

Desarrollo de Aplicaciones Web

jQuery



jQuery

[jQuery](#) → librería de funciones escrita en JavaScript

Fue creada por [John Resig](#). Su presentación tuvo lugar en enero de 2006 en el [BarCamp](#) NYC.

Se trata de un software libre y de código abierto que permite:

- Simplificación de la interacción con documentos HTML
- Manipulación del árbol DOM
- Manejo eventos
- Desarrollo de animaciones
- Interacción mediante AJAX en páginas web

Integración con HTML

- Bajar la distribución desde algún [repositorio](#) que lo contenga y añadir:

```
<script type="text/javascript" src="jquery.js"></script>
```

- Referenciar la distribución mediante:

```
<script type="text/javascript" src="http://code.jquery.com/jquery-1.2.2.min.js">  
</script>
```

Selección de Elementos

Proporciona los mecanismos de selección utilizados por CSS y XPath:

Ej.- Combinación de selectores CSS o XPath

`$(“div ul a”)`

- `$(“p”)` → Acceso a todos los párrafos del documento HTML
- `$(“#A”)` → Acceso a todos los elementos con id=“A” del documento HTML
- `$(“.B”)` → Acceso a todos los elementos de la clase class=“b” del documento HTML

Método ready()

El **método ready()** → informa de cuando el DOM del documento HTML esta listo (ready) y entonces ejecuta la función que lleva como argumento.

```
Ej.- $(document).ready( function()
{
    $("p").addClass('highlight');
}
);
```

CSS Código

```
<style type="text/css">
    .highlight {
        font-style: italic;
        background-color: #0f0;
    }
</style>
```

Manejo Contenido

- **Método `text()`** → permite obtener el texto del elemento seleccionado.
- **Método `text(texto)`** → permite modificar el contenido en texto del elemento seleccionado.
- **Método `html()`** → permite obtener el contenido en texto y etiquetas html del elemento seleccionado.
- **Método `html(texto+etiquetas html)`** → permite modificar el contenido en texto y etiquetas html del elemento seleccionado.

Manejo Contenido

Ej.- Creando contenido mediante

- Etiquetas HTML:

```
var txt("<p>Hola Mundo.</p>");
```

- Métodos DOM:

```
var txt=document.createElement("p");  
txt.innerHTML="Hola Mundo.";
```

- jQuery:

```
var txt=$("<p></p>").text("Hola Mundo.");
```

Arrays

Método `each()` → permite recorrer los elementos de un array.

```
$.each(array, function(indice, valor)
{
    -----
    -----
}
);
```


Arrays

Método `map()` → permite recorrer los elementos de un array, definiendo una función que retorna un nuevo array.

```
newArray=$.map(array, function(valor, indice)
{
    -----
    -----
    return(nuevoArray);
}
);
```

Arrays

Método `grep()` → permite recorrer los elementos de un array, definiendo una función que devuelve el resultado de filtrar los elementos del array, mediante una función definida.

```
newArray=$.grep(array, function(valor, indice)
{
    -----
    -----
    return(arrayFiltrado);
},
booleano);
```

Nota.- booleano → true o false

Eventos

Método `bind()` → permite asociar un evento determinado a un elemento concreto de la página.

```
$('.algo').bind(tipoEvento, datoPasado, manejadorEvento()  
{  
    -----  
    -----  
});
```

Eventos

- **Método** `click(manejadorEvento())` → asocia `manejadorEvento` al evento click.
- **Método** `dblclick(manejadorEvento())` → asocia `manejadorEvento` al evento doble click.
- **Método** `trigger(tipoEvento)` → dispara el evento asociado al ser ejecutada.
- **Método** `unbind(tipoEvento, manejadorEvento())` → elimina la asociación existente entre un evento y un elemento.

Eventos

- **Método** `mousedown(manejadorEvento())` → asocia el *manejadorEvento* al evento pulsar ratón.
- **Método** `mouseup(manejadorEvento())` → asocia el *manejadorEvento* al evento soltar botón del ratón.
- **Método** `mouseover(manejadorEvento())` → asocia el *manejadorEvento* a pasar por encima de un elemento.
- **Método** `focus(manejadorEvento())` → asocia el *manejadorEvento* asociado a tener elemento de texto activo.
- **Método** `blur(manejadorEvento())` → asocia el *manejadorEvento* asociado a que el elemento de texto activo deje de estarlo.

Nodos

- **Método `parent()`** → devuelve el nodo padre del elemento seleccionado.
- **Método `child()`** → devuelve el nodo hijo inmediato del elemento seleccionado.
- **Método `prepend()`** → inserta el contenido especificado al inicio del elemento seleccionado y retorna un objeto jQuery.
- **Método `prependTo()`** → inserta el elemento seleccionado al comienzo del objetivo seleccionado (elemento HTML u objeto jQuery) y retorna un objeto jQuery.
- **Método `clone()`** → añade un nodo DOM que es copia del elemento seleccionado y devuelve un objeto jQuery.

Nodos

Otros Métodos:

- `append()`
- `appendTo()`
- `before()`
- `insertBefore()`
- `after()`
- `insertAfter()`
- `remove()`
- `empty()`

Visibilidad

- **Método** `show(velocidad, funcionAlternativa())` → muestra el contenido seleccionado en respuesta a un click de ratón.
- **Método** `hide(velocidad, funcionAlternativa())` → oculta el contenido seleccionado en respuesta a un click de ratón.
- **Método** `toggle(velocidad, funcionAlternativa())` → oculta/muestra el contenido seleccionado en respuesta a un click de ratón.
- **Método** `animate({CSS nuevo}, velocidad, funcion())` → ejecuta el cambio en el contenido seleccionado desde el CSS actual hasta el nuevo CSS, a la velocidad indicada.

Ej. -

```
$( "p" ).toggle(1000);
```