Zahava Gopin

WEB 330: Discussion 3.1

Professor Krasso

1/18/23

Module Bundlers

JavaScript module bundlers begin at the entry point of your application and combine all the modules that were used. The module bundler then rewrites all the modules in a single line that can work on all browsers- even ones that don’t support modules.

Most module bundlers have the same features. However, some are easier t use than others. Three examples of module bundlers are webpack, rollup, and parcel. With webpack, the page does not require reload after a small change in the code or CSS. Additionally, every file you create becomes a module, therefor removing the possibility of overwriting global variables. Webpack helps to minimize the amount of code that you have by removing unnecessary code like, whitespace, line breaks, and changes long variable names. However, webpack is hard to use. It requires a lot of configurations and many plugins. Rollup is efficient and returns very minimal bundles. Rollup deletes unnecessary code, as well, and has scope hosting and simple API. However, rollup does not have great support for other files like images and CSS. Parcel requires no configuration. Any dependencies are installed automatically and is known for its code splitting. However, parcel bundles everything, including images and CSS together, which can be confusing and messy. When using a module bundler, you allow the code to work on all browsers regardless of their module acceptance status. Bundlers reduce extra code and make the actual files shorter. However, its important to keep bundles small as they do increase the amount of code. Modules also only load code when needed which helps reduce slow-down from unnecessary parts of an app at any given time.

Resources:

satwiksuman. (2022, August 22). *What are the advantages of using Webpack ?* GeeksforGeeks. Retrieved January 18, 2023, from https://www.geeksforgeeks.org/what-are-the-advantages-of-using-webpack/

AsyncBanana. (2021, April 19). *Why you should not use Webpack*. Medium. Retrieved January 18, 2023, from <https://javascript.plainenglish.io/why-you-should-not-use-webpack-f07f4fd7c116>

Boyle, L. (2018, February 22). *Rollup vs Webpack (JavaScript bundling in 2018)*. Medium. Retrieved January 18, 2023, from <https://medium.com/jsdownunder/rollup-vs-webpack-javascript-bundling-in-2018-b35758a2268>

Wickramarachchi, V. (2021, September 9). *Parcel vs. WebPack 2021*. Medium. Retrieved January 18, 2023, from https://levelup.gitconnected.com/parcel-vs-webpack-2021-64c347bcb31