Homework14

isAuthenticated.js

- middleware to see if a user is logged in

- calls next() when not logged in

config.json

- configuration for mysql database

- credentials, username, etc to connect to the db

passport.js

- use passport.js lib to setup authentication that is backed by mysql and uses Sequelize

- we use email to lookup a user and run validations on email and password props

models/index.js

- initialize Sequelize with credentials from config.json

models/user.js

- setup User model w/ username and password properties

- set things like validations and data type (both are String)

public/js/login.js

- setup FE JS for login functionality when DOM is ready to be manipulated by JS

- listens to the form’s submit event and logs the user in w/ the username and password data from the form

public/js/members.js

- checks which user is logged in and adds their email to the html tag w/ the class .member-name

public/js/signup.js

- signup functionality

stylesheets

- CSS styles

login.html

- login html markup

members.html

- members html

signup.html

- html markup for signup

routes/api-routes.js

- our api routes to be used via AJAX setup using ExpressJS

routes/html-routes.js

- routes that serve up our FE html when a user visits a page

package.json

- our project reps installed via NPM or Yarn

server.js

- our ExpressJS server file to be run to start the server

- run via the “npm start” / “node server.js” command