PHPUnit %s by Sebastian Bergmann and contributors.

Usage:

phpunit [options] UnitTest [UnitTest.php]

phpunit [options] <directory>

Code Coverage Options:

--coverage-clover <file> Generate code coverage report in Clover XML format

--coverage-crap4j <file> Generate code coverage report in Crap4J XML format

--coverage-html <dir> Generate code coverage report in HTML format

--coverage-php <file> Export PHP\_CodeCoverage object to file

--coverage-text=<file> Generate code coverage report in text format [default: standard output]

--coverage-xml <dir> Generate code coverage report in PHPUnit XML format

--whitelist <dir> Whitelist <dir> for code coverage analysis

--disable-coverage-ignore Disable annotations for ignoring code coverage

--no-coverage Ignore code coverage configuration

--dump-xdebug-filter <file> Generate script to set Xdebug code coverage filter

Logging Options:

--log-junit <file> Log test execution in JUnit XML format to file

--log-teamcity <file> Log test execution in TeamCity format to file

--testdox-html <file> Write agile documentation in HTML format to file

--testdox-text <file> Write agile documentation in Text format to file

--testdox-xml <file> Write agile documentation in XML format to file

--reverse-list Print defects in reverse order

Test Selection Options:

--filter <pattern> Filter which tests to run

--testsuite <name> Filter which testsuite to run

--group <name> Only runs tests from the specified group(s)

--exclude-group <name> Exclude tests from the specified group(s)

--list-groups List available test groups

--list-suites List available test suites

--list-tests List available tests

--list-tests-xml <file> List available tests in XML format

--test-suffix <suffixes> Only search for test in files with specified suffix(es). Default: Test.php,.phpt

Test Execution Options:

--dont-report-useless-tests Do not report tests that do not test anything

--strict-coverage Be strict about @covers annotation usage

--strict-global-state Be strict about changes to global state

--disallow-test-output Be strict about output during tests

--disallow-resource-usage Be strict about resource usage during small tests

--enforce-time-limit Enforce time limit based on test size

--default-time-limit=<sec> Timeout in seconds for tests without @small, @medium or @large

--disallow-todo-tests Disallow @todo-annotated tests

--process-isolation Run each test in a separate PHP process

--globals-backup Backup and restore $GLOBALS for each test

--static-backup Backup and restore static attributes for each test

--colors=<flag> Use colors in output ("never", "auto" or "always")

--columns <n> Number of columns to use for progress output

--columns max Use maximum number of columns for progress output

--stderr Write to STDERR instead of STDOUT

--stop-on-defect Stop execution upon first not-passed test

--stop-on-error Stop execution upon first error

--stop-on-failure Stop execution upon first error or failure

--stop-on-warning Stop execution upon first warning

--stop-on-risky Stop execution upon first risky test

--stop-on-skipped Stop execution upon first skipped test

--stop-on-incomplete Stop execution upon first incomplete test

--fail-on-warning Treat tests with warnings as failures

--fail-on-risky Treat risky tests as failures

-v|--verbose Output more verbose information

--debug Display debugging information

--loader <loader> TestSuiteLoader implementation to use

--repeat <times> Runs the test(s) repeatedly

--teamcity Report test execution progress in TeamCity format

--testdox Report test execution progress in TestDox format

--testdox-group Only include tests from the specified group(s)

--testdox-exclude-group Exclude tests from the specified group(s)

--no-interaction Disable TestDox progress animation

--printer <printer> TestListener implementation to use

--order-by=<order> Run tests in order: default|defects|duration|no-depends|random|reverse|size

--random-order-seed=<N> Use a specific random seed <N> for random order

--cache-result Write test results to cache file

--do-not-cache-result Do not write test results to cache file

Configuration Options:

--prepend <file> A PHP script that is included as early as possible

--bootstrap <file> A PHP script that is included before the tests run

-c|--configuration <file> Read configuration from XML file

--no-configuration Ignore default configuration file (phpunit.xml)

--no-logging Ignore logging configuration

--no-extensions Do not load PHPUnit extensions

--include-path <path(s)> Prepend PHP's include\_path with given path(s)

-d <key[=value]> Sets a php.ini value

--generate-configuration Generate configuration file with suggested settings

--cache-result-file=<file> Specify result cache path and filename

Miscellaneous Options:

-h|--help Prints this usage information

--version Prints the version and exits

--atleast-version <min> Checks that version is greater than min and exits

--check-version Check whether PHPUnit is the latest version