Welcome to Intro to Node Training

Instructor: Judy Lipinski

[judy@karmoxie.com](mailto:judy@karmoxie.com)

<http://linkedin.com/in/judylipinski>

if you want this as a PDF with links to the repo:

* [salma.khanam@ssa.gov](mailto:salma.khanam@ssa.gov); salmaknm@gmail.com
* keithbpereira@gmail.com
* quyenha3@gmail.com
* len.gorelik@ssa.gov
* [marc.grundfest@ssa.gov](mailto:marc.grundfest@ssa.gov)
* daniel.m.augustine@ssa.gov
* [Brian.ramsey@ssa.gov](mailto:Brian.ramsey@ssa.gov); musicbuilder01@gmail.com
* sean.grumbine@ssa.gov
* [michael.dorsey@ssa.gov](mailto:michael.dorsey@ssa.gov)
* navya.rajan@ssa.gov
* prkusha@gmail.com;usha.ponniah@ssa.gov
* nicholas.s.dickerson@ssa.gov
* cyril.adhikaram@ssa.gov;cyrilhai@gmail.com
* [douglas.gaede@ssa.gov](mailto:douglas.gaede@ssa.gov)
* [josephine.benedict@ssa.gov](mailto:josephine.benedict@ssa.gov)
* peter.choi@ssa.gov

To reach this document:

<http://moxie.pro/2019oct7node>

Table of Contents

[Course Slides Electronically](#_d7049h989580)

[Parking Lot](#_alen4j2mejp4)

[Tech Tips, Websites, & Advice](#_far2v4fnwrzr)

[Non-tech Things:](#_6gtotedzsb7b)

[Music Requests:](#_b2bjtpws5env)

[Modifying VSCode settings:](#_90huwx75kkj0)

# Connected to Machines:

* <http://labs.protechtraining.com>
* Username PTACCESSXXX where XXX is your number
* Everyone’s password: Pa$$w0rd

Course Lab Files:

* git clone <https://github.com/JudyLipinski/901NodeIntro-2019oct7>
* git clone <https://github.com/JudyLipinski/901NodeIntro-2019oct7>
* git clone <https://github.com/JudyLipinski/901NodeIntro-2019oct7>

# Course Slides Electronically

0 [Introduction](https://docs.google.com/presentation/d/e/2PACX-1vRJWBy6X-L799iJMdU4Mp5C7OJikRcfUQ_aWUEQPcH5u2Lunn_yrpBSSWmyWmSQLvg-DvrZln_L4Ycn/pub?start=false&loop=false&delayms=3000)

1 [Getting Started with Node](https://docs.google.com/presentation/d/e/2PACX-1vSdfN0MELzrx_1kRr0GWEmjvUTbdfOIqsYEB1peZ4lzOjPuE9SIt_Th9pyBKegvq9rzq31A7xTXu-Ii/pub?start=false&loop=false&delayms=3000)

2 [Modules, globals & event loop](https://docs.google.com/presentation/d/e/2PACX-1vQVItepizPRfGTt7YWhWgoCnyZwi-1K3Jtlwz-XdSOQQfnmNxFBXCFUvY3W9UZY3pjIAI18TFAYm_Ym/pub?start=false&loop=false&delayms=3000)

3 [Using npm](https://docs.google.com/presentation/d/e/2PACX-1vQ0578GPZHqSQW6O36OWjvBsEDcHBFra9hGerPI9mmSWCRrOa5edKwAobQbH9Ci5zXWWDN0GeE_o-kE/pub?start=false&loop=false&delayms=3000)

4 [Working with the filesystem](https://docs.google.com/presentation/d/e/2PACX-1vRU1ZkmdUxILPpPwY4LITNwDyR08N5JRnv0K1AVPYpx6VpKk9HhUVLjze7H_aTMT7UZZBCLWgtmxr1v/pub?start=false&loop=false&delayms=3000)

5 [PostgreSQL DB Access](https://docs.google.com/presentation/d/e/2PACX-1vT_EzlFkl5Jivtast7U-ErHZJCGmbrwFK440fYcvY5Dh8g1ZhOIHM1gy0YaylpONYSlJQ4rD6FkQ-J0/pub?start=false&loop=false&delayms=3000)

6 [Http Server](https://docs.google.com/presentation/d/e/2PACX-1vSRadhO8A8vstklhqwNpfbHgrjjdgFYgcEKG9mrzq85vWbDmzSzO6TESSRW7ghunuC6CUapGlVmo8k1/pub?start=false&loop=false&delayms=3000)

7 [Using the Express Module with PostgreSQL](https://docs.google.com/presentation/d/e/2PACX-1vSDAgOBpjbLQQZ8iUiDJNNDjZI-zg-DGivAwUo1UrCwqjNhnG_254dJnUI4TFyLtKIZQPSWPbOcyMNV/pub?start=false&loop=false&delayms=3000)

8 [Using Sessions](https://docs.google.com/presentation/d/e/2PACX-1vS15UGs79zO-f6ISh8uyV7AZsE9z2KRcVAfSh85BsgV53AcNliITASC5LEvVI15IVjs3PHbFf1cN8tu/pub?start=false&loop=false&delayms=3000)

9 [Validation, CSRF and Database Relations](https://docs.google.com/presentation/d/e/2PACX-1vRImocLWjz0CfylldbQWzT88oCpCqfyrUKRTttBRPNQb9x-oRZjYXjOfiGNusNNhPMbFLxMRPmxbIo3/pub?start=false&loop=false&delayms=3000)

10 [Testing in Node](https://docs.google.com/presentation/d/e/2PACX-1vSD5uJzq3C5gYmwd6ZPqzuesxVXYL2anQQU3IcQ5oJaygIKMkR4wLW6MQPkboKWMW5awdya65rIPyVJ/pub?start=false&loop=false&delayms=3000)

11 [Summary](https://docs.google.com/presentation/d/1ovhvqicLl_puDqIOWLahPU7rdc_XpzbAhhty1SY57Os/edit#slide=id.g10e9c3d72e_0_0)

# 

# Parking Lot

* Demo setting breakpoints in VSCode
* Demo - previous weather solution
* Have some fun practicing Node! npm install -g learnyounode
  + then run this command from a prompt: **learnyounode**
* Slide 45 - change rea() to area() in green box
* <https://answers.microsoft.com/en-us/windows/forum/windows_vista-files/how-can-i-recover-deleted-files-with-shift-del/13b1eaad-d41a-4e77-91c5-f0bc9590de7d>
* judy to confirm - 4 thread pool YES
* knex anonymous <https://github.com/brianc/node-postgres/issues/1062>
* Ngnx
* <https://nordvpn.com> having a special now about $4 . mo with 3 year sign up

# Tech Tips, Websites, & Advice

* **ES6 features**
  + [**https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Destructuring\_assignment**](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Destructuring_assignment)
    - in left pane - check out the JS tutorials
  + Array.prototype functions
    - <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/isArray#>
* **Ryan Dahl Talks**
  + [**10 Things I Regret About Node.js - Ryan Dahl - JSConf EU**](https://www.youtube.com/watch?v=M3BM9TB-8yA)
* **NPM**
  + <https://docs.npmjs.com/>
  + <https://docs.npmjs.com/packages-and-modules/>
  + Top npm dependencies
    - <https://www.npmjs.com/browse/depended>
* **Node Style guide**
  + <https://github.com/felixge/node-style-guide>
* **NodeJs Documentation**
  + <https://nodejs.org/api/documentation.html>
* Node single threaded info (event loop)
  + <https://hackernoon.com/demystify-the-multi-threading-in-node-js-4ff07eae9abe>
  + <https://medium.com/better-programming/is-node-js-really-single-threaded-7ea59bcc8d64>
* **Promises**
  + **JavaScript promises**
    - <https://promisesaplus.com/>
  + Promises (bluebird) <https://gist.github.com/joepie91/4c3a10629a4263a522e3bc4839a28c83#7-how-do-i-make-this-non-promises-library-work-with-promises>
  + <https://www.guru99.com/bluebird-promises.html>
  + <https://javascript.info/promise-chaining>
  + Promise cheatsheet: <https://devhints.io/bluebird>
  + Promise.try() and why we use it - TURN SYNC errors into ASYNC
    - <http://bluebirdjs.com/docs/api/promise.try.html>
  + You're Missing the Point of Promises [article](https://gist.github.com/domenic/3889970)
* **interesting questions / answers forum**
  + <https://www.quora.com/>
* **security**
  + **OWASP** [**https://www.owasp.org/index.php/Category:OWASP\_Top\_Ten\_Project**](https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project)
  + [**https://www.npmjs.com/package/csurf**](https://www.npmjs.com/package/csurf)
* **Validation** 
  + [**https://www.npmjs.com/package/checkit**](https://www.npmjs.com/package/checkit)
* **Knex**
  + [**http://knexjs.org/**](http://knexjs.org/)
* **WTF** **Javascript**

<https://www.youtube.com/watch?v=et8xNAc2ic8>

# Non-tech Things:

* GRUB HUB <http://fbuy.me/k91_U>
* [www.thejigsawpuzzles.com](http://www.thejigsawpuzzles.com)

# Music Requests:

* <https://www.youtube.com/watch?v=x8zsE5zdlsQ>
* <https://www.youtube.com/watch?v=CD-E-LDc384>
* <https://www.youtube.com/watch?v=EtI7f3Rwqkw>
* <https://www.youtube.com/watch?v=Ypn-3mX3Ooo>
* <https://www.youtube.com/results?search_query=cute+animals>

# Modifying VSCode settings:

* gear icon - then choose settings
* GUI is shown or modify the JSON object

|  |  |
| --- | --- |
| "javascript.validate.enable": false, | turns off message to convert commonJS to ES6 |
| "explorer.autoReveal": false, | won’t jump explorer selection around when selecting tabs in vscode |
| Enable preview | Disable if you do not want single click to open temporarily and get replaced |

To make cursor jump between editor and the terminal window can add a keyboard shortcut.

Use control-Shift-P and type keyboard shortcuts <https://stackoverflow.com/questions/42796887/switch-focus-between-editor-and-integrated-terminal-in-visual-studio-code>

VSCODE Keyboard shortcuts:

<https://code.visualstudio.com/shortcuts/keyboard-shortcuts-windows.pdf>

# Review Sessions

* What was something that stood out for you or helpful yesterday?
* Do you have any questions?

Fri am

* ES6 object destructuring using {propertyName1, prop2} inside a function to pull out those properties from passed in object
  + example fromserver.js passing state object to routes - that has a property of db



* + in the module that received this state object - can pull out simply the properties you want



* working with sessions in express
* error handling
* using pug as the template engine
* using express.static to add assets like public
* middleware pattern
  + indicate a path to match (optional)
  + function (request, response, next)
    - in function, can use response
      * render(view) - for template engine
      * send() - send back response
        + text
        + JSON - express knows the difference!
        + HTML
    - or use next()
      * goes to the next matching middleware
    - or use next(error)
      * go to the middleware error handler
  + function (error, req, resp, next)
    - this function is called when there is an error in the middleware
  + some considerations:
    - don’t forget to call next() or send response
      * else it hangs
    - order is important!
      * if a matching middleware executes and sends a response, no others will execute
    - how to do 404
      * just have another middleware, last in the list - and if it reaches that middleware, nothing was setup for that route - so 404
* Want test data?
  + can use faker 3rd party module
* if creating rest apis - don’t need a web front end to test
  + use postman! install this software <https://www.getpostman.com/downloads/>
  + comparing with ReadyAPI
    - <https://www.quora.com/What-is-the-major-difference-between-SoapUI-and-Postman-And-which-one-is-preferred-for-testing-REST-APIs>

Thurs am

* knex
* Postgres
* Express
  + call express() to get back app object
  + let app = express()
  + register middleware with app.use(XXXX)
  + then configure all the middleware
    - so far, we configured routes
    - path ‘/about’ leading to the right code
* nodemon
  + for being able to have server restart when we change files
* Promises
  + .then()
  + .catch()
  + .finally()

Wed am

* review of npm in web dev course
* npm setup
* order of registering event listeners makes a difference
* how will this play at SSA?
  + how to think about what we are learning as it will apply back at work
  + some areas already have Node projects
  + some do not. future journeys?
* discussing how Node handles events
  + event loop
  + internally Node uses threads
  + as developers, focus on our code being in a single thread - with objective to process things asynchronously - so it goes to the event loop
* Angular uses ES6 Module syntax
  + uses the keyword **import**
  + Node CommonJS syntax
    - uses the keyword require
    - Node experimental features have started to rewrite the APIS

Tues am

* ES6 Classes
  + using **this**
* Node
  + has modules built in: http, fs
  + limited in mathematical precision
  + commands: node file.js or node file
* Modules
  + CommonJS
    - module.exports - for export
      * export classes, functions, variables
    - require - for import
      * for custom module
      * require(‘./utility’) local files must include the path
* JavaScript
  + anonymous functions
  + arrow functions
  + review of arrays.prototype
  + backticks
    - vs single / double, can include other quote
    - concatenation
      * `you can embed ${variableName} in a string`

Ts\_tag :

// ts\_tag : returns fmt msg for use in assisting in understanding threading

// M Grundfest 20191008 v 0.5 beta

let ts\_tag= function(base) {

let m;

return ( `${m= new Date()} offset = ${m-base} ms` )

}

let local=ts\_tag;

// test

const base = new Date();

console.log("nnnn " + ts\_tag(base));

module.exports = ts\_tag;

After Lunch:

* Challenge, create a table with weather info by day
  + can do manually in GUI
  + or can use program to create it
* Read that in data, and find the hottest day and coldest day and display