

Introduzione a jQuery



Emiliano Castellina
Dipartimento di Automatica e Informatica
Politecnico di Torino

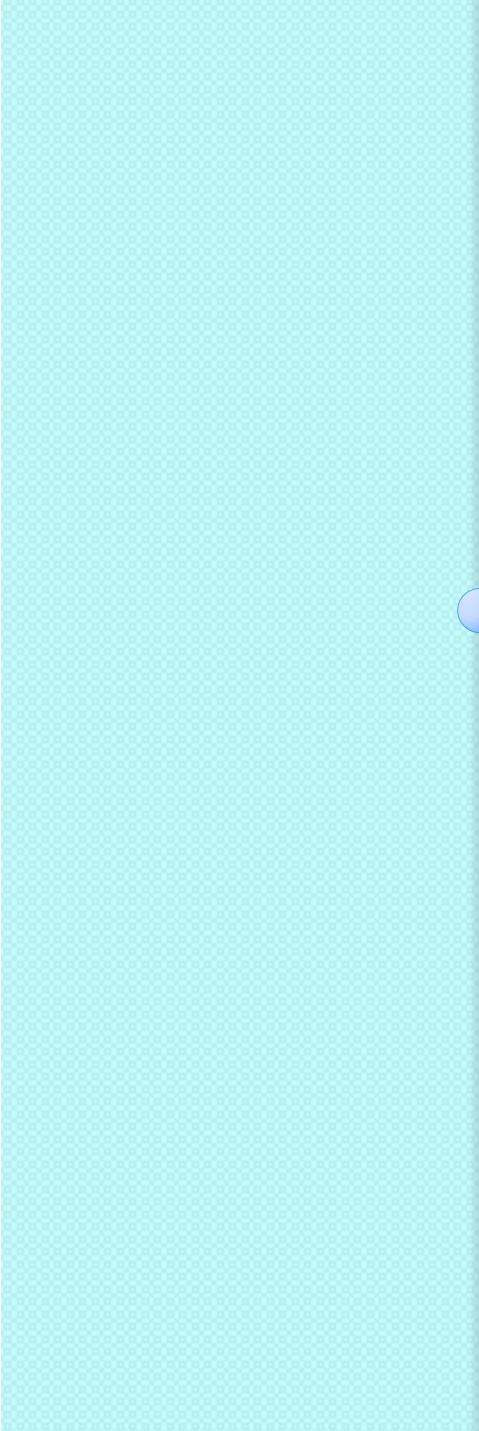


Introduzione a jQuery by Emiliano Castellina is licensed under a Creative Commons Attribuzione - Non commerciale - Condividi allo stesso modo 3.0 Unported License.



Sommario

- Introduzione
- Selettori
- Operazioni sugli oggetti selezionati
- Eventi

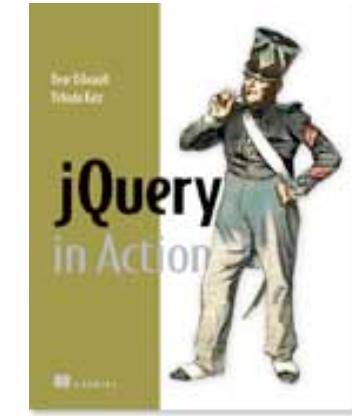


- **INTRODUZIONE**



Libri Consigliati

- **jQuery in Action**
 - Autori: Bear Bibeault and Yehuda Katz
 - Febbraio, 2008 | 376 pagine
 - ISBN: 1933988355
- **jQuery UI 1.7: The User Interface Library for jQuery**
 - Autore: Dan Wellman
 - Novembre 2009 | 392 pagine
 - ISBN : 1847199720



Link utili

- Sito web di jQuery: è possibile scaricare l'ultima versione della libreria, visualizzare la documentazione e tutorial
 - <http://jquery.com>
- Sito web di jQuery User Interface: libreria di componenti grafici di jQuery
 - <http://jqueryui.com/>
- Sito web di Eclipse: ambiente di sviluppo open source che supporta diversi linguaggi di programmazione.
 - <http://www.eclipse.org>
- Aptana: plugin di Eclipse per lo sviluppo di pagine web (html, css, javascript)
 - <http://www.aptana.com/downloads/start>



Perché jQuery?

- Motto: Scrivi Meno, Fai di più
- Facilita la selezione degli elementi del DOM tramite selettori CSS
- Semplice da imparare
- Cross-browser
- Compatibile con altre librerie Javascript

Operazioni Preliminari (1/2)

- Scaricare l'ultima versione della libreria jQuery
 - <http://code.jquery.com/jquery-1.5.1.min.js>
- Importare jQuery nella pagina web

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
    <script src="lib/jquery/jquery-1.5.1.min.js" type="text/javascript" language="JavaScript"></script>
```

ATTENZIONE!!

Non utilizzare per l'elemento script la sintassi

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

Non è supportata da molti browser!!

Operazioni Preliminari (2/2)

- Creare uno script che al caricamento della pagina visualizzi una finestra di dialogo che mostri la versione corrente di jquery
 - Aggiungere nell'header

```
<script type="text/javascript" language="JavaScript">
$(document).ready(function(){
    alert("La versione attuale di jQuery e' " + $.jquery);
</script>
```

\$ Scorciatoia per la funzione jQuery

Funzione richiamata quando il documento è completamente caricato

Stringa Versione di jQuery



Oggetto jQuery() \$()

- Permette di selezionare e Creare elementi del DOM
 - Può avere come parametro:
 - Selettore CSS \$("p.evidenziato")
 - Restituisce un array di oggetti, eventualmente vuoto
 - Oggetti javascript \$(this.value)
 - Restituisce un array di oggetti contenente l'elemento passato come parametro
 - Codice HTML \$("<div>Ciao Mondo</div>")
 - Crea un elemento del DOM e restituisce un array di oggetti contenente l'elemento appena creato



SELETTORI



Selettori CSS

`$(elemento)`

- Permette di selezionare gli elementi HTML passati come parametro
 - `$("p")` tutti i paragrafi della pagina
 - `$("a")` tutti i link (e ancore) della pagina
 - `$("a , p")` tutti i link e i paragrafi della pagina



Selettori CSS `$(#idelemento)`

- Permette di selezionare l'elemento della pagina che ha l'id passato come parametro
 - `$("#principale")` elemento della pagina con attributo id="principale"
 - `("div#principale")` div della pagina con attributo id="principale"
 - Essendo l'id univoco all'interno della pagina la selezione del div poteva essere omessa



Selettori CSS

`$(.nomeclasse)`

- Permette di selezionare gli elementi della pagina che hanno la classe passata come parametro
 - `$(".evidenziato")` elementi della pagina che hanno classe evidenziato
 - `("div.evidenziato")` tutti i div della pagina con classe evidenziato



Selettori CSS `$(elemento[attributo])`

- Permette di selezionare gli elementi della pagina che hanno l'attributo passato come parametro
 - `$("img[title]")` tutte le immagini in cui è specificato l'attributo title
 - `$("input[value]")` tutti gli elementi input in cui è specificato l'attributo value



Selettori CSS

`$(elemento[attributo=x])`

- Permette di selezionare gli elementi della pagina che hanno l'attributo passato come parametro uguale al valore X
 - `$("img[title='logo'])` tutte le immagini in cui l'attributo titolo è uguale a 'logo'
 - `$("input[type='radio'])` tutti gli elementi input di tipo radio



Selettori CSS

`$(elemento[attributo^=x])`

- Permette di selezionare gli elementi della pagina che hanno l'attributo passato come parametro che inizia per X
 - `$("img[title^='logo'])"` tutte le immagini in cui l'attributo titolo inizia con 'logo'
 - `$("a[href^='http://'])"` tutti i link con indirizzo non relativo



Selettori CSS

`$(elemento[attributo$=x])`

- Permette di selezionare gli elementi della pagina che hanno l'attributo passato come parametro che finisce per X
 - `$("img[src$='.png'])` tutte le immagini con estensione png
 - `$("a[href$='.pdf'])` tutti i link a file pdf



Selettori CSS

`$(elemento[attributo*=x])`

- Permette di selezionare gli elementi della pagina che hanno l'attributo passato come parametro che contiene la stringa X
 - `$("img[title*='logo'])"` tutte le immagini in cui l'attributo titolo contiene la stringa 'logo'
 - `$("a[href*='polito.it'])"` tutti i link che contengono nell'indirizzo la stringa 'polito.it'



Selettori CSS

`$(padre > figli)`

- Permette di selezionare gli elementi "figli" che sono discendenti diretti dell'elemento "padre"
 - `$(div > ul)` tutte le liste che sono contenute direttamente in un div
 - `$(div > li)` **errato!!!** Selezionerebbe tutti gli elementi di una lista contenuti direttamente in un div. Gli elementi di una lista devono essere contenuti in elementi **ul** o **ol**



Selettori CSS

`$(fratello+sorelle)`

- Permette di selezionare gli elementi "sorelle" che sono direttamente preceduti da un elemento "fratello"

```
$(".A + .B")  
<ul>  
    <li class="A">e1 </li>  
    <li class="B">e2 </li>  
    <li class="B">e3 </li>  
    <li class="A">e4 </li>  
    <li class="B">e5 </li>  
    <li class="C">e4 </li>  
    <li class="B">e5  
        <ul>  
            <li class="B"> e6 </li>  
        </ul>  
    </li>  
</ul>
```



Selettori CSS

`$(fratello~sorelle)`

- Permette di selezionare gli elementi "sorelle" che sono preceduti da un elemento "fratello"

```
$(".A ~ .B")  


- e1
- e2
- e3
- e4
- e5
- e4
- e5
  - e6

```

Selettori CSS

`$(padri:has(discendenti))`

- Permette di selezionare gli elementi "padri" che hanno almeno un "discendente" passato come parametro
 - `$("ul:has(a)")` tutti gli ul che hanno come discendente almeno un link
 - `$("div:has(img)")` tutti i div che hanno come discendente almeno un'immagine



Selettori di Posizione

`$(:first) $(:last)`

- Permettono di selezionare il primo `(:first)` e ultimo `(:last)` elemento di un insieme
 - `$("a:first")` il primo link della pagina
 - `$("p.evidenziato:first")` il primo paragrafo con class "evidenziato"
 - `$("img[src*=logo]:last")` ultima immagine che contenga nell'indirizzo la parola logo



Selettori di Posizione

`$(:first-child) $(:last-child)`

- Permettono di selezionare il primo `(:first-child)` e l'ultimo `(:last-child)` elemento discendente
 - `$("div:first-child")` il primo elemento contenuto in ogni **div**
 - `$("p.evidenziato:first-child")` il primo elemento contenuto in ogni paragrafo con class "evidenziato"
 - `$("li:last-child")` ultimo elemento contenuto in ogni **li**



Selettori di Posizione

`$(:even) $(:odd)`

- Permettono di selezionare gli elementi in posizioni pari `(:even)` e dispari `(:odd)`
 - `$("table.zebra tr:even")` le righe in posizione pari delle tabelle che hanno class "zebra"
 - `$("li:odd")` i list item di posizione dispari

Questo selettore utilizza come indice iniziale lo 0



Selettori di Posizione

`$(:nth-child(n))` `$(:nth-child(even|odd))`

- Permettono di selezionare gli elementi discendenti in posizioni uguale a n (`:nth-child(n)`) , dispari (`:nth-child(odd)`) o pari (`:nth-child(even)`)
 - `$("tr:nth-child(4)")` il quarto elemento contenuto in ogni **tr**
 - `$("tr:nth-childed(even)")` gli elementi in posizione pari in ogni **tr**
 - `$("tr:nth-childed(odd)")` gli elementi in posizione dispari in ogni **tr**



Selettori di Posizione

`$(:eq(n))` `$(:lt(n))` `$(:gt(n))`

- Permettono di selezionare gli elementi in posizioni uguale a n `(:eq(n))`, maggiore di n `(:gt(n))` o minore di n `(:lt(n))`
 - `$("tr:eq(4)")` il quinto elemento **tr**
 - `$("tr:lt(4)")` i primi cinque **tr**
 - `$("tr:gt(4)")` dal sesto **tr** in poi

Questo selettore utilizza come indice iniziale lo 0



Selettori Custom

- `:button` seleziona tutti i bottoni
- `:checkbox` seleziona tutte le checkbox
- `:radio` seleziona tutti i radio button
- `:checked` seleziona le check box e i radio button che sono selezionati
- `:disabled` , `:enabled` elementi di input disabilitati o abilitati
- `:not(filter)` negazione di un selettore
 - `$("li:not(.evidenziato)")` gli elementi li che non hanno classe evidenziato
- `:hidden` elementi nascosti
- `:visible` elementi visibili
- `:text` seleziona textbox
- `:filter(selettore)` applica un ulteriore selettore all'insieme selezionato
- `:contains(cerca)` seleziona gli elementi che contengono il testo "cerca"



- **OPERAZIONI SU OGGETTI
SELEZIONATI/CREATI**



Operazione su gli oggetti selezionati/creati

Navigazione del DOM

- `$(selettore).children([selettore])` **figli diretti dell'insieme selezionato**
 - È possibile filtrare i figli con un selettore
- `$(selettore).find(selettore)` **discendenti dell'insieme selezionato**
- `$(selettore).next()` elemento successivo a quello selezionato
- `$(selettore).parent([selettore])` padre dell'elemento selezionato
- `$(selettore).parents([selettore])`



Operazione su gli oggetti selezionati/creati

Manipolazione del DOM

- Inserimento di elementi
 - after, append, before, prepend, html, text, wrap, clone, value
- Cancellazione
 - empty, remove, removeAttr
- Modifica Stile
 - attr, addClass, removeClass, toggleClass, css, hide, show, toggle



Operazione su gli oggetti selezionati/creati

Esempi Manipolazione del DOM

```
$("<li>").text("testo lista").insertAfter("li:first");
//inserisce un elemento li appena creato dopo il primo li del
documento

$("div#principale").append("<p>dentro il div</p>");
//inserisce un paragrafo dentro il div che ha id principale

$("input#email").value()
//valore della input box con id email

$("span").wrap("div")
//crea un div che contiene tutti gli elementi span della pagina

$("p.evidenziato").text("Sono evidenziato");
//imposta il testo dei paragrafi di classe evidenziano con la
stringa passata come parametro

$("img.logo").css("border","solid orange 3px");
//cambia lo stile a tutte le immagini di classe logo
```



Operazione su gli oggetti selezionati/creati

`$(selettore).each()`

- Permette di scorrere il vettore di elementi selezionati e di eseguire una funzione per ogni elemento

```
$(img).each(function(i){  
    $(this).attr("alt","immagine "+i);  
})
```

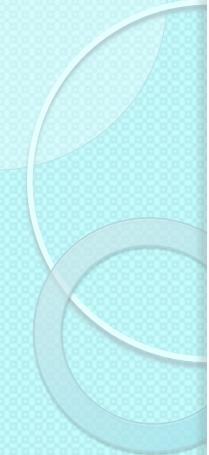


Operazione su gli oggetti selezionati/creati

Concatenazione operazioni

- E' possibile concatenare più azione su uno stesso selettore grazie al fatto che molte operazioni non alterano l'insieme di oggetti restituiti dal selettore.

```
$(.invisibile).hide().addClass("sparito")
//nasconde tutti gli elementi di classe invisibile e aggiunge
la classe sparito
$("<img>").attr("src","img/test.jpg").addClass("logo")
.appendTo("div#principale")
//crea un immagine con attributo src="img/test.jpg", di classe
logo e la appende al div con id principale
```

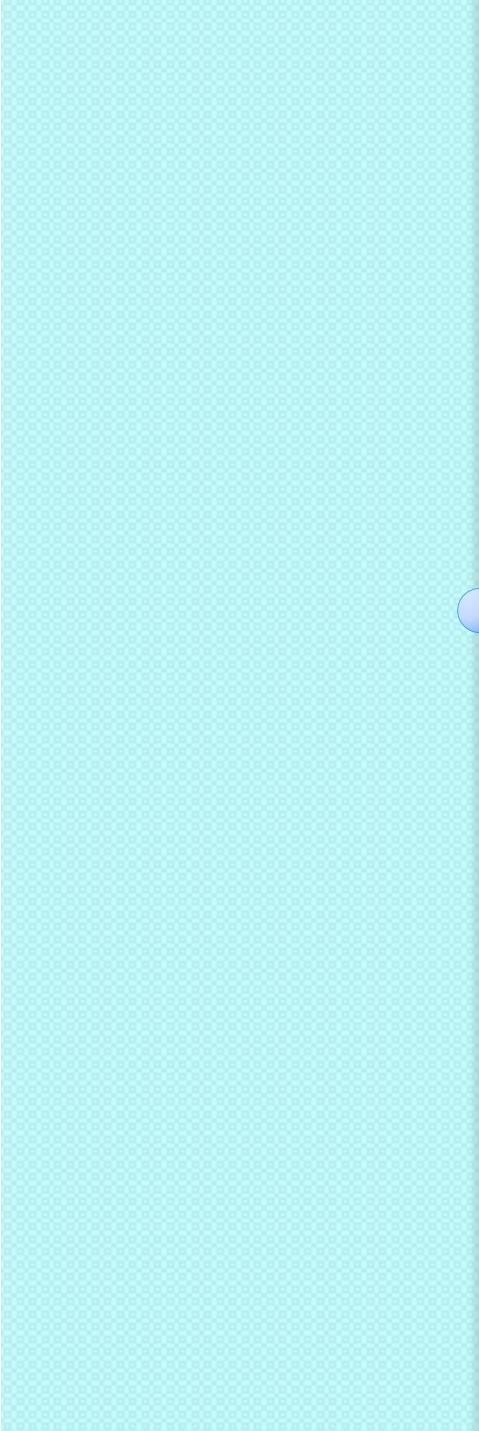


Operazione su gli oggetti selezionati/creati

Concatenazione Avanzata .end()

- Alcune funzioni alterano l'insiemi di oggetti restituiti dal selettore per cui occorre prestare attenzione!

- | | |
|--|---|
| 1. <code>\$(div p:first)</code> | 1. Seleziona il primo paragrafo di ogni DIV |
| 2. <code>.addClass("primoParagrafo")</code> | 2. Aggiunge la classe primoParagrafo |
| 3. <code>.find("a:first")</code> | 3. Seleziona il primo link del primoParagrafo di ogni div |
| 4. <code>.text("primo link")</code> | 4. Modifica il testo in "primo link" |
| 5. <code>.end()</code> | 5. Ritorna alla selezione precedente (primo paragrafo di ogni div) |
| 6. <code>.css("border", "solid 5px");</code> | 6. Modifica lo stile del bordo |



- **GESTIONE EVENTI**

Eventi supportati



Tipologia di Eventi Javascript

EVENTO	DESCRIZIONE
blur, focus	Inviati ad un elemento quando rispettivamente perde il focus od ottiene il focus.
focusin, focusout	Inviato ad un elemento se esso o un suo discendente rispettivamente ottiene o perde il focus
load	Inviato ad un elemento quando esso e tutti i suoi discendenti sono stati completamente caricati
resize	Inviato all'elemento window quando la finestra del browser ha cambiato dimensioni
scroll	Inviato ad un elemento quando l'utente ha effettuato lo scroll in un differente punto dell'elemento stesso
unload	Inviato all'oggetto window quando l'utente naviga fuori dalla pagina (chiusura del browser, click su un link, a volte anche il semplice refresh)
click, dblclick	Inviati ad un elemento quando il mouse è sopra di esso e viene effettuato un click o un doppio click
mousedown, mouseup	Inviati ad un elemento quando il mouse è sopra di esso e viene rispettivamente premuto o rilasciato il bottone del mouse
mouseover, mouseout, mousemove	Eventi inviati all'elemento in cui il puntatore del mouse entra (mouseover), in cui esce (mouseout) o in cui si sta muovendo (mousemove)
keydown, keypress, keyup	Eventi inviati quando un tasto viene premuto (keydown) è stato rilasciato (keyup) o è stato premuto (keypress)
select	Inviato ad un elemento quando viene selezionato del test all'interno di esso. Questo evento è limitato agli elementi <input type="text" /> e <textarea>
change	Evento inviato ad un elemento che ha cambiato il proprio valore. Limitato a <select> <input> <textarea>
submit	Elemento inviato quando l'utente tenta di fare il submit di un form

bind

```
$(selettore).bind(evento,[dati],function(evento){...})
```

Permette di collegare gli eventi ad un selettore e di eseguire la funzione associata

```
$(":text").bind('focusin', function(){
    this.addClass('inserimento');
}).bind('focusout',function(){
    this.removeClass('inserimento');
})
```

Per ogni casella di testo aggiunge la classe inserimento quando il focus è sull'oggetto e rimuove la stessa classe quando il focus non è più dell'oggetto

Sintassi abbreviata eventi

```
$(<selettore>).bind(<evento>, [<dati>], function(<evento>) {...})
```

```
$(<selettore>).<evento>([<dati>], function(<evento>) {...})
```

```
$("li").click(function(){$(this).hide()})
$("#mostraNascosti").mouseover(function(){
    $("li:hidden").show();
})

//nasconde i listitem al click su di essi
//quando il mouse passa sopra l'elemento con id="mostraNascondi" gli elementi li nascosti vengano visualizzati
```

```
//esempio passaggio dati alle funzioni di callback degli eventi
var messaggio="ciao mondo"
$("li").click({msg:messaggio},function(evento){
    alert(evento.data.msg)));

```

Proprietà Oggetto evento

Le funzioni di callback (richiamate al verificarsi di ogni evento) hanno come parametro un oggetto che descrive l'evento occorso.

proprietà	Descrizione
.currentTarget	Elemento del DOM che ha <i>rilanciato</i> l'evento
.data	I dati opzionali passati dalla funzione bind
.pageX	Ascissa della posizione del mouse
.pageY	Ordinata della posizione del mouse
.target	Elemento del DOM in cui è stato generato l'evento
.timeStamp	Quando si è verificato un evento (millisecondi passati dal 1 gennaio 1970)
.type	Describe la natura dell'evento
.which	Nel caso di eventi scatenati da mouse o tastiera indica (in codice) quale bottone o tasto è stato premuto. Per avere informazioni sul carattere inserito si utilizza .keyCode

Funzioni Oggetto evento

L'oggetto evento offre alcuni utili metodi

Metodo	Descrizione
.preventDefault()	Non fa eseguire la funzione di default legata all'evento. Es. disattivare tutti i link di una pagina: <code>\$("a").bind('click',function(evento){ evento.preventDefault(); })</code>
.stopPropagation()	Impedisce l'inoltro dell'evento ad ogni elemento padre
.stopImmediatePropagation()	Impedisce l'inoltro dell'evento ad ogni altro elemento del DOM

Lanciare eventi

- E' possibile tramite jQuery lanciare eventi, sia standard che personalizzati

```
$(selettore).trigger(evento, [dati])
```

Esempio

```
$("a#link").trigger('click')
```

- invia un click all'ancora con id uguale a link

- Sintassi abbreviata

```
$(selettore).evento([dati])
```

Esempio

```
 $("a#link").click()
```