




- [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

[Подпространства имён »](#)
[« Обзор](#)

- [Руководство по PHP](#)
- [Справочник языка](#)
- [Пространства имён](#)

Change language: Russian ▾

Определение пространств имён

(PHP 5 >= 5.3.0, PHP 7, PHP 8)

Хотя любой корректный PHP-код разрешается размещать внутри пространства имён, только классы (включая абстрактные и трейты), интерфейсы, функции и константы зависят от него.

Пространства имён объявляют, указывая зарезервированное слово `namespace`. Пространства имён в файлах объявляют в начале, перед любым другим кодом, кроме зарезервированного слова [declare](#).

Пример #1 Объявление единого пространства имён

```
<?php

namespace MyProject;

const CONNECT_OK = 1;
class Connection { /* ... */ }
function connect() { /* ... */ }

?>
```

Замечание: Абсолютные имена (т. е. имена, которые начинаются с обратной косой черты) не допускаются в объявлениях пространства имён, поскольку языковые конструкции с начальным слешем интерпретируются как относительные выражения пространства имён.

Только выражение `declare` разрешено помещать перед объявлением пространства имён, чтобы определить кодировку исходного файла. В дополнение, никакой код, кроме PHP-кода, нельзя размещать перед объявлением пространства имён, включая лишние пробелы:

Пример #2 Объявление простого пространства имён

```
<html>
<?php

namespace MyProject; // fatal error — объявление пространства имён должно быть первым выражением в скрипте

?>
```

Кроме того, одно и то же пространство имён, в отличие от остальных конструкций PHP, допустимо определять в нескольких файлах, что помогает распределять содержание файлов по файловой системе.

[+add a note](#)

User Contributed Notes 10 notes

[up](#)
[down](#)
203

[kuzawinski dot marcin at NOSPAM dot gmail dot com ¶](#)
9 years ago

If your code looks like this:

```
<?php
namespace NS;

?>
```

...and you still get "Namespace declaration statement has to be the very first statement in the script" Fatal error, then you probably use UTF-8 encoding (which is good) with Byte Order Mark, aka BOM (which is bad). Try to convert your files to "UTF-8 without BOM", and it should be ok.

[up](#)
[down](#)
144

[*danbattles at yahoo dot co dot uk ¶*](#)

14 years ago

Regarding constants defined with `define()` inside namespaces...

`define()` will define constants exactly as specified. So, if you want to define a constant in a namespace, you will need to specify the namespace in your call to `define()`, even if you're calling `define()` from within a namespace. The following examples will make it clear.

The following code will define the constant "MESSAGE" in the global namespace (i.e. `"\MESSAGE"`).

```
<?php
namespace test;
define('MESSAGE', 'Hello world!');
?>
```

The following code will define two constants in the "test" namespace.

```
<?php
namespace test;
define('test\HELLO', 'Hello world!');
define(__NAMESPACE__ . '\GOODBYE', 'Goodbye cruel world!');
?>
```

[up](#)

[down](#)

82

[*FatBat ¶*](#)

10 years ago

Expanding on @danbattles note, it is better to always be explicit about which constant to use.

```
<?php
namespace NS;

define(__NAMESPACE__ . '\foo', '111');
define('foo', '222');

echo foo; // 111.
echo \foo; // 222.
echo \NS\foo // 111.
echo NS\foo // fatal error. assumes \NS\NS\foo.
?>
```

[up](#)

[down](#)

7

[*aniszgzig at gmail dot com ¶*](#)

3 years ago

namespace statement is defined at first of the php files. But before namespace declaration only three elements allowed.

1.declare statement

2.spaces

3.comments

[up](#)

[down](#)

65

[*huskyr at gmail dot com ¶*](#)

14 years ago

"A file containing a namespace must declare the namespace at the top of the file before any other code"

It might be obvious, but this means that you *can* include comments and white spaces before the namespace keyword.

```
<?php
// Lots
// of
```

```
// interesting
// comments and white space
```

```
namespace Foo;
class Bar {
}
?>
```

[up](#)

[down](#)

46

[jeremeamia at gmail dot com ¶](#)

14 years ago

You should not try to create namespaces that use PHP keywords. These will cause parse errors.

Examples:

```
<?php
namespace Project/Classes/Function; // Causes parse errors
namespace Project/Abstract/Factory; // Causes parse errors
?>
```

[up](#)

[down](#)

6

[Anonymous ¶](#)

15 years ago

@ RS: Also, you can specify how your __autoload() function looks for the files. That way another users namespace classes cannot overwrite yours unless they replace your file specifically.

[up](#)

[down](#)

-1

[Baptiste ¶](#)

15 years ago

There is nothing wrong with PHP namespaces, except that those 2 instructions give a false impression of package management.

... while they just correspond to the "with()" instruction of Javascript.

By contrast, a package is a namespace for its members, but it offers more (like deployment facilities), and a compiler knows exactly what classes are in a package, and where to find them.

[up](#)

[down](#)

-1

[anisgazig at gmail dot com ¶](#)

3 years ago

Namespace name are case-insensitive.

```
namespace App
```

and

```
namespace app
```

are same meaning.

Besides, Namespace keyword are case-insensitive.

```
Namespace App
```

```
namespace App
```

and

```
NAMESPACE App
```

are same meaning.

[up](#)

[down](#)

-1

[dino at tuxweb dot it ¶](#)

1 year ago

Please note that a PHP Namespace declaration cannot start with a number.

It took some time for me to debug...

[+ add a note](#)

- [Пространства имён](#)
 - [Обзор](#)
 - [Пространства имён](#)
 - [Подпространства имён](#)
 - [Несколько пространств имён в одном файле](#)
 - [Основы](#)
 - [Пространства имён и динамические особенности языка](#)
 - [Ключевое слово namespace и константа `__NAMESPACE__`](#)
 - [Псевдонимирование и импорт](#)
 - [Глобальное пространство](#)
 - [Возврат к глобальному пространству](#)
 - [Правила разрешения имён](#)
 - [FAQ](#)
- [Copyright © 2001-2024 The PHP Group](#)
- [My PHP.net](#)
- [Contact](#)
- [Other PHP.net sites](#)
- [Privacy policy](#)

