

INSTITUTO SUPERIOR TECNOLÓGICO DEL SUR

CARRERA

DISEÑO Y PROGRAMACIÓN WEB

LENGUAJE WEB 2

“Evolución de JQuery”

PROFESOR(A) : Amado Cerpa Juan Andrés

ALUMNO : Vilca Apaza Christian

SEMESTRE : V

29/03/2021

Índice

¿Qué es JQuery?	3
Historia	3
Manipulación y cruce de DOM	3
Manejo de eventos	3
Ajax	4
Versiones y sus cambios.....	4
Origen y ventajas de jQuery sobre JavaScript.....	7
Desventajas de jQuery sobre JavaScript.....	7
¿Vale la pena seguir desarrollando en jQuery?	8
Conclusión	9
Bibliografía	10

¿Qué es JQuery?

JQuery es una biblioteca de JavaScript rápida, pequeña y rica en funciones. Hace que cosas como el recorrido y la manipulación de documentos HTML, el manejo de eventos, la animación y Ajax sean mucho más simples con una API fácil de usar que funciona en una multitud de navegadores. Con una combinación de versatilidad y extensibilidad, jQuery ha cambiado la forma en que millones de personas escriben JavaScript.

Historia

JQuery fue creado originalmente en enero de 2006 en BarCamp NYC por John Resig, influenciado por la anterior biblioteca cssQuery de Dean Edwards. Actualmente lo mantiene un equipo de desarrolladores liderado por Timmy Willison (con el motor de selección de jQuery, Sizzle, liderado por Richard Gibson).

JQuery fue originalmente licenciada bajo el CC BY-SA 2.5 , y se le otorgó la licencia MIT en 2006. A fines de 2006, tenía doble licencia bajo las licencias GPL y MIT. Como esto llevó a cierta confusión, en 2012 se eliminó la GPL y ahora solo está autorizada bajo la licencia MIT.

En 2015, JQuery se utilizó en el 63% del millón de sitios web principales (según BuiltWith) y en el 17% de todos los sitios web de Internet.¹⁵ A partir de junio de 2018, JQuery se usa en el 73% del millón de sitios web más importantes y en el 22,4% de todos los sitios web (según BuiltWith).

Desde el año 2019, JQuery se usa en el 79,47% del millón de sitios web más importantes y en el 86% de todos los sitios web (según BuiltWith), siendo la biblioteca de JavaScript más popular hasta el momento.

Manipulación y cruce de DOM

Obtenga el <button> elemento con la clase 'continue' y cambie su HTML a 'Next Step ...'!

```
$( "button.continue" ).html( "Next Step..." )
```

Manejo de eventos

Muestra el #banner-message elemento que está oculto display:none en su CSS cuando #button-containerse hace clic en cualquier botón .

```
var hiddenBox = $( "#banner-message" );  
$( "#button-container button" ).on( "click", function( event ) {  
    hiddenBox.show();  
});
```

Ajax

Llame a un script local en el servidor /api/getWeather con el parámetro de consulta zipcode=97201 y reemplace el #weather-temphtml del elemento con el texto devuelto.

```
$.ajax({  
  url: "/api/getWeather",  
  data: {  
    zipcode: 97201  
  },  
  success: function( result ) {  
    $( "#weather-temp" ).html( "<strong>" + result + "</strong> degrees" );  
  }  
});
```

Versiones y sus cambios

Fecha de versión	Número de versión	Notas adicionales
26 de agosto de 2006	1.0	Primera versión estable
31 de agosto de 2006	1.0.1	
9 de octubre de 2006	1.0.2	
27 de octubre de 2006	1.0.3	
12 de diciembre de 2006	1.0.4	Último arreglo de la versión 1.0
14 de enero de 2007	1.1	
22 de enero de 2007	1.1.1	
27 de febrero de 2007	1.1.2	
1 de julio de 2007	1.1.3	
5 de julio de 2007	1.1.3.1	
24 de agosto de 2007	1.1.4	Último arreglo de la versión 1.1
10 de septiembre de 2007	1.2	
16 de septiembre de 2007	1.2.1	
15 de enero de 2008	1.2.2	
8 de febrero de 2008	1.2.3	
19 de mayo de 2008	1.2.4	
21 de mayo de 2008	1.2.5	Arreglo por mala construcción de 1.2.4
24 de mayo de 2008	1.2.6	Último arreglo de la versión 1.2

14 de enero de 2009	1.3	Se introduce el motor de selección de Sizzle en el núcleo
21 de enero de 2009	1.3.1	
20 de febrero de 2009	1.3.2	Último arreglo de la versión 1.3
14 de enero de 2010	1.4	Crean una página web específica para jQuery 1.4
25 de enero de 2010	1.4.1	
19 de febrero de 2010	1.4.2	
16 de octubre de 2010	1.4.3	
11 de noviembre de 2010	1.4.4	Último arreglo de la versión 1.4
31 de enero de 2011	1.5	Re-escritura del módulo AJAX
24 de febrero de 2011	1.5.1	
31 de marzo de 2011	1.5.2	Último arreglo de la versión 1.5
3 de mayo de 2011	1.6	
12 de mayo de 2011	1.6.1	
30 de junio de 2011	1.6.2	
1 de septiembre de 2011	1.6.3	
12 de septiembre de 2011	1.6.4	
3 de noviembre de 2011	1.7	Añadidas las funciones .on() y .off() Mejor soporte de HTML5 para IE6/7/8 Soporte para AMD (Asynchronous Module Definition)
21 de noviembre de 2011	1.7.1	
21 de marzo de 2012	1.7.2	
9 de agosto de 2012	1.8	
30 de agosto de 2012	1.8.1	
20 de septiembre de 2012	1.8.2	
13 de noviembre de 2012	1.8.3	
1 de enero de 2013	1.9	Simplificación de la API Mejorado el soporte para CSS3 Soporte para Source map26

		Solucionados varios bugs.
30 de mayo de 2013	1.10.1	Soluciona bug grave de la versión 1.9
24 de enero de 2014	1.11	Velocidad de procesamiento aumentada Cambios a nivel de publicación de JQuery Modularidad definido por AMD Eliminados layouts forzados
20 de mayo de 2016	1.12	Cambios a nivel de publicación de JQuery Errores de la versión anterior corregidos.
18 de abril de 2013	2.0	Primer lanzamiento de las versiones 2.X Pérdida de compatibilidad con Internet Explorer (6, 7 ,8) Reducción de tamaño en comparación con la versión 1.9.1
24 de enero de 2014	2.1	Primera actualización de las versiones 2.X Menos diseños forzados Gastos generales de arranque más bajos Primera vez que se publica en NPM y Bower
8 de enero de 2016	2.2	Última actualización de las versiones 2.X Mejora de rendimiento Gastos generales de arranque más bajos Nuevas características
9 de junio de 2016	3.0	Primera actualización de las versiones 3.X Compatibilidad con JQuery UI y JQuery Mobile
7 de julio de 2016	3.1.0	Segunda actualización de las versiones 3.X Correcciones de bugs de la versión anterior
22 de septiembre de 2016	3.1.1	Correcciones de bugs de la versión anterior
16 de marzo de 2017	3.2.0	Tercera actualización de las versiones 3.X Soportes para propiedades CSS personalizadas Corrección en errores al ajustar tamaño vertical y horizontal
20 de marzo de 2017	3.2.1	Cambio en el núcleo Refuerzo para las dimensiones verticales y horizontales Cambios en los eventos
19 de enero de 2018	3.3.0	Cuarta actualización de las versiones 3.X Se eliminan varias funciones Se deja de utilizar PhantomJS para ejecutar pruebas
20 de enero de 2018	3.3.1	Cambio para adaptarse a la versión 5.X de NPM
10 de abril de 2019	3.4.0	Quinta actualización de las versiones 3.X Mejora de rendimiento en cambios de alto y ancho

		Cambio en el manejador de evento del input de tipo radio Arreglo de vulnerabilidad Eliminación de los selectores de posición
10 de abril de 2020	3.5.0	Correcciones de seguridad, métodos <code>.even ()</code> y <code>.odd ()</code> , <code>jQuery.trim</code> en desuso
2 de marzo de 2021	3.6.0	Corrección de errores, devuelve JSON cuando hay un error JSONP

Origen y ventajas de jQuery sobre JavaScript

jQuery es una librería que nos permite utilizar ciertas herramientas que facilitan el trabajo a los desarrolladores. No es otro lenguaje de programación, sino que sigue siendo JavaScript, solo que se desarrolló pensando en facilitar ciertas tareas. En su momento, esta librería tuvo bastante aceptación en el mundo web, no solo porque simplificaba muchos procesos o porque enseñó nuevos trucos a los programadores, sino por ventajas como las siguientes:

- Permite hacer operaciones del DOM con pocas líneas de código.
- Tiene una gran cantidad de plugins para descargar.
- Es fácil de aprender y utilizar.
- Tiene soporte para AJAX, lo que permite enviar peticiones al servidor y, en base a la respuesta, modificar una parte de la web sin tener la necesidad de recargar toda la página.

Desventajas de jQuery sobre JavaScript

Después de la aparición de esta librería, JavaScript siguió evolucionando, mientras que jQuery se ha quedado bastante estancado. Esto se debe a que muchos proyectos empezaron a optar por jQuery y después es bastante complicado hacer esa migración a otras librerías o frameworks de JavaScript más modernas, ya que la sintaxis de la librería jQuery no era un estándar ni tampoco se creó un estándar a partir de la misma. Esto generó bastantes complicaciones, que continúan hasta hoy en día, ya que el código jQuery suele ser un código bastante difícil de gestionar, lo que provoca algunas desventajas o problemas como:

- Ofrece muchas funcionalidades que probablemente no se necesiten.
- Ralentiza la carga de nuestra app debido a su peso de 93Kb.
- Gestionar estilos de CSS con jQuery es bastante complicado.

Durante los últimos años, con la misma evolución de JavaScript, han aparecido muchos frameworks y librerías, como React o Angular, que permiten hacer las mismas cosas que jQuery pero siguiendo un estándar y pudiendo mantener un código más limpio.

¿Vale la pena seguir desarrollando en jQuery?

[Joshua Daza Rincon](#), Desarrollador web UI - UX en Globant (2018-presente).

Esta pregunta persigue una respuesta obvia, pero en realidad no lo es tanto, la mayoría que no abordan proyectos grandes acusan de obsoleto JQuery, pero en la realidad agiliza el flujo y estandariza código haciendo más legible a mediano plazo. Claro está que la evolución de frameworks y librerías como Angular, React, Vue, Polymer que brindan posibilidades reactivas en tu proyecto, funcionalidades SPA y amigable a necesidades de funcionales más exigentes. JQuery brinda alternativas que en ES6 ya son resueltas. Pero aun así el uso de esta librería requiere de un tiempo prudente ya que se encuentra presente en varios entornos por defecto (CMS, Plugins, librerías y Sitios web.) cross browsing y eventualmente será una amada pieza en nuestro cada vez más grande museo digital.

Conclusión

Jquery tuvo una gran evolución durante todo su tiempo de uso desde 2006 y aun sigue siendo usado en la mayoría de sitios webs, pero como toda tecnología siempre abra algo mejor en el futuro y en el caso de jquery el mismo javascript es mucho mejor en la actualidad, también tenemos tecnologías nuevas q son mucho mejores como VueJS, Angular, React, Svelte y muchas mas que dan un nuevo gusto al desarrollo y diseño web.

Para mi si es bueno aprender JQuery para proyectos antiguos o para proyectos nuevos que no tenga muchos requisitos.

Bibliografía

<https://jquery.com/>

<https://es.wikipedia.org/wiki/JQuery>

<https://openwebinars.net/blog/origen-de-jquery-y-diferencias-con-javascript/>

<https://es.quora.com/Vale-la-pena-seguir-desarrollando-en-jQuery>

<https://www.todojs.com/10-aniversario-de-jquery/>

<https://en.wikipedia.org/wiki/JQuery>