jQuery AJAX JSON

JSON:

* Method for sending, receiving, & storing data – Text written with JS object notation
* Objects & Arrays nested inside of each other
* Backend team builds the API for us

AJAX:

* Loading data in the background and display it on the webpage, without reloading the whole page
* Serialize: taking a complex object on the server and break it down to a rawer form – text, JSON
* Deserialize: taking raw data converting it back into the object graph you had before
* XMLHttpRequest: web browsers have this built in tool
* Will establish a connection with the URL you specify
* Then lets us send & receive data

jQuery & AJAX:

* With the jQuery AJAX methods, you can request text, HTML, XML, or JSON from a remote server using both HTTP Get and HTTP Post - And you can load the external data directly into the selected HTML elements of your web page!
* GitHubGist ???

Node.js Tutorial

**Node.JS** Runtime – node.exe

* A JavaScript runtime built on Chrome’s V8 JavaScript engine
* Node is essentially JS running on the server
* The runtime environment provides the built-in libraries that are available to the program at runtime (during execution)
* Node.js uses an Event Driven, Nonblocking I/O model
* Asynchronous
* Single thread

Traditional server-side technology – ex: PHP application on Apache server

* Synchronous
* Each connection / request spawns a new thread which takes up system memory
* Eventually will max out
* Essentially needs to wait for one process to stop before starting the next

REPL –

* Read, Eval, Print, Loop
* Allows us to run JS in the Command Line
* JavaScript without the browser / DOM functionality

NPM package manager –

* Allows us to install Packages & Modules for node
* Execute ‘npm init’ in cmd prompt

Package.JSON File –

* Main manifest file you must have in your node apps
* Goes to root of your package/application
* Tells the NPM how your package is structured & how to install it

Node.js most useful for:

* Restful API’s, Backend Applications
* Anything NOT CPU intensive

Building a Basic Web Server –

JQuery Ajax Node.js Live-Server

NPM Live Server:

* To install 🡪 cmd prompt – npm install live-server -g // npm i live-server
* Check version 🡪 npm -v
* Run server 🡪 live-server
* Specify port 🡪 live-server --port= port #
* Specify directory index 🡪 live-server --entry-point= index.html (name of file)
* Cannot GET/ ???????????????????????

AJAX: (MVC jQuery) <https://www.youtube.com/watch?v=_89ngzQIDG8>

* jQuery Links🡪 <https://developers.google.com/speed/libraries/>
* Asynchronous JavaScript And XML (XML portion doesn’t always apply anymore)
* Making a JS call to a web server & getting a response back in some sort of metadata JSON, XML
* Making an asynchronous request to a web server a receiving response back w/out refreshing the page
* \*Don’t have to refresh the page
* Doesn’t have to be asynchronous but usually is

MVC:

* Every View has a:
* Model: info stored on the back end
* Controller:

Restful:

* Confining yourself to Get, Post, Update, Delete / CRUD