**What is Node.js?**

According to the documentation, “Node.js is an open-source and cross-platform JavaScript runtime environment” (Introduction to Node.js, n.d.). I always thought of Node as the “server-side” of JavaScript that does the heavy lifting and allows web developers to create more robust applications.

**Node Use**

Node is a platform used to run JavaScript outside of the browser (Hahn, 2016). This means that JavaScript can be used anywhere other languages are utilized. A good example is the use of Node competing alongside the language C as a viable option in embedded systems (Guinard, 2016). So, not only is Node used to create a seamless connection between client-side and server-side but also serves to create code for many other devices.

**Node vs JavaScript**

The main difference is that Node is an environment for JavaScript and JavaScript is a language. JavaScript is used in Node to make things happen like using JavaScript in the browser with differences in what is available for the language to access.

**Chrome V8 Engine**

The V8 engine enables JavaScript to be converted to machine code which is why Node is a powerful tool for the language (Tripathi, 2017). Basically, it takes the high-level language of JavaScript and translates it into a low-level language the CPU understands. The V8 engine also defines standards according to ECMAScript and can be used as a standalone environment to create and execute JavaScript code separate from the browse (Tripathi, 2017).

**What is npm?**

When Node is installed, a second program is made available as well going by the name npm. This program, npm, is Node’s unofficial package manager (Hahn, 2016). When a project needs third-party modules, npm is used to install these modules to the project.

**npm Use**

Just like any programming language, there are many libraries that can save time and offer invaluable tools for creating projects. Once a package has been defined inside the project, npm can be used to install the dependencies from the package.json file (Hahn, 2016).

**References**

Guinard, D. (2016, June 18). Node. Js for embedded systems – web of things. https://webofthings.org/2016/06/18/node-js-for-embedded-systems/

Hahn, E. (2016). Express in action: Writing, building, and testing Node.js applications. Manning Publications.

Introduction to Node.js. (n.d.). Retrieved August 11, 2022, from https://nodejs.dev/learn/introduction-to-nodejs/

Tripathi, M. (2017, December 20). Understanding how the chrome v8 engine translates javascript into machine code. https://www.freecodecamp.org/news/understanding-the-core-of-nodejs-the-powerful-chrome-v8-engine-79e7eb8af964/