---

layout: essay

type: essay

title: Reflection on Javascript

date: 2016-09-4

labels:

- Learning

- Javascript

---

<img class="ui medium left circular floated image" src="../images/Java Logo.png">

Javascript has been a brand new programing language that I have recently begun to learn. From what I've used and done with it so far, I can say that I am enjoying it. My career of programming began first with Java. Java is a simple, yet powerful programming language that is great for beginners to learn. It has access to classes, loops, arrays, objects, and various other things that give a programmer access to useful tools without having to worry too much about things like the assembler. Garbage collection was a big thing that I liked about Java and how Java, in general, makes the overall programming process simplier and "almost" pain-free.

From Java, I had learned C#, C, and C++. All these languages are very powerful in that they give the programmer more control over how objects are made, where they are made in the stack/heap, and many other hands-on types of tools. With these powerful tools, also means more headaches to make sure the code creates things in the right places and removes them properly. Juggling these types of responsibilities puts more pressure on the programmer on top of the pressure of creating code.

Javascript is a "best of both worlds" kind of language in that it seems to really streamline the coding process. It has almost all the same tools of Java and C++, but a few extra benifits that make it stand out. Javascript having "Prototype Inheritance" means that the need to have interfaces, abstract classes, and deconstructors is eliminated. This removes some of the load off the programmer having to worry about arbitrary stuff, rather than more important parts of their algorithm.

In addition to being introduced to Javascript, I have recently been undergoing an activity called "athletic software engineering". Athletic software engineering is a time-based programming exercise that involves solving a programming program as fast as possible while still being correct. It is a stressful experience, but through repition it has slowly been improving my programming skills. The time aspect makes it so that you think with all your might in the few minutes time constraint and find the solution in a timely manner. I'm certain that this mimics real-world conditions of solving complex coding problems within time constratins. In no way can you say to your boss that you've failed to find the solution and need more time. Thus this exercise is proving a good prepatory exercise to entering the field of software engineering.