

Creare uno shortcode in WordPress

domenica, 04 dicembre 2016

Uno shortcode è una sorta di "codice segnaposto" che permette di richiamare funzioni più o meno complesse all'interno dei nostri contenuti. L'esigenza di creare uno shortcode nasce dall'impossibilità di inserire funzioni o codice eseguibile all'interno dell'editor o dei campi di input di Wordpress. Ma come fare se hai la necessità di inserire nel testo dei tuoi articoli o delle tue pagine un contenuto variabile o risultante da una serie di operazioni e funzioni?



Sommario

Come creare uno shortcode

Diversi tipi di shortcode

Un esempio pratico: self-closing

Gestire gli attributi

Un esempio pratico: enclosing

Come creare uno shortcode

Per creare uno shortcode bisogna avvalersi della funzione `add_shortcode()`, che accetta 2 parametri:

\$tag: il nome dello shortcode, tutto minuscolo senza spazi
(string) - (obbligatorio)
default: nessuno

\$function: la funzione che genera il contenuto
(callable) - (obbligatorio)
default: nessuno

Per utilizzare lo shortcode nell'editor, è sufficiente inserirlo (tra parentesi quadre) come segnaposto e al momento della renderizzazione Wordpress lo sostituirà con il valore restituito dalla funzione di richiamo.

Diversi tipi di shortcode

Gli shortcode si possono dividere in due categorie sostanziali:

- *self-closing*: prevedono solo il tag di apertura, senza quello di chiusura
- *enclosing*: prevedono un tag di apertura ed uno di chiusura

Nel secondo caso, tutto ciò che è all'interno dei tag di apertura e chiusura viene passato alla funzione che genera il contenuto attraverso il parametro `$content`.

Un esempio pratico: self-closing

Ipotesizza, ad esempio, di voler inserire semplicemente nei tuoi contenuti il nome del tuo sito. Cosa succederebbe se lo cambiassi in futuro? Dovresti modificare tutte le pagine e gli articoli in cui esso è presente!

Con uno shortcode è possibile ovviare al problema, magari prelevando il nome del sito dalle opzioni di Wordpress, inserendo nel file `functions.php` del tema attivo, che trovi nello spazio FTP del tuo sito, in:

```
wp-content/themes/{nome_del_tema}/
```

le seguenti righe di codice:

```
// Crea lo shortcode [nomesito]  
add_shortcode( 'nomesito', function() {  
    return get_bloginfo( 'name' );  
} );
```

Ti basta inserire dove desideri il segnaposto `[nomesito]` e vedrai comparire nei contenuti nel frontend il nome del sito, così come settato nelle impostazioni del tuo WordPress.

Gestire gli attributi

Qualora ne avessi bisogno, c'è la possibilità di utilizzare dei parametri per aumentare la flessibilità degli shortcode.

Rifacendosi all'esempio precedente, ipotizza, ad esempio, di voler inserire il nome del sito completo di link, ma solo in alcuni casi. Inutile creare due shortcode che fanno sostanzialmente cose simili, ti basta utilizzare i parametri:

```
// Crea lo shortcode [nomesito] con parametri
add_shortcode( 'nomesito', function( $atts ) {
    extract( shortcode_atts( array(
        'addlink' => false
    ), $atts ) );

    if ( $addlink ) {
        return '<a href="' . home_url() . '">' . get_bloginfo( 'name' ) . '</a>';
    }
    return get_bloginfo( 'name' );
} );
```

Notiamo la differenza sostanziale: i parametri, assegnati alla variabile `$atts` sotto forma di array, vengono filtrati dalla funzione `shortcode_atts()` che controlla l'esistenza dei parametri ed eventualmente assegna un valore di default ad ognuno di essi.

Nel nostro esempio, qualora non specificassimo il parametro `addlink`, questo verrebbe automaticamente considerato `false`.

Ma come passare un parametro allo shortcode? Niente di più semplice! Ti basterà richiamarlo in questo modo: `[nomesito addlink="true"]`.

Un esempio pratico: enclosing

Ipotesi, ad esempio, di voler inserire all'interno dei tuoi articoli dei contenuti accessibili solo agli utenti loggati ed inaccessibile agli altri. Ecco come fare:

```
// Crea lo shortcode [privato]
add_shortcode( 'privato', function( $atts, $content = null ) {
    if ( is_user_logged_in() ) {
        return '<div>' . $content . '</div>';
    }
    return '';
} );
```

Ti basta adesso utilizzare lo shortcode in questo modo `[privato]` e tutto ciò che è contenuto tra il tag di apertura e quello di chiusura del tuo shortcode sarà invisibile agli utenti non loggati.

Esempio:

Questo contenuto è visibile a tutti gli utenti

`[privato]`Questo contenuto invece è visibile solo agli utenti loggati`[/privato]`

Anche questo contenuto è visibile a tutti gli utenti