**Introducing Node and Express**

[Node.js](https://nodejs.org/en/) allows us to write JavaScript code that executes on a server.

Node uses the [V8 JavaScript engine](https://developers.google.com/v8/) from Chrome. Written in C++, V8 is a [just-in-time compiler](https://en.wikipedia.org/wiki/Just-in-time_compilation) that takes the JavaScript you write in your web apps and turns it into machine-level instructions.

[Express](http://expressjs.com/) is a "minimalist web framework for Node.js" that simplifies creating modern server-side web applications in Node.

We'll use Express in all of the apps we create in this course. You'll be learning how to create Express apps that:

* serve static assets like HTML, CSS, client-side JavaScript, and images
* expose RESTful APIs that clients can make requests to in order to retrieve data
* persist data to and retrieve data from databases

This course will dive into the devilish details, but in this reading we'll get our first exposure to a simple Express app. Don't expect to understand each line of code here, but instead focus on building up a mental model of the different parts of an Express app, and how they interact with one another.

**A simple counter app**

The app below uses [Glitch](https://glitch.com/about/), which is the cloud-based Node environment we'll use in this course to demonstrate and practice server-side programming concepts in Node. If you've used [repl.it](https://repl.it/" \t "_blank), [Codepen](https://codepen.io/" \t "_blank), or [JSBin](https://jsbin.com/" \t "_blank) for front-end prototyping, the look and feel of Glitch will be familiar, with the key difference that Glitch is for both server and client-side programming.

**var express = require('express');**

**var app = express();**

**app.get('/', function(req, res) {**

**res.render('index', {**

**title: "hello i am here",**

**nav: nav**

**});**

**});**

**app.get('/books', function(req, res) {**

**res.send('books');**

**});**

**app.listen(port, function(err) {**

**console.log('running');**

**});**