

Jquery

- Jquery
 - Jquery è un Framework
 - PRINCIPALI CARATTERISTICHE
 - Utilizziamo la libreria javascript jQuery per:
 - Partenza del codice quando il documento (il DOM) è pronto
 - INCLUDERE JQUERY NELLE PAGINE WEB
 - "Filosofia Jquery"
 - Selezionare gli elementi
 - selettori css 2 / css 3
 - jQuery supporta poi alcuni selettori custom:
 - Accedere e modificare le classi CSS
 - Gestire gli stili Css con Jquery
 - Aggiungere e rimuovere contenuto e inserire elementi
 - Ottenere l'elemento html
 - Ottenere solo il testo contenuto nel tag
 - Cambiare l'HTML e/o il Testo
 - Far seguire (appendere) del contenuto ad un elemento
 - Altri comandi
 - Gli Eventi: intercettare il click del mouse con jQuery
 - Mostrare e nascondere elementi con jQuery
 - Animate
 - Oggetto jQuery() \$()
 - Selettori TAG HTML
 - Selettori di ID
 - Selettori di CLASSE
 - Selettori per ATTRIBUTI
 - Selettori GERARCHICI
 - Selettori di Posizione
 - Selettori Cardinali di Posizione

- Selettori Custom
- Operazione sugli oggetti selezionati/creati
- Operazione sugli oggetti selezionati/creati
- Operazione sugli oggetti selezionati/creati
- Operazione sugli oggetti selezionati/creati
- Operazione sugli oggetti selezionati/creati
- Eventi supportati
- Tipologia di Eventi Javascript
- on
- Sintassi abbreviata eventi
- Proprietà Oggetto evento
- Funzioni Oggetto evento
- Lanciare eventi

Jquery è un Framework

- write less, do more

Jquery è un Framework (una collezione o libreria di codice che agevola la risoluzione di una classe di problemi) che permette di interagire , con semplicità e rapidità, sul DOM (Document Object Model) * di una pagina HTML

PRINCIPALI CARATTERISTICHE

- Cross browser (supportato dai principali browser)
- Grande community = Grande disponibilità di plugin
- Leggero (opinabile...)
- Documentazione chiara e completa
- Semplice
- Utilizza selettori CSS (anche Css3)
- Sintassi compatta, permette di ridurre la quantità di codice necessario rispetto all'uso di Javascript
- Può gestire tutti gli elementi che compongono una pagina web (div, immagini, span, p ...)

Utilizziamo la libreria javascript jQuery per:

- Partenza del codice quando il documento (il DOM) è pronto
- Selezionare gli elementi
 - selettori supportati: CSS2, CSS3, custom
- Accedere e modificare le classi CSS
- Gestire gli stili CSS
- Aggiungere e rimuovere contenuto e inserire elementi
- Ottenere l'elemento html
- Ottenere il testo contenuto nell'elemento
- Cambiare l'HTML e/o il testo contenuto
- Aggiungere nuovo contenuto ad un elemento esistente
- Gestire gli eventi
- Mostrare e nascondere elementi
- Aggiungere animazioni ed effetti
- Gestire le chiamate asincrone (AJAX)
- Creare e processare dati inJSON

Partenza del codice quando il documento (il DOM) è pronto

```
$(document).ready(function() {  
  //codice da eseguire al caricamento del documento  
});
```

INCLUDERE JQUERY NELLE PAGINE WEB

1. Scarichiamo l'ultima versione di JQuery dal sito <http://jquery.com/> il file avrà un nome nella forma jquery-x.x.-min.js
2. Per usarlo nelle nostre pagine web dobbiamo linkarlo attraverso il comando `<script>...</script>` nella sezione `<head />` della nostra pagina web

```
<html>
<head>
<script src="percorso/al/file/jquery-x.x.js">
</script>
</head>
<body> ... </body>
</html>
```

"Filosofia JQuery"

```
//Cerca questo elemento
$("p")
//Fai questa azione
.html("Hello World");
```

```
<html>
<head>
<script src="percorso/jquery.js"></script>
<script type="javascript/text">
$(document).ready(function() {
$("p").html("Hello World");
}
</script>
</head>
<body>
<div>
<p></p>
</div>
</body>
</html>
```

Selezionare gli elementi

La principale caratteristica della funzione `jQuery()` (abbreviata in `$()`) è quella di selezionare elementi HTML per modificarli: basta passare un parametro stringa (tra apici doppi `"` o singoli `'`) come negli esempi che seguono:

```
$("div");  
$("#mioId");  
$(".miaClasse");  
$("#p#mioId");  
$("ul li a.nav");
```

selettori css 2 / css 3

jQuery supporta tutti i selettori anche quelli avanzati e quelli del CSS3

```
$("p > a");  
$("input[type=text]");  
$("a:first");  
$("p:odd");  
$("li:first-child");
```

jQuery supporta poi alcuni selettori custom:

```
$(":animated");  
$(":button"); selects any button elements (inputs or  
buttons)  
$(":radio");  
$(":checkbox"); selects checkboxes  
$(":checked");  
$(":header"); selects header elements (h1, h2, h3,
```

etc.)

Accedere e modificare le classi CSS

Con jQuery è facile aggiungere, scambiare o rimuovere classi CSS: ecco i tre metodi

```
//aggiunge la classe "content" da tutti i <div>
$("div").addClass("content");
//rimuove la classe "content" da tutti i <div>
$("div").removeClass("content");
//fa lo switch della classe "content" da tutti i
<div>
$("div").toggleClass("content");
```

Se un elemento è di una determinata classe allora esegui il codice:

```
if ($("#mioElemento").hasClass("content")) {
    //codice
}
```

Gestire gli stili Css con JQuery

Ecco come modificare i css con jQuery:

```
$("p").css("width", "400px");
$("#mioElemento").css("color", "blue");
$("ul").css("border", "solid 1px #ccc");
```

Aggiungere e rimuovere contenuto e inserire elementi

Ottenere l'elemento html

```
var mioElementHTMLo = $("#mioElemento").html();
```

Ottenere solo il testo contenuto nel tag

```
var mioElementHTMLo = $("#mioElemento").text();
```

Cambiare l'HTML e/o il Testo

```
$("#mioElemento").html("<p>Questo è il nuovo  
contenuto</p>");  
$("#mioElemento").text("Questo è il nuovo  
contenuto."); // questo sostituisce il testo presente
```

Far seguire (appendere) del contenuto ad un elemento

```
$("#mioElemento").append("<p>Questo lo aggiungo.  
</p>");  
$("p").append("<p>Questo pure.</p>"); // Aggiunge  
questo contenuto ad ogni paragrafo
```

Altri comandi

```
appendTo()
```

```
prepend()  
prependTo()  
before()  
insertBefore()  
after()  
insertAfter()
```

- lavorano in modo simile con specifiche peculiarità
- Ulteriori informazioni jQuery.com

Gli Eventi: intercettare il click del mouse con jQuery

```
$("#a").click(function() {  
    // il codice presente viene eseguito onclick!  
});
```

- Il codice dentro `function()` viene eseguito quando un link viene cliccato
- Altri comuni eventi inclusi in jQuery
 - `blur()`
 - `focus()`
 - `hover()`
 - `keydown()`
 - `load()`
 - `mousemove()`
 - `resize()`
 - `scroll()`
 - `submit()`
 - `select()`

Mostrare e nascondere elementi con jQuery


```
$("#mioElemento").hide("slow", function() {  
  // nasconde  
})  
  
//mostra  
$("#mioElemento").show("fast", function() {  
  // qui dentro viene eseguito una volta che l'elemento  
  è stato mostrato  
})  
  
// scambia: se è nascosto mostra e se è visibile  
nasconde  
$("#mioElemento").toggle(1000, function() {  
})
```

```
$("#mioElemento").fadeOut("slow", function() {  
  //fai qualcosa quando termina l'effetto  
})  
  
$("#mioElemento").fadeIn("fast", function() {  
  //fai qualcosa quando termina l'effetto  
})
```

```
$("#mioElemento").fadeTo(2000, 0.4, function() {  
  //fai qualcosa quando termina l'effetto  
})
```

```
$("#mioElemento").slideDown("fast", function() {  
  //fai qualcosa quando termina l'effetto  
})
```

```
$("#mioElemento").slideUp("slow", function() {  
    //fai qualcosa quando termina l'effetto  
})  
  
$("#mioElemento").slideToggle(1000, function() {  
    //fai qualcosa quando termina l'effetto  
})
```

Animate

```
$("#mioElemento").animate(  
    {  
        opacity: .3,  
        width: "500px",  
        height: "700px"  
    }, 2000, function() {  
        // fai qualcosa quando termina l'animazione  
    }  
);
```

Oggetto jQuery() \$()

- jQuery() oppure \$() 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 \$("Ciao Mondo") Crea un elemento del DOM e restituisce un array di oggetti contenente l'elemento appena creato

Selettori TAG HTML

```
$(elemento)
```

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

Selettori di ID

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

Selettori di CLASSE

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

Selettori per ATTRIBUTI

| selettore | descrizione |
|--------------------------------------|--|
| <code>\$(elemento[attributo])</code> | elementi della pagina che hanno l'attributo passato come |

| | parametro |
|--|--|
| <code>\$(elemento[attributo=x])</code> | elementi della pagina che hanno l'attributo passato come parametro uguale al valore x |
| <code>\$(elemento[attributo^=x])</code> | elementi della pagina che hanno l'attributo passato come parametro che inizia per x |
| <code>\$(elemento[attributo\$=x])</code> | elementi della pagina che hanno l'attributo passato come parametro che finisce per x |
| <code>\$(elemento[attributo*=x])</code> | elementi della pagina che hanno l'attributo passato come parametro che contiene la stringa x |
| <code>\$("input[value]")</code> | elementi input in cui è specificato l'attributo value |
| <code>\$("input[type='radio']")</code> | elementi input di tipo radio |
| <code>\$("a[href^='http://']")</code> | i link con indirizzo non relativo |
| <code>\$("a[href*='it.wikipedia.org']")</code> | i link che contengono nell'indirizzo la stringa 'it.wikipedia.org' |
| <code>\$("a[href\$='.pdf']")</code> | i link a file pdf |
| <code>\$("img[title]")</code> | elementi img in cui è specificato l'attributo title |
| <code>\$("img[title='logo']")</code> | le immagini in cui l'attributo titolo è uguale a 'logo' |
| <code>\$("img[src\$='.png']")</code> | le immagini con estensione png |
| <code>\$("img[title^='logo']")</code> | le immagini in cui l'attributo titolo |

inizia con 'logo'

`$("img[title*='logo']")`

le immagini in cui l'attributo titolo contiene la stringa 'logo'

Selettori GERARCHICI

| selettore | descrizione |
|---|--|
| <code>\$(padre > figli)</code> | elementi "figli" che sono discendenti diretti dell'elemento "padre" |
| <code>\$(fratello+sorelle)</code> | |
| <code>\$(".A + .B")</code> | Permette di selezionare gli elementi "sorelle" che sono direttamente preceduti da un elemento "fratello" |
| <code>\$(fratello~sorelle)</code> | |
| <code>\$(".A ~ .B")</code> | Permette di selezionare gli elementi "sorelle" che sono preceduti da un elemento "fratello" |
| <code>\$(padri:has(descendenti))</code> | Permette di selezionare gli elementi "padri" che hanno almeno un "discendente" passato come parametro |
| <code>\$("ul:has(a)")</code> | tutti gli ul che hanno come discendente almeno un link |
| <code>\$("div:has(img)")</code> | tutti i div che hanno come discendente almeno un'immagine |
| <code>\$(div > li)</code> | errato!!! Selezionerebbe tutti gli elementi di una lista contenuti direttamente in un div. |

Selettori di Posizione

| selettore | descrizione |
|---|--|
| Questi selettori sono zero-based | utilizzano come indice iniziale lo 0 |
| <code>\$(:first) \$(:last)</code> | Permettono di selezionare il primo (:first) e ultimo (:last) elemento di un insieme il primo link della pagina |
| <code>\$("p.evidenziato:first")</code> | il primo paragrafo con class "evidenziato" |
| <code>\$("img[src*=logo]:last")</code> | ultima immagine che contenga nell'indirizzo la parola logo |
| <code>\$(:first-child) \$(:last-child)</code> | Per selezionare il primo (:first-child) e l'ultimo (:last-child) elemento discendente |
| <code>\$("p.evidenziato:first-child")</code> | Per selezionare il primo elemento contenuto in ogni paragrafo con class "evidenziato" |
| <code>\$("li:last-child")</code> | Per selezionare l'ultimo elemento contenuto in ogni li |
| <code>\$("div:first-child")</code> | Per selezionare il primo elemento contenuto in ogni div |
| <code>\$(:even) \$(:odd)</code> | Permettono di selezionare gli elementi in posizioni pari (:even) e dispari (:odd) |
| <code>\$("li:odd")</code> | i list item di posizione dispari |
| <code>\$("table.zebra tr:even")</code> | le righe in posizione pari delle tabelle che hanno class "zebra" |

Selettori Cardinali di Posizione

- `$(:nth-child(n))` e `$(:nth-child(even|odd))` Permettono di selezionare gli elementi discendenti in posizioni uguale a n

| selettore | descrizione |
|---------------------------------------|--|
| <code>\$("tr:nth-child(4)")</code> | il quarto elemento contenuto in ogni tr |
| <code>\$("tr:nth-child(even)")</code> | gli elementi in posizione pari in ogni tr |
| <code>\$("tr:nth-child(odd)")</code> | dispari in ogni tr gli elementi in posizione |

- Questi selettori sono zero-based: utilizzano come indice iniziale lo 0
- `$(:eq(n))`, `$(:lt(n))` e `$(:gt(n))` Permettono di selezionare gli elementi in posizioni uguale, maggiore o minore di n

| selettore | descrizione |
|-----------------------------|-----------------------|
| <code>\$("tr:eq(4)")</code> | il quinto elemento tr |
| <code>\$("tr:lt(4)")</code> | i primi cinque tr |
| <code>\$("tr:gt(4)")</code> | dal sesto tr in poi |

Selettori Custom

| selettore | descrizione |
|------------------------|--|
| <code>:button</code> | seleziona tutti i bottoni |
| <code>:checkbox</code> | seleziona tutte le checkbox |
| <code>:radio</code> | seleziona tutti i radio button |
| <code>:checked</code> | seleziona le check box e i radio button che sono selezionati |

| | |
|---|--|
| :disabled, :enabled | elementi di input disabilitati o abilitati |
| \$(<code>"li:not(.evidenziato)"</code>) | 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" |

Operazione sugli oggetti selezionati/creati

Navigazione del DOM

`$(selettore).children([selettore])` figli diretti dell'insieme selezionato

`$(selettore).next()` elemento successivo a quello selezionato

`$(selettore).parent([selettore])` padre dell'elemento selezionato

`$(selettore).parents([selettore])` `$(selettore).find(selettore)`

Operazione sugli oggetti selezionati/creati

| Manipolazione del DOM | metodi disponibili |
|------------------------------|---|
| Inserire elementi | after, append, before, prepend, html, text, wrap, clone, value |
| Cancellare elementi | empty, remove, removeAttr |
| Modifica elementi | attr, addClass, removeClass, toggleClass, css, hide, show, toggle |

Operazione sugli 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("dentro il div");  
//inserisce un paragrafo dentro il div che ha id  
principale
```

```
$("#input#email").val()  
//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
```

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

```
$(selettore).each()
```

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

Operazione sugli oggetti selezionati/creati

- Concatenazione operazioni
- E' possibile concatenare più azioni su uno stesso selettore,
- se le operazioni precedenti non alterano l'insieme di oggetti restituiti dal selettore.
- nasconde tutti gli elementi di classe invisibile e aggiunge la classe sparito

```
$(.invisibile).hide().addClass("sparito")
```

- crea un immagine con attributo src="img/test.jpg", di classe logo e la appende al div con id principale

```
$("").attr("src","img/test.jpg").addClass("logo").appendTo("div#principale")
```

Operazione sugli oggetti selezionati/creati

- Concatenazione Avanzata .end()
- NB: alcune funzioni alterano l'insiemi di oggetti restituiti dal selettore

| selettore | descrizione |
|--|--|
| <code>\$(div p:first)</code> | Seleziona il primo paragrafo di ogni DIV |
| <code>.addClass("primoParagrafo")</code> | Aggiunge la classe primoParagrafo |

| | |
|----------------------------------|---|
| <code>.find("a:first")</code> | Seleziona il primo link del primo Paragrafo di ogni div |
| <code>.text("primo link")</code> | Modifica il testo in "primo link" |

Ritorna alla selezione precedente (primo `.css("border","solid 5px");`|Modifica lo stile del bordo

Eventi supportati

- dbleclick
- load
- unload
- submit
- focus
- blur
- change
- select
- resize
- keydown
- keypress
- keyup
- scroll
- mousemove
- mousedown
- mouseup
- mouseover
- error

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, dbclick | 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. |
| change | Evento inviato ad un elemento che ha cambiato il proprio valore. |
| submit | Evento inviato quando l'utente tenta di fare il submit di un form |

on

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

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

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

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

```
//nasconde i listitem al click su di essi
```

```
$( "li" ).click( function() { $( this ).hide() } );

$( "#mostraNascosti" ).mouseover( function() {
    //quando il mouse passa sopra l'elemento con
    id="mostraNascondi" gli elementi li nascosti vengano
    visualizzati
    $( "li:hidden" ).show() ;
} );

//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 rilanciato l'evento |
| .data | I dati opzionali passati dalla funzione on |
| .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 | Descrive la natura dell'evento |

Nel caso di eventi scatenati da mouse o tastiera

| | |
|----------|---|
| .which | indica (in codice) quale bottone o tasto è stato premuto. |
| .keyCode | Per avere informazioni sul carattere inserito si utilizza |

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:

```
$( "a" ).on( 'click', function( evento ) {  
    evento.preventDefault();  
});
```

- **.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()**

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