**The following pages will be dedicated to the understanding and persistent pursuit of becoming a fluent server side programmer.**

**This will aid me towards achieving a MERN stack developer role.**

**Created by Jonathan Mohabir**

[**https://github.com/EthanRBrown/web-development-with-node-and-express-2e**](https://github.com/EthanRBrown/web-development-with-node-and-express-2e)

**companion repo for code understanding.**

**Chapter 01 introducing express:**

* The age of JavaScript is truly among us.
* Thanks to node, knowing this language allows a front end developer to incorporate some server-side capabilities of his application he will be creating.

* This is truly an exciting time to be apart of internet technology.
* JavaScript started in 1996, but a lot of its features that were pushing for greater use among programmers came during the year of 2005.
* From there… in 2009 we then see the release of NODE JS. Node js is a JavaScript runtime environment. Node package manager contains packages of things that can help us develop applications for the web.
* The internet today is estimated to be worth 2.1 trillion dollars….. you can get in on a piece of that pie… if you know how.

**LOOKING AT EXPRESS**

* Express is simply a web application framework.

* Express allows us to actually write server side code .
* Express simply **accepts HTTP requests from the client, and then returns an HTTP response.**
* Single page applications are applications used by users which can allow for functionality of rendering data from a server, and contain a front end for displaying HTML , CSS, media assets, and JavaScript DOM manipulation.

**Chapter: 02 – Getting started with Node.**

* Node is best started when its actually being coded… so in that case refer to Jonathan\_coding\_prac repo.

* Inside chapter 02 of the orielly\_book directory contains this chapters code.
* Comments also included within the code.
* To start a simple server with node, the http module is required to be imported. As we are writing http requests and responses.
* Require is a keyword within node which allows us to work with different modules.
* When starting a server: declare an http constant which allow for methods of this module to be used.
* Set a Port for the app to be started onto.
* Express does better handling for this so just understanding the basics from this node file.
* A switch case was included into the create server method to demonstrate routing capabilities of the node system.