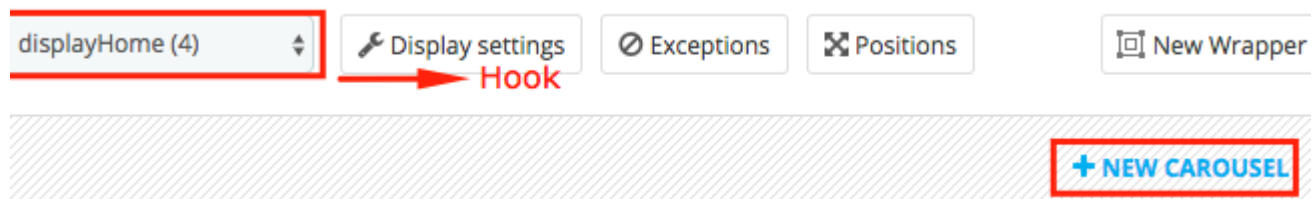# EASY CAROUSELS

Installation and configuration guide

Installation

Module is installed in a regular way. Simply upload your archive and click install. Once module is installed, you will have 3 carousels, grouped in tabs displayed on home page.

Adding new carousels

First of all you should select the hook, where carousel will be displayed. After that click 'New carousel', adjust settings and click 'Save'. That's it, carousel is ready.

Note: Carousels can be put together in wrappers. Each wrapper can have a custom class, and each carousel can have its own class as well, so you can build complex and responsive layouts using these classes.

Carousel settings

Most settings are self explanatory. Here is the description for some specific options:

Visible rows: If you want to rotate more than one row in a single carousel, you can increase this value. In this case carousel will look like rotating a grid of items.

Normalize heights: This option activates additional script that forces same height for all elements within carousel. In most cases this option is not required. You should use it only if layout is broken by element heights difference.

Min slide width (px): Carousel layout is dynamically adjusted for current screen resolution. You can set different number of visible items for different resolutions. In some cases items can get too narrow in order to fit in container width. For example if you want to fit 5 items in 480px, each item should be not more than 96px. This is obviously not enough for displaying product information. So, if you set min slide width to 200px, the script will decrease number of displayed elements in order to make sure, each of them has a minimal width of 200px.
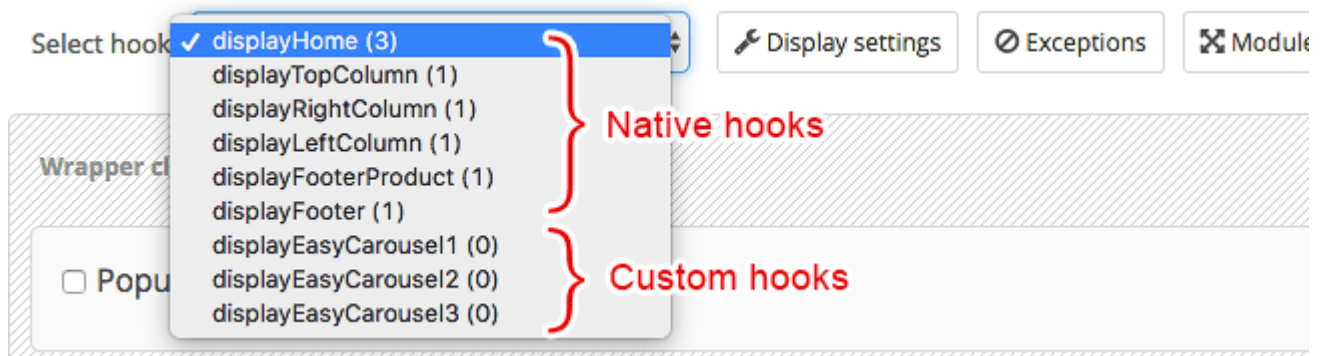
Special settings for mobile devices

You can adjust settings for better performance on mobile devices. For example, instead of loading **standard carousel script** you can arrange items as a **horizontal scrollable list**, that behaves very smooth on mobile devices. Or, you can load **small_deault** images instead of **home_default**, etc...

Available hooks

Carousels can be placed in native Prestashop hooks or custom hooks ('displayEasyCarousel1/2/3').
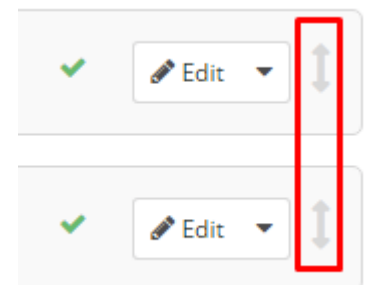

Native hooks can be found in standard positions in most of themes. Custom hooks can be added anywhere in your tpl files. For example, you want to display your carousel just above product listing on category page. Most themes don't have native hooks, available for that position. Here is what you should do:

1) Select hook 'displayEasyCarousel1'
2) Add carousel(s) to this hook
3) Open the category.tpl, and insert the following code just above the place where product-list is rendered: {hook h='displayEasyCarousel1'}.

In the same way you can add 2 other custom hooks to different positions in your theme.

Carousels ordering within a hook

When you have more than one carousel in a hook, you might want to change the ordering. That's pretty simple. In order to do that, you just click on the draggable area on the right of a carousel block and move it among other blocks using drag-n-drop. New ordering is applied dynamically. Page reload is not required

Customizing carousel templates

You can override module files in a standard way, by placing customized versions of these files in your theme folder.

For example, if you want to override /views/templates/hook/carousel.tpl, you should add customized version here: /themes/your_theme/modules/easycarousels/views/templates/hook/carousel.tpl

Since v2.5.0 some template files have extendable structure. It means you can override only selected blocks of those files. No need to override whole files.

For example you need custom layout only for product buttons. In this case you should add the following code to /product-item-16.tpl, located in your theme:

```
{extends file='modules/easycarousels/views/templates/hook/product-item-16.tpl'}

{block name='product_buttons'}
    {* your custom buttons layout here*}
{/block}
```

Here is the list of extendable template files:

- /views/templates/hook/item.tpl *(used for displaying categories/manufacturers/suppliers)*
- /views/templates/hook/product-item-16.tpl *(used to display products in PS 1.6)*
- /views/templates/hook/product-item-17.tpl *(used to display products in PS 1.7)*

Keeping customized files up-to-date

Sometimes, after upgrading the module you can get this kind of notification:

It means, that you have custom version of /product-item-16.tpl in your theme folder, and original file was upgraded in the new module version. Most probably some new features/variables were added to this file, so in order to benefit from those new features you should update the customized version of the file.

So, you open the original file, and copy new features/variables to your customized file. In some cases you don't need to copy anything. But in either case you should add comment {* since x.x.x *} to the last line of customized file. This comment identifies that you have checked your custom file, and warning will not be displayed anymore.

Exporting/Importing carousels

In order to export all current carousel data, just click "Export carousels" and save the file. This file contains all multilingual data including hook positions and page exceptions.

In order to import, just upload the file using "Import carousels" button. You can use this file on the same store as a backup, or you can upload it to any other store. When you upload the file, data is processed in a smart way to synchronize with installed languages/shops.

Note: If you are using multistore, data is imported only to shops that are currently selected. It may be a single shop, a group of shops, or all shops.

If you have any questions or suggestions, feel free to contact the developer