www.7Learn.com

PHP Predefined Debugging Constants

The constants below are always available as part of the PHP core.

Constant	Description
E_ERROR	Fatal run-time errors. These indicate errors that can not be recovered from, such as a memory allocation problem. Execution of the script is halted.
E_WARNING	Run-time warnings (non-fatal errors). Execution of the script is not halted.
E_PARSE	Compile-time parse errors. Parse errors should only be generated by the parser.
E_NOTICE	Run-time notices. Indicate that the script encountered something that could indicate an error, but could also happen in the normal course of running a script.
E_CORE_ERROR	Fatal errors that occur during PHP's initial startup. This is like an **E_ERROR**, except it is generated by the core of PHP.
E_CORE_WARNING	Warnings (non-fatal errors) that occur during PHP's initial startup. This is like an **E_WARNING**, except it is generated by the core of PHP.
E_COMPILE_ERROR	Fatal compile-time errors. This is like an **E_ERROR**, except it is generated by the Zend Engine.
E_COMPILE_WARNING	Compile-time warnings (non-fatal errors). This is like an**E_WARNING**, except it is generated by the Zend Scripting Engine.
E_USER_ERROR	User-generated error message. This is like an **E_ERROR**, except it is generated in PHP code by using the PHP functiontrigger_error().
E_USER_WARNING	User-generated warning message. This is like an **E_WARNING**, except it is generated in PHP code by using the PHP functiontrigger_error().
E_USER_NOTICE	User-generated notice message. This is like an **E_NOTICE**, except it is generated in PHP code by using the PHP functiontrigger_error().
E_STRICT	Enable to have PHP suggest changes to your code which will ensure the best interoperability and forward compatibility of your code.
E_RECOVERABLE_ERROR	Catchable fatal error. It indicates that a probably dangerous error occurred, but did not leave the Engine in an unstable state. If the error is not caught by a user defined handle
E_DEPRECATED	Run-time notices. Enable this to receive warnings about code that will not work in future versions.
E_USER_DEPRECATED	User-generated warning message. This is like an **E_DEPRECATED**, except it is generated in PHP code by using the PHP function trigger_error().
E_ALL	All errors and warnings, as supported, except of level**E_STRICT** prior to PHP 5.4.0.

The above values (either numerical or symbolic) are used to build up a bitmask that specifies which errors to report. You can use the bitwise operators to combine these values or mask out certain types of errors. Note that only '\|', '~', '!', '^' and '&' will be understood within php.ini.