How TypeScript Improves JavaScript

JavaScript is a very powerful coding language that has been used for many years. It has many applications and is versatile and flexible enough to be used with many different browsers, operating systems and code bases. Many developers like to use JavaScript because it is easy to use and easy to learn. It has many benefits but the biggest issues with JavaScript is that it isn’t very scalable.

When the creators of JavaScript started it, they never imagined its wide-spread use and therefore, they didn’t create it with scalability in mind. Because it has been used so much and in so many applications, websites and technologies, Microsoft decided to try to make it more scalable. They could’ve gone out and create a new language but instead, they created TypeScript. A language that adds to JavaScript. A language that uses JavaScript but is a statically types subset of JavaScript.

Just like JavaScript, TypeScript can be used on any browser, any operating system and on any host server. Just like JavaScript, this makes it flexible, versatile and easy for developers to use. Because it is written on top of JavaScript, it also is easy to code if you already know JavaScript.

When the creators sought out to create TypeScript they wanted to make sure any one could use it or add to it so they kept it an open source platform. This allows for a community of people to improve upon it over time.

Two key benefits of TypeScript are; it is enabled by static types and it was created with features from the future.

Static types in TypeScript declare variables with a specified typed. While this is optional in TypeScript, the biggest benefit is that it debugs the code as you’re writing it. JavaScript will only debug during compiling so this is a big advantage when you think about quality of your code. Specifying types also makes your code more readable and easier to write which helps when scaling code on a large project.

The other benefit of TypeScript is that is was made with features from the future. With JavaScript, there is always a feature gap. A feature gap is when a browser or operating system is not compatible with your version of JavaScript. With TypeScript, it will detect the feature gap for you and guide you on how to make sure that you are forward compatible with any new browsers or operating systems.

As someone new to development, I have really enjoyed the benefits of TypeScript. With JavaScript, you can have a very small error that derails your entire code and throws an error message or causes it to not compile at all. With TypeScript you are able to debug as you go and make sure that when you get to the point of compiling your code, you have most of your bugs worked out.

One thing that would make TypeScript better would be debugging any syntax errors in the code. Sometimes, I forget to close a bracket or forget a semi-colon and it will bring down my entire application or code. It would be great to have that level of “spell check” for your code as you write it. Maybe someone in the open source community will add that feature in the future.

Overall, the many benefits of using Typescript instead of JavaScript outweigh the small amount of time it takes to learn the language on top of JavaScript.