

JavaScript 2015 (I). Introducción.



¿Qué es JavaScript 2015? Bueno. Empecemos por el principio. Parto de la base de que eres desarrollador web (ya sea profesional o aficionado) y que estás familiarizado con JavaScript. Seguro que estás harto de usarlo. Puede que sólo lo uses a nivel básico, o que seas un avezado escritor de códigos de script del lado del cliente. Da igual. La especificación de JavaScript ha permanecido (casi) inamovible durante más de cinco años. Eso, en este sector, es inconcebible. Las razones por las que esto ha ocurrido así son, básicamente, tres:

En su origen, la versión 5 de JavaScript era mucho más potente y flexible de lo que la mayoría de la gente cree. Por esta razón, ya se sabía que duraría mucho tiempo.

Las carencias que presentaba el lenguaje en sí eran suplidas, de largo, con el uso de frameworks que han ido apareciendo en el mercado, como Angular, MooTools o jQuery.

Ciertos navegadores (y todos sabemos de cual hablo, ¿verdad, señor Gates?) han marcado siempre una tendencia involucionista al negarse a adaptar su operativa a las normalizaciones y especificaciones aprobadas por el consorcio. Dada la cuota de mercado que este navegador llegó a tener en su momento, era muy difícil que ningún producto dependiente de un navegador evolucionara. afortunadamente, su cuota de mercado es cada vez menor, lo que les obliga a adaptar más rápidamente las normalizaciones (dentro de lo que es "normal" para explorer, claro).

JAVASCRIPT 2015

Se le conoce como JavaScript 2015, JavaScript 6, ECMAScript 6, JS6, ES6... Da igual. Nos referimos a la versión actual de JavaScript. Si quieres conocer un poco de la historia de este lenguaje, te sugiero que visites [esta página](#). En esta serie de artículos vamos a conocer las principales novedades que nos aporta la actual versión del lenguaje.

Por supuesto, no vamos a hacer un estudio detallado de todas y cada una de las características de JavaScript. En primer lugar, porque la actual versión incorpora, desde luego, todas las funcionalidades que ya tenía JavaScript 5. Nosotros nos centraremos en las novedades más relevantes. De otro modo, tendríamos que escribir un hermoso par de volúmenes, lo que queda fuera del propósito de este blog.

Lo primero es que le echemos un vistazo a la página oficial de ECMAScript, en la URL <http://ecma-international.org/>. Está en inglés, por supuesto, pero creo que todos los que andamos en este oficio estamos acostumbrados a eso. En esta web podrás encontrar varias cosas interesantes. Por ejemplo, en lo que a documentación se refiere, puedes visitar <https://www.ecma-international.org/publications/standards/Ecma-262.htm>. En esta página encontrarás dos ediciones de la última revisión del manual (la 7, de Junio de 2016, en el momento de escribir esto): una en pdf para descargar y otra en HTML para revisar online.

¿COMPATIBLE?

La compatibilidad con los navegadores o, mejor dicho, la compatibilidad de los navegadores con las nuevas características, determinará, en gran medida, la aceptación de esta versión por parte de los desarrolladores. A día de hoy, los navegadores más extendidos, como Chrome, Firefox u Opera aceptan la mayoría de las nuevas características. La tendencia involucionista, cómo siempre, la marca Explorer. Sólo la versión 11 acepta algunas de las nuevas características de JS6. Por lo tanto, si tu público objetivo se centra en versiones anteriores de este navegador (vale, lo llamaremos así), deberás retrasar un poco la implementación de lo que comentaremos en estos artículos.

En todo caso, si alguna sentencia te ofrece dudas, visita <http://caniuse.com/>, donde podrás verificar que navegadores aceptan cada

una de las nuevas sentencias que veremos.

JAVASCRIPT EXPERIMENTAL

Una alternativa para probar todas las novedades de JS6 es activar en tu navegador la funcionalidad de javascript experimental. En Chrome, por ejemplo, se activa tecleando, en la barra de direcciones, `chrome://flags` para acceder a la página que permite configurar características avanzadas del navegador. Después, busca [javascript experimental](#), y pulsa el enlace de habilitar. A continuación pulsa un botón que te aparece en la parte inferior para reiniciar el navegador con el javascript experimental habilitado.

No obstante, ten en cuenta que esto puede servirte para poder "jugar" con JS6. Sin embargo, cuando desarrolles, no deberás tener esta función habilitada. Esto se debe a que los visitantes de tu página, evidentemente, no la tienen y, por supuesto, debes desarrollar algo que les funcione a ellos igual que a ti.

Todas estas cuestiones han hecho que, cuando salió JS6 en Junio de 2015, fuera todo más teórico que práctico. A día de hoy, se puede hablar de estos conceptos como algo utilizable (salvo IE 10 y anteriores, claro).