

# Innamorarsi di jQuery

Finalmente potete utilizzare segmenti di codice per scrivere un HTML semplice e semantico, e potete potenziarlo con un CSS da esperti per trasformare le vostre idee di progettazione in fantastici siti web in grado di affascinare gli utenti. Avete però compreso che un design brillante e un codice HTML impeccabile, da soli, non bastano per cercare di creare il Facebook o il Twitter del futuro. Qual è, quindi, la tessera mancante del puzzle del front end?

È JavaScript, il linguaggio di scripting considerato a lungo la pecora nera delle applicazioni di sviluppo web. JavaScript vi consente di aggiungere comportamenti complessi, interazioni sofisticate e ulteriore vivacità al vostro sito. Per conquistare questo gigante dormiente, dovete mettervi d'impegno e dedicare i prossimi anni a imparare i linguaggi di programmazione e le relative funzioni, classi, schemi di progettazione, prototipi, chiusure e altro ancora.

In alternativa esiste un segreto che alcuni dei nomi più noti del Web, come Google, Digg, WordPress e Amazon, saranno contenti di condividere con voi, e si chiama jQuery. I progettisti e gli sviluppatori in tutto il mondo utilizzano la libreria di jQuery per implementare in modo elegante e rapido le loro idee sull'interazione, completando il rompicapo dello sviluppo web.

In questo capitolo darete un'occhiata a ciò che rende jQuery così speciale e a come integra HTML e CSS in un modo più naturale rispetto al vostro vecchio amico e acerrimo nemico JavaScript. Vedrete anche cosa occorre per implementare e far funzionare jQuery con i siti attuali.

## In questo capitolo

- **Cosa rende così speciale jQuery**
- **Qual è il lato negativo?**
- **Download e inclusione di jQuery**
- **Anatomia di uno script jQuery**
- **Frammenti di HTML: il DOM**
- **Se scegliete di accettarlo...**

## Cosa rende così speciale jQuery

Abbiamo detto che jQuery semplifica l'utilizzo del DOM, l'aggiunta di effetti e l'esecuzione delle richieste Ajax, ma cosa lo rende più appetibile che non scrivere una libreria personale o utilizzare una delle altre librerie di JavaScript, altrettanto eccellenti?

jQuery rende queste operazioni così semplici da essere addirittura divertente, tanto che dovrete spesso rinunciare a qualcuna delle vostre creazioni bizzarre, indossare i panni del progettista web serio e contenervi un po' (ah, quante cose interessanti potreste realizzare se il buon gusto non fosse una barriera!). Esistono tuttavia numerosi fattori da prendere in considerazione se state per investire il vostro tempo prezioso nell'apprendimento di come funziona una libreria di JavaScript.

## Compatibilità tra browser

Oltre a essere molto piacevole da utilizzare, uno dei più grandi vantaggi di jQuery consiste nel fatto che gestisce molti dei problemi più critici di compatibilità tra browser. Chiunque abbia scritto seriamente codice JavaScript in passato sa che le incoerenze tra i browser possono portare alla pazzia. Per esempio, un design che risulta perfetto in Mozilla Firefox e Internet Explorer 8 va a rotoli in Internet Explorer 7; o magari un componente dell'interfaccia al quale avete dedicato giorni di lavoro funziona benissimo in tutti i browser principali tranne che in Opera su Linux. E il cliente utilizza proprio Opera su Linux. Questi tipi di problemi non sono mai facili da individuare, oltre che difficili da debellare totalmente.

Anche quando i problemi di compatibilità tra i browser sono relativamente semplici da gestire, dovete sempre ricorrere a tutta la vostra conoscenza: quando sono le undici della sera prima del lancio di un progetto importante, potete solo sperare di ricordarvi perché in un browser è presente uno strano baco che vi siete dimenticati di testare.

Il team di jQuery è perfettamente consapevole dei problemi tra i browser e, fattore ancora più rilevante, comprende perché si verificano; avendo scritto questa informazione nella libreria, jQuery risolverà automaticamente gli avvertimenti. La maggior parte del codice scritto verrà eseguita nello stesso modo in tutti i browser principali, compreso in quel piantagrane di Internet Explorer 6 così amato da tutti.

Questa caratteristica da sola consentirà allo sviluppatore medio di evitarsi dei gran mal di testa. Ovviamente il vostro scopo è quello di essere sempre aggiornati sugli ultimi sviluppi e le best practice del vostro settore, ma lasciare il compito di individuare i problemi dei browser al team di jQuery, che ne risolve un numero sempre maggiore con ogni nuova versione, vi permette di avere più tempo per implementare le vostre idee.

## Selettori CSS3

Rendere le tecnologie attuali compatibili tra i browser è utile, ma jQuery fornisce anche un supporto completo per la specifica dei prossimi selettori CSS3. Sì, persino in Internet Explorer 6.0! Potete così ottenere un vantaggio sul futuro imparando a utilizzare i selettori CSS3 proprio adesso nel codice di produzione. La selezione degli elementi che desiderate modificare sta alla base della potenza di jQuery e i selettori CSS3 forniscono ancora più strumenti da utilizzare.

## Utility vantaggiose

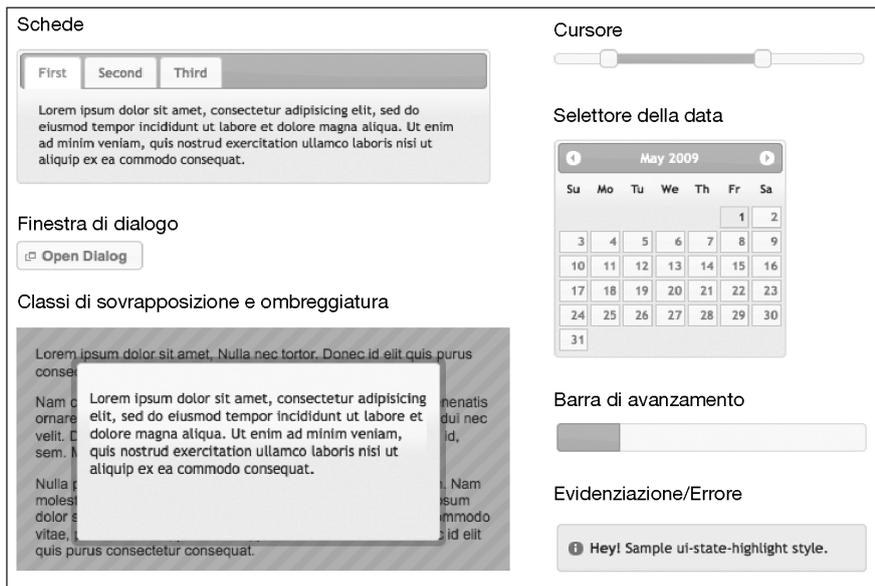
È anche inclusa una serie di utility che implementano funzioni comuni utili per scrivere codice jQuery (o addirittura mancanti in JavaScript), quali il taglio delle stringhe, la possibilità di estendere gli oggetti in modo semplice e altro ancora. Queste funzioni da sole sono particolarmente utili, ma consentono un'integrazione uniforme tra jQuery e JavaScript, il che dà come risultato un codice più semplice da scrivere e gestire.

Un'utility degna di nota è la funzione `supports`, che verifica se determinate caratteristiche sono disponibili nel browser attuale dell'utente. Per tradizione, gli sviluppatori ricorrono al rilevamento del browser per determinare quale viene utilizzato dall'utente finale, in base alle informazioni fornite dal browser stesso, per risolvere problemi noti. Si è sempre trattato di una procedura lacunosa e incline a errori. Mediante la funzione dell'utility `supports` di jQuery potete capire se una data caratteristica è disponibile per l'utente o se determinate prestazioni di un'applicazione sono destinate a sparire sui browser più vecchi o non conformi agli standard.

## Interfaccia utente di jQuery

jQuery è già stato utilizzato per creare widget ed effetti notevoli, alcuni dei quali sufficientemente utili da giustificare l'inclusione nella libreria *core*. Tuttavia, il team di jQuery ha saggiamente deciso che, per fare in modo che la libreria *core* rimanga mirata, è necessario separare costrutti di livello superiore e di includerli in pacchetti in una libreria nuova al livello superiore di jQuery.

Questa libreria è denominata *interfaccia utente di jQuery*, e comprende una serie di effetti utili e widget avanzati accessibili e altamente personalizzabili mediante l'utilizzo di temi. Alcune di queste caratteristiche sono illustrate nella Figura 1.1.



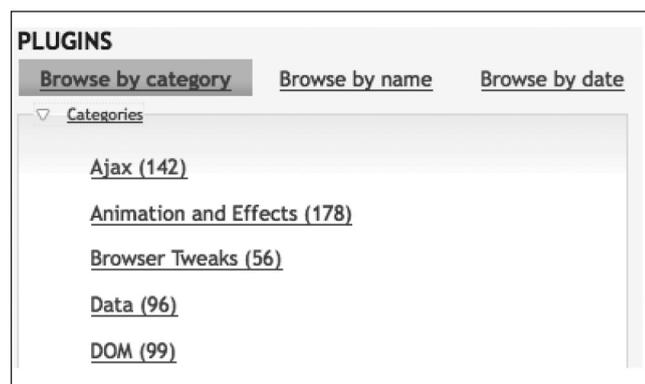
**Figura 1.1** Alcuni widget dell'interfaccia utente di jQuery.

Menu a fisarmonica, cursori, finestre di dialogo, selettori di date e altro ancora, sono tutti pronti per essere utilizzati immediatamente. Potreste dedicare tempo a crearne di personalizzati in jQuery, come quelli mostrati, ma i controlli dell'interfaccia utente di jQuery possono essere configurati e sono sufficientemente sofisticati che il vostro tempo potrebbe essere meglio speso in altro modo, e più precisamente implementando i vostri requisiti di progetto unici invece di assicurarvi che il selettore di date personalizzato venga visualizzato correttamente in browser diversi.

Nel corso del libro utilizzerete una serie di funzionalità dell'interfaccia utente di jQuery. Vi occuperete anche dell'integrazione di alcuni dei temi non comuni disponibili e apprenderete come crearne di personalizzati mediante lo strumento *ThemeRoller* dell'interfaccia utente.

## Plug-in

Il team di jQuery ha dedicato molta attenzione alla creazione di una libreria proprietaria estendibile. Includendo solo una serie di caratteristiche core e fornendo un framework per estendere la libreria, ha semplificato la creazione di plug-in che si possono riutilizzare in tutti i progetti di jQuery, oltre che condividerli con altri sviluppatori. Dalla libreria core di jQuery sono state omesse molte funzionalità più comuni, che sono state relegate tra i plug-in. Non preoccupatevi, poiché si tratta di un vantaggio, non di un difetto. Tutte le funzionalità ulteriori richieste possono essere incluse in modo semplice pagina per pagina così che la larghezza di banda e la quantità di codice siano ridotti al minimo. Fortunatamente, molte persone hanno tratto vantaggio dall'estendibilità di jQuery, quindi esistono già centinaia di eccellenti plug-in che possono essere scaricati dal repository (*deposito*) di plug-in di jQuery, a cui se ne aggiungono sempre di nuovi. Nella Figura 1.2 potete vederne un esempio.



**Figura 1.2** Il repository dei plug-in di jQuery.

Ogni volta che dovete affrontare un'attività o un problema, vale la pena verificare prima se esiste un plug-in che potrebbe essere adatto alle vostre esigenze. Questo perché quasi tutte le funzionalità che potrebbero essere necessarie sono probabilmente già state

trasformate in un plug-in e sono disponibili per essere utilizzate subito. Anche se fosse necessario svolgere del lavoro personalmente, il repository dei plug-in è spesso il luogo migliore da cui partire.

## Tenere il markup pulito

Separare il comportamento di scripting dalla presentazione della pagina è cruciale nel gioco dello sviluppo web, anche se presenta le sue sfide. jQuery favorisce l'affrancamento del markup dallo scripting inline grazie alla sua abilità di collegare in modo semplice gli elementi alla pagina e associarvi il codice in modo naturale, similmente a quanto avviene nel codice CSS. A jQuery manca un meccanismo per aggiungere codice inline, e questa separazione porta a un codice più snello, chiaro e gestibile. Di conseguenza è semplice effettuare le operazioni nel modo giusto e quasi impossibile portarle a termine nel modo sbagliato.

jQuery non si limita a mescolarsi con il codice HTML esistente in una pagina, ma può anche aggiungere nuovi elementi della pagina e frammenti di documenti tramite una serie di utili funzioni, come quelle per inserire, aggiungere e anteporre nuovi segmenti di HTML in un punto qualsiasi della pagina. Potete persino sostituire, rimuovere o clonare elementi esistenti, tutte caratteristiche che consentono di migliorare in modo progressivo i siti, fornendo così un'esperienza completa agli utenti i cui browser lo consentono e un'esperienza accettabile per tutti gli altri.

## Adozione diffusa

Se passate in Google Trends tutte le librerie di JavaScript a cui riuscite a pensare (<http://www.google.com/trends/>), noterete l'aumento esponenziale della notorietà di jQuery. I numeri elevati sono utili quando si tratta di librerie, dato che la popolarità è uguale a uno sviluppo del codice più attivo e a una vasta gamma di interessanti strumenti di terzi.

Molti dei grandi protagonisti del Web stanno passando a jQuery: IBM, Netflix, Google (che utilizza e ospita la libreria di jQuery) e persino Microsoft, che ora include jQuery con il proprio framework MVC. Con una gamma così vasta di grandi aziende dalla propria parte, è certo che jQuery sarà tra noi ancora a lungo, quindi il tempo e gli sforzi che investirete nel relativo apprendimento varranno la pena.

La popolarità di jQuery ha anche creato una community di grandi dimensioni sorprendentemente generosa. A prescindere dal livello di competenza, troverete altri sviluppatori sufficientemente pazienti da aiutarvi a risolvere qualsiasi problema incontriate. Questo spirito di assistenza e condivisione si è anche diffuso a Internet, sviluppando un'enciclopedia di tutorial, blog e documentazione di alta qualità.

## Qual è il lato negativo?

Quasi non esiste un lato negativo. Gli argomenti principali a sfavore dell'utilizzo di qualsiasi libreria di JavaScript sono sempre stati la velocità e le dimensioni. Alcuni sostengono che il ricorso a una libreria comporti un download eccessivo per le pagine, mentre secondo altri le librerie hanno prestazioni mediocri se confrontate con il codice

personalizzato più snello. Anche se vale la pena prendere in considerazione questi argomenti, la loro importanza è relativa.

Innanzitutto, per quanto riguarda le dimensioni, jQuery è leggero. La libreria core di jQuery ha sempre avuto dimensioni alquanto ridotte, pari a circa 19 KB per gli elementi di base, meno di un'immagine JPG media. Tutti gli elementi ulteriori che il vostro progetto necessita, quali plug-in o componenti dell'interfaccia utente, possono essere aggiunti a moduli, per poter calcolare agevolmente le dimensioni della larghezza di banda.

La velocità, come le dimensioni, sta diventando un problema minore dato che le specifiche hardware dei computer aumentano e i motori JavaScript dei browser sono sempre più prestanti. Ovviamente ciò è ben lontano dall'implicare che jQuery sia lento (il team di jQuery sembra ossessionato dalla velocità); ogni nuova versione è più veloce della precedente, quindi ogni vantaggio che potrebbe derivare dall'utilizzare codice JavaScript personalizzato si riduce progressivamente.

Quando si tratta delle librerie di JavaScript concorrenti (e ne esistono molte là fuori), jQuery è lo strumento migliore per svolgere il suo compito principale, ossia manipolare il DOM, aggiungere effetti e creare richieste Ajax. Molte delle librerie disponibili sono inoltre di qualità eccellente e primeggiano in altri campi, come la programmazione basata su classi complesse. Vale sempre la pena esaminare le alternative, ma se i motivi qui elencati vi stimolano, jQuery è probabilmente la direzione da seguire.

Basta parlare. È il momento per jQuery di passare ai fatti!

## Download e inclusione di jQuery

Prima di innamorarvi di jQuery, dovete procurarvi la versione più recente del codice e aggiungerlo alle vostre pagine web. Esistono alcuni modi per effettuare questa operazione, ciascuno con un paio di opzioni disponibili. Qualunque sia la vostra scelta, dovrete includere jQuery nella vostra pagina HTML, come fareste per qualsiasi altro file sorgente JavaScript.

### **SUGGERIMENTO: È SOLO JAVASCRIPT!**

---

Non dimenticate che jQuery è solo JavaScript! In superficie può sembrare che agisca in modo diverso, ma sotto sotto è scritto in JavaScript e, di conseguenza, è incapace di effettuare qualsiasi operazione che il buon vecchio JavaScript non è in grado fare. Ciò significa che lo includerete nelle vostre pagine nello stesso modo in cui avverrebbe per qualsiasi altro file di JavaScript.

## Download di jQuery

Si tratta del metodo più comune di acquisire la libreria di jQuery. La versione più recente è sempre disponibile sul sito web di jQuery (<http://jquery.com>). Il grosso e luccicante pulsante di download vi porterà al repository del codice di Google, dove potete trovare la versione del livello più recente di "compressione della produzione".

Fate clic sul collegamento di download e salvate il file di JavaScript in una nuova cartella di lavoro, pronti per utilizzarlo. Dovete collocarlo dove i vostri file HTML siano in

grado di vederlo, in genere in una directory *script* o *javascript* nella cartella principale del documento del sito. Per l'esempio riportato di seguito, manterrete una certa semplicità e collegherete la libreria nella stessa directory del file HTML.

Per fare in modo che tutto funzioni, dovete indicare al file HTML di includere la libreria di jQuery. Questa operazione viene effettuata mediante un tag `script` all'interno della sezione `head` del documento HTML. L'elemento `head` di un file HTML di base che comprende jQuery potrebbe essere simile al seguente:

```
<head>
  <title>Hello jQuery world!</title>
  <script type='text/javascript' src='jquery-1.4-min.js'></script>
  <script type='text/javascript' src='script.js'></script>
</head>
```

Il primo tag di scripting della pagina carica la libreria di jQuery e il secondo è diretto a un file `script.js`, il punto dal quale viene eseguito il codice jQuery. È tutto. Ora potete iniziare a utilizzare jQuery.

Sebbene il download del file di jQuery sia l'approccio più comune, esistono altre opzioni, che esamineremo rapidamente prima di continuare. Se volete iniziare subito a utilizzare jQuery, potete tranquillamente ignorare la parte restante di questo paragrafo.

## CDN di Google

Un metodo alternativo per includere la libreria di jQuery è la rete CDN (*Content Delivery Network*) di Google. Una CDN è una rete di computer progettati in modo specifico per fornire contenuti agli utenti in modo rapido e scalabile. Questi server vengono spesso distribuiti geograficamente, con ogni richiesta gestita dal server più vicino della rete.

Google ospita nella propria CDN numerose librerie open source note, incluso jQuery e l'interfaccia utente di jQuery che vedrete a breve. Quindi, invece di ospitare i file di jQuery nel server web come in precedenza, avete la possibilità di lasciare che Google utilizzi parte della vostra larghezza di banda. Sarete avvantaggiati dalla velocità e dall'affidabilità della grande infrastruttura di Google, con l'ulteriore possibilità di utilizzare sempre la versione più recente di jQuery.

Un altro vantaggio derivante dall'utilizzo della CDN di Google consiste nel fatto che molti utenti avranno già scaricato jQuery da Google durante la visualizzazione di un altro sito. Di conseguenza, verrà caricato dalla cache quando visiteranno il vostro sito, dato che l'URL al file di JavaScript sarà lo stesso, e i tempi di caricamento saranno notevolmente più rapidi. Con lo stesso metodo potete anche includere la più considerevole libreria dell'interfaccia utente di jQuery, cosa per cui vale la pena di prendere in considerazione la CDN di Google per i vostri progetti, perché vi consentirà di risparmiare denaro e aumentare le prestazioni quando il vostro lavoro verrà diffuso.

Vi sono vari modi di includere jQuery dalla CDN di Google. Utilizzerete ora il metodo più semplice, seppur leggermente meno flessibile, basato sul percorso:

```
<head>
  <title>Hello jQuery world!</title>
  <script type="text/javascript" src="http://ajax.googleapis.com/ajax/libs/jquery/1.4.0/
  ↪jquery.min.js"></script>
  <script type='text/javascript' src='script.js'></script>
</head>
```

Sembra sospettosamente simile all'esempio originale, ma invece di dirigere il tag `script` a una copia locale di jQuery, lo dirige a uno dei server di Google.

### **SUGGERIMENTO: OTTENERE L'ULTIMA VERSIONE CON LA CDN DI GOOGLE**

Se osservate con attenzione l'URL che rimanda ai server di Google, noterete che la versione di jQuery viene specificata da uno degli elementi del percorso (1.4.0 in questo esempio). Se desiderate utilizzare la versione più recente, tuttavia, potete rimuovere un numero dalla fine della stringa della versione, per esempio 1.4, e otterrete la versione più recente disponibile della serie (1.4, 1.4.1, 1.4.2 e così via). Potete persino utilizzare il numero intero (1), nel qual caso Google vi fornirà la versione più recente anche quando verrà rilasciato jQuery nella versione 1.5 e successive. Fate comunque attenzione: non è necessario aggiornare i file HTML quando viene rilasciata una nuova versione di jQuery, ma sarà necessario cercare qualsiasi modifica della libreria che potrebbe influire sulla funzionalità esistente.

Se desiderate esaminare il metodo di caricamento di Google per l'inclusione delle librerie (leggermente più complesso), potete trovare molte informazioni sulla CDN di Google nel relativo sito web (<http://code.google.com/apis/ajaxlibs/documentation/>).

## **Nightly e versioni secondarie**

Ulteriori opzioni avanzate per ottenere jQuery sono elencate nella pagina della documentazione ufficiale di download di jQuery ([http://docs.jquery.com/Downloading\\_jQuery](http://docs.jquery.com/Downloading_jQuery)). La prima di queste opzioni è costituita dalle *build nightly*. Si tratta di build automatizzate della libreria di jQuery che includono tutto il nuovo codice aggiunto o modificato nel corso della giornata. Ogni notte le versioni di sviluppo più recenti vengono rese disponibili per il download e possono essere incluse nello stesso modo della libreria stabile e regolare. E se ogni singola nightly è ancora troppo poco per voi, potete utilizzare il repository delle versioni secondarie per recuperare il codice sorgente aggiornato all'ultimo minuto. La *versione secondaria* è un sistema di controllo delle versioni open source utilizzato dal team di jQuery. Ogni volta che uno sviluppatore invia una modifica a jQuery, potrete scaricarla all'istante.

Fate tuttavia attenzione, perché le librerie di jQuery delle versioni nightly e secondarie spesso non sono state verificate. Possono, e spesso è effettivamente così, contenere errori e sono soggette a modifiche frequenti. A meno che non stiate cercando di lavorare sulla libreria di jQuery stessa, è preferibile ignorare queste opzioni.

## **Compresso o con compresso?**

Se avete dato un'occhiata alla pagina di download di jQuery, potreste aver notato un paio di opzioni di formato di download diverse, vale a dire il formato compresso (*ridotto*) e non compresso (anche noto come *sviluppo*).

In genere è consigliabile utilizzare la versione compressa per il codice di produzione, dove spazi e interruzioni di riga sono stati rimossi e i nomi delle variabili abbreviati. Otterrete la stessa libreria di jQuery, ma contenuta in un file di JavaScript molto più piccolo dell'originale: ciò è molto utile per ridurre i costi della larghezza di banda e per velocizzare le richieste delle pagine per l'utente finale.

Lo svantaggio del file compresso è la leggibilità. Se esaminate il file di jQuery compresso nell'editor di testi, vedrete che è praticamente illeggibile e rappresentato da una singola riga di JavaScript ingarbugliata. Nella maggior parte dei casi la leggibilità della libreria è irrilevante, ma se siete interessati al modo in cui jQuery funziona effettivamente, la versione di sviluppo non compressa è un esempio di codice JavaScript commentato, intellegibile ed eccellente.

## Anatomia di uno script jQuery

Dopo aver incluso jQuery nella pagina web, esaminerete cosa è in grado di fare. La sintassi di jQuery potrebbe sembrare alquanto strana la prima volta, ma in realtà è abbastanza diretta e, ancora meglio, decisamente coerente. Dopo aver scritto i primi comandi, lo stile e la sintassi vi rimarranno impressi nella memoria e vorrete continuare a scriverne altri.

### L'alias di jQuery

L'inclusione di jQuery nella vostra pagina vi consente l'accesso a un'unica funzione magica denominata, stranamente, `jQuery`. È attraverso quest'unica funzione che jQuery espone centinaia di potenti strumenti per consentire di aggiungere un'altra dimensione alle pagine web.

Dato che una singola funzione agisce da gateway all'intera libreria di jQuery, sono poche le probabilità che i nomi delle funzioni della libreria entrino in conflitto con altre librerie o con il vostro codice JavaScript. In caso contrario, si potrebbe verificare una situazione come la seguente. Supponete che jQuery abbia definito una funzione denominata `hide` e che anche voi disponiate di una funzione denominata `hide` nel vostro codice: una delle funzioni verrebbe sovrascritta, portando a eventi ed errori imprevisi.

Si dice che la libreria di jQuery è contenuta nel *namespace* di jQuery. Si tratta di un eccellente approccio per utilizzare in modo corretto altro codice in una pagina, ma se utilizzerete spesso jQuery, diventerà rapidamente fastidioso dover digitare il nome della funzione di jQuery completo per ogni comando da utilizzare. Per risolvere questo problema, jQuery fornisce un alias più breve per accedere alla libreria, vale a dire `$`.

Il simbolo del dollaro è un nome di variabile JavaScript breve, valido e di bell'aspetto. Potrebbe sembrare un indice di pigrizia – dopotutto con l'alias risparmiate solo cinque lettere – ma una pagina intera di jQuery conterrà grandi quantità di chiamate alla libreria e mediante l'alias il codice diventerà molto più leggibile e gestibile.

#### **AVVERTIMENTO: UTILIZZO DI PIÙ LIBRERIE**

Il motivo principale per cui è consigliabile utilizzare la chiamata `jQuery` completa invece dell'alias è nel caso di più librerie JavaScript nella stessa pagina, tutte che combattono per il controllo del nome della funzione con il simbolo del dollaro. `$` è un nome di funzione comune in molte librerie, spesso utilizzato per selezionare gli elementi. Se si verificano problemi con più librerie, fate riferimento al paragrafo relativo a come evitare i conflitti nel Capitolo 9.

## Sezionare un'istruzione jQuery

Sapete che i comandi di jQuery iniziano con una chiamata alla funzione `jQuery` o al relativo alias. Esaminerete ora i componenti rimanenti di un'istruzione jQuery. La Figura 1.3 mostra entrambe le varianti della stessa istruzione jQuery con il nome di funzione completo o l'alias `$`.

selettore	azione	parametri
<code>jQuery('p')</code>	<code>.css</code>	<code>('color', 'blue');</code>
<code>\$('#p')</code>	<code>.css</code>	<code>('color', 'blue');</code>

**Figura 1.3** Una tipica istruzione jQuery.

Ogni comando è costituito da quattro parti: la funzione jQuery o l'alias corrispondente, i selettori, le azioni e i parametri. Conoscete già la funzione jQuery, quindi osserverete ora ciascuno degli altri elementi. Innanzitutto si utilizza un selettore per selezionare uno o più elementi nella pagina web. Successivamente si sceglie un'azione da applicare a ogni elemento selezionato. Nel corso del libro vedrete un numero sempre maggiore di azioni mentre implementate gli effetti. Infine specificherete alcuni parametri per indicare a jQuery il modo esatto in cui applicare l'azione prescelta. Ogni volta che utilizzate il codice jQuery, provate a dividerlo in queste parti. Semplificherà notevolmente la comprensione iniziale.

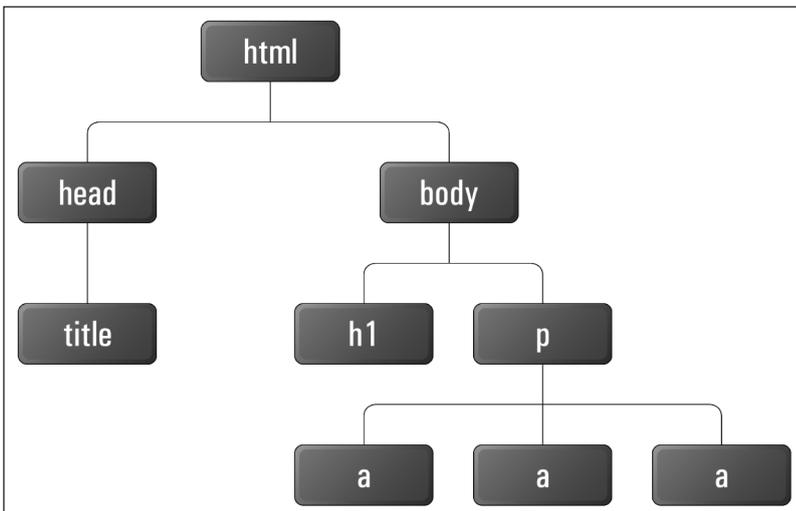
Nell'esempio precedente avete chiesto al selettore di selezionare tutti i tag di paragrafo, i tag HTML `<p>`, della pagina. Successivamente avete scelto l'azione `css` di jQuery, che viene utilizzata per modificare una proprietà CSS degli elementi del paragrafo selezionati inizialmente. Infine avete passato alcuni parametri per impostare la proprietà `color` di CSS sul valore `blue`. Il risultato finale? Tutti i paragrafi sono diventati blu. Ulteriori informazioni sui selettori e sull'azione `css` sono disponibili nel Capitolo 2.

L'esempio ha passato due parametri (`color` e `blue`) all'azione `css`, ma il numero di parametri passati a un'azione può variare. Alcuni richiedono zero parametri, altri accettano serie diverse di parametri per modificare un intero gruppo di proprietà contemporaneamente, altri ancora richiedono di specificare un'altra funzione JavaScript per eseguire il codice quando si verifica un evento, come il clic su un elemento. Tutti i comandi seguono però questa struttura di base.

## Frammenti di HTML: il DOM

jQuery è stato progettato per integrarsi in modo uniforme con il codice HTML e CSS. Se siete esperti dei selettori CSS e sapete, per esempio, che `div#heading` fa riferimento a un elemento `div` con un `id` di `heading`, potete ignorare questo paragrafo. In caso contrario, un breve corso intensivo sui selettori CSS e sul modello DOM (*Document Object Model*) può esservi utile.

Il DOM non riguarda in modo specifico jQuery, ma è un modo standard per rappresentare gli oggetti in HTML che tutti i produttori di browser hanno concordato di seguire. Una buona conoscenza pratica del DOM garantirà una transizione senza problemi a jQuery. Il DOM è costituito da frammenti di HTML ottimizzato. È una rappresentazione gerarchica del markup HTML, in cui ogni elemento, per esempio `div` o `p`, ha un elemento *genitore*, o contenitore, e può anche avere uno o più elementi *figlio* annidati. Ogni elemento può avere un `id` e/o uno o più attributi `class`, che in genere vengono assegnati nel file sorgente HTML. Quando il browser legge una pagina HTML e crea il DOM, lo visualizza come pagina web che comprende oggetti che possono rappresentare una pagina statica o, fatto ancora più interessante, essere manipolati dal codice. Un frammento di DOM di esempio è illustrato nella Figura 1.4. Come potete vedere, `body` dispone di due elementi figlio, `h1` e `p`. Questi due elementi, grazie al fatto di essere contenuti nello stesso elemento genitore, sono detti *sibling* (consanguinei).



**Figura 1.4** Esempio di frammento di DOM.

L'`id` di un elemento identifica in modo univoco l'elemento nella pagina:

```
<div id="footer">Come back and visit us soon!</div>
```

A `div` è stato assegnato l'`id` di `footer`. Utilizza un `id` perché è univoco. Deve essercene solo uno nella pagina. Il DOM consente anche di assegnare lo stesso nome a più elementi della pagina mediante l'attributo `class`. Ciò viene in genere effettuato per elementi che condividono una caratteristica:

```
<p class="warning">Sorry, this field must be filled in!</p>
<span class="warning">Please try again</span>
```

In questo esempio più elementi della stessa pagina vengono classificati come “warning” (avvertimenti). Ogni CSS applicato a `warningclass` sarà applicato a entrambi gli elementi. Più attributi `class` nello stesso elemento, quando necessari, sono separati da spazi.

Quando scrivete il vostro codice CSS, potete collegare gli elementi per `id` con un simbolo di cancelletto (`#`) o per `class` con un punto (`.`):

```
#footer { border: 2px solid black }  
.warning { color: red }
```

Queste regole del codice CSS assegneranno un bordo nero all'elemento con un `id` pari a `footer` e garantiranno che tutti gli elementi con una `class` di tipo `warning` verranno visualizzati in rosso.

Quando si tratta di scrivere codice jQuery, scoprirete che conoscere i selettori CSS e il DOM è importante, perché jQuery utilizza la stessa sintassi di CSS per selezionare gli elementi della pagina da manipolare. Dopo aver compreso la selezione, il resto è facile, grazie a jQuery.

## Se scegliete di accettarlo...

jQuery è un prodotto stabile e maturo pronto all'uso in siti web di qualsiasi dimensione, come dimostrato dalla sua adozione da parte di alcuni dei giganti di Internet. Ciononostante è ancora un progetto dinamico in costante sviluppo e miglioramento, con ogni nuova versione che offre potenziamenti delle prestazioni e ulteriori funzionalità intelligenti. Non esiste un momento migliore per imparare a utilizzare jQuery!

Mentre leggete il libro, vedrete che il motto "meno si scrive, più si fa" di jQuery risulterà vero. Si tratta di una libreria semplice e divertente con una curva di apprendimento graduale che vi consentirà di effettuare molte operazioni interessanti con una quantità esigua di codice. Proseguendo sul percorso verso jQuery, speriamo che acquistiate anche un po' di rispetto e comprensione per lo stesso JavaScript.

Nel Capitolo 2 vi tufferete in jQuery e inizierete a utilizzarlo per aggiungere splendore al sito web del vostro cliente. A proposito del cliente, è giunto il momento di incontrarlo...