

Creating your own theme

Table of contents:

- [Creating your own theme](#)
 - [Getting Started](#)
 - [Cleaning up](#)
 - [Creating content](#)
 - [Design!](#)

Creating your own theme

Building a PrestaShop is a complex endeavour. For instance, in comparison with a WordPress theme, which could work with a single `index.php` file, a minimal PrestaShop requires many more pages: home page, product page, user account pages, cart page, order process pages, etc. Building a theme for an e-commerce site implies a much more complex and intertwined set of pages and templates.

This is why we advise to start your own theme by using the foundations laid by the default theme. Complete and proven, PrestaShop's default theme ensures that all necessary pages are already in places, leaving you the freedom to rework the page display, to use your own images, to enhance it with your own scripts.

Getting Started

The first step is plain in simple: make a straight copy of PrestaShop's default theme folder.

In PrestaShop 1.4, the default theme is called "prestashop", and is located in the `/themes/prestashop` folder.

In PrestaShop 1.5, the default theme is called "default", and is located in the `/themes/default` folder.

Give your copy of the folder its own name: it should be the final name of your theme, in a single lowercase word. Make sure to check on the Addons website that no other theme is already using that name, even more so if you plan to eventually sell that theme online.

Place the renamed folder in the `/themes` folder of your installation PrestaShop.

Cleaning up

Both the default theme and your installation of PrestaShop contain a payload of content and styles that are not necessarily useful to your own theme. For instance, many modules are installed and activated by default by PrestaShop. While some are necessary for the proper functioning of a complete store, others can simply be put aside while you build the theme.

The necessary module templates are:

Module name	Why it is necessary
blockcart	Displays the whole order & payment process.
blockmyaccount	Displays the user creation process.

You simply cannot sell a product if your theme does not support these modules.

There also are modules which, while not necessary for a functioning store, should still be included when designing a theme. You should try your best to build your theme with these modules in mind.

The "Must have" module templates are:

Module name	Why it is necessary
blockcategories	Displays the product categories.
blockcms	Lists and displays the CMS pages (i.e. Terms & Conditions, Legal notice, etc.).
blockcontact	Displays the Customer Service information.
blockcontactinfos	Displays the stores contact info.
blockmyaccountfooter	Displays links to the user's account pages in the footer.
blocksearch	Displays the search engine and its results.
blocktags	Displays the product tags.
homefeatured	Displays featured products.

Finally, these modules are not as important as the others ones, but bring a lot of value to your store, and helps your customers discover products and learn more about your store. Again, you should design your store with these modules activated.

The "Good to have" module templates are:

Module name	Why it is necessary
blockbestsellers	Displays the best-selling product.
blocklayered	Displays layered navigation filters.
blocklinks	Displays additional custom links.
blockmanufacturer	Lists and displays the manufacturers/brands of the store's products.
blocknewproductcs	Displays the newest products.
blocknewsletter	Displays a form where customers can subscribe to your store's newsletter.
blockrss	Displays the content of an RSS feed from another site.
blocksocial	Displays information about your store's social networking pages.
block specials	Displays the current discounts.
blocstore	Displays a link to the store located.
blocksupplier	Lists and displays the suppliers of the store's products.
blockviewed	Lists the products that the customer viewed last.
blockwishlist	Displays the customer's wishlists.
productcomments	Displays a comment section in each product page.

All these module templates are included by default in the default theme's `/modules` folder, because they are front-end features that are needed by that theme. You can safely disable/uninstall any other module in the back-office "Modules" page. This enables you to start on a somewhat clean slate.

i A fully clean slate would be to disable all modules and re-install them one by one, enabling you to integrate them into your design while building your theme. This is a good way to work, as you it helps you know which content broke your page layout, but it takes longer to reach your goal. Keep a known set of essential modules helps you build your theme faster while making sure it will work in most configuration.

The point here is that a theme must come packed with custom templates for the default modules. These template files are stored in the `/modules` folder of the theme's folder. For instance, in the default theme, the template file for the Layered Navigation module can be found in the following path: `/themes/default/modules/blocklayered/blocklayered.tpl` It is your duty as a theme developer to build templates for at least all the default PrestaShop modules, along with the additional modules that you plan on providing along with your theme's files. If all you want to change in a module's front-office appearance is its CSS file, you should put your customized version in this folder: `/themes/YOUR_THEME/css/modules/blocklayered/blocklayered.css`. Just make sure to use the same file path.

Creating content

Your theme will display content taken from the PrestaShop database. Whether you plan on keeping the theme to yourself or share/sell it for others to use, you simply cannot start designing it without content, along with the activation of some key features that any store might use, along with yourself.

The demo data installed with PrestaShop is enough to help with it, as it features products, categories, stores, etc. Starting with a fresh installation of PrestaShop gives you a head-start with demo content, while empty stores will require you to start adding content (either fake or real) to the store in order to actually see your theme react to it.

Design!

Now that the default theme has been turned into a folder of its own, it is time for you to explore its files: Smarty templates, CSS rules, JavaScript codes, location of the hooks and content blocks... Everything can be changed, and it is up to you to rework it the way you want!