sitepoint premium

Handout

Introduction to JavaScript
Lesson 1 - Step 1
M. David Green

The History of JavaScript

Before we get started, I'd like to tell you a little bit about the history of JavaScript.

JavaScript changes very quickly. As you look through examples of Java Script code from different places on the Internet and read tutorials, you're going to find that people's coding styles and even the examples sometimes won't work in every version of every browser that you have.

JavaScript was actually invented in **1995** for Netscape. Brendan Eich created a language called **LiveScript** that was supposed to be a programming language to exist in the browser and help developers manipulate the content of web pages. Just before LiveScript was actually released, the marketing people at Netscape decided to rename it **JavaScript** to take advantage of the popularity of the Java programming language. That decision has made it very confusing for many people. One of my favourite sayings is that Java is to JavaScript as cat is to catalog: there really is no relationship between the two languages.

JavaScript was developed in order to be easy to use. It was supposed to be a coding language that people could use directly in web pages to manipulate the elements there without really understanding the broad fundamentals of programming.

JavaScript is compiled directly in the browser. When you write a piece of JavaScript code, the browser interprets that code in a way that it can execute. This means that once you have a browser installed on your system, you have a JavaScript interpreter and compiler right there at your fingertips.

JavaScript was also designed to be cross-platform. Just like the browser the idea is that no matter what computer you're working with you should be able to create a JavaScript program and have it run consistently.

One of the early issues that hit the JavaScript world was standardization. JavaScript was developed by Netscape, and Netscape had its own browser as well as Microsoft. Microsoft wanted to customize the programming environment to take advantages of special features that they had in their own browser. They developed variations called **VBScript** and **JScript**, which weren't completely compatible with the general JavaScript that Netscape was developing.As a result, developers for years had to choose between working in JavaScript or in JScript or VBScript or creating subsets of those that were compiled in different ways to make them functional in both environments. Ultimately **European Computer Manufacturers Association** stepped in and decided to establish a standard for JavaScript.

The Evolution of JavaScript.

The evolution of JavaScript led to **ECMAScript-262 Edition 1**. This standard is actually broad enough: it also covers languages such as **ActionScript**, which powers Adobe's Flash.

Over the years, JavaScript has continued to evolve. A few years ago, the JavaScript family was split between **ECMAScript 3.1** and JavaScript 4. Ultimately JavaScript 4 was abandoned in favor of ECMAScript 3.1, and currently browsers support ECMAScript 5.

The problem is that when you browse through tutorials that are on the Internet, you may find code that's intended to support different versions of JavaScript, and that could be a little bit confusing. But if you keep in mind that this is how the language evolved, you can watch out for some of these landmines, and be aware that the more recent the tutorial, the more likely it is to support the environment that you're trying to develop for.

Our Focus in this Course

We're going to discuss what a variable is, how statements work, how functions work. I will give you the basis that you need in order to understand how to use and interpret JavaScript code. In this course, you won't be copying and pasting examples. You're going to be writing code from scratch. So let's get started!