

Code Style. PHP

File | Settings | Project Settings | Code Style - PHP



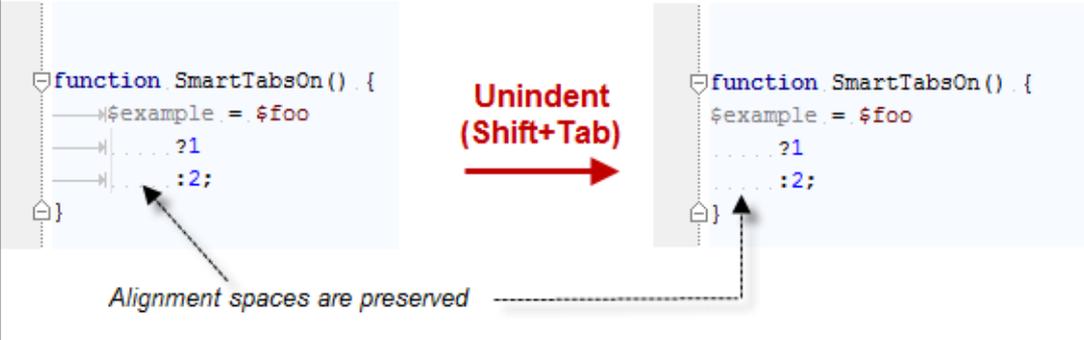
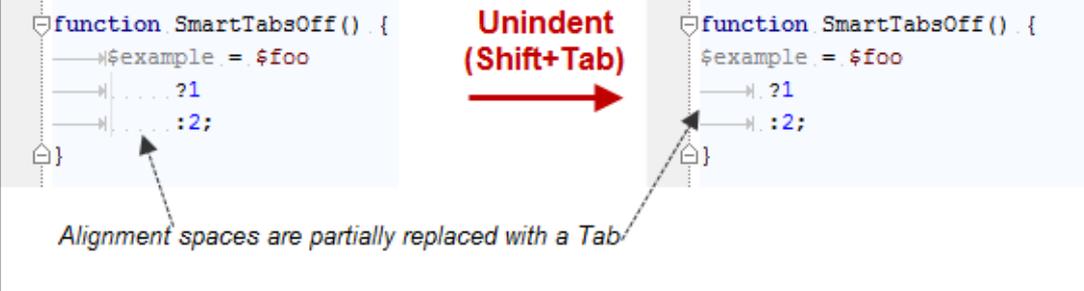
Use this page to configure [formatting](#) options for PHP files. View the result in the **Preview** pane on the right.

On this page:

- [Tabs and Indents](#)
- [Spaces](#)
- [Wrapping and Braces](#)
- [Blank Lines](#)
- [Arrangement](#)
- [PHPDoc](#)
- [Other](#)
- [Set From](#)

Tabs and Indents

Item	Description
Use tab character	<ul style="list-style-type: none">■ If this check box is selected, tab characters are used:<ul style="list-style-type: none">■ On pressing the Tab key■ For indentation■ For code reformatting■ When the check box is cleared, IntelliJ IDEA uses spaces instead of tabs.

Item	Description
Smart tabs	<ul style="list-style-type: none"> <li data-bbox="352 192 1414 322">■ If this check box is selected, IntelliJ IDEA inserts tabs for indentation and reformatting, but fine alignment to a necessary column is done <i>only</i> via spaces without replacing them with tabs. This is done in order to preserve visual representation of the source code, when the Tab Size is changed. <div data-bbox="395 349 1489 808" style="border: 1px solid black; padding: 10px; text-align: center;"> <p>Smart Tabs check box is selected</p>  <p>The diagram shows two code snippets for a function <code>SmartTabsOn()</code>. The left snippet shows code with spaces for alignment: <code>\$example = \$foo</code>, <code>... ?1</code>, and <code>... :2;</code>. A red arrow labeled "Unindent (Shift+Tab)" points to the right snippet, where the same code is shown but with tabs inserted for indentation. A dashed arrow points from the text "Alignment spaces are preserved" to the spaces in the left snippet.</p> </div> <ul style="list-style-type: none"> <li data-bbox="352 824 1493 920">■ If this check box is cleared, only tabs are used. This means that a group of spaces that fits the specified tab size is automatically replaced with a tab, which may result in breaking fine alignment. <div data-bbox="395 947 1489 1357" style="border: 1px solid black; padding: 10px; text-align: center;"> <p>Smart Tabs check box is not selected</p>  <p>The diagram shows two code snippets for a function <code>SmartTabsOff()</code>. The left snippet shows code with spaces for alignment: <code>\$example = \$foo</code>, <code>... ?1</code>, and <code>... :2;</code>. A red arrow labeled "Unindent (Shift+Tab)" points to the right snippet, where the same code is shown but with tabs inserted for indentation, breaking the fine alignment. A dashed arrow points from the text "Alignment spaces are partially replaced with a Tab" to the spaces in the left snippet.</p> </div> <p data-bbox="347 1379 1393 1442">The Smart Tabs check box is available if the Use Tab Character check box is selected.</p>
Tab size	In this text box, specify the number of spaces included in a tab.
Indent	In this text box, specify the number of spaces (or tabs if the Use Tab Character check box is selected) to be inserted for each indent level.

Spaces

Use this tab to specify where you want spaces in your code. To have IntelliJ IDEA automatically insert a space at a location, select the check box next to this location in the list. The results are displayed in the **Preview** pane.

Wrapping and Braces

In this tab, customize the code style options, which IntelliJ IDEA will apply on [reformatting the source code](#). The left-hand pane contains the list of exceptions (**Keep when reformatting**), and placement and alignment options for the various code constructs (lists, statements, operations, annotations, etc.). The right-hand pane shows preview.

Alignment takes precedence over [indentation options](#).

Keep when reformatting

Use the check boxes to configure exceptions that IntelliJ IDEA will make when reformatting the source code. For example, by default, the *Line brakes* check box is selected. If your code contains lines that are shorter than a standard convention, you can convert them by disabling the *Line brakes* check box before you [reformat the source code](#).

Blank Lines

Use this tab to define where and how many blank lines you want IntelliJ IDEA to retain and insert in your code after reformatting. For each type of location, specify the number of blank lines to be inserted. The results are displayed in the **Preview** pane.

Item	Description
Keep Maximum Blank Lines	In this area, specify the number of blank lines to be kept after reformatting in the specified locations.
Minimum Blank Lines	In the text boxes in this area, specify the number of blank lines to be present in the specified locations. <div style="border: 1px solid orange; padding: 5px; margin-top: 10px;">These settings do not influence the number of blank lines before the first and after the last item.</div>

Arrangement

This tab lets you define a set of rules that rearranges your code according to your preferences.

Item	Description
Grouping Rules	Use this area to set the grouping rules. <ul style="list-style-type: none">■ <i>Keep getters and setters together</i> Select this check box to keep getter and setter methods together. By default, this check box is selected.■ <i>Keep overridden methods together</i> Select this check box to group the overridden methods together by class and interface. In <i>order</i>: list, select <i>keep</i> or <i>by name</i> options.■ <i>Keep dependent methods together</i> Select this check box to group the dependent methods together. In <i>order</i>: list, select <i>depth-first</i> or <i>breadth-first</i> options.

Item	Description
Matching rules	<p>Use this area to define elements order as a list of rules, where every rule has a set of matches such as modifier or type.</p> <ul style="list-style-type: none"> ■  - use this button to add a rule. The empty rule dialog window opens. ■  - use this button to remove the rule from the list. ■  - use this button to edit an existing rule. To see this button, navigate to the rule that you want to edit and click on the button. In pop-up window that opens, modify the rule fields. ■  - use these buttons to move the selected rule up or down.
Empty rule	<p>Use this window to create a new matching rule or edit an existing one. You can select from the following filters:</p> <ul style="list-style-type: none"> ■ Type - use this filter to choose classes or methods for your rule. ■ Name - use this field to specify entry names in the rule. This filter matches only entry names such as field names, method names, class names and etc. The filter supports regular expressions and uses a standard syntax. The match is performed against the entire name. ■ Order - use this option to select the sorting order for the rule. This option is useful when more than one element uses the same matching rule. In this case, selecting <i>Keep order</i> will keep the same order as was set before the rearrangement and selecting <i>Order by Name</i> will sort the elements with the same matching rule by their names.
	<p>This icon appears when you select <i>Order by Name</i> in Order option. The icon indicates that the items in this rule are alphabetized.</p>

PHPDoc

In this tab, configure the code style to be applied inside [PHPDoc](#) comments.

Item	Description
Align parameter names	Select this check box to have the <code>&lt;paramname></code> elements aligned.
Keep blank lines	Select this check box to suppress removing blank lines automatically.
Blank lines around parameters	Select this check box to have a blank line inserted above and below the section with <code>@param</code> tags.
Blank line before the first tag	Select this check box to have an blank line inserted above the first PHPDoc tag.
Align tag comments	Select this check box to have the <code>description</code> elements aligned.

Other

Item	Description
Indent code in PHP tags	Select this check box to have the code enclosed in <code><?php></code> tags indented against the opening <code><?php</code> tag.
Align key-value pairs	Select this check box to have the <code>=></code> separators in key-value assignments aligned.
Align consecutive assignments	Select this check box to have the <code>=</code> operators in two or more consecutive assignments aligned.
Convert True/False constants to upper case	Select this check box to have the <code>true</code> and <code>false</code> constants displayed in the upper case.
Convert Null constant to upper case	Select this check box to have the <code>null</code> constant displayed in the upper case.

Set From

Click this link to choose the base for the current language default code style from the pop-up list, that appears. The list contains two options:

- **Language:** choose this option to inherit the coding style settings from another language. Select the source language from the list, that opens. So doing, only the settings that are applicable to the current language are taken. All the other settings are not affected.
- **Predefined code style:** choose this option to use the coding standards defined for a specific framework. Select one of the following frameworks from the list:
 - [PEAR](#)
 - [Zend](#)
 - [Drupal](#)

This link appears in the upper-right corner of the language-specific code style page, when applicable.

Click **Reset** to discard changes and return to the initial set of code style settings.

See Also

Procedures:

- [PHP-Specific Guidelines](#)
- [Configuring Code Style](#)

Reference:

- [Code Style](#)

Web Resources:

- [Developer Community](#)