

## Come sviluppare una navigazione dinamica e mantenibile

In questo articolo vedremo com'è possibile realizzare una **navigazione che sia in grado di evidenziare la pagina corrente** nelle voci del menù come in [questo esempio](#). Come puoi vedere, lo stile della voce del menù applicato alla pagina visualizzata è diverso dalle altre voci.

Per ottenere questo risultato in un modo elegante e soprattutto **mantenibile** dovremo costruire la navigazione "*dinamicamente*", ovvero attraverso la programmazione e nella fattispecie con PHP. Anche se i passaggi sono spiegati in modo approfondito, sono comunque necessarie delle conoscenze medie di questo linguaggio per poter seguire e comprendere appieno l'argomento trattato dall'articolo.

### Prepariamo le pagine (X)HTML

Iniziamo a preparare le pagine. Nulla di complesso, un elemento per la navigazione a sinistra ed uno per i contenuti a destra.

Iniziamo con la pagina *index.php*:

Come detto nulla di speciale. Ora puoi salvare questa pagina anche come *profile.php*, *contact.php*, *newsletter.php*, avendo cura di modificare il titolo contenuto nei tag *h1*.

### Il foglio di stile

Passiamo ora al foglio di stile (da me nominato *style.css*) che come puoi vedere l'ho già collegato al documento nell'head delle pagine appena realizzate.

Anche il foglio di stile è piuttosto semplice. Negli elementi della lista eliminiamo il marcatore per poi mostrarlo invece negli elementi con classe *active*. **Questa classe è quella che attribuiremo alla voce di menù relativa alla pagina visualizzata.**

### Rendere la navigazione mantenibile

Visto che, come detto precedentemente, la navigazione verrà creata tramite script PHP, è importante **tenere fuori dal codice elementi che potrebbero necessitare di aggiunte o modifiche**. Questo piccolo sforzo sull'altare della [mantenibilità](#) ci potrebbe rendere molto felici in futuro.

Crea dunque il file *navigation.txt* nel quale scrivere le voci del menù ed il relativo link separati dal segno ":".

## Sviluppare la funzione "Navigation()"

Crea ora il file *functions.php*. Ho scelto questo nome in quanto potremmo utilizzarlo non solo per generare la navigazione, ma anche per portarci appresso una serie di altre funzioni da utilizzare nelle nostre pagine.

Iniziamo a sviluppare la funzione. Quello che dovremo fare é:

- Stabilire in che pagina ci troviamo (*index.php, contact.php,...*);
- Leggere il contenuto del file *navigation.txt* ed elaborarlo in modo da disporre delle voci e dei relativi link;
- Verificare se il link corrisponde al nome della pagina visualizzata, nel qual caso dovremo aggiungere la classe *active*;

Ma prima di tutto é necessario che vi spieghi cos'è un **array** in quanto ci sarà molto utile. L'array é un tipo di variabile che contiene una collezione di *valori* associati ad una *chiave*. Detto così può sembrare molto complesso, dunque farò subito un esempio pratico.

Dichiariamo l'array *\$colori*

Ora, la variabile

restituirà *rosso* (il primo elemento di un array ha chiave 0), mentre

restituirà *giallo* (come dire che la *chiave* "2" ha *valore* "giallo")

Un array può essere anche di tipo *associativo*:

In questo caso la chiave non é più numerica ma inserita **esplicitamente**, dunque:

restituirà il nome *Maurizio*.

Mi fermo qui. In realtà un array può raggiungere un livello di complessità enorme; posso ad esempio inserire come valore di un array un altro array. Si parlerà in questo caso di *array multidimensionali*. Potremmo inoltre dire che ci sono altri modi di valorizzare e manipolare gli array, ma ora non ci interessa.

Iniziamo ora a dichiarare la funzione *Navigation()* ed a procurarci il **nome della pagina attiva**.

`$_SERVER` é una array associativo creato automaticamente, sempre e ovunque disponibile (tecnicamente si dice un *array superglobale*) che contiene diverse indicazioni riguardanti il server. La chiave `PHP_SELF` restituisce il percorso/nomeFile della pagina visualizzata.

Tramite la funzione `basename()` "depuriamo" il risultato dal percorso ottenendo così quello che vogliamo: **il nome della pagina visualizzata** che inseriremo nella variabile `$activePage`.

Dovremo ora leggere ed elaborare il contenuto del file `navigation.txt`. La via migliore é utilizzare la funzione `file()`.

Questa funzione legge il contenuto del file che passiamo come argomento e restituisce un array basato sulle righe di questo documento (ogni riga un valore).

Ad esempio `$rows[0]` conterrà `home:index.php`

A questo punto quello che dobbiamo fare é:

- Per ogni valore dell'array `$rows` "spezzare" la stringa sul segno ":" in modo da ottenere il nome da visualizzare nel menù e il link al quale dovrà puntare;
- Verificare se il nome della pagina corrisponde alla pagina visualizzata (se si aggiunge la classe `active`, se no stampare il link);

Per fare questo utilizzeremo un ciclo `foreach`, si tratta di un ciclo appositamente studiato per "sfogliare" gli array.

Prima di iniziare questo ciclo stampo il tag di apertura della lista.

In sintesi la sintassi di `foreach` significa questo:

"Esegui il codice contenuto nelle parentesi graffe (che non abbiamo ancora scritto) tante volte quanti sono i valori dell'array `$rows`. Ad ogni ciclo fai assumere alla variabile `$row` uno dei valori dell'array `$rows`."

Dunque al primo passaggio `$row` sarà uguale a `$rows[0]`, al secondo passaggio `$row` sarà uguale a `$rows[1]`, e così via.

Adesso dovremo "spezzare" sul ":" la stringa (nome:link). Utilizzeremo la funzione `explode()` che serve proprio a questo. Infatti `explode()` prende due argomenti: il separatore (nel nostro caso il ":") e la stringa. Restituisce un array i cui valori sono le parti di stringa spezzati.

Ad esempio al primo passaggio la situazione sarà questa:

Quindi `$result[0]` conterrà `home` e `$result[1]` conterrà `index.php`

Dunque il nostro ciclo sarà così:

Valorizzo *\$page* con il primo valore della riga e *\$link* con il secondo. Applico la funzione `trim()` per eliminare eventuali spazi vuoti prima e dopo.

E finalmente andiamo a verificare se si tratta della pagina attualmente visualizzata tramite la struttura di controllo `if/else`.

Se lo è stamperò il tag

*con classe active, altrimenti lo stamperò "liscio" ma avendo cura di dichiarare il link.*

*Una volta terminato il ciclo (quindi una volta stampata tutta la navigazione), chiuderemo il tag `ul`.*

Ed ecco la funzione completa:

Ora non resta che **includere il file `functions.php` in tutte le pagine ed invocare la funzione `Navigation()`**, in questo modo:

## Conclusione

In questo articolo abbiamo visto una semplice possibilità per **rendere la nostra navigazione dinamica** aggiungendo un *effetto active* al menù. Abbiamo inoltre implementato un metodo basilare per separare i *valori* della navigazione dal *codice* che la produce aumentando la [mantenibilità](#). Infatti, se hai necessità di aggiungere una voce al menù, non dovrai fare altro che aprire il file `navigation.txt` ed aggiungere la voce nella posizione desiderata. Aggiungere, rimuovere, spostare saranno operazioni molto semplici e veloci. Unico aspetto al quale prestare un minimo di attenzione: dopo l'ultima riga del file `navigation.php` **non dovranno esserci delle interruzioni di riga** in quanto potrebbero essere interpretati come una voce di navigazione vuota.