




- [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](#)

[Garbage Collection](#)
[DTrace Dynamic Tracing](#)

[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

[require »](#)

[« declare](#)

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

Change language: Russian

return

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

`return` возвращает управление программой модулю, из которого была вызвана функция. Выполнение программы продолжается с инструкции, следующей за местом вызова.

Если вызывается из функции, выражение `return` немедленно прекращает выполнение текущей функции и возвращает свой аргумент как значение данной функции. `return` также завершит выполнение выражения [eval\(\)](#) или всего файла скрипта.

Если вызывается из глобальной области видимости, выполнение текущего файла скрипта прекращается. Если текущий файл скрипта был подключён с помощью функций [include](#) или [require](#), тогда управление возвращается к файлу, который вызывал текущий. Более того, если текущий файл скрипта был подключён с помощью [include](#), тогда значение переданное `return` будет возвращено в качестве значения вызова [include](#). Если `return` вызывается из главного файла скрипта, тогда выполнение скрипта прекращается. Если текущий файл скрипта был вызван конфигурационными опциями [auto_prepend_file](#) или [auto_append_file](#) из файла настроек *php.ini*, тогда выполнение этого скрипта прекращается.

Для более детальной информации смотрите раздел [Возвращаемые значения](#).

Замечание: Заметьте, что `return` является языковой конструкцией, а не функцией и круглые скобки, окружающие аргументы, не являются необходимостью. Мало того, они тут не приветствуются.

Замечание: Если параметры не указаны, тогда круглые скобки должны быть опущены и будет возвращён `null`. Вызов `return` со скобками, но без аргументов вызовет синтаксическую ошибку.

Начиная с PHP 7.1.0, операторы возврата без аргумента в функциях, объявляющих тип возврата, вызывают `E_COMPILE_ERROR`, если только тип возврата не `void`, в этом случае такую же ошибку вызывают операторы возврата с аргументом.

[+ add a note](#)

User Contributed Notes 2 notes

[up](#)

[down](#)

180

[warhog at warhog dot net ¶](#)

18 years ago

for those of you who think that using `return` in a script is the same as using `exit` note that: using `return` just exits the execution of the current script, `exit` the whole execution.

look at that example:

```
a.php
<?php
include("b.php");
echo "a";
?>
```

```
b.php
<?php
echo "b";
return;
?>
```

(executing a.php:) will echo "ba".

whereas (b.php modified):

```
a.php
```

```
<?php
include("b.php");
echo "a";
?>
```

```
b.php
<?php
echo "b";
exit;
?>
```

(executing a.php:) will echo "b".

[up](#)

[down](#)

73

[J.D. Grimes](#)

10 years ago

Note that because PHP processes the file before running it, any functions defined in an included file will still be available, even if the file is not executed.

Example:

```
a.php
<?php
include 'b.php';
```

```
foo();
?>
```

```
b.php
<?php
return;
```

```
function foo() {
echo 'foo';
}
?>
```

Executing a.php will output "foo".

[+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](#)

