



- [Downloads](#)
- [Documentation](#)
- [Get Involved](#)
- [Help](#)
- 

[Dutch PHP Conference 2024](#)

[Getting Started](#)

[Introduction](#)

[A simple tutorial](#)

[Language Reference](#)

[Basic syntax](#)

[Types](#)

[Variables](#)

[Constants](#)

[Expressions](#)

[Operators](#)

[Control Structures](#)

[Functions](#)

[Classes and Objects](#)

[Namespaces](#)

[Enumerations](#)

[Errors](#)

[Exceptions](#)

[Fibers](#)

[Generators](#)

[Attributes](#)

[References Explained](#)

[Predefined Variables](#)

[Predefined Exceptions](#)

[Predefined Interfaces and Classes](#)

[Predefined Attributes](#)

[Context options and parameters](#)

[Supported Protocols and Wrappers](#)

[Security](#)

[Introduction](#)

[General considerations](#)

[Installed as CGI binary](#)

[Installed as an Apache module](#)

[Session Security](#)

[Filesystem Security](#)

[Database Security](#)

[Error Reporting](#)

[User Submitted Data](#)

[Hiding PHP](#)

[Keeping Current](#)

[Features](#)

[HTTP authentication with PHP](#)

[Cookies](#)

[Sessions](#)

[Dealing with XForms](#)

[Handling file uploads](#)

[Using remote files](#)

[Connection handling](#)

[Persistent Database Connections](#)

[Command line usage](#)

[Function Reference](#)

[Affecting PHP's Behaviour](#)
[Audio Formats Manipulation](#)
[Authentication Services](#)
[Command Line Specific Extensions](#)
[Compression and Archive Extensions](#)
[Cryptography Extensions](#)
[Database Extensions](#)
[Date and Time Related Extensions](#)
[File System Related Extensions](#)
[Human Language and Character Encoding Support](#)
[Image Processing and Generation](#)
[Mail Related Extensions](#)
[Mathematical Extensions](#)
[Non-Text MIME Output](#)
[Process Control Extensions](#)
[Other Basic Extensions](#)
[Other Services](#)
[Search Engine Extensions](#)
[Server Specific Extensions](#)
[Session Extensions](#)
[Text Processing](#)
[Variable and Type Related Extensions](#)
[Web Services](#)
[Windows Only Extensions](#)
[XML Manipulation](#)
[GUI Extensions](#)

Keyboard Shortcuts

? This help
j Next menu item
k Previous menu item
g p Previous man page
g n Next man page
G Scroll to bottom
g g Scroll to top
g h Goto homepage
g s Goto search
(current page)
/ Focus search box

[while »](#)
[« elseif/else if](#)

- [Руководство по PHP](#)
- [Справочник языка](#)
- [Управляющие конструкции](#)

Альтернативный синтаксис управляющих структур

(PHP 4, PHP 5, PHP 7, PHP 8)

PHP предлагает альтернативный синтаксис для некоторых его управляющих структур, а именно: `if`, `while`, `for`, `foreach` и `switch`. В каждом случае основной формой альтернативного синтаксиса является изменение открывающей фигурной скобки на двоеточие (`:`), а закрывающей скобки на `endif;`, `endwhile;`, `endfor;`, `endforeach;` или `endswitch;` соответственно.

```
<?php if ($a == 5): ?>
А равно 5
<?php endif; ?>
```

В приведённом выше примере, блок HTML "А равно 5" вложен внутрь структуры `if`, написанной с альтернативным синтаксисом. Блок HTML будет показан только если переменная `$a` равна 5.

Альтернативный синтаксис также применяется и к `else` и `elseif`. Ниже приведена структура `if` с `elseif` и `else` в альтернативном формате:

```
<?php
if ($a == 5):
echo "а равно 5";
echo "...";
elseif ($a == 6):
echo "а равно 6";
echo "!!!";
else:
echo "а не равно ни 5 ни 6";
endif;
?>
```

Замечание:

Смешивание синтаксиса в одном и том же блоке управления не поддерживается.

Внимание

Любой вывод (включая пробельные символы) между выражением `switch` и первым `case` приведут к синтаксической ошибке. Например, данный код не будет работать:

```
<?php switch ($foo): ?>
<?php case 1: ?>
...
<?php endswitch; ?>
```

В то же время следующий пример будет работать, так как завершающий перевод строки после выражения `switch` считается частью закрывающего `?>` и следовательно ничего не выводится между `switch` и `case`:

```
<?php switch ($foo): ?>
<?php case 1: ?>
...
<?php endswitch; ?>
```

Смотрите также [while](#), [for](#) и [if](#) для дальнейших примеров.

[+add a note](#)

User Contributed Notes 1 note

[up](#)
[down](#)
20
[toxyy](#) 1
1 year ago

I feel compelled to give a more elegant way using heredoc than the other comment:

```
<ul>
<?php foreach($list as $item): echo
<<<ITEM
<li id="itm-$item[number]">Item $item[name]</li>
ITEM;
endforeach; ?>
</ul>
```

Which works better with multi line blocks, as you only need one overall php tag.

(please don't omit the closing tag despite it being legal, personal preference)

[+add a note](#)

- [Управляющие конструкции](#)
 - [Введение](#)
 - [if](#)
 - [else](#)
 - [elseif/else if](#)
 - [Альтернативный синтаксис управляющих структур](#)
 - [while](#)
 - [do-while](#)
 - [for](#)
 - [foreach](#)
 - [break](#)
 - [continue](#)
 - [switch](#)
 - [match](#)
 - [declare](#)
 - [return](#)
 - [require](#)
 - [include](#)
 - [require_once](#)
 - [include_once](#)
 - [goto](#)
- [Copyright © 2001-2024 The PHP Group](#)
- [My PHP.net](#)
- [Contact](#)
- [Other PHP.net sites](#)
- [Privacy policy](#)

