

## Personalizzare PrestaShop: la lista prodotti nel backend

*sabato, 22 luglio 2017*

PrestaShop è sicuramente una dei CMS più utilizzati per creare in maniera rapida e funzionale un e-commerce di successo, ma iniziando a lavorarci con frequenza ci si rende conto che spesso è necessario espanderne le funzionalità per renderne l'utilizzo più comodo ed agevole.

Ad esempio, ti è mai capitato di avere centinaia, se non migliaia di prodotti in catalogo e volerli filtrare per brand nel backend?



### Sommario

Elenco prodotti predefinito

L'override delle classi di PrestaShop

Ricostruiamo la cache delle classi

## Elenco prodotti predefinito

Chi ha utilizzato PrestaShop almeno una volta, sa che non possibile farlo nell'installazione di default. Sembra strano, eppure è proprio così!

The screenshot shows the PrestaShop 1.6.1.15 admin interface. The top navigation bar includes 'Dashboard', 'Catalog', 'Orders', 'Customers', 'Price Rules', 'Modules and Services', 'Shipping', 'Localization', 'Preferences', 'Advanced Parameters', and 'Administration'. The 'Products' page is active, displaying a dashboard with four key metrics: 'Out of stock items' at 13.33%, 'Average Gross Margin %' at 70%, 'Purchased references' at 0% of your Catalog, and 'Disabled Products' at 0%. Below the dashboard is a table of products with columns for ID, Image, Name, Reference, Category, Base price, Final price, Quantity, and Status. The table lists four products: Faded Short Sleeves T-shirt, Blouse, Printed Dress, and Printed Dress.

ID	Image	Name	Reference	Category	Base price	Final price	Quantity	Status
1		Faded Short Sleeves T-shirt	demo_1	T-shirts	\$16.51	\$16.51	1799	✓
2		Blouse	demo_2	Blouses	\$27.00	\$27.00	1799	✓
3		Printed Dress	demo_3	Casual Dresses	\$26.00	\$26.00	899	✓
4		Printed Dress	demo_4	Evening Dresses	\$50.99	\$50.99	900	✓

Ma niente paura, la soluzione è a portata di mano!

## L'override delle classi di PrestaShop

Il sistema di espansione utilizzato da PrestaShop è davvero molto molto semplice. Basta un minimo accenno ai concetti di ereditarietà delle classi, tra le prime nozioni di programmazione ad oggetti, che non approfondiremo però in questa sede per non perdere il focus della guida.

Nel caso ipotizzato, il **Controller** che si occupa di visualizzare l'elenco prodotti nella sezione admin è **AdminProductsController**, che si trova nella cartella:

```
controllers/admin/
```

nella root nello spazio FTP del tuo sito.

Non ci resta che "estenderlo" aggiungendo il campo che ci occorre. Per farlo, crea un nuovo file:

```
override/controllers/admin/AdminProductsController.php
```

ed aggiungi il seguente codice:

```
<?php

class AdminProductsController extends AdminProductsControllerCore
{

    public function __construct()
    {
        parent::__construct();
        $this->_join .= ' LEFT JOIN `'. _DB_PREFIX_ . 'manufacturer` m ON a.`id_manufacturer`
= m.id_manufacturer ';
        $this->fields_list['manufacturer'] = array(
            'title' => $this->l( 'Manufacturer' ),
            'filter_key' => 'm!name'
        );
    }

}
```

In pratica, dopo aver richiamato la funzione costruttore della classe "padre"

**AdminProductsControllerCore**, da cui deriva la nuova classe appena definita, abbiamo aggiunto una **join** alla query di default del Controller, così da avere accesso anche ai campi relativi ai produttori (manufacturer) ed estrapolarne il nome.

L'ultima istruzione, estende la lista di colonne da visualizzare nell'elenco, a cui aggiungiamo appunto la colonna "manufacturer" definendo il titolo e il corrispondente campo da cui estrapolare il valore, così come definito nella tabella appena collegata.

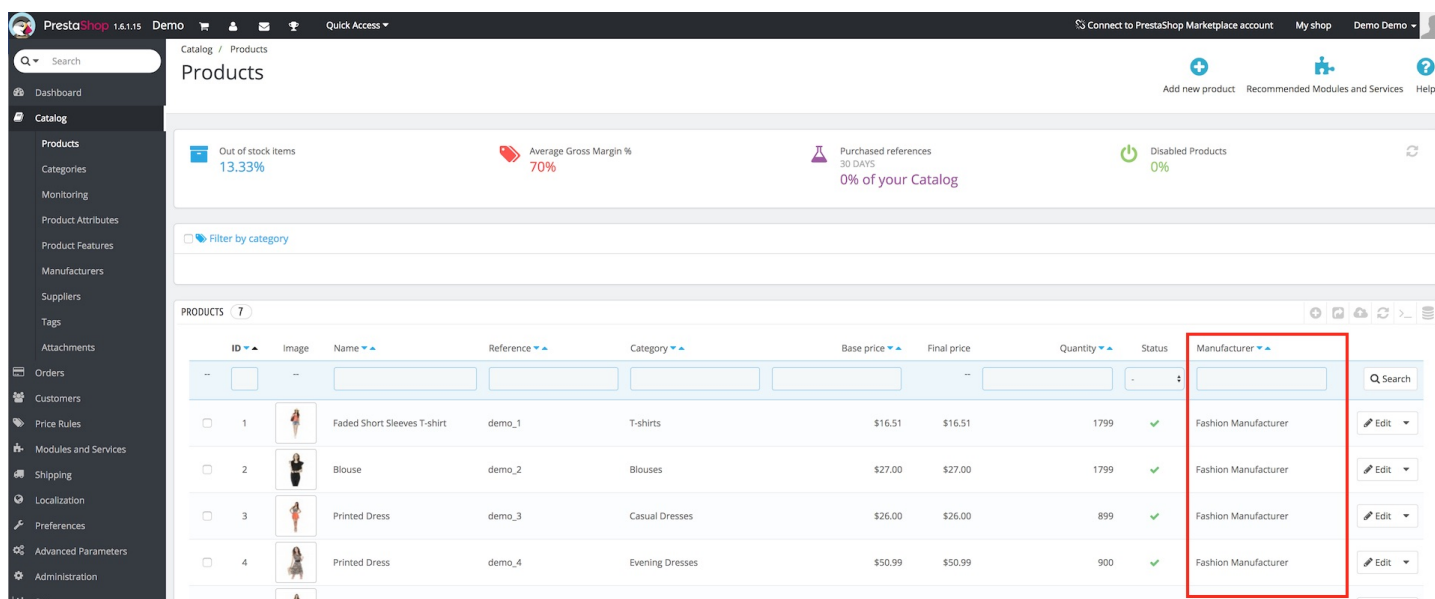
Ma un attimo! Ti starai chiedendo per quale motivo la colonna non è comparsa nell'elenco. Tranquillo, non hai fatto alcun errore (lo dico sulla fiducia, eh). Bisogna effettuare un ultimo piccolo passaggio per rendere effettive le modifiche.

## Ricostruiamo la cache delle classi

Tieni sempre a mente che ogni qual volta crei manualmente l'override di una classe di PrestaShop, bisogna far ricostruire la cache delle classi utilizzate. Per farlo, è necessario rimuovere il file:

`cache/class_index.php`

Basta un refresh della pagina e noterai che il file viene immediatamente rigenerato. Da questo momento, PrestaShop si accorderà delle modifiche da te effettuate ed il gioco è fatto!



The screenshot shows the PrestaShop back office interface. The top navigation bar includes 'PrestaShop 1.6.1.15 Demo', 'Quick Access', and 'Connect to PrestaShop Marketplace account'. The main content area is titled 'Products' and features a dashboard with three key metrics: 'Out of stock items 13.33%', 'Average Gross Margin % 70%', and 'Purchased references 30 DAYS 0% of your Catalog'. Below the dashboard is a 'Filter by category' section. The main table displays a list of products with columns for ID, Image, Name, Reference, Category, Base price, Final price, Quantity, Status, and a newly added 'Manufacturer' column. The 'Manufacturer' column is highlighted with a red box, and its values are 'Fashion Manufacturer' for all four products listed. The table also includes search and edit options for each row.

ID	Image	Name	Reference	Category	Base price	Final price	Quantity	Status	Manufacturer	
1		Faded Short Sleeves T-shirt	demo_1	T-shirts	\$16.51	\$16.51	1799	✓	Fashion Manufacturer	Edit
2		Blouse	demo_2	Blouses	\$27.00	\$27.00	1799	✓	Fashion Manufacturer	Edit
3		Printed Dress	demo_3	Casual Dresses	\$26.00	\$26.00	899	✓	Fashion Manufacturer	Edit
4		Printed Dress	demo_4	Evening Dresses	\$50.99	\$50.99	900	✓	Fashion Manufacturer	Edit

In questo modo, potrai ordinare o filtrare per brand, magari combinando il nuovo filtro con gli altri per estrapolare i dati che preferisci.