Vuoi imparare jQuery ma non sai da dove iniziare? Le animazioni personalizzate

Nella scorsa lezione abbiamo dato un'occhiata veloce alle principali animazioni predefinite presenti in jQuery. Ma come fare ad applicare delle animazioni personalizzate? Oggi scoprirai come fare: iniziamo!

La funzione ".animate()"

Oltre a tutti metodi di animazione già visti, jQuery fornisce la funzione .animate() che ci permette di progettare le nostre animazioni. Questa funzione prende in ingresso quattro paramentri:

- Un array con le proprietà e i rispettivi valori (simili a quello utilizzato per la funzione css. *Obbligatorio*
- Una stringa (slow,normal,fast) oppure il numero di millisecondi rappresentanti la durata dell'animazione. *Opzionale*
- Una stringa che descrive come deve essere svolta l'animazione (detta anche *easing type*). *Opzionale*
- Una funzione, da eseguire al termine dell'animazione. *Opzionale*

La sintassi è dunque la seguente:

Per il momento ignorerò gli ultimi due parametri, e mi concentrerò solo sui primi due, dato che sono più che sufficienti per le prime animazioni.

Come utilizzarla?

L'utilizzo della funzione animate è identico a quello della funzione css, con la differenza che in questo caso l'elemento **passerà da un valore all'altro gradualmente, senza un brusco salto.** Facciamo un esempio dunque. Prendiamo questo markup:

Vogliamo aumentare le dimensioni dell'elemento #box. Potremmo farlo con la funzione css in questo modo:

Come puoi vedere, non si riesce nemmeno a notare che c'è un passaggio da un div 50x50 ad uno 400x400. Proviamo allora con la funzione animate:

Così va molto meglio! Con questa funzione possiamo animare qualsiasi proprietà CSS che abbia un valore numerico, quindi margini, padding, bordi, altezza, larghezza, posizioni, opacità sono tutti candidati ideali. L'animazione avrà come punto iniziale quello specificato nel foglio di stile, e come punto finale quello specificato nel file JavaScript.

Nell'animazione precedente abbiamo visto come animare contemporaneamente due proprietà: infatti il nostro "box" cresceva allo stesso modo sia in altezza che il larghezza. Possiamo animare una proprietà (o gruppi di proprietà) per volta concatenando più chiamate alla funzione animate:

Posso utilizzare solo valori numerici?

No. Per le proprietà height, width e opacity oltre a valori numerici, è possibile specificare **anche dei valori testuali**:

- **show**: aumenta gradualmente la proprietà, se è già al massimo non fa nulla
- **hide**: stessa cosa di show, però diminuisce il valore fino ad azzerarlo. Se è già zero non fa nulla
- **toggle**: alterna show e hide. Se la proprietà è visibile, la nasconde, altrimenti la mostra

Grazie a quest'ultima proprietà possiamo realizzare animazioni ancora più complesse con meno righe di codice, che tengono conto del contesto. Prendiamo ad esempio questo markup:

Possiamo nascondere e mostrare il secondo paragrafo semplicemente, con la funzione animate:

Ottenendo questo risultato. Ti avverto che questa non è una soluzione ottimale, dato che nel caso JavaScript fosse disattivato, non sarebbe possibile mostrare il paragrafo.

Quando si utilizza la funzione animate, però, bisogna stare attenti a non andar contro le regole dei CSS. Prendiamo come esempio questo semplice labirinto. L'HTML è una semplice immagine e un div. Il CSS associato è il seguente:

Il codice che effettua l'animazione è invece il seguente:

Come puoi vedere, ho utilizzato la funzione animate per modificare le proprietà top e left, ma questo non avrebbe prodotto alcun movimento, se nel CSS non avessi posizionato assolutamente il div. **È importante dunque pianificare con attenzione le animazioni**.

Un'altra cosa che puoi notare nel codice JavaScript precedente, è l'utilizzo dell'operatore +=. Questo operatore permette di aggiungere una quantità relativa: invece di muovere l'elemento alla posizione "60px", la muove di 60px rispetto a quella attuale.

Conclusioni

Abbiamo fatto un sacco di strada fino ad ora. Siamo partiti dai concetti base di jQuery, passando per la gestione degli eventi, le animazioni e, quest'oggi, le animazioni personalizzate. Questi concetti dovrebbero bastarti per cominciare a creare le tue prime funzioni utilizzando questo fantastico framework. Nelle prossime lezioni vedrai come questi concetti vengono applicati nel "mondo reale": ti mostrerò infatti alcuni esempi di codice jQuery funzionanti al 100%.

- Lezione 1: introduzione e primi concetti fondamentali
- Lezione 2: gli eventi (concetti basilari)
- Lezione 3: alternare le funzioni, le variabili, primi effetti
- Lezione 4: gli stili inline, gli effetti predefiniti
- Lezione 5: le animazioni personalizzate