

L'Oscura Arte del Drag and Drop: Come Trascinare e Rilasciare Oggetti?



Quella di trascinare e rilasciare oggetti in una pagina web è una proprietà molto affascinante e di sicuro effetto. Chi ha familiarità con il pannello di amministrazione di WordPress saprà ad esempio che per attivare un widget, basta trascinarlo sulla sidebar nella posizione desiderata. Questo rende certamente molto semplice ed intuitiva tale operazione. In questo articolo **apprenderemo le basi di questa procedura** piuttosto complessa, ma che grazie a jQuery si trasforma in un gioco da ragazzi. L'idea ti stuzzica? Addentriamoci dunque nella lettura dell'articolo.

Le user interface di jQuery, una risorsa di enorme valore

Le User Interface (UI) sono una raccolta di solide estensioni che aiutano nella risoluzione di problemi piuttosto comuni. Possiamo ad esempio (tramite una riga di codice), installare un datepicker (sono quei piccoli calendari che compaiono nei form quando dobbiamo inserire una data) altamente configurabile sia nell'estetica sia nella funzionalità: [vedere per credere](#).

Di questa raccolta fanno parte i metodi *draggable()* e *droppable()* che andremo ora ad analizzare nel dettaglio.

Prima di iniziare, dai uno sguardo alla [pagina di esempio](#). Come puoi vedere si tratta di una pagina con cinque elementi:

- Un contenitore (sulla destra) che rappresenta l'area nella quale gli elementi potranno essere rilasciati.
- Un elemento non trascinabile.

- Un elemento trascinabile ma il cui rilascio nell'area a destra non viene intercettato.
- Un elemento trascinabile (unicamente sull'asse orizzontale) il cui rilascio viene intercettato.
- Un elemento trascinabile che durante il trascinamento diventa opaco, il cui rilascio viene intercettato, ed una volta rilasciato rimane "incollato" al contenitore. Se notate l'elemento precedente, una volta rilasciato, viene intercettato; ma in seguito è ancora possibile riposizionarlo anche fuori dal contenitore.

Quanto hai visto è quello che stiamo per fare insieme. Iniziamo come sempre ad impostare la pagina html.

Prepariamo la pagina

Adesso inseriamo nel body del documento i quattro elementi di sinistra e l'area di rilascio a destra.

Come vedi, all'interno dell'elemento `#left_block` dichiaro i quattro elementi. Nota le classi applicate ad ognuno di essi; la classe `box` è quella che ne definisce l'aspetto, mentre le altre serviranno per definirne le proprietà (vedremo in seguito) del trascinamento.

Andiamo ora a scrivere il CSS nel file `screen.css`

Inoltre centro il contenuto all'interno di un contenitore, posiziono gli elementi e gli attribuisco un minimo di stile.

Tra poco vedremo cosa definisce la classe `hover`.

Nell'header del nostro documento dobbiamo ancora richiamare jQuery e le estensioni che costituiscono le user interface, contenute nel file `jquery-ui-1.7.2.custom.min.js`.

Inoltre includeremo il foglio di stile `screen.css` che abbiamo precedentemente creato.

Possiamo dare uno sguardo al documento che ora dovrebbe presentarsi così

A questo punto la pagina dovrebbe apparire come in [questo esempio](#).

Ora che abbiamo concluso la preparazione, passiamo finalmente alla realizzazione del codice per attivare le funzionalità di trascinamento e rilascio degli oggetti.

Come rendere un elemento "droppabile" (rilasciabile)

La prima cosa che faremo sarà rendere il blocco di destra **un'area dove è possibile rilasciare un oggetto** utilizzando il metodo *droppable()*.

Tutti i blocchi di codice che aggiungeremo ora, vanno scritti all'interno della funzione principale.

Vediamo cosa significa questo codice:

- nella prima riga diciamo di rendere droppabile l'elemento con id *right_block*.
- In seguito passiamo una serie di parametri il primo dei quali è *accept*. Nel nostro caso è valorizzato con *.rilasciabile*; ciò significa che **sarà possibile rilasciare unicamente gli elementi con classe *rilasciabile***.
- Il secondo parametro è *hoverClass*. Definisce la classe CSS da applicare quando l'elemento trascinato si trova sopra l'area di rilascio. Nella fattispecie, la classe *hover* che abbiamo definito precedentemente e che modifica il colore del background. Segnalare quando un elemento è in zona utile per il rilascio è una questione di usabilità.
- Il terzo parametro è *tolerance* e definisce la circostanza nella quale un elemento è da considerarsi in zona di rilascio. Abbiamo quattro possibilità:
 - **fit**: quando tutto l'elemento trascinato è all'interno dell'area rilasciabile.
 - **pointer**: quando il puntatore del mouse è all'interno dell'area rilasciabile;
 - **touch**: quando l'elemento trascinato tocca l'area rilasciabile;
 - **intersect**: quando almeno la metà dell'elemento trascinato si trova all'interno dell'area rilasciabile;
- Infine, con il parametro *drop* possiamo definire la **funzione da eseguire al momento del rilascio**. In questa funzione, il parametro che abbiamo definito con *ui*, diventerà l'oggetto che ci permetterà di **manipolare e accedere all'elemento trascinato** ed infine rilasciato al quale ci si potrà riferire con *ui.draggable*. Noi semplicemente ci limiteremo a sostituire la scritta contenuta nell'elemento.

Questa riga di codice riportata sopra significa: nell'elemento appena rilasciato, trova il tag *p* e sostituisci il contenuto di questo tag con la scritta "*rilasciato!*".

Come rendere gli elementi trascinabili?

Per rendere trascinabile un oggetto non dobbiamo fare altro che applicarvi il metodo *draggable()* che ora andremo ad analizzare. Iniziamo quindi con l'applicare il metodo *draggable()* all'elemento con classe "*trascinabile*" (secondo elemento) in questo modo:

Se provi ora il tuo script, vedrai che questo elemento è **diventato trascinabile e può essere posizionato ovunque**. Ma trascinandolo sull'area di rilascio non succede nulla, perché?

Abbiamo definito che **solo** gli elementi aventi classe *rilasciabile* possono essere rilasciati e non è il caso di questo elemento.

Come rendere gli oggetti trascinabili su un unico asse?

Passiamo ora al terzo elemento.

L'abbiamo reso trascinabile ed abbiamo aggiunto la proprietà *axis* impostata su *x*. Ciò significa che questo elemento è trascinabile unicamente sull'asse orizzontale. Non vi era nessuna necessità di assegnare questa impostazione, l'ho fatto per puro scopo didattico e mostrarti con quale semplicità si può ulteriormente personalizzare il trascinamento.

Se entriamo nell'area di rilascio con questo elemento, vedremo il colore del background cambiare, e se lo rilasciamo potremo notare che la scritta si modifica, **a testimoniare che l'evento è stato intercettato**.

In questo caso però, se provi a trascinare l'elemento all'esterno dell'area, ti renderai conto che questa operazione è possibile.

Ma cosa dobbiamo fare in caso volessimo che l'elemento rilasciato rimanga in qualche modo legato al contenitore?

Come trascinare un elemento e legarlo al contenitore dopo il rilascio?

Applichiamo questa proprietà all'ultimo elemento:

Abbiamo impostato il parametro *opacity* su *0.5*. Questo significa che durante il trascinamento, **l'oggetto diventerà opaco**. Anche in questo caso è stata utilizzata questa proprietà per puro scopo didattico.

Ma quello che veramente ci interessa è il parametro *revert* impostato su *invalid*. In questo modo, una volta rilasciato l'elemento **non sarà più possibile trascinarlo all'esterno**. Un altro effetto di questa impostazione è che, qualora l'elemento venisse trascinato e rilasciato all'esterno dell'area preposta, esso **si riposizionerà nel punto di partenza**.

Ecco la funzione completa

Conclusion

In questo articolo abbiamo visto le basi di una tecnica che ci permette di aggiungere delle funzionalità dinamiche e di effetto ai nostri siti. Io ho utilizzato il *drag and drop* nel pannello di amministrazione di un sito nel quale, tutto quello che è eliminabile, può essere trascinato nel cestino (utenti, contenuti, ecc), [qualcosa del genere](#).

Dal lato utente, un'applicazione diffusa è il "*carrello della spesa*"; funzionalità della quale puoi vedere un [esempio molto raffinato](#) realizzato con jQuery e con i principi che abbiamo visto in questo articolo.

E tu lo utilizzi? Puoi farci degli esempi concreti di come hai applicato o come potresti applicare questa tecnica?