

Highlight widgets usage

Contents

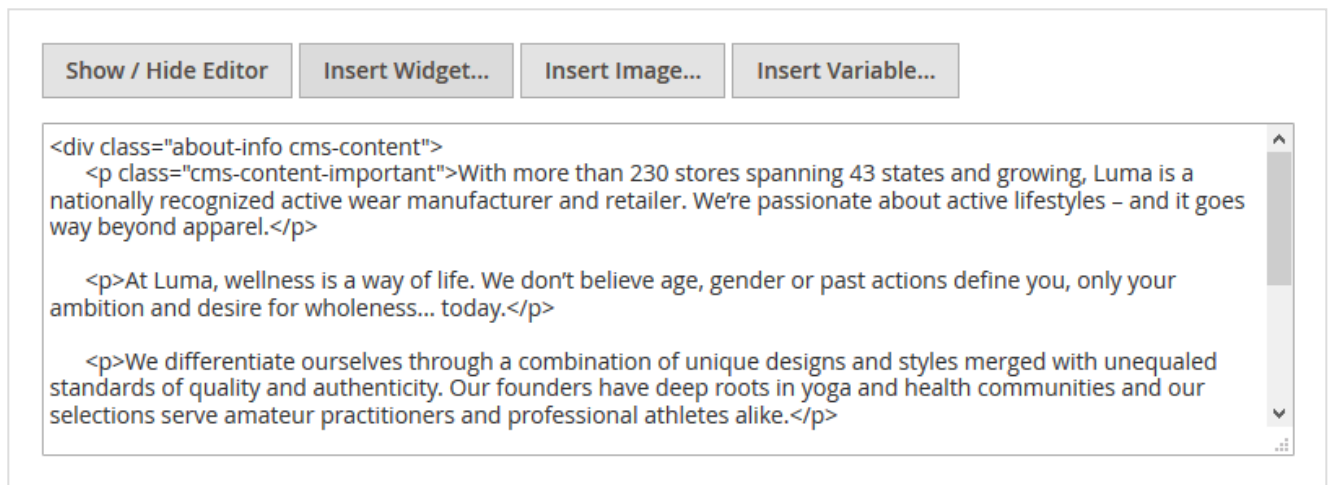
1. [WYSIWYG Editor](#)
2. [Widgets Page](#)
3. [XML Layout Update](#)
4. [PHP Inline Code](#)

WYSIWYG Editor

Highlight widgets may be inserted into custom content using Magento's built-in WYSIWYG Editor. This method is recommended to insert widget into cms pages and blocks. It also may be used with any our or third-party extension that utilize `Insert Widget` button.

Example below shows how to add widget into CMS Page content

1. Press `Insert Widget` button.



2. Select one of available highlight widgets in `Widget Type` dropdown.

Insert Widget...

Widget

Widget Type *

- Please Select --
- Highlight: All Products
- Highlight: Attribute Date
- Highlight: Attribute Yes/No
- Highlight: Bestsellers
- Highlight: Featured Products
- Highlight: New Products
- Highlight: Onsale Products (Special Price)
- Highlight: Popular Products

Insert Widget

3. Fill **Widget Options** form with values you'd like to. See the [Widget Settings](#) page, if you are not sure about some options.

Widget Options

Title

Use Ajax Carousel

Number of Products to Display *

Number of Columns to Display *
Used, when template value is set to 'Grid'

Number of Pages to Display *

Sort Order

Sort Direction *

Template

Listing Mode	Grid	▼
	Sidebar templates does not support this option	
Custom Template	<input type="text"/>	
	Used, when template value is set to 'Custom'	
Conditions *	If ALL of these conditions are TRUE :	
	⊕	
Cache Lifetime (Seconds)	<input type="text"/>	
	86400 by default, if not set. To refresh instantly, clear the Blocks HTML Output cache.	
Hidden, when filter or pagination is used	No	▼
CSS Class	<input type="text"/>	
	Class names for customization purposes. You can view ready to use css class helpers at http://docs.swissuplabs.com/m2/extensions/highlight/widgets/css-helpers/	

4. Press `Insert Widget` button at the right top corner of widget popup.
5. Save your cms page.




Widgets Page

Magento's `Content > Widgets` page allows easily add custom content to almost all frontend pages. This method is highly recommended to insert highlight widgets.

Example below shows how to add widget to the Product Page

1. Navigate to `Content > Widgets` and press `Add Widget` button in the right top corner of the page.

Widgets



 admin ▾

Add Widget

Search
Reset Filter

2 records found 20 ▾ per page < 1 of 1 >

Widget ID ↓	Widget	Type	Design Theme	Sort Order
<input type="text"/>	<input type="text"/>	<input style="width: 100%;" type="text"/>	<input style="width: 100%;" type="text"/>	<input type="text"/>
1	Contact us info	CMS Static Block	Magento Luma	0
2	Footer Links	CMS Static Block	Magento Luma	0

- Select one of available highlight widgets in `Widget Type` dropdown and your theme in `Design Package/Theme` dropdown and press `Continue`.

Settings

Type * ▾

Design Theme * ▾

Continue

- Now you should fill the form with two tabs: `Frontend Properties` and `Widget Options`. Lets fill the first one:

Storefront Properties

Type

Design Package/Theme

Widget Title *

Assign to Store Views *
Main Website
Main Website Store
Default Store View



Sort Order

Sort Order of widget instances in the same container

Layout Updates

Display on

Products	Container	Template
<input checked="" type="radio"/> All <input type="radio"/> Specific Products	<input type="text" value="Main Content Area"/>	<input type="text" value="Grid Template"/>

Add Layout Update

And the second one:

Widget Options

Title

Use Ajax Carousel

Number of Products to Display *

Number of Columns to Display *

Used, when template value is set to 'Grid'

Number of Pages to Display *

Sort Order

Sort Direction *



Listing Mode

Sidebar templates does not support this option

Custom Template

Used, when template value is set to 'Custom'

Conditions * If ANY of these conditions are TRUE :

- Category is current+ 
- 

Cache Lifetime (Seconds)

86400 by default, if not set. To refresh instantly, clear the Blocks HTML Output cache.

Hidden, when filter or pagination is used

CSS Class

Class names for customization purposes. You can view ready to use css class helpers at <http://docs.swissuplabs.com/m2/extensions/highlight/widgets/css-helpers/>

See the [Widget Settings](#) page, if you are not sure about some options.

4. Save widget and navigate to product page.

XML Layout Update

- See the [Available Widgets](#) page, for full block type reference.
- See the [Widget Settings](#) page, for full options reference.

Here is an xml layout skeleton, that could be used to write code to call any kind of highlight widget:

```
<referenceContainer name="content">
  <block class="Swissup\Highlight\Block\ProductList\Featured" name="highlight.featured">
    <action method="setTemplate">
      <argument name="template"
xsi:type="string">Swissup_Highlight::product/widget/column/list.phtml</argument>
    </action>
    <action method="setTitle">
      <argument name="title" xsi:type="string">Featured Products</argument>
    </action>
    <action method="setProductsCount">
      <argument name="products_count" xsi:type="number">4</argument>
    </action>
    <action method="setCssClass">
      <argument name="css_class" xsi:type="string">highlight-featured</argument>
    </action>
    <action method="setPageTitle">
      <argument name="page_title" xsi:type="string">See All Featured Products</argument>
    </action>
  </block>
</referenceContainer>
```

PHP Inline Code

- See the [Available Widgets](#) page, for full block type reference.
- See the [Widget Settings](#) page, for full options reference.

Here is an xml layout skeleton, that could be used to write code to call any kind of highlight widget:

```
<?php
echo $this->getLayout()
    ->createBlock('Swissup\Highlight\Block\ProductList\All')
    ->setTemplate('Swissup_Highlight::product/widget/content/grid.phtml')
    ->setProductsCount(20)
    ->setTitle('Featured Products')

    // show pager
    ->setShowPager(true)
    ->setProductsPerPage(5)
    ->setPageVarName('page-highlight-all')

    // show link to custom page
    ->setShowPageLink(true)
    ->setPageLinkTitle('See all Featured Products')
    ->setPageUrl('test/test.html')

    // render block
    ->toHtml();
?>
```