



- [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

[Ошибки в PHP 7 »](#)

[« Ошибки](#)

- [Руководство по PHP](#)
- [Справочник языка](#)
- [Ошибки](#)

Change language: Russian

ОСНОВЫ

PHP сообщает об ошибках в ответ на некоторые внутренние ошибочные обстоятельства. Они могут быть использованы для уведомления о разных состояниях, а также могут выводиться на экран и записываться в логи по желанию.

Каждая ошибка, генерируемая PHP, обязательно содержит информацию о своём типе. [В этом списке](#) перечислены все типы ошибок, а также описания их поведений и провоцирующие их причины.

Обработка ошибок

PHP, по умолчанию, может обрабатывать любые ошибки в соответствии со своей конфигурацией, если обработчик ошибок не установлен. О каких ошибках сообщать и какие ошибки игнорировать указывается в параметре [error_reporting](#) конфигурации `php.ini`, или во время исполнения программы с помощью вызова [error_reporting\(\)](#). Настоятельно рекомендуется заранее конфигурировать `php.ini`, так как некоторые ошибки могут произойти до начала выполнения вашего скрипта.

На стадии разработки, обязательно настройте параметр [error_reporting](#) на значение `E_ALL`, так как вам необходимо знать обо всех ошибках для их решения на этой стадии. Когда продукт готов, вы можете изменить значение на менее подробное, вроде `E_ALL & ~E_NOTICE & ~E_DEPRECATED`, но во многих случаях значение `E_ALL` также желательно, так как оно может предупреждать заранее о других возможных ошибках.

Дальнейшая работа PHP с появившимися ошибками зависит от двух других параметров в `php.ini`. Параметр [display_errors](#) определяет, включать ли данные ошибки в вывод скрипта или нет. Когда продукт готов, обязательно выключите этот параметр, так как в описании ошибки может содержаться конфиденциальная информация, вроде паролей базы данных. Во время разработки данный параметр лучше включить для решения проблем, провоцирующих ошибки.

Кроме отображения ошибок, их можно записывать в лог с помощью включённого параметра [log_errors](#). Файл лога указывается в параметре [error_log](#). Данная функция незаменима на стадии конечного продукта, так как позволяет видеть информацию об ошибках и составлять основанные на них отчёты вдали от глаз пользователя.

Пользовательская обработка ошибок

Если вас не устраивает работа предустановленного обработчика ошибок, вы также можете обрабатывать множество видов ошибки с помощью своего обработчика, который можно установить вызовом функции [set_error_handler\(\)](#). Некоторые ошибки не могут обрабатываться пользовательским обработчиком, но те, которые могут, обрабатываются по вашему желанию: например, ваш скрипт может выдавать пользователю любую отдельную страницу ошибки, а подробную информацию записывать в лог или отсылать по почте.

[+add a note](#)

User Contributed Notes

There are no user contributed notes for this page.

- [Ошибки](#)
 - [ОСНОВЫ](#)
 - [Ошибки в PHP 7](#)
- [Copyright © 2001-2024 The PHP Group](#)
- [My PHP.net](#)
- [Contact](#)
- [Other PHP.net sites](#)
- [Privacy policy](#)

