Welcome to Node Training

**You can reach this document at:**[**shorturl.at/wzGH5**](http://shorturl.at/wzGH5)

**Instructor:** Christopher Cioffi

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

<https://www.linkedin.com/in/cioffichristopher/>

**Repository**:

For README.md on your right side monitor click to the GitHub page

<https://github.com/JudyLipinski/901NodeIntro>

# **Connecting to Remote Machine**

* [HTML 5 Remote Lab Connection PDF](http://www.protechtraining.com/downloads/HTML_5_Remote_Lab_Connection.pdf) (Use for class)
* List of machine assignments: [link](https://docs.google.com/spreadsheets/d/1SJKjfzlkC7FFHOLYTHb7rFojQ2mnONExRk4qTltsiQk/edit?usp=sharing) (choose a machine by adding your name)
* To bring up your Virtual Machine:
* Visit <http://labs.protechtraining.com>
* Supply your username & password

# **Slides:**

[Ch00: Introduction](https://docs.google.com/presentation/d/e/2PACX-1vRJWBy6X-L799iJMdU4Mp5C7OJikRcfUQ_aWUEQPcH5u2Lunn_yrpBSSWmyWmSQLvg-DvrZln_L4Ycn/pub?start=false&loop=false&delayms=3000)

[Ch01: Getting Started with Node](https://docs.google.com/presentation/d/e/2PACX-1vSdfN0MELzrx_1kRr0GWEmjvUTbdfOIqsYEB1peZ4lzOjPuE9SIt_Th9pyBKegvq9rzq31A7xTXu-Ii/pub?start=false&loop=false&delayms=3000)

[Ch02: Modules, globals & event loop](https://docs.google.com/presentation/d/e/2PACX-1vQVItepizPRfGTt7YWhWgoCnyZwi-1K3Jtlwz-XdSOQQfnmNxFBXCFUvY3W9UZY3pjIAI18TFAYm_Ym/pub?start=false&loop=false&delayms=3000)

[Ch03: Using npm](https://docs.google.com/presentation/d/e/2PACX-1vQ0578GPZHqSQW6O36OWjvBsEDcHBFra9hGerPI9mmSWCRrOa5edKwAobQbH9Ci5zXWWDN0GeE_o-kE/pub?start=false&loop=false&delayms=3000)

[Ch04: Working with the filesystem](https://docs.google.com/presentation/d/e/2PACX-1vRU1ZkmdUxILPpPwY4LITNwDyR08N5JRnv0K1AVPYpx6VpKk9HhUVLjze7H_aTMT7UZZBCLWgtmxr1v/pub?start=false&loop=false&delayms=3000)

[Ch05: PostgreSQL DB Access](https://docs.google.com/presentation/d/e/2PACX-1vT_EzlFkl5Jivtast7U-ErHZJCGmbrwFK440fYcvY5Dh8g1ZhOIHM1gy0YaylpONYSlJQ4rD6FkQ-J0/pub?start=false&loop=false&delayms=3000)

[Ch06: Http Server](https://docs.google.com/presentation/d/e/2PACX-1vSRadhO8A8vstklhqwNpfbHgrjjdgFYgcEKG9mrzq85vWbDmzSzO6TESSRW7ghunuC6CUapGlVmo8k1/pub?start=false&loop=false&delayms=3000)

[Ch07: 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)

[Ch08: Using Sessions](https://docs.google.com/presentation/d/e/2PACX-1vS15UGs79zO-f6ISh8uyV7AZsE9z2KRcVAfSh85BsgV53AcNliITASC5LEvVI15IVjs3PHbFf1cN8tu/pub?start=false&loop=false&delayms=3000)

[Ch09: Validation, CSRF and Database Relations](https://docs.google.com/presentation/d/e/2PACX-1vRImocLWjz0CfylldbQWzT88oCpCqfyrUKRTttBRPNQb9x-oRZjYXjOfiGNusNNhPMbFLxMRPmxbIo3/pub?start=false&loop=false&delayms=3000)

[Ch10: Testing in Node](https://docs.google.com/presentation/d/e/2PACX-1vSD5uJzq3C5gYmwd6ZPqzuesxVXYL2anQQU3IcQ5oJaygIKMkR4wLW6MQPkboKWMW5awdya65rIPyVJ/pub?start=false&loop=false&delayms=3000)

[Ch11: Summary](https://docs.google.com/presentation/d/e/2PACX-1vQbsgUpCpbZd5sOMxGGya0Vy_5ZAfDjhxuBDtfjB-p1A-TQ-u5ghKxEDsfEw29HtBn4EWzCH3rhv8Fe/pub?start=false&loop=false&delayms=3000)

If you need to reboot your VM:

* Execute this command in the command line
  + Shutdown -r -t 0

System source checklist:

* Set resolution on monitors to RECOMMENDED
* Shortcut through back of building
* Windows
  + command to get list of files? dir
  + shortcuts how to copy, paste, select all?
    - undo - ctrl+Z
    - copy - ctrl + C
    - paste ctrl + V
    - redo - ctrl + Y (undo your ctrl+Z)
    - save - CTRL + s

# OPTIONAL use of GitHub:

Visit [https://GitHub.com/](https://github.com/) and sign into an existing account of yours or sign up for a new account.

If signing up for a new account, you will need to provide an email that you can check and verify, on your phone or by website. If you cannot, the instructor will provide you an email to you. [yournamehere@karmoxie.com](mailto:xxx@karmoxie.com)

# **Parking Lot:**

# **Tech Resources:**

<https://nodejs.org/en/docs/guides/>

* Interactive terminal for node exercises, similar to javascripting
  + <https://github.com/workshopper/learnyounode>

# **Node Guides:**

* <https://nodejs.org/en/docs/guides/>

## Node.js core concepts (Intermediate)

* + [Overview of Blocking vs Non-Blocking](https://nodejs.org/en/docs/guides/blocking-vs-non-blocking/)
  + [The Node.js Event Loop, Timers, and process.nextTick()](https://nodejs.org/en/docs/guides/event-loop-timers-and-nexttick/)
  + [Don't Block the Event Loop (or the Worker Pool)](https://nodejs.org/en/docs/guides/dont-block-the-event-loop/)
  + [Timers in Node.js](https://nodejs.org/en/docs/guides/timers-in-node/)

# **Music Requests:**

* <https://radio.garden/>
* cho yong pil: <https://www.youtube.com/watch?v=oqiDWM0OOD4>
* <https://www.youtube.com/watch?v=7t3xBqAWLaU>
* Together ~~ The Confluence ~~ Rahul Sharma and Richard Clayderman
  + <https://www.youtube.com/watch?v=zAWv2U8kKAY>

# **Misc:**

* [How-a-programmer-almost-broke-the-internet-by-deleting-11-lines-of-code](https://www.sciencealert.com/how-a-programmer-almost-broke-the-internet-by-deleting-11-lines-of-code)
* Baby Elon Musk Montage Deepfake
  + <https://www.youtube.com/watch?v=WHwQeetjLwk&feature=youtu.be>
* Mike Tyson Family Matters Deepfake
  + <https://youtu.be/zWOY10DyPnU>
* Chase bank Canadian credit card debt forgiveness
  + <https://www.cbc.ca/news/business/chase-bank-amazon-visa-marriott-credit-card-debt-1.5239411>
* <http://voidcanvas.com/npm-packages-you-wont-believe-exist/>

Bonus Extra Credit Project:

* You will create a new website project based on a hobby of yours
* You’ll create an optimal solution of best practices using everything we’ve learned in this course
* In the future, you can use this project as a reference for when you’re using Node on the job
* As you complete each lab, you will have learned more of what you can add to your website, use your free time to do so
* Requirements:
  + Client Side:
    - You can continue to use pug or use Angular/React
    - Use REST API
  + Create REST API
    - Be creative and think of what types of request your client side would need to make to your backend if you were creating a realistic website for your hobby
    - Have something that requires CRUD on database
  + Create node server
    - Ensure you’re following best practices like we’re teaching you in class, ie. project directory/file structure
    - Use express
    - nodemon
    - Use knex and pg
  + PostgreSQL
    - Create a new DB to interact with, can be on same server as course files

# **Tuesday Review:**

* Something new or interesting
* Any Questions
* VS Code Integrated terminal
  + Open in terminal
  + Powershell vs bash
* Using node command to run JS in terminal
* Arrays
  + Prototype functions
    - Sort
    - Slice
    - Filter
    - Push / pop
    - Find
    - Map
* Javasripting
* Code Runner extension
* Modules
* Class syntax
* Prototype functions
  + Only saved once in memory no matter how many instances of obj
* Arrow functions
  + function (param0) { }
  + (param0) => { }

# **Wednesday Review:**

1. Something new or interesting
2. Any Questions for us?

* Npm install (easy!)
* Open source
* Package.json
  + All dependencies
  + Scripts
  + Devdependencies
  + Meta data about the project
* Semantic versioning semver
  + 1.0.4
* Package-lock.json
* Modules
  + Modules.exports
  + require(‘http’)
* Callbacks
* eventEmitter
* Globals
  + process
  + Require
* Moment package

# **Thursday Review:**

1. Something new or interesting
2. Any Questions for us?

* Nodemon
  + Automatic restart, detects changes
* Promisify
  + Bluebird
  + require(‘util’)
* Streams
  + Read, write, duplex, transform
* require(‘http’)
* Knex
  + Query builder
  + require(‘pg’)
* PostgreSQL
  + pgAdmin (client GUI)
* require(‘fs’)
* Promise.try() {
  + Asynchronous fs.readFile
  + then() =>
  + catch()
  + finally()

# **Friday Review:**

1. Something new or interesting
2. Any Questions for us?

* next()
* Middleware
  + Logging
  + Manipulating request
  + Getting data
  + Routing ‘/about’
  + Errors
* Templates and engine
  + View
* Routes
* Promises
* Express
* Session data
* PostgreSQL
* Knex
  + Migrations
* body-parser
* Cookies
* express.static(
  + Js
  + CSS
  + Favicon.ico
* Custom errors
  + Duplicate
  + Validatoin error