

Guida allo sviluppo di plugin per wordpress: Gli strumenti di base

Nel [precedente articolo](#) ci siamo lasciati con la realizzazione dell'intestazione del nostro plugin. Ora dovremo iniziare a lavorare sulle funzioni fondamentali che ci permetteranno di “*inserirci*” nell'esecuzione di WordPress per “*aggiungere*” quello che ci serve.

Per fare questo dovremo dichiarare delle funzioni e dire a WordPress **dove** o **quando** eseguirle. Detto così può sembrare complicato, ma vediamo subito di chiarire con un po' di pratica.

add_action: Come aggiungere codice al runtime di WordPress?

La prima funzione che trattiamo in questo articolo é [add_action\(\)](#) che ci permette di definire **dove** o **quando** una funzione deve essere eseguita. La funzione `add_action()` ha questo formato:

```
add_action('dove o quando', 'funzione da eseguire');
```

Dunque, come secondo argomento della funzione dobbiamo passare il nome della funzione (da noi definita) che intendiamo eseguire.

Il primo parametro invece si riferisce ad un “gancio” messo a disposizione da WordPress; [qui trovi l'elenco di questi ganci](#) (la pagina continua molto oltre il primo gruppo di ganci).

`wp_footer` ad esempio si riferisce al gancio `wp_footer` del tema. Lo utilizzeremo se vogliamo che la nostra funzione sia eseguita in quel punto (**dove**).

Oppure potremo utilizzare `publish_post` se vogliamo eseguire una funzione nel momento in cui viene pubblicato un nuovo articolo (**quando**). Ad esempio se volessimo che alla pubblicazione di un nuovo articolo venga inviata un'email di notifica all'amministratore, dovremo scrivere qualcosa del genere:

```
add_action('publish_post', 'funzione_che_invia_un_email');
```

Vediamo subito un esempio tanto semplice quanto inutile. Facciamo in modo che il nostro plugin inserisca un commento nell'header del nostro sito.

Innanzitutto dichiariamo una funzione che stampa un commento.

! best practice !

A questo punto vediamo una prima importantissima regola: il nome delle funzioni deve essere un po' “complicato” e deve avere un prefisso (io uso l'acronimo del nome del plugin).

Questo per **limitare al massimo il rischio che un altro plugin o un tema contengano funzioni con nomi uguali alle nostre** (eventualità che provocherebbe un fatal error).

Bene. Ora abbiamo la nostra funzione che stampa a video un commento. Non dobbiamo fare altro che *dire* a WordPress che vogliamo eseguirla nell'header del nostro sito utilizzando il gancio *wp_head*, in questo modo.

Attiviamo il plugin (se non lo abbiamo ancora fatto) e proviamo a caricare una pagina del nostro sito. Visualizzando il sorgente vedrai che il commento é stato inserito nell'header.

Cancelliamo ora quanto abbiamo scritto e dedichiamoci ad un'altra fondamentale funzione.

add_filter: Come manipolare i contenuti?

Questa funzione si utilizza prevalentemente per apportare delle modifiche (filtri) ai **dati che verranno salvati nel database o ai dati in uscita da esso**, o più in generale a qualsiasi tipo di dato. Anche *add_filter* ha una [lunga serie di ganci](#). A differenza di *add_action* questi ganci non si riferiscono a quello che ho chiamato (per semplificare) un *dove* o un *quando* ma a un **“cosa”**. In pratica questi ganci ci indicano **a cosa** vogliamo aggiungere il filtro che definiremo nella funzione.

Facciamo subito un esempio. Poniamo di voler aggiungere un filtro che trasformi tutti i titoli dei nostri post da:

Titolo del post

a

Titolo: Titolo del post

In pratica vogliamo aggiungere “Titolo: “ prima del titolo di ciascun post.

Il “gancio” di cui avremo bisogno é *the_title*, che fa appunto riferimento al titolo del post.

Iniziamo con lo scrivere la semplice funzione per implementare questo filtro.

Bisogna sapere che *add_filter* passerà come argomento alla nostra funzione il contenuto del gancio. Non dovremo fare altro che recuperarlo e modificarlo, in questo modo.

Ed ora aggiungiamo il filtro

Ed ecco fatto. Il titolo di ogni nostro post verrà passato come argomento della funzione *yfb_filter_title_test* che provvederà a modificarlo.

Ora che abbiamo trattato queste due fondamentali funzioni, siamo pronti per continuare il nostro

plugin. Cancella tutto, mantenendo solo l'intestazione che abbiamo realizzato nel precedente articolo.

Come mostrare il nostro box della licenza?

Quello che dobbiamo fare per raggiungere lo scopo che ci siamo prefissati all'inizio di questa guida é aggiungere le informazioni relative alla licenza alla fine di ogni articolo. Iniziamo dunque a sviluppare la funzione.

Come vedi aggiungiamo il testo della licenza e il link per la chiusura del box alla fine del contenuto dell'articolo (per il momento in questo modo, più avanti quel testo lo renderemo disponibile come opzione nel pannello amministrativo di WordPress).

Ed ora non dovremo fare altro che aggiungere questo filtro al gancio che fa riferimento al contenuto dell'articolo (*the_content*), in questo modo:

A questo punto possiamo testare il nostro primo plugin. Alla fine di ogni articolo adesso dovrebbe apparire il testo della licenza.

Come aggiungere lo stile?

Nella cartella *yiw-license-plugin* creiamo la cartella *css* e al suo interno il file *style.css*.
Diamo ora un minimo di forma al nostro box

Ora abbiamo il nostro foglio di stile e dovremmo includerlo nell'header del sito.

Qualcuno a questo punto potrebbe pensare di scrivere una funzione come questa

E passarla tramite *add_action* al gancio *wp_head*.
Funzionerebbe, ma non é il modo corretto

! best practice !

I fogli di stile vanno inclusi tramite un'apposita funzione, *wp_enqueue_style*. Questa funzione passa diversi parametri; quelli che ci interessano sono i primi due.

Il primo é il nome univoco del foglio di stile ed il secondo é l'url.

Quindi per includere il nostro foglio di stile, procederemo in questo modo:

Anche se potrebbe andare non ci siamo ancora, il tutto può ancora essere ottimizzato.

! best practice !

WordPress raccomanda in questo caso di utilizzare la costante predefinita `WP_PLUGIN_URL` che contiene l'url della cartella plugin. Dunque procediamo in questo modo:

Ora possiamo ricaricare il nostro sito; il foglio di stile verrà correttamente caricato producendo un risultato simile a questo:

Il testo di questo articolo é di proprietà di Maurizio Tarchini ed é rilasciato sotto licenza Creative Commons. Sei libero di ridistribuirlo e riprodurlo....

[chiudi](#)

Riporto anche lo stato attuale del nostro plugin:

Conclusioni

In questo articolo abbiamo approfondito le due funzioni cardine nell'ambito dello sviluppo di plugin. Abbiamo inoltre visto il modo corretto di includere un foglio di stile.

Nel prossimo tutorial implementeremo la chiusura del box con animazione jQuery.

Una cosa del tutto superflua ma mi serve per mostrarti come funziona l'inclusione di script e per parlare delle funzioni preposte a questo scopo.

Arrivato a questo punto cosa ne pensi? Non é così difficile come può sembrare, vero?

Capitoli di questa guida

1. [Introduzione](#)
2. **Gli strumenti di base**
3. [La gestione degli script](#)
4. [Il pannello di amministrazione \(1/3\)](#)
5. [Il pannello di amministrazione \(2/3\)](#)
6. [Il pannello di amministrazione \(3/3\)](#)

7. Condividere il plugin

La mascotte dell'articolo è stata fornita da [@Fotolia](#)