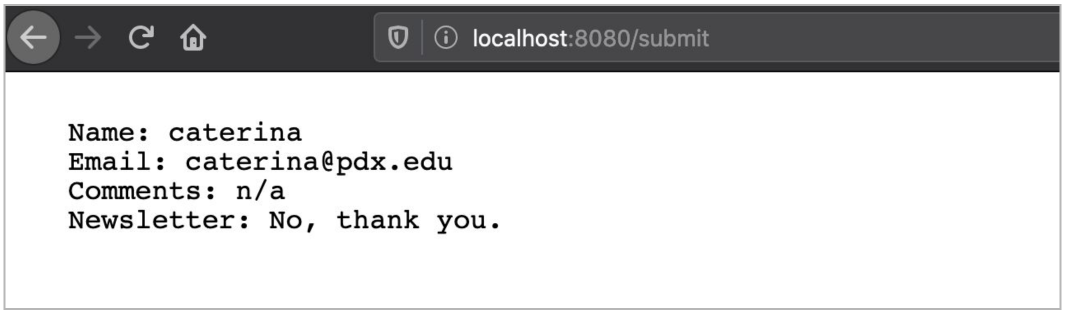
first	1
second	2
third	3

- ❑ **Exercise 03 - Form Exercise**
    - ❑ Use files **03-form.js, index.html** and Express.js
    - ❑ In this version of the form exercise, you will create the necessary client and server files to submit a form.
    - ❑ In your **index.html**, you will need to reuse the form or the Bootstrap form you created in a previous assignment. You will create an Express.js server and submit user input to the server (name, email, comments, and whether the newsletter checkbox is checked).
        - ❑ You can assign an action and a method to the form element.
            - ❑ For example: <**form** action="/submit" method="post">

        - ❑ Middleware**:** Depending on how you implement this, you may need to use middleware for serving static pages and for parsing incoming request body data.

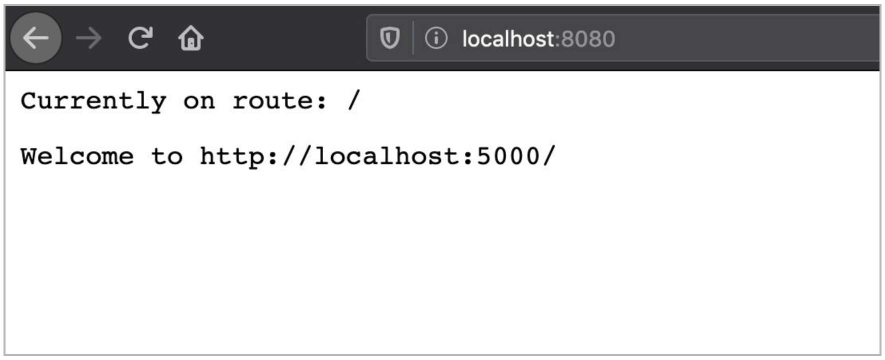


```
Name: caterina
Email: caterina@pdx.edu
Comments: I have a question about the class.
Newsletter: Yes, I would like to sign up for the newsletter.
```



```
Name: caterina
Email: caterina@pdx.edu
Comments: n/a
Newsletter: No, thank you.
```

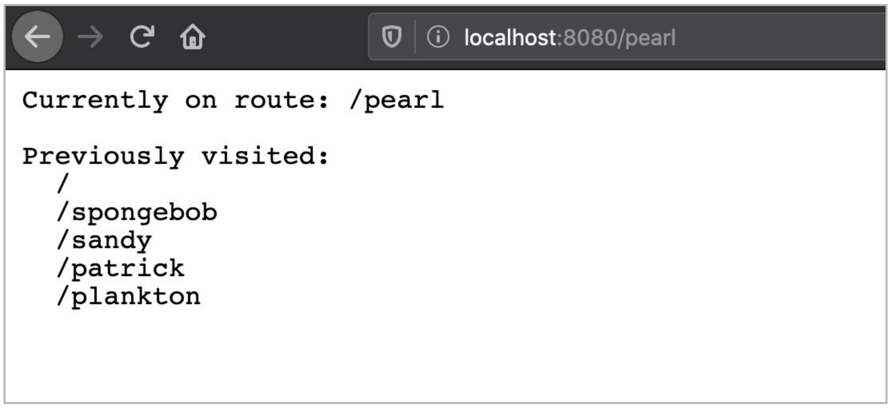
- ❑ **Exercise 04 - Express Sessions**
    - ❑ Use file **04-sessions.js, express-sessions,** and Express.js in this exercise
    - ❑ In this exercise, you will keep track of all the pages a user visits, and send those to the client via an HTTP response.
        - ❑ Start by sending back to the client the current route.
        - ❑ If the user visits the domain name for the first time, send back “Welcome to http://locahost:5000/”



```
Currently on route: /

Welcome to http://localhost:5000/
```

        - ❑ If the user visits multiple paths within your domain name (for example /spongebob, /sandy, /patrick, plankton, and /pearl), send back all the previous paths visited.



```
Currently on route: /pearl

Previously visited:
  /
  /spongebob
  /sandy
  /patrick
  /plankton
```

- ❑ **Exercise 05 - Templates**
    - ❑ Use file **05-templates.js,** Express.js, and the Pug Templating Engine in this exercise
    - ❑ Recreate the REST Countries exercise from the previous assignment, this time using one or more of the frameworks above.
        - ❑ Use the REST Countries API (https://restcountries.eu/) to make a GET request for all the countries in the world.
        - ❑ Create three routes:
            - ❑ **/main** - return the list of all the countries in the world and their capitals.
            - ❑ **/populous** - return the list of countries with a population of least 20 million, starting with the most populous.
            - ❑ **/regions** - with the number of countries in each region of the world, according to the API data.

- ❑ **Exercises 06 - Tutorials**
    - ❑ Complete one of the tutorials below. Create a folder called **06 - <tutorial name>** (i.e. 06-react if you are going through the React tutorial), and upload all files from your finished code (minus the node_modules/). Make

sure you also include instructions for how to run this tutorial in your Readme file.

- ❑ **React - Reactjs.org:**
  - ❑ Intro to React: https://reactjs.org/tutorial/tutorial.html
    - ❑ Tic Tac Toe Game: https://codepen.io/gaearon/pen/gWWZgR
- ❑ **Vue - Vue Vixens**
  - ❑ Mini Workshop 1 - Build a Simple Pet Fetching Web Application: https://workshops.vuevixens.org/workshops/vue/minis/mini1.html
- ❑ **Angular Tutorial - Angular.io**
  - ❑ Basic Angular Application: https://angular.io/start
- ❑ **Express - MDN:**
  - ❑ Express Tutorial: Creating a skeleton website
  - ❑ Complete Part 2 - Part 7 of this tutorial
  - ❑ https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express_Nodejs/skeleton_website
- ❑ **Google Cloud Platform - Firebase**
  - ❑ Firebase Web CodeLab: https://codelabs.developers.google.com/codelabs/firebase-web/index.html#0
- ❑ **Google Cloud Platform - Google Maps**
  - ❑ Build a Nearby Business Search service with Google Maps Platform: https://codelabs.developers.google.com/codelabs/google-maps-nearby-search-js/index.html

❑ **Exercises 07 - Graduate Students ONLY**

- ❑ Complete another tutorial from the list above