

PHPUnit Manual

- PHPUnit 3.7
English
- PHPUnit 3.6
English
- PHPUnit 3.7
Japanese
- PHPUnit 3.6
Japanese
- PHPUnit 3.7
French
- PHPUnit 3.6
French

1. Automating Tests

2. PHPUnit's Goals

3. Installing PHPUnit

4. Writing Tests for PHPUnit

Test Dependencies

Data Providers

Testing Exceptions

Testing PHP Errors

Testing Output

Assertions

assertArrayHas
assertClassHas
assertClassHas
assertContains
assertContains
assertCount()
assertEmpty()
assertEqualXML
assertEquals(
assertFalse()
assertFileEquals
assertFileExists
assertGreaterThan
assertGreaterThan
assertInstanceOf
assertInternalType
assertLessThan
assertLessThan
assertNull()
assertObjectHas
assertRegExp(
assertStringMatches
assertStringMatches
assertSame()
assertSelectC
assertSelectE
assertSelectR
assertStringE
assertStringE

Prev

Next

Chapter 3. Installing PHPUnit

PHPUnit should be installed using the PEAR Installer, the backbone of the [PHP Extension and Application Repository](#) that provides a distribution system for PHP packages.

Caution

Depending on your OS distribution and/or your PHP environment, you may need to install PEAR or update your existing PEAR installation before you can proceed with the instructions in this chapter.

sudo pear upgrade PEAR usually suffices to upgrade an existing PEAR installation. The [PEAR Manual](#) explains how to perform a fresh installation of PEAR.

Note

PHPUnit 3.6 requires PHP 5.2.7 (or later) but PHP 5.3.9 (or later) is highly recommended.

PHP_CodeCoverage, the library that is used by PHPUnit to collect and process code coverage information, depends on Xdebug 2.0.5 (or later) but Xdebug 2.1.3 (or later) is highly recommended.

The following two commands (which you may have to run as `root`) are all that is required to install PHPUnit using the PEAR Installer:

```
pear config-set auto_discover 1
pear install pear.phpunit.de/PHPUnit
```

The following optional packages are available:

DbUnit

DbUnit port for PHP/PHPUnit to support database interaction testing.

This package can be installed using the following command:

```

assertStringS
assertTag()
assertThat()
assertTrue()
assertXmlFile
assertXmlStri
assertXmlStri

```

Error output

Edge cases

5. The Command-Line Test Runner

Command-Line switches

6. Fixtures

More setUp() than tearDown()

Variations

Sharing Fixture

Global State

7. Organizing Tests

Composing a Test Suite Using the Filesystem

Composing a Test Suite Using XML Configuration

8. Database Testing

Supported Vendors for Database Testing

Difficulties in Database Testing

The four stages of a database test

1. Clean-Up Database

2. Set up fixture

3–5. Run Test, Verify outcome and Teardown

Configuration of a PHPUnit Database TestCase

Implementing getConnection

Implementing getDataSet()

What about the

pear install phpunit/DbUnit

PHPUnit_Selenium

Selenium RC integration for PHPUnit.

This package can be installed using the following command:

```
pear install phpunit/PHPUnit_Selenium
```

PHPUnit_Story

Story-based test runner for Behavior-Driven Development with PHPUnit.

This package can be installed using the following command:

```
pear install phpunit/PHPUnit_Story
```

PHPUnit_TestListener_DBUS

A TestListener that sends events to DBUS.

This package can be installed using the following command:

```
pear install phpunit/PHPUnit_TestListener_DBUS
```

PHPUnit_TestListener_XHProf

A TestListener that uses XHProf for automated profiling of the tested code.

This package can be installed using the following command:

```
pear install phpunit/PHPUnit_TestListener_XHProf
```

PHPUnit_TicketListener_Fogbugz

A ticket listener that interacts with the Fogbugz issue API.

This package can be installed using the following command:

```
pear install phpunit/PHPUnit_TicketListener_Fogbugz
```

PHPUnit_TicketListener_GitHub

A ticket listener that interacts with the GitHub issue API.

This package can be installed using the following command:

```
pear install phpunit/PHPUnit_TicketListener_GitHub
```

PHPUnit_TicketListener_GoogleCode

A ticket listener that interacts with the Google Code issue API.

This package can be installed using the following command:

```
pear install phpunit/PHPUnit_TicketListener_GoogleCode
```

Database Schema
(DDL)?

Tip: Use your own
Abstract Database
TestCase

Understanding DataSets
and DataTables

Available
Implementations

Beware of Foreign
Keys

Implementing your
own
DataSets/DataTable

The Connection API
Database Assertions
API

Asserting the
Row-Count of a
Table

Asserting the State
of a Table

Asserting the
Result of a Query

Asserting the State
of Multiple Tables

Frequently Asked
Questions

Will PHPUnit (re-)create the database
schema for each
test?

Am I required to
use PDO in my
application for the
Database
Extension to
work?

What can I do,
when I get a “Too
much
Connections”
Error?

How to handle
NULL with Flat
XML / CSV
Datasets?

9. Incomplete and Skipped
Tests

Incomplete Tests
Skipping Tests

PHPUnit_TicketListener_Trac

A ticket listener that interacts with the Trac issue API.

This package can be installed using the following command:

pear install phpunit/PHPUnit_TicketListener_Trac

PHP_Invoker

A utility class for invoking callables with a timeout. This package is required to enforce test timeouts in strict mode.

This package can be installed using the following command:

pear install phpunit/PHP_Invoker

After the installation you can find the PHPUnit source files inside your local PEAR directory; the path is usually */usr/lib/php/PHPUnit*.

Prev

Next

10. Test Doubles

Stubs	
Mock Objects	
Stubbing and Mocking	
Web Services	
Mocking the Filesystem	

11. Testing Practices

During Development	
During Debugging	

12. Test-Driven Development

BankAccount Example	
---------------------	--

13. Behaviour-Driven Development

BowlingGame Example	
---------------------	--

14. Code Coverage Analysis

Specifying Covered Methods	
Ignoring Code Blocks	
Including and Excluding Files	
Edge cases	

15. Other Uses for Tests

Agile Documentation	
Cross-Team Tests	

16. Skeleton Generator

Generating a Test Case Class Skeleton	
Generating a Class Skeleton from a Test Case Class	

17. PHPUnit and Selenium

Selenium Server	
Installation	
PHPUnit_Extensions_Selenium2Tests	
PHPUnit_Extensions_Selenium2Tests	

18. Logging

Test Results (XML)	
Test Results (TAP)	
Test Results (JSON)	
Code Coverage (XML)	

Code Coverage (TEXT)

19. Extending PHPUnit

Subclass
PHPUnit_Framework_T
Write custom assertions
Implement
PHPUnit_Framework_T
Subclass
PHPUnit_Extensions_Te
Implement
PHPUnit_Framework_T

A. Assertions

B. Annotations

@assert
@author
@backupGlobals
@backupStaticAttr
@codeCoverageIgnor
@covers
@dataProvider
@depends
@expectedExceptio
@expectedExceptio
@expectedExceptio
@group
@outputBuffering
@runTestsInSepara
@runInSeparatePro
@test
@testdox
@ticket

C. The XML Configuration

File

PHPUnit
Test Suites
Groups
Including and Excluding
Files for Code Coverage
Logging
Test Listeners
Setting PHP INI
settings, Constants and
Global Variables
Configuring Browsers
for Selenium RC

D. Index

E. Bibliography

