#### **Dynamic Web Server Instructions**

Task

Create a dynamic web server about US energy consumption from 1960 to 2018. Your server will serve static files (e.g. CSS, JavaScript, Jpeg, and Png files) a well as dynamic routes for pages on energy consumption for a particular year, energy consumption for a particular state, and energy consumption for a particular energy source.

*NOTE*: you are allowed to use any Node.js modules (built-in, installed via npm, or written yourself) to help develop your dynamic web server. You are also allowed to use any CSS / client-side JavaScript libraries to help design your web pages.

About the Data Set

*US Energy Database*:

I have downloaded data from the U.S. Energy Information Administration public dataset located at [https://www.eia.gov/state/seds/seds-data-complete.php?sid=US#StatisticsIndicators (Links to an external site.)](https://www.eia.gov/state/seds/seds-data-complete.php?sid=US#StatisticsIndicators) and stored them in a SQLite3 database.

The database has 2 tables as follows:

* States:
  + state\_abbreviation (TEXT) - two character abbreviation for a state
  + state\_name (TEXT) - full name of the state
* Consumption:
  + year (INTEGER) - year for consumption measurements in YYYY format
  + state\_abbreviation (TEXT) - two character abbreviation for a state
  + coal (INTEGER) - amount of energy produced by coal (in billion Btu)
  + natural\_gas (INTEGER) - amount of energy produced by natural gas (in billion Btu)
  + nuclear (INTEGER) - amount of energy produced by nuclear energy (in billion Btu)
  + petroleum (INTEGER) - amount of energy produced by petroleum (in billion Btu)
  + renewable (INTEGER) - amount of energy produced by renewable energy (in billion Btu)

Dynamic Web Server (50 pts)

To earn 38/50 points (grade: C)

* Package.json
  + Fill out the author and contributors sections in package.json (author should be whoever's GitHub account is used to host the code, contributors should be all group members)
  + Fill out the URL of the repository
  + Ensure all used modules downloaded via NPM are in the dependencies object
* Energy Consumption by Year
  + Implement a dynamic route for years 1960 - 2018 (/year/xxxx)
    - Page should be based on a single a template HTML file. The final HTML response should contain the following:
      * Year
      * Table with proper state consumptions for each energy source (including total of all 5)  
        Columns = State, coal, natural gas, nuclear, petroleum, renewable, and total  
        Rows = each State
      * Graph of energy consumption for the entire US (bar chart, pie chart, ...)
    - Dynamically populate the the template to include the specific year being viewed
    - Dynamically populate the body of a table with proper state consumptions for each energy source (including total of all 5) for the selected year
    - Dynamically populate some JavaScript variables in a script tag within the HTML template with proper US totals for energy consumption of each energy source (to be used for creating the graph)
* Energy Consumption by State
  + Implement a dynamic route for each State (/state/XX)
    - Page should be based on a single template HTML file. The final HTML response should contain the following:
      * State
      * Image to represent the State
      * Table with proper yearly consumptions for each energy source (including total of all 5)  
        Columns = year, coal, natural gas, nuclear, petroleum, renewable, and total  
        Rows = each year
      * Graph of percentage of energy consumption of each energy type for each year (percent stacked area chart)
    - Dynamically populate the the template to include the specific State being viewed (two character abbreviation)
    - Statically populate the image src with a generic image to represent all States (map of whole US, ...)
    - Dynamically populate the body of a table with proper yearly consumptions for each energy source (including total of all 5) for the selected State
    - Dynamically populate some JavaScript variables in a script tag within the HTML template with proper yearly energy consumption of each energy source (to be used for creating the graph)
* Energy Source page
  + Implement a dynamic route for each energy source (/energy/xxxxx)
    - Page should be based on a single template HTML file. The final HTML response should contain the following:
      * Type of energy
      * Image to represent the energy source
      * Table with proper yearly consumptions for each state (including total of all 51 - counting Washington DC)  
        Columns = year, AK, AL, AR, ... WY  
        Rows = each year
      * Graph of each State's yearly consumption for selected energy source (line chart)
    - Dynamically populate the the template to include the specific energy source being viewed
    - Statically populate the image src with a generic image to represent energy
    - Dynamically populate the body of a table with proper yearly consumptions for each State (including total of all 51 - counting Washington DC) for the selected energy source
    - Dynamically populate some JavaScript variables in a script tag within the HTML template with proper yearly energy consumption of selected energy source for each State (to be used for creating the graph)
* Site Navigation
  + Create a standard navigation to enable a user to select whether they want to view data by year, State, or energy source
    - Allow user to then select which year, State, or energy source
  + Home page should reroute to /year/2018

To earn a grade of A or B

* 3 pts: create dynamically populated 'previous' and 'next' links in the year.html, state.html, and energy.html templates that link to the previous or next page in its respective sequence
  + Link can either be disabled or circle around when at the first/last item
* 3 pts: dynamically populate States to include the **full name** (rather than abbreviation) of the specific state being viewed
* 3 pts: send a proper 404 error if the requested year, state, or energy source does not exist in the database
  + Can be plain text, but should be customized to the request (e.g. "Error: no data for state FB", or "Error: no data for year 2020")
* 3 pts: create a set of images (one for each state, and one for each energy source). Dynamically populate the <img> src and alt in the state.html and energy.html templates.
  + Make sure that you do not infringe copyrights - either create your own images, or find royalty free images and follow any stipulations the creators provide (e.g. citing where you got the image on your page)

Tip: I would suggest using an existing library for generating the graphs on your HTML pages. Some possible choices are:

* ChartJS ([https://www.chartjs.org/ (Links to an external site.)](https://www.chartjs.org/))
* AnyChart ([https://www.anychart.com/ (Links to an external site.)](https://www.anychart.com/))
* CanvasJS ([https://canvasjs.com/ (Links to an external site.)](https://canvasjs.com/))
* Plotly ([https://plotly.com/javascript/ (Links to an external site.)](https://plotly.com/javascript/))
* ...

Starter Code

Download [Project2-DynamicServer.zip](https://stthomas.instructure.com/courses/36471/files/3358413/download?wrap=1)U,{5ca8eeb7-30dd-492b-bce6-6ca84ace4b3c}{193},0.6666666666666666,0.6666666666666666 [Download Project2-DynamicServer.zip](https://stthomas.instructure.com/courses/36471/files/3358413/download?download_frd=1): contains starting server code and client code as well as a template for your package.json.

Note: the server will use the sqlite3 NPM module to interface with our database. The API for this module can be found on their GitHub page: [https://github.com/mapbox/node-sqlite3/wiki/API (Links to an external site.)](https://github.com/mapbox/node-sqlite3/wiki/API). Of particular interest will be the `Database#all(sql, [param, ...], [callback])` method. Also look at the `Database#run(sql, [param, ...], [callback])` method to see how to use the param optional parameter when constructing your queries.

Submission

Code should be saved in a repository on GitHub. Do NOT add your node\_modules directory to your repository. This is what package.json is for - it will store which modules you use for your project. In order to submit, you should enter the the project's GitHub URL for the assignment (in Canvas). I will be doing the following to assess your assignment:

1. git clone [https://github.com/*<user>*/*<project*](https://github.com/<user>/<project)*>*
2. cd *<project>*
3. npm install
4. node server.js

IMPORTANT: Only one group member should submit the GitHub URL. Every member should submit a checklist of what you feel you have accomplished from the rubric above (including who did what), and include your total expected score. This can be as a text entry submission (if not submitting the URL), or as a comment once you submit the URL.

Group

Andrew Buirge, Chase Rapp & Blake Robinson