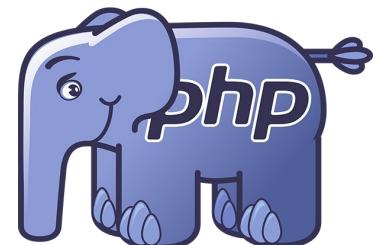


<? echo
“The PHP Contributor
Roadmap”;
?>

by Juan Carlos Morales



Who am I?



Juan Carlos Morales
Software Engineer



juan-morales



jcm-jcm

Who am I?



Juan Morales

juan-morales

I just love software development, that's it

[Edit profile](#)

29 followers · 47 following

📍 Argentina

✉️ jcmargentina@gmail.com

🔗 <https://www.linkedin.com/in/jcmcvm-en>

𝕏 @jcm_dev_

Pinned

[Customize your pins](#)

[php-src](#) Public

Forked from [php/php-src](#)

The PHP Interpreter

● C

[php/php-src](#) Public

The PHP Interpreter

● C ⭐ 38.4k 📈 7.8k

[PHP-Internals-Book](#) Public

Forked from [phpinternalsbook/PHP-Internals-Book](#)

PHP Internals Book

● Python

[kinimodemeyer/tradebyte-sdk-php](#) Public

SDK to access API endpoints of Tradebyte

● PHP ⭐ 14 📈 6

[laravel-workshop-blog](#) Public

Workshop for beginners on Laravel Framework. Presentation, Video, and Code is included

● PHP ⭐ 4

[juan-morales.github.io](#) Public

● HTML

2,214 contributions in the last year

[Contribution settings](#) ▾

2024



2023

2022

2021

History

It was year 2022, at the IPC (International PHP Conference) in Berlin, Germany.



The image shows a YouTube video player interface. The main video frame displays a presentation in a conference room. A man is standing on a stage, and the screen behind him shows the text "How to design for the metaverse.". The audience is seated in rows of chairs, facing the stage. The YouTube interface includes a search bar at the top, a sidebar with a list of conference sessions, and a video control bar at the bottom.

International PHP Conference Spring 2022

International PHP Conference - 1/7

- 1 How to design for the metaverse? | Guillaume Vaslin 30:09 International PHP Conference
- 2 Building Recommender System in PHP8 | Mihailo Joksimovic 49:29 International PHP Conference
- 3 PHP Module als Rundum-Sorglos-Pakete entwickeln | Ralf... 32:15 International PHP Conference
- 4 PHPUnit 10: Why It's Delayed, What It Brings | Sebastian... 49:17 International PHP Conference
- 5 Asynchronous PHP | Florian Engelhardt 35:16 International PHP Conference
- 6 PHP Fibers | Tomasz Turkowski 38:55 International PHP Conference
- 7 Git Scenarios: How Do I Fix That? | Rob Richardson 54:07 International PHP Conference

0:24 / 30:08 - Intro >

How to design for the metaverse? | Guillaume Vaslin

IPC International PHP Conference 5.14 K suscriptores

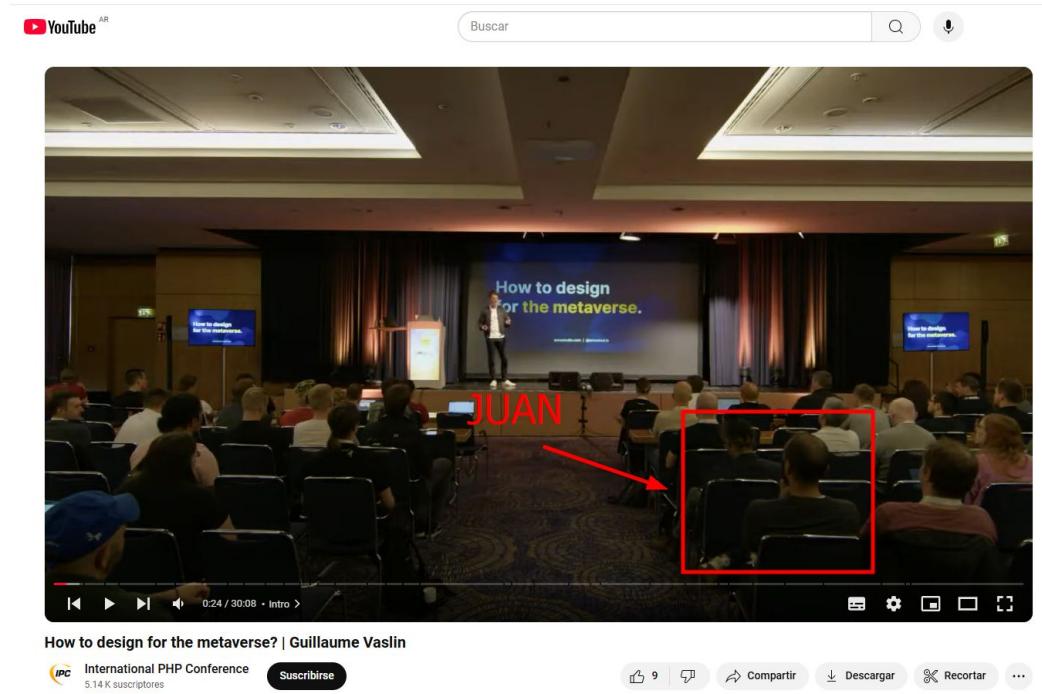
Suscribirse

Compartir Descargar Recortar

9 49:29 jazz lofi radio beats to chill/study to

History

It was year 2022, at the IPC (International PHP Conference) in Berlin, Germany.



History

It was year 2022, at the IPC (International PHP Conference) in Berlin, Germany.



Jose M. Valera Reales



Sebastian Bergmann

What is this presentation about?

- This presentation will try to give you a better understanding about different ways someone can contribute to the PHP Project, **specially by coding**
- Will provide online resources to learn about
- Instructions to setup your local environment
- Roadmap made by experience to contribute to the code of PHP, **the way I made it!**

My PHP contributions

The screenshot shows a GitHub repository page for 'php / php-src'. The 'Pull requests' tab is selected, showing 538 pull requests. The search bar contains 'ispr author:@me'. The pull requests are filtered by 'Extension: json' and 'Extension: filter'. Several pull requests are highlighted with blue boxes and arrows pointing to them:

- 1 Open, 23 Closed
- better json error message** (Extension: json) #14672 (Approved)
- QA - finfo_open - increase coverage** (Extension: fileinfo) #13839 (Approved)
- QA - Bug fix - filter_input_array - solution 2** (Extension: filter) #13804 (Approved)
- [On Hold - Ignore] QA - Bug fix - filter input array - solution 1** (Extension: filter) #13803 (Closed)
- QA - Test - filter_input_array - FILTER_NULL_ON_FAILURE** (Extension: filter) #13792 (Closed)
- QA - filter_input - LCOV - FILTER_NULL_ON_FAILURE** (Extension: filter) #9737 (Approved)
- RFC - json_validate()** (Extension: json, RFC) #9399 (Approved)
- QA - json_decode - max depth allowed** (Extension: json) #9359 (Closed)
- RFC - Work in progress - is_json()** (Extension: json, RFC, SAPF: fuzzer) #9355 (Closed)
- QA - adjust set/get priority tests to consider environment variables and root user** (Extension: pcntl) #9284 (Approved)
- Fix pcntl get/set priority** (Extension: pcntl) #9044 (Approved)

Roadmap to start contributing to the code of PHP

1 Join the community

2 Read contributor information

3 Setup local environment

4 Learn PHP Tests (PHPT)

5 Fix a bug

6 Deep Dive into PHP Internals

Always

Read the existing documentation and learn from it, while walking the roadmap

6.1 Make an RFC

6.2 Write an Extension

Roadmap to start contributing to the code of PHP

1

Join the community

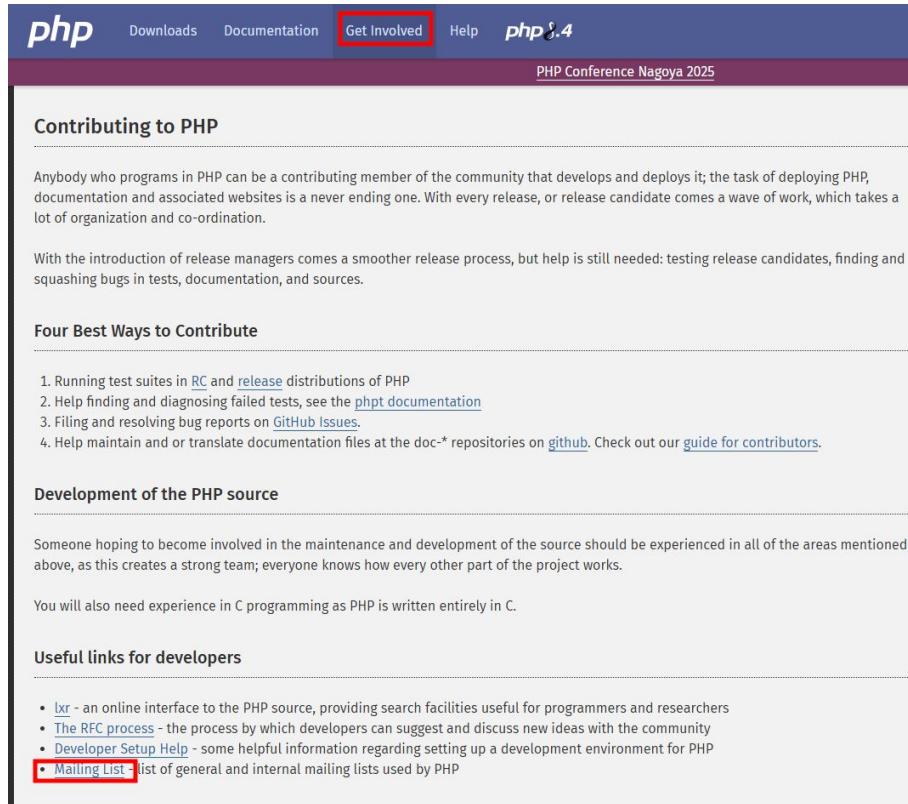
Always

Read the existing documentation and learn from it, while walking the roadmap

Roadmap: 1 Join the Community

- Joining the community is the first step.
- Will help you to put yourself into the radar of “what is going on at the moment?”.
- You will put yourself up-to-date with the actual discussions related to features, changes, RFCs, etc.
- Also, if in the future you work in an RFC or new extension, you will need to be in touch with the community, either because of help, or in the case of RFCs you need to discuss it with the Community as part of the procedure.

Roadmap: 1 Join the Community - Internals Mailing List



The screenshot shows the 'Contributing to PHP' page on the official PHP website. The 'Get Involved' button in the top navigation bar is highlighted with a red box. The 'Mailing List' link in the 'Useful links for developers' section is also highlighted with a red box.

Contributing to PHP

Anybody who programs in PHP can be a contributing member of the community that develops and deploys it; the task of deploying PHP, documentation and associated websites is a never ending one. With every release, or release candidate comes a wave of work, which takes a lot of organization and co-ordination.

With the introduction of release managers comes a smoother release process, but help is still needed: testing release candidates, finding and squashing bugs in tests, documentation, and sources.

Four Best Ways to Contribute

1. Running test suites in [RC](#) and [release](#) distributions of PHP
2. Help finding and diagnosing failed tests, see the [phpt documentation](#)
3. Filing and resolving bug reports on [GitHub Issues](#).
4. Help maintain and or translate documentation files at the doc-* repositories on [github](#). Check out our [guide for contributors](#).

Development of the PHP source

Someone hoping to become involved in the maintenance and development of the source should be experienced in all of the areas mentioned above, as this creates a strong team; everyone knows how every other part of the project works.

You will also need experience in C programming as PHP is written entirely in C.

Useful links for developers

- [lxr](#) - an online interface to the PHP source, providing search facilities useful for programmers and researchers
- [The RFC process](#) - the process by which developers can suggest and discuss new ideas with the community
- [Developer Setup Help](#) - some helpful information regarding setting up a development environment for PHP
- [Mailing List](#) - list of general and internal mailing lists used by PHP

Roadmap: 1 Join the Community - Internals Mailing List

Internals Mailing Lists					
	Moderated	Archive	Newsgroup	Normal	Digest
Internals list <internals@lists.php.net> A medium volume list for those who want to help out with the development of PHP	no	<u>yes</u>	<u>yes http</u>	<input checked="" type="radio"/>	<input type="radio"/>
Windows Internals list <internals-win@lists.php.net> A low volume list for those who want to help out with the development of PHP on Windows	no	n/a	<u>yes http</u>	<input type="radio"/>	<input type="radio"/>
Git commit list <php-cvs@lists.php.net> All commits to internals (php-src) and the Zend Engine are posted to this list automatically	yes	<u>yes</u>	<u>yes http</u>	<input type="radio"/>	n/a
Git pull requests <git-pulls@lists.php.net> Pull requests from Github	no	n/a	<u>yes http</u>	<input type="radio"/>	n/a
Quality Assurance list <php-qa@lists.php.net> List for the members of the PHP-QA Team	no	<u>yes</u>	<u>yes http</u>	<input type="radio"/>	n/a
General bugs <php-bugs@lists.php.net> General bug activity are posted here	no	n/a	<u>yes http</u>	<input type="radio"/>	n/a
PHP Standardization and interoperability list <standards@lists.php.net> Development of language standards	no	n/a	<u>yes http</u>	<input type="radio"/>	n/a
PHP internal website mailing lists					
	Moderated	Archive	Newsgroup	Normal	Digest
PHP php.net internal infrastructure discussion <php-webmaster@lists.php.net> List for discussing and maintaining the php.net web infrastructure. For general PHP support questions, see "General Mailing Lists" or the support page	no	n/a	<u>yes http</u>	<input type="radio"/>	n/a
PHP documentation mailing lists					
	Moderated	Archive	Newsgroup	Normal	Digest
Documentation discussion <phpdoc@lists.php.net>	no	<u>yes</u>	<u>yes http</u>	<input type="radio"/>	n/a

Roadmap: 1 Join the Community - Internals Mailing List

Email:

You will be sent a confirmation mail at the address you wish to be subscribed or unsubscribed, and only added to the list after following the directions in that mail.

Warning: The PHP users mailing list gets close to 1500-2000 messages per month currently. Do the math. That translates to about 60 messages per day. If your mailbox can't handle this traffic you might want to consider subscribing to the digest list instead (two messages per day), using the news server, or reading the mailing list using the archives.

Subscribe via Email

If you experience trouble subscribing via the form above, you may also subscribe by sending an email to the list server. To subscribe to any mailing list, send an email to *listname+subscribe@lists.php.net* (substituting the name of the list for *listname* – for example, *php-general+subscribe@lists.php.net*).

Mailing list options

All of the mailing lists hosted at [lists.php.net](#) are managed using the [mlmmj](#) mailing list software. There are a variety of commands you can use to modify your subscription. For a full overview, send a message to php-whatever+help@lists.php.net (as in, *php-general+help@lists.php.net*).

Subscribing

- The normal mailing list, where you receive every message separately:
Email: php-listname+subscribe@lists.php.net
- The daily digest list, where you receive an email with every message for a whole day at once:
Email: php-listname+subscribe-digest@lists.php.net
- The no email list, where you receive no emails from the list, but you can still post to it:
Email: php-listname+subscribe-nomail@lists.php.net

Unsubscribing

Email: php-listname+unsubscribe@lists.php.net

Roadmap: 1 Join the Community - Internals Mailing List

- ★ Daniel, Tim 3  [PHP-DEV] [RFC] [Discussion] Attributes on compile-time constants - Hi again, Based on feedback to my last thread[1] that an RFC would be needed to allo...

- ★ mickmackusa .. Ilija 12 [PHP-DEV] Inaccurate documentation on return values from native functions - I want to point out some misleading documentation so that those who are b...

- ★ Daniel .. Christoph 5 [PHP-DEV] Adding attributes to compile-time constants - Hi internals, For the last few weeks, I've been working on a patch[1] that I just uploaded that adds s...

- ★ Máté .. Larry, Tim 41  Re: [PHP-DEV] [RFC] [Discussion] Add WHATWG compliant URL parsing API - Hi Everyone, I've been working on a new RFC for a while now, and time has co...

- ★ Sergey, Jakub 2 [PHP-DEV] PHP 8.2 Active Support Closed Reminder - Hi, all! Just a friendly reminder, according to the supported versions page [1], active support for PHP-8....

 signature.ascDownload
- ★ sam@melroseandco.uk [PHP-DEV] opcache_is_script_cached(...) for file cache - Hello, I'd like to propose a version of `opcache_is_script_cached(...)` that supports checking the fil...

- ★ Tim, Juliette 5  [PHP-DEV] [VOTE] Support Closures in constant expressions - Hi we just started the vote the the "Support Closures in constant expressions" RFC. Please fin...

- ★ Eric, Tim 2  [PHP-DEV] [RFC] [Discussion] Persistent curl share handle improvement - Hello again internals, Similar to the Random Extension improvement RFC (<https://...>)

- ★ Theodore .. Christo. 15 [PHP-DEV] Deprecate alternate switch case syntax? - Hello Internals, I recently found out (after 15 years of being a PHP developer) that PHP allows using a s...

- ★ aggelos .. Christoph 4 [PHP-DEV] Needs Feedback - Yield without value in reference generator function does not create notice - Hello everyone, In this issue #16761 it was verifi...

- ★ Larry, Tim 3 [PHP-DEV] [RFC] Static property asymmetric visibility - Static property asymmetric visibility was left out of the original RFC, because it seemed like it would ...

- ★ Volodymyr .. Juris 6 [PHP-DEV] Return When - Hi PHP Mailing list, My username is vvolynets. I have an idea which will save a lot of code. I am proposing to add": return when [condi...

- ★ Eugene, Tim, Marco 5 [PHP-DEV] iterator_count signature for iterables - Hello internals! Currently, iterator_count has the following signature: iterator_count(Traversable|array \$itera...

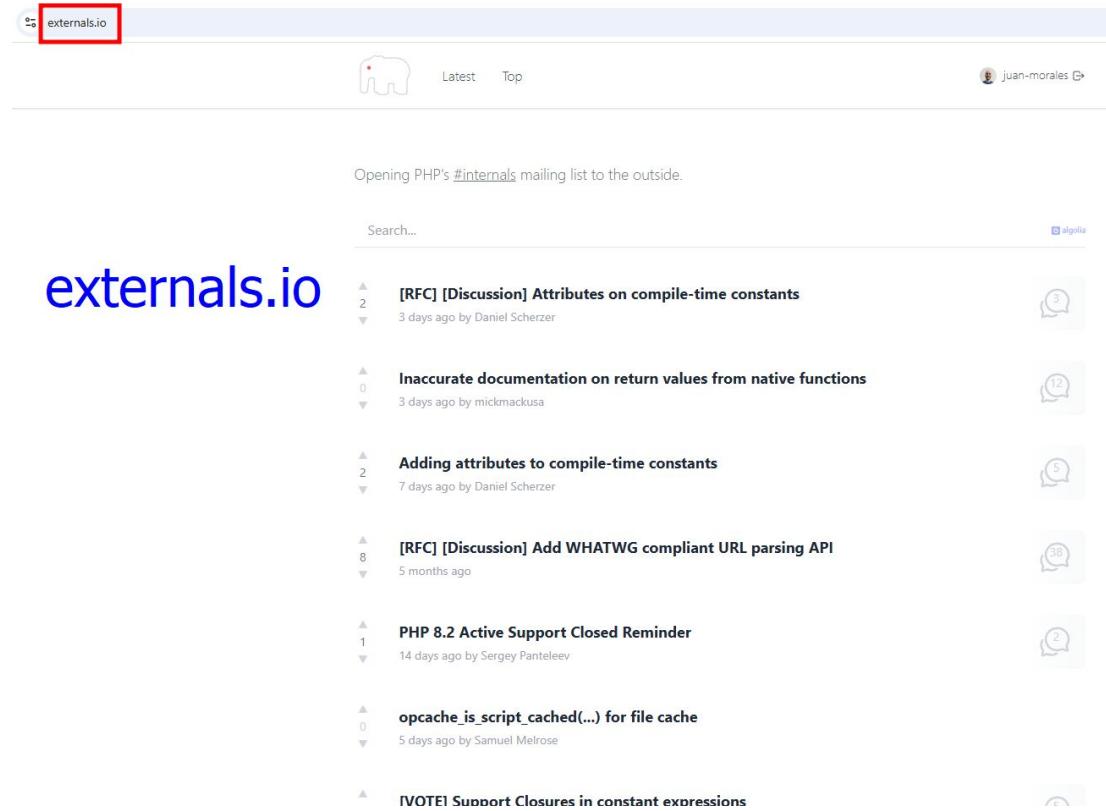
- ★ Rob .. Niels, Larry 17 [PHP-DEV] [RFC] Data Classes - Hello internals, Born from the Records RFC (<https://wiki.php.net/rfc/records>) discussion, I would like to introduce to you a com...

- ★ Larry .. Deri., John 9 [PHP-DEV] [RFC][Vote] Policy on third party code - I have opened the vote on the proposed third-patry-code policy clarification. <https://wiki.php.net/rfc/third...>

- ★ Rob .. Michał, Larry 9 [PHP-DEV] RFC: Records - Hello internals, I'm ready as I'm going to be to introduce to you: "Records" <https://wiki.php.net/rfc/records>! Records allow for a lightw...

- ★ Ilija .. Daniil 8 [PHP-DEV] opcache_compile_file() declares top-level functions - Hi everyone We recently received a bug report regarding the behavior of opcache_compli...

Roadmap: 1 Join the Community - Mailing List

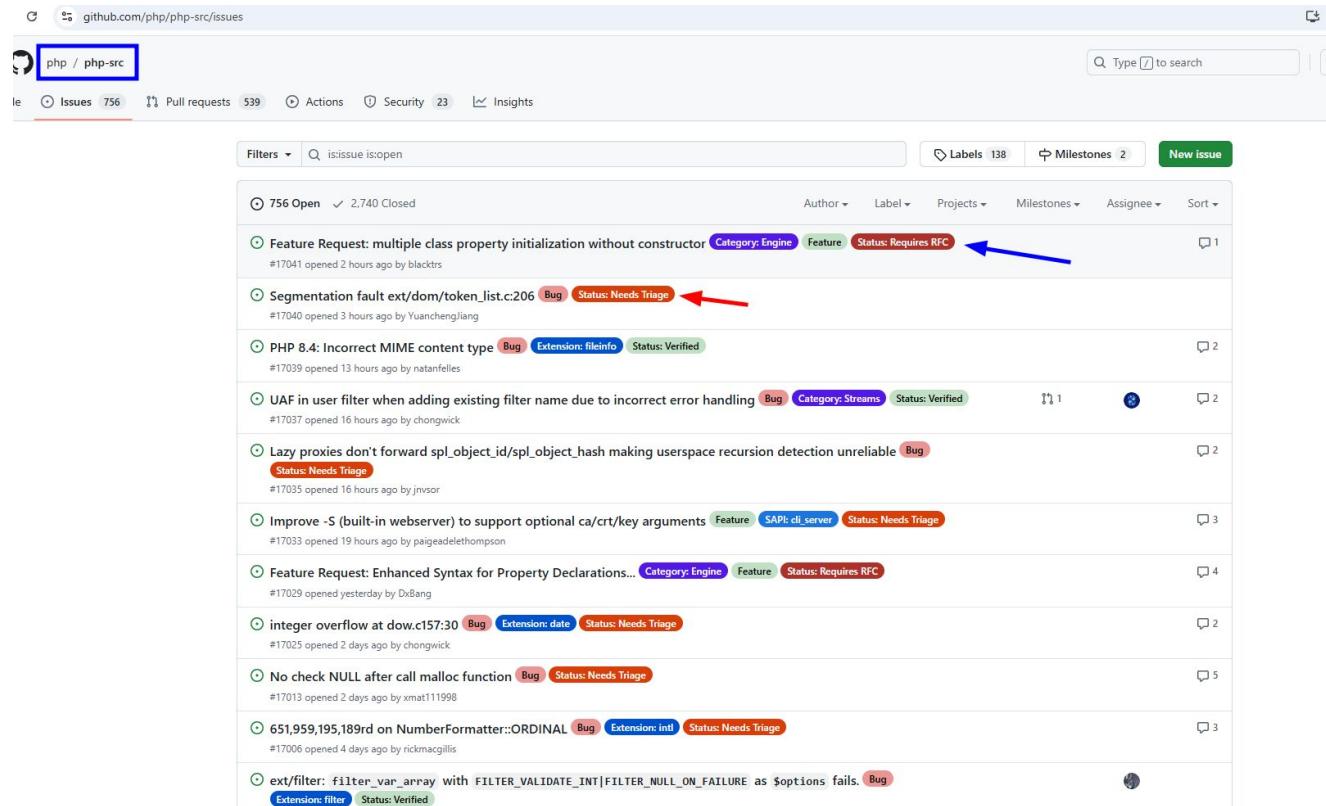


The screenshot shows the externals.io homepage with a red box highlighting the URL in the address bar. The page features a header with a logo, 'Latest', 'Top', and a user profile for 'juan-morales'. Below the header is a search bar and the 'algolia' logo. The main content area displays a list of PHP mailing list discussions:

- [RFC] [Discussion] Attributes on compile-time constants**
3 days ago by Daniel Scherzer
- Inaccurate documentation on return values from native functions**
3 days ago by mickmackusa
- Adding attributes to compile-time constants**
7 days ago by Daniel Scherzer
- [RFC] [Discussion] Add WHATWG compliant URL parsing API**
5 months ago
- PHP 8.2 Active Support Closed Reminder**
14 days ago by Sergey Panteleev
- opcache_is_script_cached(...) for file cache**
5 days ago by Samuel Melrose
- IVOTEI Support Closures in constant expressions**

Each discussion item includes a reply icon with a number indicating the count of replies.

Roadmap: 1 Join the Community - Github



github.com/php/php-src/issues

php / php-src

Issues 756 Pull requests 539 Actions Security 23 Insights

Filters isissue isopen

756 Open 2,740 Closed

Author Label Projects Milestones Assignee Sort

Feature Request: multiple class property initialization without constructor Category: Engine Feature Status: Requires RFC

Segmentation fault ext/dom/token_list.c:206 Bug Status: Needs Triage

PHP 8.4: Incorrect MIME content type Bug Extension: fileinfo Status: Verified

UAF in user filter when adding existing filter name due to incorrect error handling Bug Category: Streams Status: Verified

Lazy proxies don't forward spl_object_id/spl_object_hash making userspace recursion detection unreliable Bug Status: Needs Triage

Improve -S (built-in webserver) to support optional ca/crt/key arguments Feature SAPI: cli_server Status: Needs Triage

Feature Request: Enhanced Syntax for Property Declarations... Category: Engine Feature Status: Requires RFC

integer overflow at dow.c157:30 Bug Extension: date Status: Needs Triage

No check NULL after call malloc function Bug Status: Needs Triage

651,959,189rd on NumberFormatter::ORDINAL Bug Extension: intl Status: Needs Triage

ext/filter: filter_var_array with FILTER_VALIDATE_INT|FILTER_NULL_ON_FAILURE as \$options fails. Bug Extension: filter Status: Verified

Roadmap to start contributing to the code of PHP

1

Join the community

2

Read contributor
information

Always

Read the existing
documentation and
learn from it, while
walking the roadmap

Roadmap: 2 Read the contributor information

The screenshot shows the PHP 'Get Involved' page. The URL 'php.net/get-involved.php' is in the address bar. The page has a blue header with 'php' logo, 'Downloads', 'Documentation', 'Get Involved' (which is highlighted with a red box), 'Help', and 'php 8.4'. The sub-header says 'PHP Conference Nagoya 2025'. The main content is divided into sections: 'Contributing to PHP', 'Four Best Ways to Contribute', 'Development of the PHP source', and 'Useful links for developers'. A 'Table of Contents' sidebar on the right also lists these sections. Red boxes highlight the 'Get Involved' button in the header and the 'Table of Contents' sidebar.

Contributing to PHP

Anybody who programs in PHP can be a contributing member of the community that develops and deploys it; the task of deploying PHP, documentation and associated websites is a never ending one. With every release, or release candidate comes a wave of work, which takes a lot of organization and co-ordination.

With the introduction of release managers comes a smoother release process, but help is still needed: testing release candidates, finding and squashing bugs in tests, documentation, and sources.

Four Best Ways to Contribute

1. Running test suites in [RC](#) and [release](#) distributions of PHP
2. Help finding and diagnosing failed tests, see the [phpt documentation](#)
3. Filing and resolving bug reports on [GitHub Issues](#).
4. Help maintain and or translate documentation files at the doc-* repositories on [github](#). Check out our [guide for contributors](#).

Development of the PHP source

Someone hoping to become involved in the maintenance and development of the source should be experienced in all of the areas mentioned above, as this creates a strong team; everyone knows how every other part of the project works.

You will also need experience in C programming as PHP is written entirely in C.

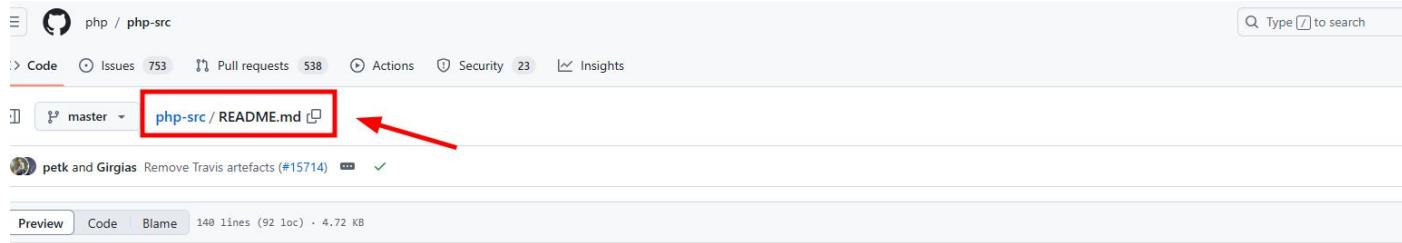
Useful links for developers

- [lxr](#) - an online interface to the PHP source, providing search facilities useful for programmers and researchers
- [The RFC process](#) - the process by which developers can suggest and discuss new ideas with the community
- [Developer Setup Help](#) - some helpful information regarding setting up a development environment for PHP
- [Mailing List](#) - list of general and internal mailing lists used by PHP

Table of Contents

- Contributing to PHP
- Four Best Ways to Contribute
- Development of the PHP source
- Useful links for developers

Roadmap: 2 Read the contributor information



The PHP Interpreter

PHP is a popular general-purpose scripting language that is especially suited to web development. Fast, flexible and pragmatic, PHP powers everything from your blog to the most popular websites in the world. PHP is distributed under the [PHP License v3.01](#).

 Push  passing  fuzzing

Documentation

The PHP manual is available at [php.net/docs](#).

Installation

Prebuilt packages and binaries

Prebuilt packages and binaries can be used to get up and running fast with PHP.

For Windows, the PHP binaries can be obtained from [windows.php.net](#). After extracting the archive the *.exe files are ready to use.

Roadmap: 2 Read the contributor information

php-src / README.md

Blame 140 lines (92 loc) · 4.72 KB

Contributing

The PHP source code is located in the Git repository at github.com/php/php-src. Contributions are most welcome by forking the repository and sending a pull request.

Discussions are done on GitHub, but depending on the topic can also be relayed to the official PHP developer mailing list internals@lists.php.net.

New features require an RFC and must be accepted by the developers. See [Request for comments - RFC](#) and [Voting on PHP features](#) for more information on the process.

Bug fixes don't require an RFC. If the bug has a GitHub issue, reference it in the commit message using `GH-NNNNNN`. Use `#NNNNNN` for tickets in the old [bugs.php.net](#) bug tracker.

Fix GH-7815: `php_uname` doesn't recognise latest Windows versions
Fix #55371: `get_magic_quotes_gpc()` throws deprecation warning

See [Git workflow](#) for details on how pull requests are merged.

Guidelines for contributors

See further documents in the repository for more information on how to contribute:

- [Contributing to PHP](#)
- [PHP coding standards](#)
- [Internal documentation](#)
- [Mailing list rules](#)
- [PHP release process](#)

Credits

For the list of people who've put work into PHP, please see the [PHP credits page](#).

Roadmap to start contributing to the code of PHP

1 Join the community

2 Read contributor
information

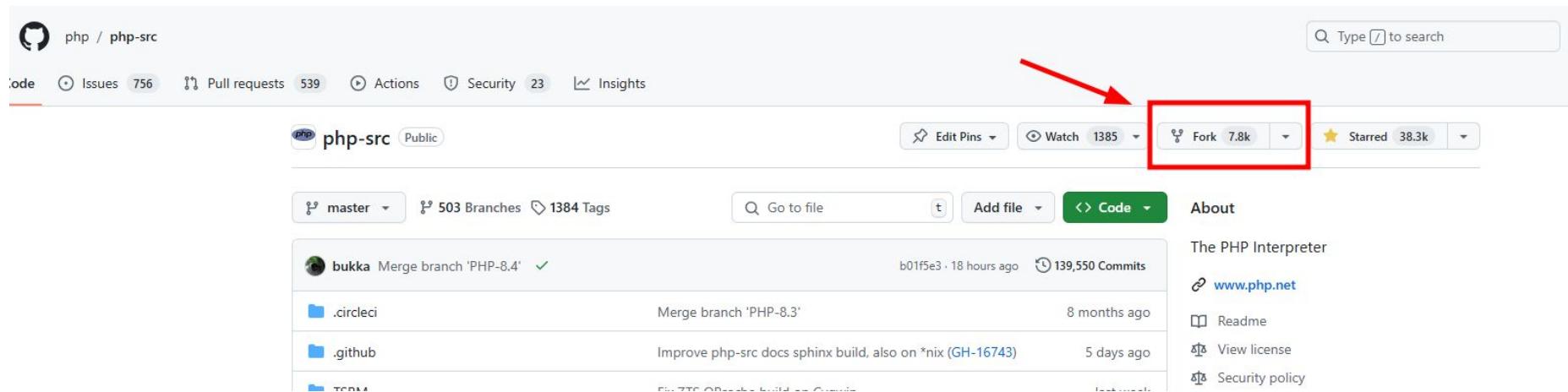
3 Setup local
environment

Always

Read the existing
documentation and
learn from it, while
walking the roadmap

Roadmap: 3 Setup your local environment

Fork the php-src repository



Code Issues 756 Pull requests 539 Actions Security 23 Insights

php / php-src Type / to search

php-src Public

Edit Pins Watch 1385 Fork 7.8k Starred 38.3k

master 503 Branches 1384 Tags Go to file Add file Code About

bukka Merge branch 'PHP-8.4' b01f5e3 · 18 hours ago 139,550 Commits

.circleci Merge branch 'PHP-8.3' 8 months ago

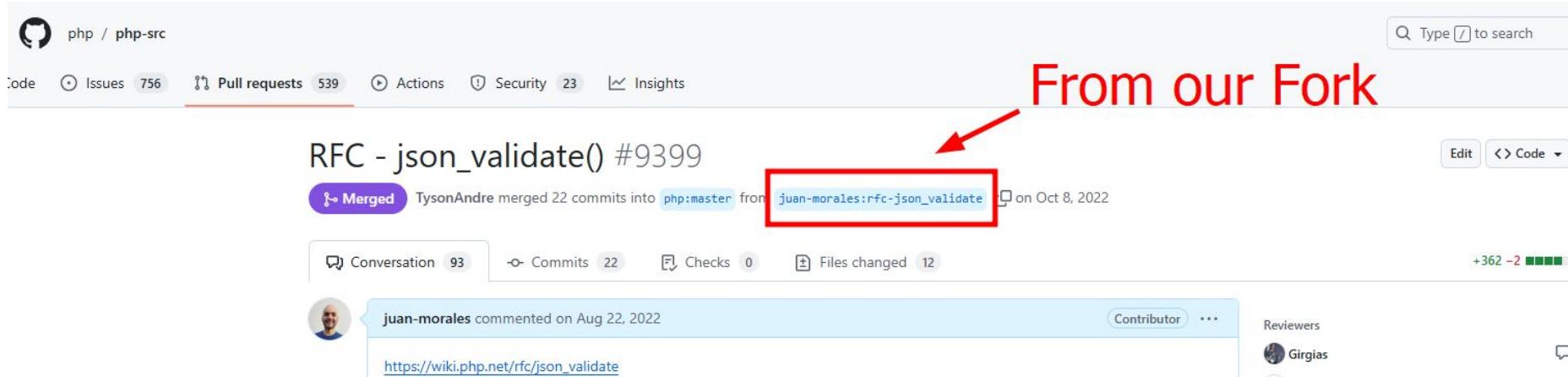
.github Improve php-src docs sphinx build, also on *nix (GH-16743) 5 days ago

TCO4

The PHP Interpreter
www.php.net
Readme
View license
Security policy

Roadmap: 3 Setup your local environment

Fork the php-src repository



From our Fork

RFC - json_validate() #9399

Merged TysonAndre merged 22 commits into `php:master` from `juan-morales:rfc-json_validate` on Oct 8, 2022

Conversation 93 Commits 22 Checks 0 Files changed 12

juan-morales commented on Aug 22, 2022

https://wiki.php.net/rfc/json_validate

Contributor

Reviewers

Girgias

Roadmap: 3 Setup your local environment

Clone your fork locally in your development machine



```
git clone git@github.com:juan-morales/php-src.git
```

Roadmap: 3 Setup your local environment

Install development requirements and dependencies

```
sudo apt install -y pkg-config \
  build-essential \
  autoconf \
  bison \
  re2c \
  libxml2-dev \
  libsqlite3-dev \
  lcov
```

Roadmap: 3 Setup your local environment

Configure and Build PHP from source code

```
$ make clean  
$ ./buildconf --force  
$ ./configure --enable-debug --enable-gcov --without-sqlite3 --without-pdo-sqlite --enable-mbstring --enable-bcmath --enable-ftp --enable-pcntl --enable-sockets --enable-zts --with-pear --with-mysqli
```

Roadmap: 3 Setup your local environment

Configure and Build PHP from source code



```
make -j`nproc`
```

Roadmap: 3 Setup your local environment

Run the PHP Tests



`make test`



`make TEST_PHP_ARGS=-j`nproc` test`

Roadmap: 3 Setup your local environment

...

All Set!

Roadmap to start contributing to the code of PHP

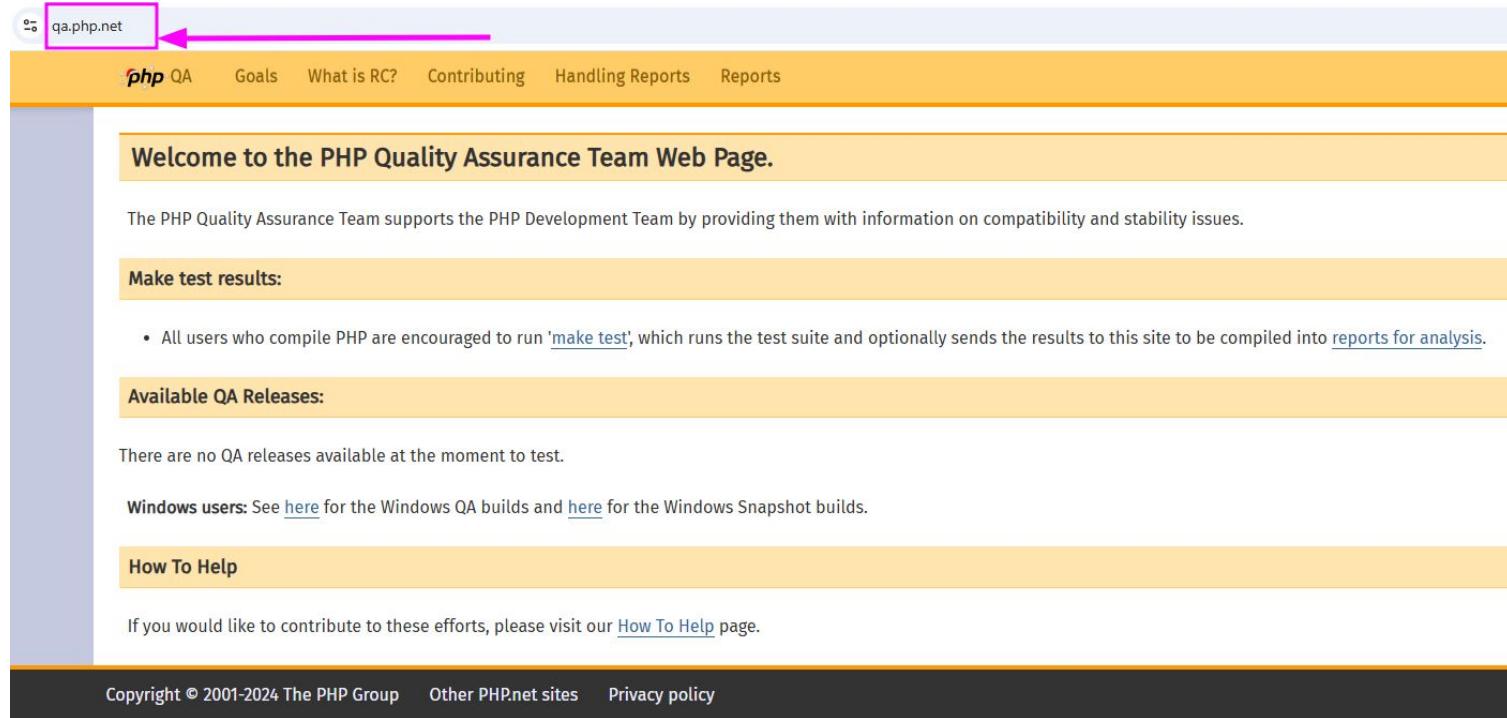
- 1 Join the community
- 2 Read contributor information
- 3 Setup local environment
- 4 Learn PHP Tests (PHPT)

Always

Read the existing documentation and learn from it, while walking the roadmap

Roadmap: 4 Learn PHP Tests - PHPT

Official documentation



The screenshot shows the homepage of the PHP Quality Assurance Team website at qa.php.net. The page has a light blue header with the URL. Below it is an orange navigation bar with links: 'php QA', 'Goals', 'What is RC?', 'Contributing', 'Handling Reports', and 'Reports'. The main content area has a light orange background. It starts with a welcome message: 'Welcome to the PHP Quality Assurance Team Web Page.' followed by a paragraph about the team's role. There are three main sections: 'Make test results:', 'Available QA Releases:', and 'How To Help'. Each section contains a list of items or links. At the bottom, there is a footer with copyright information and links to other PHP.net sites and privacy policy.

qa.php.net

php QA Goals What is RC? Contributing Handling Reports Reports

Welcome to the PHP Quality Assurance Team Web Page.

The PHP Quality Assurance Team supports the PHP Development Team by providing them with information on compatibility and stability issues.

Make test results:

- All users who compile PHP are encouraged to run '[make test](#)', which runs the test suite and optionally sends the results to this site to be compiled into [reports for analysis](#).

Available QA Releases:

There are no QA releases available at the moment to test.

Windows users: See [here](#) for the Windows QA builds and [here](#) for the Windows Snapshot builds.

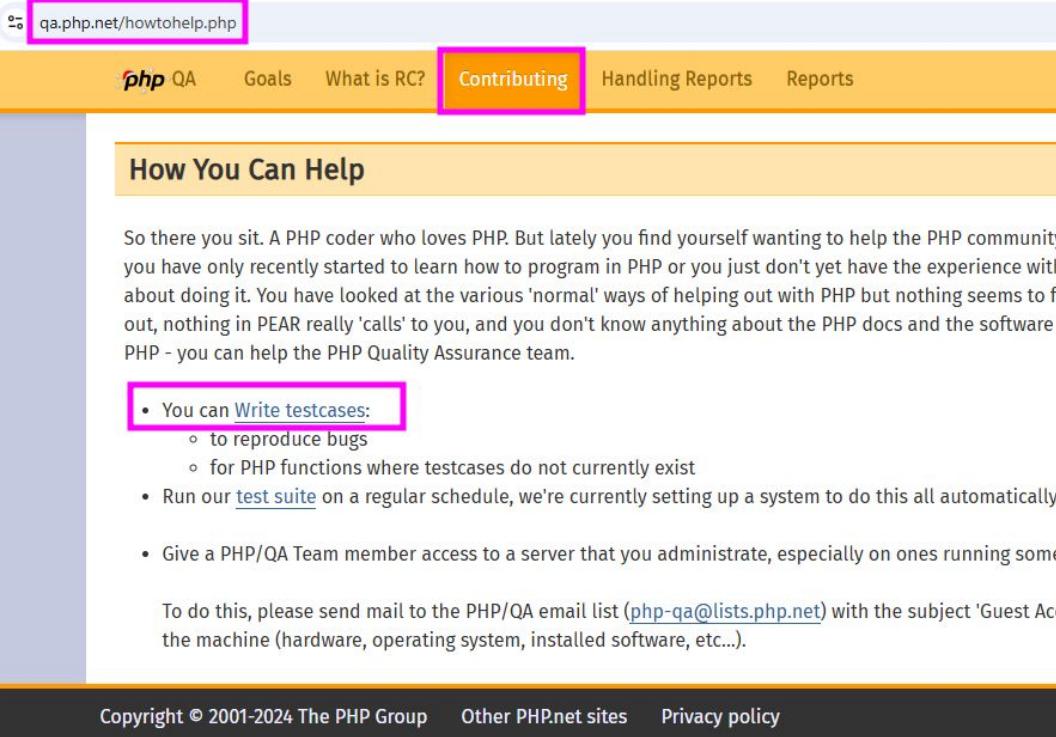
How To Help

If you would like to contribute to these efforts, please visit our [How To Help](#) page.

Copyright © 2001-2024 The PHP Group [Other PHP.net sites](#) [Privacy policy](#)

Roadmap: 4 Learn PHP Tests - PHPT

Official documentation



qa.php.net/howtohelp.php

Contributing

How You Can Help

So there you sit. A PHP coder who loves PHP. But lately you find yourself wanting to help the PHP community. You have only recently started to learn how to program in PHP or you just don't yet have the experience with it. You are about doing it. You have looked at the various 'normal' ways of helping out with PHP but nothing seems to fit. Out, nothing in PEAR really 'calls' to you, and you don't know anything about the PHP docs and the software it depends on - you can help the PHP Quality Assurance team.

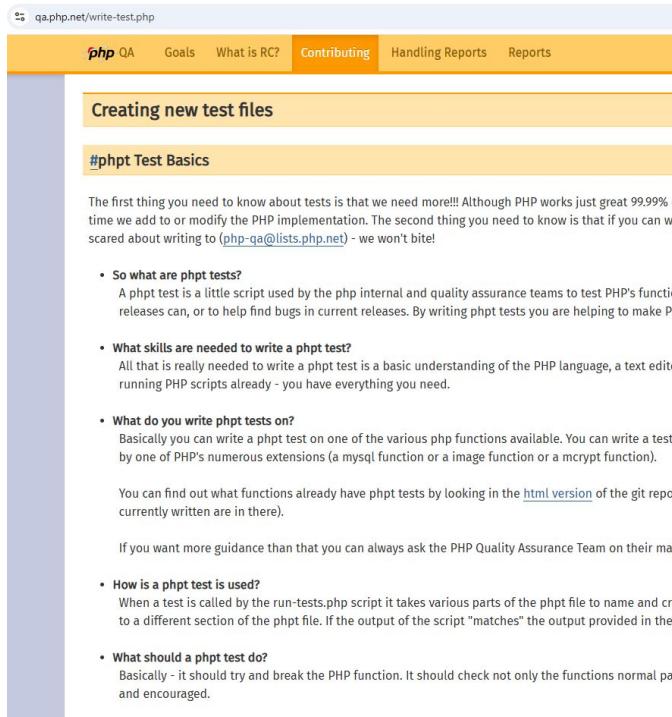
- You can Write testcases:
 - to reproduce bugs
 - for PHP functions where testcases do not currently exist
- Run our test suite on a regular schedule, we're currently setting up a system to do this all automatically.
- Give a PHP/QA Team member access to a server that you administrate, especially on ones running some of the PHP code.

To do this, please send mail to the PHP/QA email list (php-qa@lists.php.net) with the subject 'Guest Account'. Please include information about the machine (hardware, operating system, installed software, etc...).

Copyright © 2001-2024 The PHP Group [Other PHP.net sites](#) [Privacy policy](#)

Roadmap: 4 Learn PHP Tests - PHPT

Official documentation



qa.php.net/write-test.php

php QA Goals What is RC? Contributing Handling Reports Reports

Creating new test files

#phpt Test Basics

The first thing you need to know about tests is that we need more!!! Although PHP works just great 99.99% of time we add to or modify the PHP implementation. The second thing you need to know is that if you are scared about writing to (php-qa@lists.php.net) - we won't bite!

- **So what are phpt tests?**
A phpt test is a little script used by the php internal and quality assurance teams to test PHP's function releases can, or to help find bugs in current releases. By writing phpt tests you are helping to make PHP
- **What skills are needed to write a phpt test?**
All that is really needed to write a phpt test is a basic understanding of the PHP language, a text editor running PHP scripts already - you have everything you need.
- **What do you write phpt tests on?**
Basically you can write a phpt test on one of the various php functions available. You can write a test covering one of PHP's numerous extensions (a mysql function or a image function or a mcrypt function).

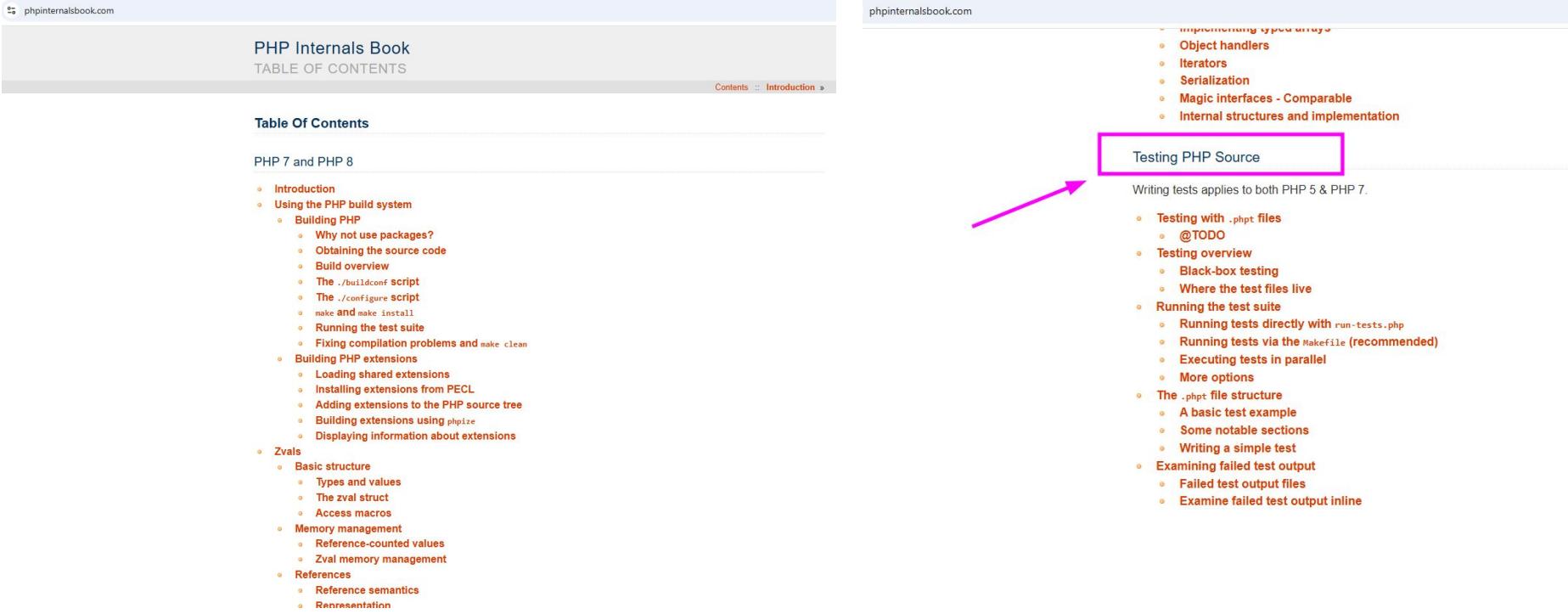
You can find out what functions already have phpt tests by looking in the [html version](#) of the git repository. Currently written are in there).

If you want more guidance than that you can always ask the PHP Quality Assurance Team on their mailing list.

- **How is a phpt test is used?**
When a test is called by the run-tests.php script it takes various parts of the phpt file to name and create a different section of the phpt file. If the output of the script "matches" the output provided in the phpt file then the test is considered to have passed.
- **What should a phpt test do?**
Basically - it should try and break the PHP function. It should check not only the functions normal parameters and encouraged.

Roadmap: 4 Learn PHP Tests - PHPT

PHP Internals Book



The image shows two screenshots of the PHP Internals Book website. The left screenshot shows the Table of Contents, and the right screenshot shows a detailed section on testing.

Table of Contents (Left Screenshot):

- PHP 7 and PHP 8
 - Introduction
 - Using the PHP build system
 - Building PHP
 - Why not use packages?
 - Obtaining the source code
 - Build overview
 - The `./buildconf` script
 - The `./configure` script
 - make AND make install
 - Running the test suite
 - Fixing compilation problems and make clean
 - Building PHP extensions
 - Loading shared extensions
 - Installing extensions from PECL
 - Adding extensions to the PHP source tree
 - Building extensions using `phpize`
 - Displaying information about extensions
 - Zvals
 - Basic structure
 - Types and values
 - The zval struct
 - Access macros
 - Memory management
 - Reference-counted values
 - Zval memory management
 - References
 - Reference semantics
 - Representation

Testing PHP Source (Right Screenshot):

Testing PHP Source

Writing tests applies to both PHP 5 & PHP 7.

- Testing with `.phpt` files
 - @TODO
- Testing overview
 - Black-box testing
 - Where the test files live
- Running the test suite
 - Running tests directly with `run-tests.php`
 - Running tests via the `Makefile` (recommended)
 - Executing tests in parallel
 - More options
- The `.phpt` file structure
 - A basic test example
 - Some notable sections
 - Writing a simple test
- Examining failed test output
 - Failed test output files
 - Examine failed test output inline

Roadmap: 4 Learn PHP Tests - PHPT

Write a QA Test to increase source code test coverage



```
make TEST_PHP_ARGS=-j`nproc` test
```

```
make lcov
```

Then check in folder **lcov_html** the file **index.html**

Roadmap: 4 Learn PHP Tests - PHPT

Write a QA Test to increase source code test coverage

```
PASS Fundamental breakpoints functionality [sapi/phpdbg/tests/breakpoints_001.php]
PASS A script with die() must end "normally" [sapi/phpdbg/tests/normal_exit.php]
PASS Test phpdbg_break_next() function [sapi/phpdbg/tests/phpdbg_break_next.php]
PASS Preserve breakpoints on restart [sapi/phpdbg/tests/breakpoints_002.php]
PASS Extension loading [sapi/cli/tests/ext_loading.php]
PASS Getting executable lines from custom wrappers [sapi/phpdbg/tests/phpdbg_get_executable_stream.php]
PASS Test deleting breakpoints [sapi/phpdbg/tests/breakpoints_003.php]
PASS Test phpdbg_*oplog() functions [sapi/phpdbg/tests/phpdbg_oplog_001.php]
PASS Test opcode breakpoints [sapi/phpdbg/tests/breakpoints_004.php]
PASS Ensure a single Date header is present [sapi/cli/tests/gh12363.php]
PASS phpdbg_end_oplog() alone must not crash [sapi/phpdbg/tests/phpdbg_oplog_002.php]
PASS Test breakpoint into function context [sapi/phpdbg/tests/breakpoints_005.php]
PASS Basic function breakpoints [sapi/phpdbg/tests/breakpoints_006.php]
PASS GH-13113: Missing syntax error in CLI-server router script [sapi/cli/tests/gh13113.php]
SKIP GH-14189 (PHP Interactive shell input state incorrectly handles quoted heredoc literals.) [sapi]
PASS Basic method breakpoints [sapi/phpdbg/tests/breakpoints_007.php]
PASS Test namespaced and non-lowercase breakpoint names [sapi/phpdbg/tests/breakpoints_008.php]
PASS Bug GH-16137 duplicate *Forwarded* HTTP headers values. [sapi/cli/tests/gh16137.php]
PASS Test parsing longer opcode [sapi/phpdbg/tests/breakpoints_009.php]
PASS STD handles can be deliberately closed 001 [sapi/cli/tests/gh8827-001.php]
PASS Bug #73615 (phpdbg without option never load .phdbginit at startup) [sapi/phpdbg/tests/bug73615.php]
PASS STD handles can be deliberately closed 002 [sapi/cli/tests/gh8827-002.php]
PASS STD handles can be deliberately closed 003 [sapi/cli/tests/gh8827-003.php]
PASS Bug #73704 (phpdbg shows the wrong line in files with shebang) [sapi/phpdbg/tests/bug73704.php]
PASS Bug #73794 (Crash (out of memory) when using run # and command separator) [sapi/phpdbg/tests/bug73794.php]
PASS GHSA-4w77-75f9-2c8w (Heap-Use-After-Free in sapi read_post_data Processing in CLI SAPI Interface) [sapi]
PASS Bug #73927 (phpdbg fails with windows error prompt at "watch array") [sapi/phpdbg/tests/bug73927.php]
PASS basic function [sapi/cli/tests/php_cli_server_001.php]
$_SERVER variable [sapi/cli/tests/php_cli_server_002.php]
PASS Bug #5726 (Changing the working directory makes router script inaccessible) [sapi/cli/tests/php]
PASS Bug #5747 (request headers missed in $_SERVER) [sapi/cli/tests/php_cli_server_004.php]
PASS Post a file [sapi/cli/tests/php_cli_server_005.php]
PASS include(jing files should not raise "too many open files" error [sapi/phpdbg/tests/bug76801.php]
PASS Bug #5755 (SegFault when outputting header WWW-Authenticate) [sapi/cli/tests/php_cli_server_006.php]
PASS Bug #78297: Include unexistent file memory leak (Heap-Use-After-Free in sapi read_post_data Processing in CLI SAPI Interface) [sapi]
PASS Bug #81135: unknown help topic file assertion [sapi/phpdbg/tests/bug81135.php]
PASS Bug #5758 (Digest Authenticate missed in 5.4) [sapi/cli/tests/php_cli_server_007.php]
PASS Cleaning must preserve breakpoints [sapi/phpdbg/tests/clean_001.php]
PASS Test clearing breakpoints [sapi/phpdbg/tests/clean_001.php]
PASS Test # delimiter parsing and execution [sapi/phpdbg/tests/delimiter.php]
PASS SERVER_PROTOCOL header availability [sapi/cli/tests/php_cli_server_008.php]
PASS PATH_INFO (relevant to #60112) [sapi/cli/tests/php_cli_server_009.php]
PASS Bug #60108 ( $_SERVER["PHP_SELF"] incorrect) [sapi/cli/tests/php_cli_server_010.php]
PASS Bug #60159 (Router returns false, but POST is not passed to requested resource) [sapi/cli/tests/php]
SKIP FPM: Ping on the status invisible pool [sapi/fpm/tests/status_ping.php] reason: php-fpm binary
SKIP FPM: Status listen test [sapi/fpm/tests/status_listen.php] reason: php-fpm binary not found
SKIP FPM: Status pool is inheriting FPM setting overrides test - expose_php = on [sapi/fpm/tests/status.php]
SKIP FPM: Status pool is inheriting FPM setting overrides test - expose_php = off [sapi/fpm/tests/status.php]
SKIP FPM: Status basic test [sapi/fpm/tests/status_basic.php] reason: php-fpm binary not found
SKIP FPM: UNIX socket filename is too far for test [sapi/fpm/tests/socket_uds_too_long-filename-test.php]
SKIP FPM: UNIX socket filename is too far for start [sapi/fpm/tests/socket_uds_too-long-filename-start.php]
SKIP FPM: UNIX socket owner and group settings can be numeric [sapi/fpm/tests/socket_uds_numeric-ugid.php]
SKIP FPM: UNIX socket owner and group settings can be numeric [sapi/fpm/tests/socket_uds_numeric-ugid.php]
```

```
PASS File upload greater than 2G [sapi/cli/tests/upload_2G.php]
PASS Bug #69900 Commandline input/output weird behaviour with STDIO [ext/streams]
SKIP Bug #81145 (copy() and stream_copy_to_stream() fail for +4GB files)
PASS Phar::buildFromIterator() RegexIterator(RecursiveIteratorIterator),
SKIP Bind limits [ext/mysql/tests/mysql_stmt_bind_limits.php] reason:
SKIP mysql_fetch_array() - large packages (to test compression) [ext/mysql]
=====
TIME END 2024-12-06 21:20:03
=====
TEST RESULT SUMMARY
=====
Exts skipped : 38 (enchant, sodium, calendar, pgsql, curl, exif, geoip, msg, odbc, soap, ldap, dba, zend_test, snmp, bz2, gd, openssl, xsl, dl, tokenizer)
Exts tested : 31
=====
Number of tests : 20043 15002
Tests skipped : 5041 ( 25.2%) ( 0.0%)
Tests warned : 0 ( 0.0%) ( 0.0%)
Tests failed : 0 ( 0.0%) ( 0.0%)
=====
expected fail : / ( 0.0%) ( 0.0%)
Tests passed : 14995 ( 74.8%) (100.0%)
Time taken : 488.016 seconds
=====
EXPECTED FAILED TEST SUMMARY
=====
Test open_basedir configuration [tests/security/open_basedir_linkinfo.php]
bugs 48111 and 52176.
Interfaces don't set prototypes to their parent method [zend/tests/Interface.php]
uses prototype to verify LSP, but misses A::__construct() which has a string signature.
Bug #67296 (filter_input doesn't validate variables) [ext/filter/tests/filter.php]
Bug #42718 (unsafe_raw filter not applied when configured as default filter)
Bug #67167: filter_var(null,FILTER_VALIDATE_BOOLEAN,FILTER_NULL_ON_FAILURE).
Return type check elision [ext/opcache/tests/opt/verify_return_type.php]
updateTimestamp never called when session data is empty [ext/session/test.php]
=====
This report can be saved and used to open an issue on the bug tracker at
https://github.com/php/php-src/issues
This gives us a better understanding of PHP's behavior.
Do you want to save this report in a file? [Yn]: n
```

Roadmap: 4 Learn PHP Tests - PHPT

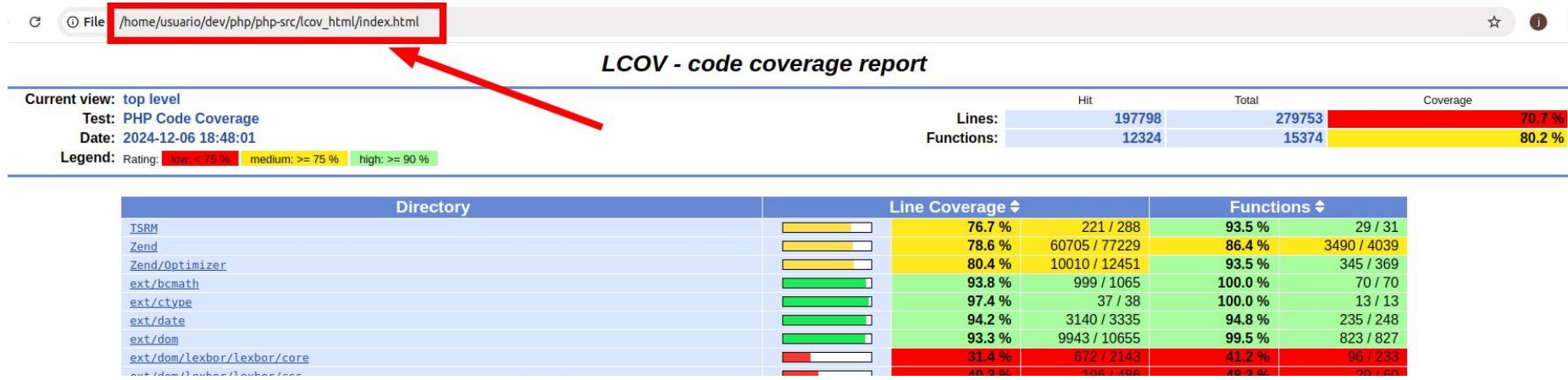
Write a QA Test to increase source code test coverage

```
PASS Fundamental breakpoints functionality [sapi/phpdbg/tests/breakpoints_001.php]
PASS A script with die() must end "normally" [sapi/phpdbg/tests/normal_exit.php]
PASS Test phpdbg_break_next() function [sapi/phpdbg/tests/phpdbg_break_next.php]
PASS Preserve breakpoints on restart [sapi/phpdbg/tests/breakpoints_002.php]
PASS Extension loading [sapi/cli/tests/ext_loading.php]
PASS Getting executable lines from custom wrappers [sapi/phpdbg/tests/phpdbg_get_executable_stream.php]
PASS Test deleting breakpoints [sapi/phpdbg/tests/breakpoints_003.php]
PASS Test phpdbg_*oplog() functions [sapi/phpdbg/tests/phpdbg_oplog_001.php]
PASS Test opcode breakpoints [sapi/phpdbg/tests/breakpoints_004.php]
PASS Ensure a single Date header is present [sapi/cli/tests/gh12363.php]
PASS phpdbg_end_oplog() alone must not crash [sapi/phpdbg/tests/phpdbg_oplog_002.php]
PASS Test breakpoint into function context [sapi/phpdbg/tests/breakpoints_005.php]
PASS Basic function breakpoints [sapi/phpdbg/tests/breakpoints_006.php]
PASS GH-13113: Missing syntax error in CLI-server router script [sapi/cli/tests/gh13113.php]
SKIP GH-14189 (PHP Interactive shell input state incorrectly handles quoted heredoc literals.) [sapi]
PASS Basic method breakpoints [sapi/phpdbg/tests/breakpoints_007.php]
PASS Test namespaced and non-lowercase breakpoint names [sapi/phpdbg/tests/breakpoints_008.php]
PASS Bug GH-16137 duplicate *Forwarded* HTTP headers values. [sapi/cli/tests/gh16137.php]
PASS Test parsing longer opcode [sapi/phpdbg/tests/breakpoints_009.php]
PASS STD handles can be deliberately closed 001 [sapi/cli/tests/gh8827-001.php]
PASS Bug #73615 (phpdbg without option never load .phdbginit at startup) [sapi/phpdbg/tests/bug73615.php]
PASS STD handles can be deliberately closed 002 [sapi/cli/tests/gh8827-002.php]
PASS STD handles can be deliberately closed 003 [sapi/cli/tests/gh8827-003.php]
PASS Bug #73704 (phpdbg shows the wrong line in files with shebang) [sapi/phpdbg/tests/bug73704.php]
PASS Bug #73794 (Crash (out of memory) when using run # and command separator) [sapi/phpdbg/tests/bug73794.php]
PASS GHSA-4w77-75f9-2c8w (Heap-Use-After-Free in sapi read_post_data Processing in CLI SAPI Interface) [sapi]
PASS Bug #73927 (phpdbg fails with windows error prompt at "watch array") [sapi/phpdbg/tests/bug73927.php]
PASS basic function [sapi/cli/tests/php_cli_server_001.php]
$_SERVER variable [sapi/cli/tests/php_cli_server_002.php]
PASS Bug #5726 (Changing the working directory makes router script inaccessible) [sapi/cli/tests/php]
PASS Bug #5747 (request headers missed in $_SERVER) [sapi/cli/tests/php_cli_server_004.php]
PASS Post a file [sapi/cli/tests/php_cli_server_005.php]
PASS include(jing files should not raise "too many open files" error [sapi/phpdbg/tests/bug76801.php]
PASS Bug #5755 (SegFault when outputting header WWW-Authenticate) [sapi/cli/tests/php_cli_server_006.php]
PASS Bug #78297: Include unexistent file memory leak (Heap-Use-After-Free in sapi read_post_data Processing in CLI SAPI Interface) [sapi]
PASS Bug #81135: unknown help topic file assertion [sapi/phpdbg/tests/bug81135.php]
PASS Bug #5758 (Digest Authenticate missed in 5.4) [sapi/cli/tests/php_cli_server_007.php]
PASS Cleaning must preserve breakpoints [sapi/phpdbg/tests/clean_001.php]
PASS Test clearing breakpoints [sapi/phpdbg/tests/clean_001.php]
PASS Test # delimiter parsing and execution [sapi/phpdbg/tests/delimiter.php]
PASS SERVER_PROTOCOL header availability [sapi/cli/tests/php_cli_server_008.php]
PASS PATH_INFO (relevant to #60112) [sapi/cli/tests/php_cli_server_009.php]
PASS Bug #60108 ( $_SERVER["PHP_SELF"] incorrect) [sapi/cli/tests/php_cli_server_010.php]
PASS Bug #60159 (Router returns false, but POST is not passed to requested resource) [sapi/cli/tests/php]
SKIP FPM: Ping on the status invisible pool [sapi/fpm/tests/status_ping.php] reason: php-fpm binary
SKIP FPM: Status listen test [sapi/fpm/tests/status_listen.php] reason: php-fpm binary not found
SKIP FPM: Status pool is inheriting FPM setting overrides test - expose_php = on [sapi/fpm/tests/status.php]
SKIP FPM: Status pool is inheriting FPM setting overrides test - expose_php = off [sapi/fpm/tests/status.php]
SKIP FPM: Status basic test [sapi/fpm/tests/status_basic.php] reason: php-fpm binary not found
SKIP FPM: UNIX socket filename is too far for test [sapi/fpm/tests/socket_uds_too_long-filename-test.php]
SKIP FPM: UNIX socket filename is too far for start [sapi/fpm/tests/socket_uds_too_long-filename-start.php]
SKIP FPM: UNIX socket owner and group settings can be numeric [sapi/fpm/tests/socket_uds_numeric-ugid.php]
SKIP FPM: UNIX socket owner and group settings can be numeric [sapi/fpm/tests/socket_uds_numeric-ugid.php]
```

```
PASS File upload greater than 2G [sapi/cli/tests/upload_2G.php]
PASS Bug #69900 Commandline input/output weird behaviour with STDIO [ext/streams]
SKIP Bug #81145 (copy() and stream_copy_to_stream() fail for +4GB files)
PASS Phar::buildFromIterator() RegexIterator(RecursiveIteratorIterator),
SKIP Bind limits [ext/mysql/tests/mysql_stmt_bind_limits.php] reason:
SKIP mysql_fetch_array() - large packages (to test compression) [ext/mysql]
=====
TIME END 2024-12-06 21:20:03
=====
TEST RESULT SUMMARY
=====
Exts skipped : 38 (enchant, sodium, calendar, pgsql, curl, exif, geoip, msg, odbc, soap, ldap, dba, zend_test, snmp, bz2, gd, openssl, xsl, dl, tokenizer)
Exts tested : 31
=====
Number of tests : 20043 15002
Tests skipped : 5041 ( 25.2%) ( 0.0%)
Tests warned : 0 ( 0.0%) ( 0.0%)
Tests failed : 0 ( 0.0%) ( 0.0%)
=====
expected fail : / ( 0.0%) ( 0.0%)
Tests passed : 14995 ( 74.8%) (100.0%)
Time taken : 488.016 seconds
=====
EXPECTED FAILED TEST SUMMARY
=====
Test open_basedir configuration [tests/security/open_basedir_linkinfo.php]
bugs 48111 and 52176.
Interfaces don't set prototypes to their parent method [zend/tests/Interface.php]
uses prototype to verify LSP, but misses A::__construct() which has a string signature.
Bug #67296 (filter_input doesn't validate variables) [ext/filter/tests/filter.php]
Bug #42718 (unsafe_raw filter not applied when configured as default filter)
Bug #67167: filter_var(null,FILTER_VALIDATE_BOOLEAN,FILTER_NULL_ON_FAILURE).
Return type check elision [ext/opcache/tests/opt/verify_return_type.php]
updateTimestamp never called when session data is empty [ext/session/test.php]
=====
This report can be saved and used to open an issue on the bug tracker at
https://github.com/php/php-src/issues
This gives us a better understanding of PHP's behavior.
Do you want to save this report in a file? [Yn]: n
```

Roadmap: 4 Learn PHP Tests - PHPT

Write a QA Test to increase source code test coverage



Results may vary depending on how
PHP was configured while building
it. (the ./configure part)

Roadmap: 4 Learn PHP Tests - PHPT

Example

QA - Test Cov - ext:pcntl - pcntl_signal() - max signal allowed #8956

 Merged Girgias merged 4 commits into [php:master](#) from [juan-morales:qa-pcntl_signal-max-signal](#)  on Jul 11, 2022

Conversation 12 · Commits 4 · Checks 0 · Files changed 1

 juan-morales commented on Jul 8, 2022 · Contributor · 

Extension: pcntl
Function: pcntl_signal
Description:
When provided the `$signal` parameter is greater than allowed in operating system, then expect an error.
I hard-coded the number 1000000 on purpose because I cannot find a constant or function that can give me the max allowed in order to increment it.
Please, all advices are welcome.
I also attach what was covered in the following screenshots:

Before

```
1:  /*{{{ Error handling */
2:  4:      if (zend_error_get_error() != E_ERROR)
3:  5:          php_error_doref(NULL, E_WARNING, "Error has occurred: (%d) %s", errno, strerror(errno));
4:  6:      efree(argv);
5:  7:      RETURN_FALSE;
6:  8:  /* }}} */
```

76:  `/* {{{ Assigns a system signal handler to a PHP function */
8: 9: PHP_FUNCTION(pcntl_signal)
9: 10: {
10: 11: zval *handle;
11: 12: zend_long signo;
12: 13: bool restart_syscalls = 1;
13: 14: bool restart_syscalls_is_null = 1;
14: 15: char *error = NULL;
15: 16: }`

Reviewers
 Girgias
 cmb69

Assignees
No one assigned

Labels
 Extension: pcntl

Projects
None yet

Milestone
No milestone

Development
Successfully merging this pull request

Roadmap: 4 Learn PHP Tests - PHPT

Example

```
939      :
940      : /* {{{ Assigns a system signal handler to a PHP function */
941 76 : PHP_FUNCTION(pcntl_signal)
942      :
943      :     zval *handle;
944      :     zend_long signo;
945 76 :     bool restart_syscalls = 1;
946 76 :     bool restart_syscalls_is_null = 1;
947 76 :     char *error = NULL;
948      :
949 76 :     ZEND_PARSE_PARAMETERS_START(2, 3)
950 48 :         Z_PARAM_LONG(signo)
951 44 :         Z_PARAM_ZVAL(handle)
952 44 :         Z_PARAM_OPTIONAL
953 44 :         Z_PARAM_BOOL_OR_NULL(restart_syscalls, restart_syscalls_is_null)
954 76 :     ZEND_PARSE_PARAMETERS_END();
955      :
956 44 :     if (signo < 1) {
957 8 :         zend_argument_value_error(1, "must be greater than or equal to 1");
958 8 :         RETURN_THROWS();
959      :
960      :
961 36 :     if (signo >= PCNTL_G(num_signals)) {
962 0 :         zend_argument_value_error(1, "must be less than %d", PCNTL_G(num_signals));
963 0 :         RETURN_THROWS();
964      :
965      :
966 36 :     if (!PCNTL_G(spares)) {
967      :         /* since calling malloc() from within a signal handler is not portable
```

Roadmap: 4 Learn PHP Tests - PHPT

Example

name of a userland php function, because of PHP_FUNCTION()

```
939 :     :
940 :     : /* {{{ Assigns a system signal handler to a PHP function */
941 : 76 :     : PHP_FUNCTION(pcntl_signal)
942 : 76 :     :
943 :     :     zval *handle;
944 :     :     zend_long signo;
945 : 76 :     :     bool restart_syscalls = 1;
946 : 76 :     :     bool restart_syscalls_is_null = 1;
947 : 76 :     :     char *error = NULL;
948 :     :
949 : 76 :     :     ZEND_PARSE_PARAMETERS_START(2, 3)
950 : 48 :     :         Z_PARAM_LONG(signo)
951 : 44 :     :         Z_PARAM_ZVAL(handle)
952 : 44 :     :         Z_PARAM_OPTIONAL
953 : 44 :     :         Z_PARAM_BOOL_OR_NULL(restart_syscalls, restart_syscalls_is_null)
954 : 76 :     :     ZEND_PARSE_PARAMETERS_END();
955 :
956 : 44 :     :     if (signo < 1) {
957 : 8 :         :         zend_argument_value_error(1, "must be greater than or equal to 1");
958 : 8 :         :         RETURN_THROWS();
959 :     :     }
960 :
961 : 36 :     :     if (signo >= PCNTL_G(num_signals)) {
962 : 0 :         :         zend_argument_value_error(1, "must be less than %d", PCNTL_G(num_signals));
963 : 0 :         :         RETURN_THROWS();
964 :     :     }
965 :
966 : 36 :     :     if (!PCNTL_G(spares)) {
967 :         :         /* since calling malloc() from within a signal handler is not portable
```

some variables

parsing parameters

if parameter "signo" is bigger than "max allowed" we inform "that" and returns

Roadmap: 4 Learn PHP Tests - PHPT

Example

Change language: [English](#)

pcntl_signal

(PHP 4 >= 4.1.0, PHP 5, PHP 7, PHP 8)

`pcntl_signal` – Installs a signal handler

Description

```
pcntl_signal(int $signal, callable|int $handler, bool $restart_syscalls = true): bool
```

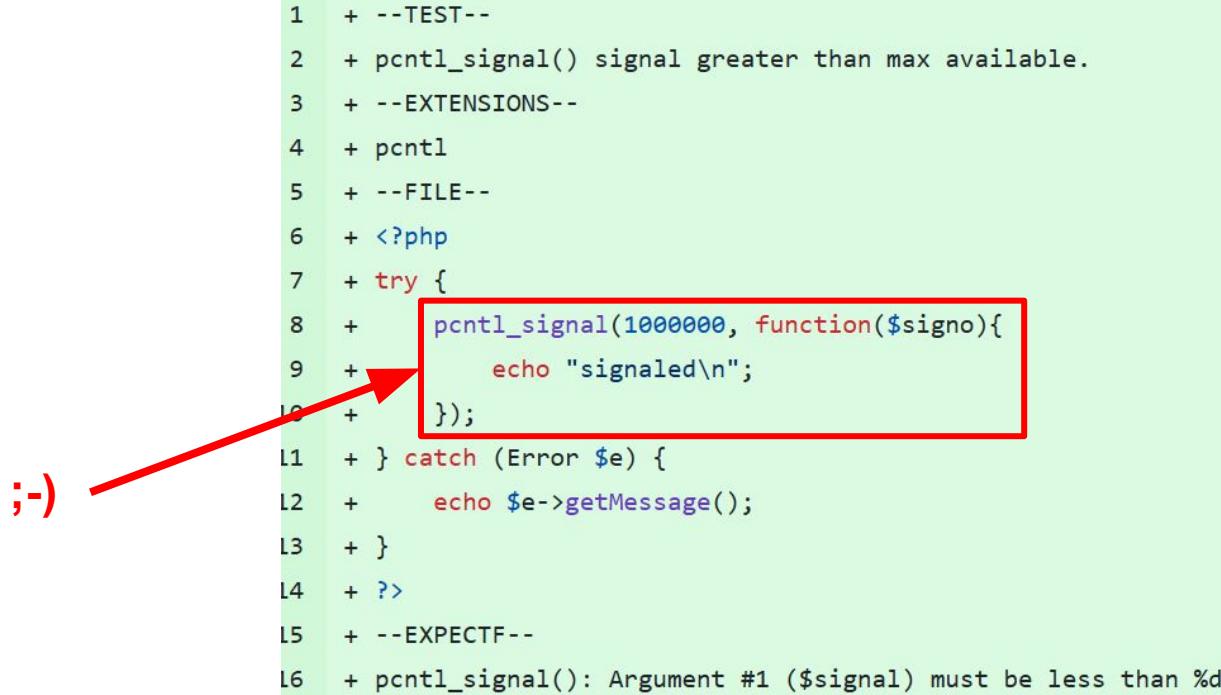
The `pcntl_signal()` function installs a new signal handler or replaces the current signal handler for the signal indicated by `signal`.

Roadmap: 4 Learn PHP Tests - PHPT

Example

```
1  + --TEST--
2  + pcntl_signal() signal greater than max available.
3  + --EXTENSIONS--
4  + pcntl
5  + --FILE--
6  + <?php
7  + try {
8  +     pcntl_signal(1000000, function($signo){
9  +         echo "signaled\n";
10 +    });
L1 + } catch (Error $e) {
L2 +     echo $e->getMessage();
L3 + }
L4 + ?>
L5 + --EXPECTF--
L6 + pcntl_signal(): Argument #1 ($signal) must be less than %d
```

;-)



Roadmap: 4 Learn PHP Tests - PHPT

Example

```
940  : /* {{{ Assigns a system signal handler to a PHP function */
941 82 : PHP_FUNCTION(pcntl_signal)
942  :
943  :     zval *handle;
944  :     zend_long signo;
945 82 :     bool restart_syscalls = 1;
946 82 :     bool restart_syscalls_is_null = 1;
947 82 :     char *error = NULL;
948  :
949 82 :     ZEND_PARSE_PARAMETERS_START(2, 3)
950 54 :         Z_PARAM_LONG(signo)
951 50 :         Z_PARAM_ZVAL(handle)
952 50 :         Z_PARAM OPTIONAL
953 50 :             Z_PARAM_BOOL_OR_NULL(restart_syscalls, restart_syscalls_is_null)
954 82 :     ZEND_PARSE_PARAMETERS_END();
955  :
956 50 :     if (signo < 1) {
957 8 :         zend_argument_value_error(1, "must be greater than or equal to 1");
958 8 :         RETURN_THROWS();
959  :
960  :
961 42 :     if (signo >= PCNTL_G(num_signals)) {
962 6 :         zend_argument_value_error(1, "must be less than %d", PCNTL_G(num_signals));
963 6 :         RETURN_THROWS();
964  :
965  :
```

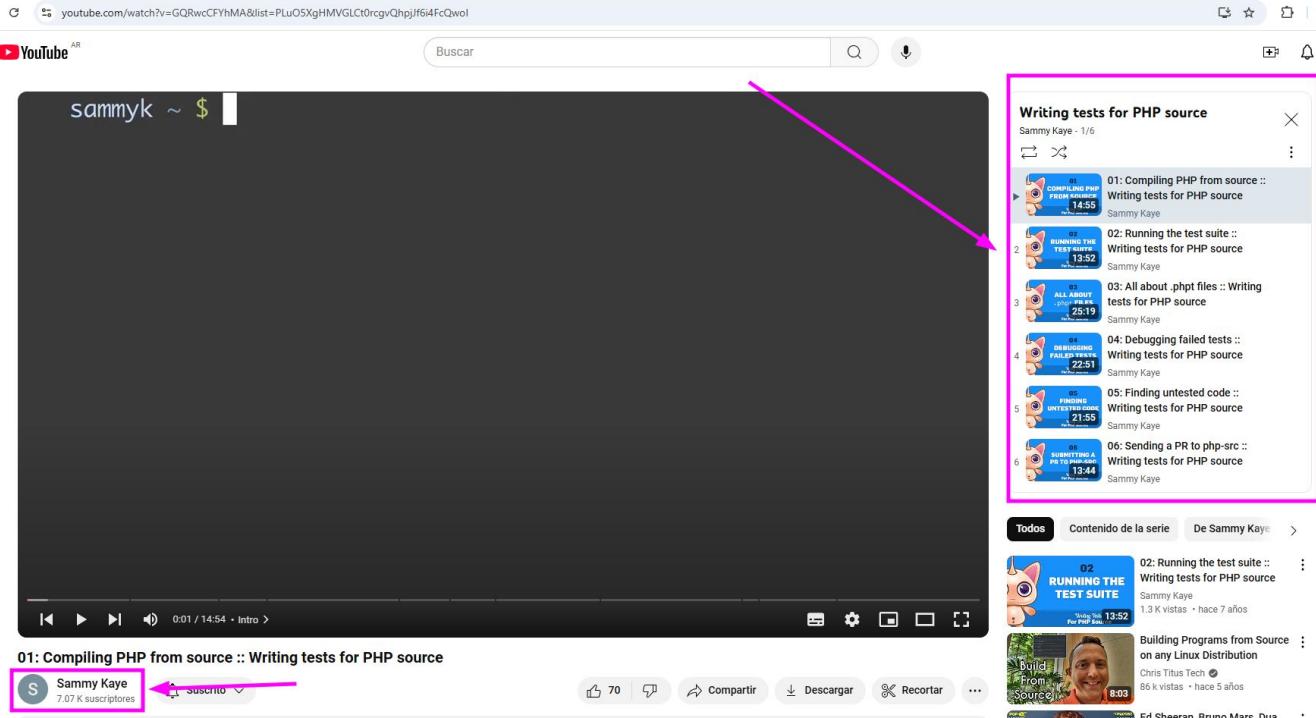
Roadmap: 4 Learn PHP Tests - PHPT

Example

- Include all information possible in your pull request and provide coverage before and after your test
- **Did you read my comment in the example? Nice idea for an RFC right? ;)**
- **Nice opportunity to enhance the documentation right? :D**
- **According to my experience writing coverage (QA) tests is a smooth way to learn about PHPs internals**

Roadmap: 4 Learn PHP Tests - PHPT

Nice tutorial by Sammy Kaye on YouTube



The image shows a YouTube video player with a dark theme. On the left, a terminal window displays the command `sammyk ~ $`. A pink arrow points from the terminal area to a sidebar on the right. The sidebar is titled "Writing tests for PHP source" and lists six video thumbnails, each with a title, duration, and the name "Sammy Kaye". The titles are: 01: Compiling PHP from source :: Writing tests for PHP source (14:55), 02: Running the test suite :: Writing tests for PHP source (13:52), 03: All about .php files :: Writing tests for PHP source (25:19), 04: Debugging failed tests :: Writing tests for PHP source (22:31), 05: Finding untested code :: Writing tests for PHP source (21:55), and 06: Sending a PR to php-src :: Writing tests for PHP source (13:44). Below the sidebar, there are navigation buttons for "Todos", "Contenido de la serie", and "De Sammy Kaye". At the bottom of the video player, there is a "Suscribir" button with a "7.07 K subscribers" count, and a list of recommended videos on the right.

01: Compiling PHP from source :: Writing tests for PHP source

Sammy Kaye 7.07 K subscribers

02: Running the test suite :: Writing tests for PHP source

Sammy Kaye 1.3 K vistas • hace 7 años

Building Programs from Source :: on any Linux Distribution

Chris Titus Tech 86 k vistas • hace 5 años

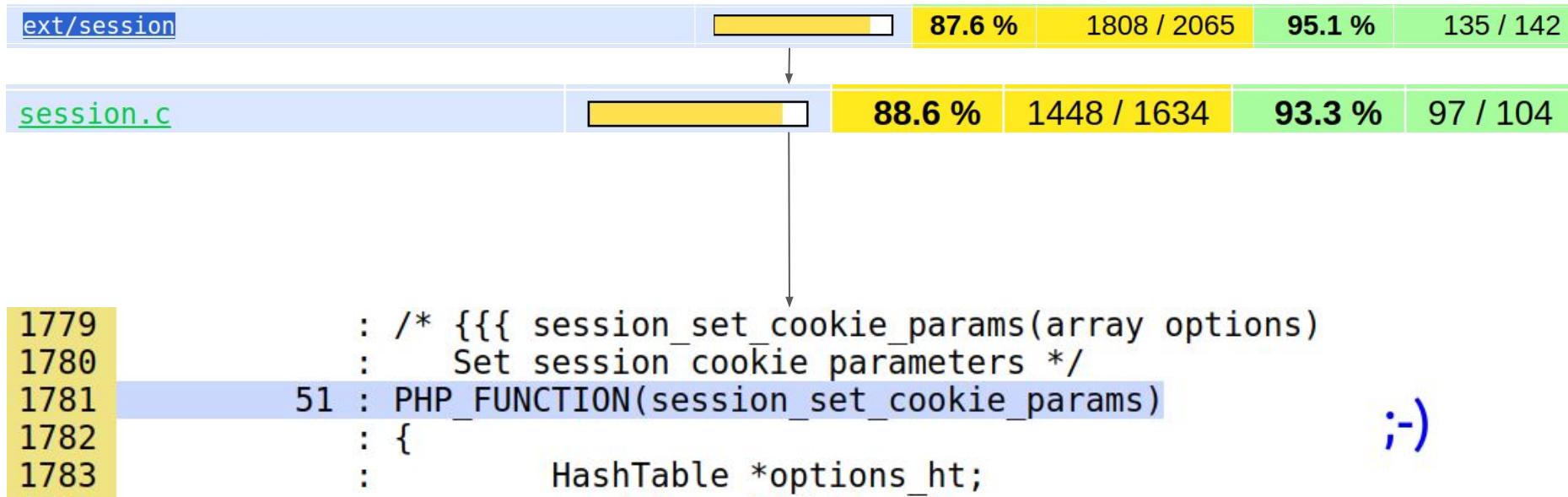
Ed Sheeran Bring Me The Horizon

Roadmap: 4 Learn PHP Tests - PHPT

- Writing QA Tests will give you confidence and better understanding of what you read in documentation related to PHP's internals
- You will move through the source code with more knowledge about it
- In order to understand how the test should be written in order to cover PHP's C code, you need to understand what the C code is doing, in order to check how to cover that code from a PHP script. This process accelerates the learning process of PHP Core Code and gives better understanding of related documentation

Roadmap: 4 Learn PHP Tests - PHPT

Want to write a test ? :D



Roadmap: 4 Learn PHP Tests - PHPT

a question ...

What could happen while writing tests
for uncovered code ?

Roadmap to start contributing to the code of PHP

- 1 Join the community
- 2 Read contributor information
- 3 Setup local environment
- 4 Learn PHP Tests (PHPT)
- 5 Fix a bug

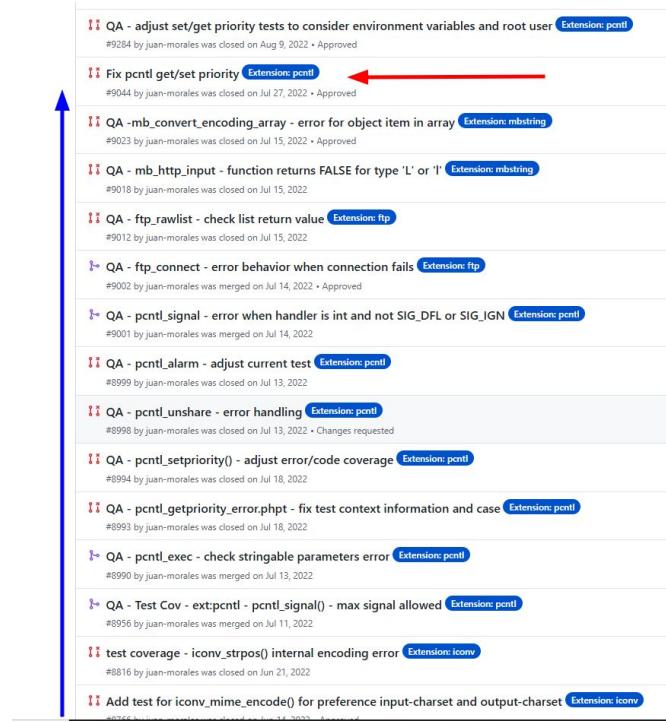
Always

Read the existing documentation and learn from it, while walking the roadmap

Roadmap: 5 Fix a bug

Overview

While writing test for uncovered code, could be that you discover a bug ;)



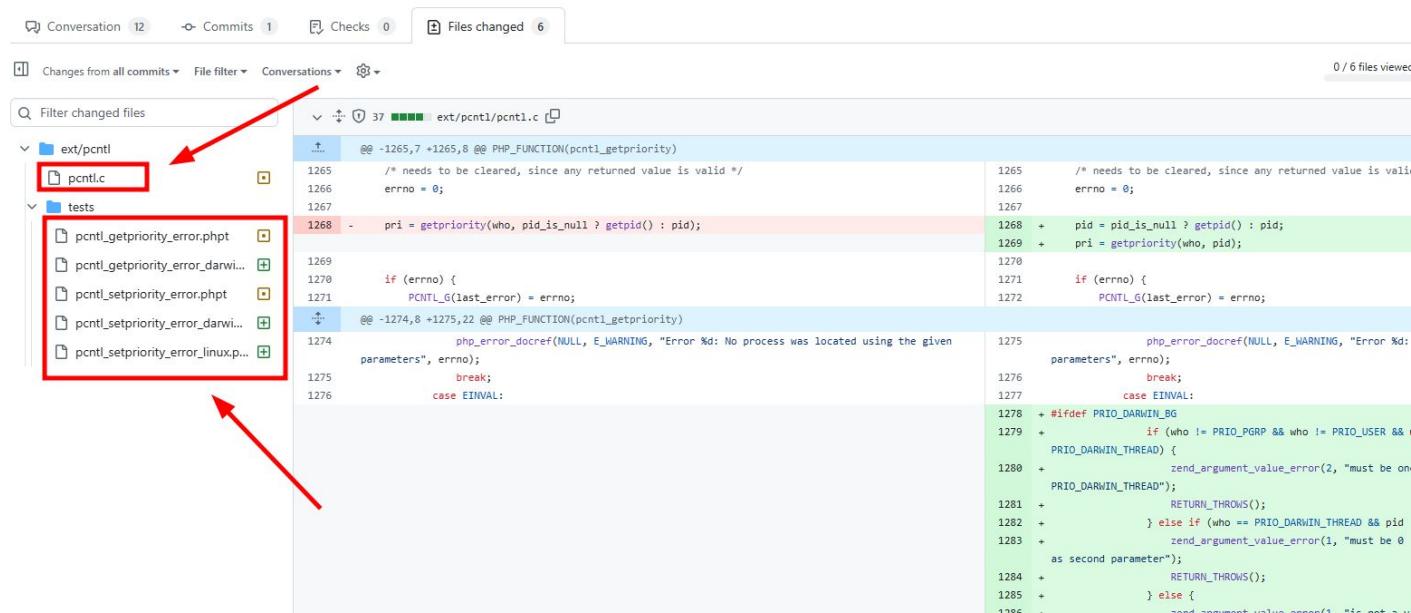
II	QA - adjust set/get priority tests to consider environment variables and root user	Extension: pcntl
II	Fix pcntl_get/set priority	Extension: pcntl
II	QA - mb_convert_encoding_array - error for object item in array	Extension: mbstring
II	QA - mb_http_input - function returns FALSE for type 'L' or 'I'	Extension: mbstring
II	QA - ftp_rawlist - check list return value	Extension: ftp
II	QA - ftp_connect - error behavior when connection fails	Extension: ftp
II	QA - pcntl_signal - error when handler is int and not SIG_DFL or SIG_IGN	Extension: pcntl
II	QA - pcntl_alarm - adjust current test	Extension: pcntl
II	QA - pcntl_unshare - error handling	Extension: pcntl
II	QA - pcntl_setpriority() - adjust error/code coverage	Extension: pcntl
II	QA - pcntl_getpriority_error.php - fix test context information and case	Extension: pcntl
II	QA - pcntl_exec - check stringable parameters error	Extension: pcntl
II	QA - Test Cov - ext:pcntl - pcntl_signal() - max signal allowed	Extension: pcntl
II	test coverage - iconv_strpos() internal encoding error	Extension: iconv
II	Add test for iconv_mime_encode() for preference input-charset and output-charset	Extension: iconv

Roadmap: 5 Fix a bug

Overview

Fix pcntl get/set priority #9044

 **Closed** juan-morales wants to merge 1 commit into `php:master` from `juan-morales:fix-pcntl_get/set-priority` 



Roadmap: 5 Fix a bug

Overview

Fix pcntl get/set priority #9044

A screenshot of a GitHub pull request interface. The title is "Fix pcntl get/set priority #9044". The status is "Closed" with a note that "juan-morales wants to merge 1 commit into php:master from juan-morales:fix-pcntl_get/set-priority". The commit list shows 1 commit with 6 files changed. The commit message is "@ -1265,7 +1265,8 @@ PHP_FUNCTION(pcntl_getpriority)". The file changes are shown in a diff view. A red arrow points from the text "Little bit of C" to the file "pcntl.c". Another red arrow points from the text "And many tests :D" to the test files in the "tests" directory.

Little bit of C

And many tests :D

```
diff --git a/ext/pcntl/pcntl.c b/ext/pcntl/pcntl.c
@@ -1265,7 +1265,8 @@ PHP_FUNCTION(pcntl_getpriority)
1265  /* needs to be cleared, since any returned value is valid */
1266  errno = 0;
1267
1268 - pri = getpriority(who, pid_is_null ? getpid() : pid);
1269
1270     if (errno) {
1271         PCNTL_G(last_error) = errno;
1272
1273     @@ -1274,8 +1275,22 @@ PHP_FUNCTION(pcntl_getpriority)
1274         php_error_docref(NULL, E_WARNING, "Error %d: No process was located using the given
1275         parameters", errno);
1276         break;
1277     case EINVAL:
1278
1279     + #ifdef PRIO_DARWIN_BG
1280     +         if (who != PRIO_PGRP && who != PRIO_USER && !PRIO_DARWIN_THREAD) {
1281         zend_argument_value_error(2, "must be on
1282         PRIO_DARWIN_THREAD");
1283
1284     +             RETURN_THROWS();
1285     +         } else if (who == PRIO_DARWIN_THREAD && pid
1286     +             zend_argument_value_error(1, "must be 0
1287     +             as second parameter");
1288
1289     +             RETURN_THROWS();
1290     +         } else {
1291
1292             zend_argument_value_error(1, "pid not >
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1499
1500
1500
1501
1501
1502
1502
1503
1503
1504
1504
1505
1505
1506
1506
1507
1507
1508
1508
1509
1509
1510
1510
1511
1511
1512
1512
1513
1513
1514
1514
1515
1515
1516
1516
1517
1517
1518
1518
1519
1519
1520
1520
1521
1521
1522
1522
1523
1523
1524
1524
1525
1525
1526
1526
1527
1527
1528
1528
1529
1529
1530
1530
1531
1531
1532
1532
1533
1533
1534
1534
1535
1535
1536
1536
1537
1537
1538
1538
1539
1539
1540
1540
1541
1541
1542
1542
1543
1543
1544
1544
1545
1545
1546
1546
1547
1547
1548
1548
1549
1549
1550
1550
1551
1551
1552
1552
1553
1553
1554
1554
1555
1555
1556
1556
1557
1557
1558
1558
1559
1559
1560
1560
1561
1561
1562
1562
1563
1563
1564
1564
1565
1565
1566
1566
1567
1567
1568
1568
1569
1569
1570
1570
1571
1571
1572
1572
1573
1573
1574
1574
1575
1575
1576
1576
1577
1577
1578
1578
1579
1579
1580
1580
1581
1581
1582
1582
1583
1583
1584
1584
1585
1585
1586
1586
1587
1587
1588
1588
1589
1589
1590
1590
1591
1591
1592
1592
1593
1593
1594
1594
1595
1595
1596
1596
1597
1597
1598
1598
1599
1599
1600
1600
1601
1601
1602
1602
1603
1603
1604
1604
1605
1605
1606
1606
1607
1607
1608
1608
1609
1609
1610
1610
1611
1611
1612
1612
1613
1613
1614
1614
1615
1615
1616
1616
1617
1617
1618
1618
1619
1619
1620
1620
1621
1621
1622
1622
1623
1623
1624
1624
1625
1625
1626
1626
1627
1627
1628
1628
1629
1629
1630
1630
1631
1631
1632
1632
1633
1633
1634
1634
1635
1635
1636
1636
1637
1637
1638
1638
1639
1639
1640
1640
1641
1641
1642
1642
1643
1643
1644
1644
1645
1645
1646
1646
1647
1647
1648
1648
1649
1649
1650
1650
1651
1651
1652
1652
1653
1653
1654
1654
1655
1655
1656
1656
1657
1657
1658
1658
1659
1659
1660
1660
1661
1661
1662
1662
1663
1663
1664
1664
1665
1665
1666
1666
1667
1667
1668
1668
1669
1669
1670
1670
1671
1671
1672
1672
1673
1673
1674
1674
1675
1675
1676
1676
1677
1677
1678
1678
1679
1679
1680
1680
1681
1681
1682
1682
1683
1683
1684
1684
1685
1685
1686
1686
1687
1687
1688
1688
1689
1689
1690
1690
1691
1691
1692
1692
1693
1693
1694
1694
1695
1695
1696
1696
1697
1697
1698
1698
1699
1699
1700
1700
1701
1701
1702
1702
1703
1703
1704
1704
1705
1705
1706
1706
1707
1707
1708
1708
1709
1709
1710
1710
1711
1711
1712
1712
1713
1713
1714
1714
1715
1715
1716
1716
1717
1717
1718
1718
1719
1719
1720
1720
1721
1721
1722
1722
1723
1723
1724
1724
1725
1725
1726
1726
1727
1727
1728
1728
1729
1729
1730
1730
1731
1731
1732
1732
1733
1733
1734
1734
1735
1735
1736
1736
1737
1737
1738
1738
1739
1739
1740
1740
1741
1741
1742
1742
1743
1743
1744
1744
1745
1745
1746
1746
1747
1747
1748
1748
1749
1749
1750
1750
1751
1751
1752
1752
1753
1753
1754
1754
1755
1755
1756
1756
1757
1757
1758
1758
1759
1759
1760
1760
1761
1761
1762
1762
1763
1763
1764
1764
1765
1765
1766
1766
1767
1767
1768
1768
1769
1769
1770
1770
1771
1771
1772
1772
1773
1773
1774
1774
1775
1775
1776
1776
1777
1777
1778
1778
1779
1779
1780
1780
1781
1781
1782
1782
1783
1783
1784
1784
1785
1785
1786
1786
1787
1787
1788
1788
1789
1789
1790
1790
1791
1791
1792
1792
1793
1793
1794
1794
1795
1795
1796
1796
1797
1797
1798
1798
1799
1799
1800
1800
1801
1801
1802
1802
1803
1803
1804
1804
1805
1805
1806
1806
1807
1807
1808
1808
1809
1809
1810
1810
1811
1811
1812
1812
1813
1813
1814
1814
1815
1815
1816
1816
1817
1817
1818
1818
1819
1819
1820
1820
1821
1821
1822
1822
1823
1823
1824
1824
1825
1825
1826
1826
1827
1827
1828
1828
1829
1829
1830
1830
1831
1831
1832
1832
1833
1833
1834
1834
1835
1835
1836
1836
1837
1837
1838
1838
1839
1839
1840
1840
1841
1841
1842
1842
1843
1843
1844
1844
1845
1845
1846
1846
1847
1847
1848
1848
1849
1849
1850
1850
1851
1851
1852
1852
1853
1853
1854
1854
1855
1855
1856
1856
1857
1857
1858
1858
1859
1859
1860
1860
1861
1861
1862
1862
1863
1863
1864
1864
1865
1865
1866
1866
1867
1867
1868
1868
1869
1869
1870
1870
1871
1871
1872
1872
1873
1873
1874
1874
1875
1875
1876
1876
1877
1877
1878
1878
1879
1879
1880
1880
1881
1881
1882
1882
1883
1883
1884
1884
1885
1885
1886
1886
1887
1887
1888
1888
1889
1889
1890
1890
1891
1891
1892
1892
1893
1893
1894
1894
1895
1895
1896
1896
1897
1897
1898
1898
1899
1899
1900
1900
1901
1901
1902
1902
1903
1903
1904
1904
1905
1905
1906
1906
1907
1907
1908
1908
1909
1909
1910
1910
1911
1911
1912
1912
1913
1913
1914
1914
1915
1915
1916
1916
1917
1917
1918
1918
1919
1919
1920
1920
1921
1921
1922
1922
1923
1923
1924
1924
1925
1925
1926
1926
1927
1927
1928
1928
1929
1929
1930
1930
1931
1931
1932
1932
1933
1933
1934
1934
1935
1935
1936
1936
1937
1937
1938
1938
1939
1939
1940
1940
1941
1941
1942
1942
1943
1943
1944
1944
1945
1945
1946
1946
1947
1947
1948
1948
1949
1949
1950
1950
1951
1951
1952
1952
1953
1953
1954
1954
1955
1955
1956
1956
1957
1957
1958
1958
1959
1959
1960
1960
1961
1961
1962
1962
1963
1963
1964
1964
1965
1965
1966
1966
1967
1967
1968
1968
1969
1969
1970
1970
1971
1971
1972
1972
1973
1973
1974
1974
1975
1975
1976
1976
1977
1977
1978
1978
1979
1979
1980
1980
1981
1981
1982
1982
1983
1983
1984
1984
1985
1985
1986
1986
1987
1987
1988
1988
1989
1989
1990
1990
1991
1991
1992
1992
1993
1993
1994
1994
1995
1995
1996
1996
1997
1997
1998
1998
1999
1999
2000
2000
2001
2001
2002
2002
2003
2003
2004
2004
2005
2005
2006
2006
2007
2007
2008
2008
2009
2009
2010
2010
2011
2011
2012
2012
2013
2013
2014
2014
2015
2015
2016
2016
2017
2017
2018
2018
2019
2019
2020
2020
2021
2021
2022
2022
2023
2023
2024
2024
2025
2025
2026
2026
2027
2027
2028
2028
2029
2029
2030
2030
2031
2031
2032
2032
2033
2033
2034
2034
2035
2035
2036
2036
2037
2037
2038
2038
2039
2039
2040
2040
2041
2041
2042
2042
2043
2043
2044
2044
2045
2045
2046
2046
2047
2047
2048
2048
2049
2049
2050
2050
2051
2051
2052
2052
2053
2053
2054
2054
2055
2055
2056
2056
2057
2057
2058
2058
2059
2059
2060
2060
2061
2061
2062
2062
2063
2063
2064
2064
2065
2065
2066
2066
2067
2067
2068
2068
2069
2069
2070
2070
2071
2071
2072
2072
2073
2073
2074
2074
2075
2075
2076
2076
2077
2077
2078
2078
2079
2079
2080
2080
2081
2081
2082
2082
2083
2083
2084
2084
2085
2085
2086
2086
2087
2087
2088
2088
2089
2089
2090
2090
2091
2091
2092
2092
2093
2093
2094
2094
2095
2095
2096
2096
2097
2097
2098
2098
2099
2099
20100
20100
20101
20101
20102
20102
20103
20103
20104
20104
20105
20105
20106
20106
20107
20107
20108
20108
20109
20109
20110
20110
20111
20111
20112
20112
20113
20113
20114
20114
20115
20115
20116
20116
20117
20117
20118
20118
20119
20119
20120
20120
20121
20121
20122
20122
20123
20123
20124
20124
20125
20125
20126
20126
20127
20127
20128
20128
20129
20129
20130
20130
20131
20131
20132
20132
20133
20133
20134
20134
20135
20135
20136
20136
20137
20137
20138
20138
20139
20139
20140
20140
20141
20141
20142
20142
20143
20143
20144
20144
20145
20145
20146
20146
20147
20147
20148
20148
20149
20149
20150
20150
20151
20151
20152
20152
20153
20153
20154
20154
20155
20155
20156
20156
20157
20157
20158
20158
20159
20159
20160
20160
20161
20161
20162
20162
20163
20163
20164
20164
20165
20165
20166
20166
20167
20167
20168
20168
20169
20169
20170
20170
20171
20171
20172
20172
20173
20173
20174
20174
20175
20175
20176
20176
20177
20177
20178
20178
20179
20179
20180
20180
20181
20181
20182
20182
20183
20183
20184
20184
20185
20185
20186
20186
20187
20187
20188
20188
20189
20189
20190
20190
20191
20191
20192
20192
20193
20193
20194
20194
20195
20195
20196
20196
20197
20197
20198
20198
20199
20199
20200
20200
20201
20201
20202
20202
20203
20203
20204
20204
20205
20205
20206
20206
20207
20207
20208
20208
20209
20209
20210
20210
20211
20211
20212
20212
20213
20213
20214
20214
20215
20215
20216
20216
20217
20217
20218
20218
20219
20219
20220
20220
20221
20221
20222
20222
20223
20223
20224
20224
20225
20225
20226
20226
20227
20227
20228
20228
20229
20229
20230
20230
20231
20231
20232
20232
20233
20233
20234
20234
20235
20235
20236
20236
20237
20237
20238
20238
20239
20239
20240
20240
20241
20241
20242
20242
20243
20243
20244
20244
20245
20245
20246
20246
20247
20247
20248
20248
20249
20249
20250
20250
20251
20251
20252
20252
20253
20253
20254
20254
20255
20255
20256
20256
20257
20257
20258
20258
20259
20259
20260
20260
20261
20261
20262
20262
20263
20263
20264
20264
20265
20265
20266
20266
20267
20267
20268
20268
20269
20269
20270
20270
20271
20271
20272
20272
20273
20273
20274
20274
20275
20275
20276
20276
20277
20277
20278
20278
20279
20279
20280
20280
20281
20281
20282
20282
20283
20283
20284
20284
20285
20285
20286
20286
20287
20287
20288
20288
20289
20289
20290
20290
20291
20291
20292
20292
20293
20293
20294
20294
20295
20295
20296
20296
20297
20297
20298
20298
20299
20299
20300
20300
20301
20301
20302
20302
20303
20303
20304
20304
20305
20305
20306
20306
20307
20307
20308
20308
20309
20309
20310
20310
20311
20311
20312
20312
20313
20313
20314
20314
20315
20315
20316
20316
20317
20317
20318
20318
20319
20319
20320
20320
20321
20321
20322
20322
20323
20323
20324
20324
20325
20325
20326
20326
20327
20327
20328
20328
20329
20329
20330
20330
20331
20331
20332
20332
20333
20333
20334
20334
20335
20335
20336
20336
20337
20337
20338
20338
20339
20339
20340
20340
20341
20341
20342
20342
20343
20343
20344
20344
20345
20345
20346
20346
20347
20347
20348
20348
20349
20349
20350
20350
20351
20351
20352
20352
20353
20353
20354
20354
20355
20355
20356
20356
20357
20357
20358
20358
20359
20359
20360
20360
20361
20361
20362
20362
20363
20363
20364
20364
20365
20365
20366
20366
20367
20367
20368
20368
20369
20369
20370
20370
20371
20371
20372
20372
20373
20373
20374
20374
20375
20375
20376
20376
20377
20377
20378
20378
20379
20379
20380
20380
20381
20381
20382
20382
20383
20383
20384
20384
20385
20385
20386
20386
20387
20387
20388
20388
20389
20389
20390
20390
20391
20391
20392
20392
20393
20393
20394
20394
20395
20395
20396
20396
20397
20397
20398
20398
20399
20399
20400
20400
20401
20401
20402
20402
20403
20403
20404
20404
20405
20405
20406
20406
20407
20407
20408
20408
20409
20409
20410
20410
20411
20411
20412
20412
20413
20413
20414
20414
20415
20415
20416
20416
20417
20417
20418
20418
20419
20419
20420
20420
20421
20421
20422
20422
20423
20423
20424
20424
20425
20425
20426
20426
20427
20427
20428
20428
20429
20429
20430
20430
20431
20431
20432
20432
20433
20433
20434
20434
20435
20435
20436
20436
20437
20437
20438
20438
20439
20439
20440
20440
20441
20441
20442
20442
20443
20443
20444
20444
20445
20445
20446
20446
20447
20447
20448
20448
20449
20449
20450
20450
20451
20451
20452
20452
20453
20453
20454
20454
20455
20455
20456
20456
20457
20457
20458
20458
20459
20459
20460
20460
20461
20461
20462
20462
20463
20463
20464
20464
20465
20465
20466
20466
20467
20467
20468
20468
20469
20469
20470
20470
20471
20471
20472
20472
20473
20473
20474
20474
20475
20475
20476
20476
20477
2
```

Roadmap: 5 Fix a bug

Overview

cmb69 left a comment

Thank you for the PR! I'll try to do a full review tomorrow, but left some comments for things that caught my attention while skimming over the patch.

1 like

ext/pcntl/pcntl.c Outdated

```
...     ... @@ -1274,8 +1275,22 @@ PHP_FUNCTION(pcntl_getpriority)
1274 1275     php_error_docref(NULL, E_WARNING, "Error %d: No process was located using
1275 1276     break;
1276 1277     case EINVAL:
1278 +     #ifdef PRIO_DARWIN_BG
```

cmb69 on Jul 19, 2022

Preprocessor directives should always start at the beginning of a line (there may be whitespace between the `#` and the symbol, but in this case this would not be helpful:

Suggested change

```
- #ifdef PRIO_DARWIN_BG
+ #ifdef PRIO_DARWIN_BG
```

Commit suggestion

Same for the other new directives in this file.

Fix pcntl get/set priority #9044

Closed juan-morales wants to merge 1 commit into [php:master](#) from [juan-morales:fix-pcntl_get/set-priority](#)

Conversation 12 12 Commits 1 Checks 0 Files changed 6

juan-morales commented on Jul 19, 2022 · edited

Code

Extension: pcntl
Functions: pcntl_setpriority(), pcntl_getpriority()

The problem

The PHP code to handle the errors for the mentioned functions is Linux aware only. Mac OS has a different behavior and uses the same error code for different things; plus FreeBSD does not respond in the same way as Linux or Mac under certain circumstances.

Because this feature (managing processes) is very OS dependent we cannot fully test the whole code, and also we cannot manage it in a uniform way (one code for all of them).

Documentation per OS related to possible error codes that can be return for the getpriority() and setpriority() functions

Errors per OS:

Mac

ERRORS

Getpriority() and setpriority() will fail if:

[EINVAL]
[EINVAL]
[EINVAL]
[ESRCH]

which is not one of PRIO_PROCESS, PRIO_PGRP, PRIO_USER, or PRIO_DARWIN_THREAD.

which is not a valid process, process group, or user ID.

who is not 0 when which is PRIO_DARWIN_THREAD.

No process can be located using the which and who values specified.

Reviewers

cmb69
bukka

Assignees

No one assigned

Labels

Extension: pcntl

Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request.

None yet

Notifications

Unsub

Roadmap: 5 Fix a bug

Where to look for bugs?



PHP Bug Tracking System

Before you report a bug, please make sure you have completed the following steps:

- Used the form above or our [advanced search page](#) to make sure nobody has reported the bug already.
- Make sure you are using the latest stable version or a build from Git, if similar bugs have recently been fixed and committed.
- Read our tips on [how to report a bug that someone will want to help fix](#).
- Read the [security guidelines](#), if you think an issue might be security related.
- See how to get a backtrace in case of a crash: [for *NIX](#) and [for Windows](#).
- Make sure it isn't a support question. For support, see the [support page](#).

Once you've double-checked that the bug you've found hasn't already been reported, and that you have collected all the information you need, you're ready to report it.

Search the Bug System

You can search all of the bugs that have been reported on our [advanced search page](#), or use the form at the top of the page for a basic search.

If you have 10 minutes to kill and you want to help us out, grab a random open bug and see if you can help resolve it. We have made it easy to do so.

Common searches

- [Most recent open bugs \(all\)](#)
- [Most recent open bugs \(all\) with patch or pull request](#)
- [Most recent open bugs \(PHP 7.2\)](#)
- [Most recent open bugs \(PHP 7.3\)](#)
- [Most recent open bugs \(PHP 7.4\)](#)
- [Most recent open bugs \(PHP 8.0\)](#)
- [Open Documentation bugs](#)
- [Open Documentation bugs \(with patches\)](#)

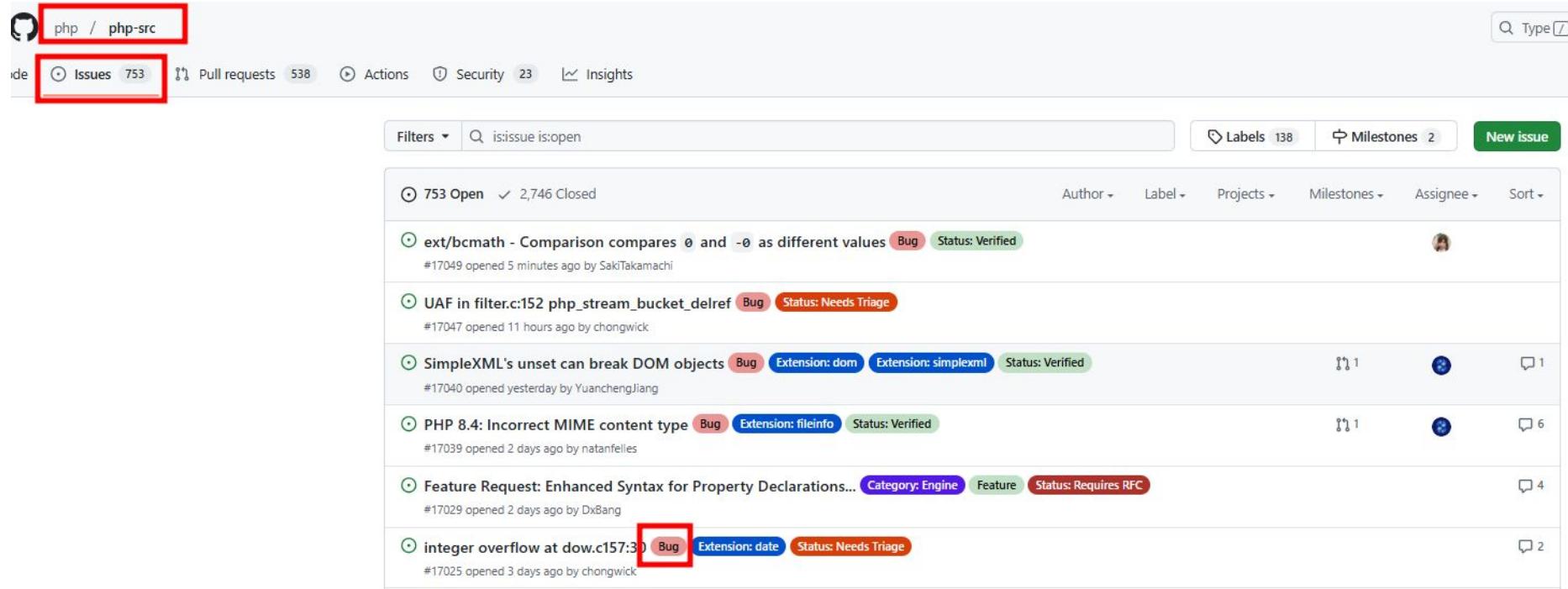
Bug System Statistics

You can view a variety of statistics about the bugs that have been reported on our [bug statistics page](#).



Roadmap: 5 Fix a bug

Where to look for bugs?



php / php-src

Issues 753

Filters is:issue is:open

753 Open ✓ 2,746 Closed

ext/bcmath - Comparison compares 0 and -0 as different values Bug Status: Verified

UAF in filter.c:152 php_stream_bucket_delref Bug Status: Needs Triage

SimpleXML's unset can break DOM objects Bug Extension: dom Extension: simplexml Status: Verified

PHP 8.4: Incorrect MIME content type Bug Extension: fileinfo Status: Verified

Feature Request: Enhanced Syntax for Property Declarations... Category: Engine Feature Status: Requires RFC

integer overflow at dow.c:157:30 Bug Extension: date Status: Needs Triage

Roadmap to start contributing to the code of PHP

- 1 Join the community
- 2 Read contributor information
- 3 Setup local environment
- 4 Learn PHP Tests (PHPT)
- 5 Fix a bug
- 6 Deep Dive into PHP Internals

Always

Read the existing documentation and learn from it, while walking the roadmap

6.1

Make an RFC

6.2

Write an Extension

Roadmap: 6 PHP Internals - RFCs

RFCs - the process, the key

- RFC: Request For Change
- An RFC is ideally accompanied by pull requests, showing the change in discussion
- There is a formal and structured procedure to create RFCs, as stated in “Contributing section of PHP’s repository” ...

Feature requests

Feature requests are generally submitted in the form of [Requests for Comments \(RFC\)](#), ideally accompanied by [pull requests](#). You can find the extremely large list of RFCs that have been previously considered on the [PHP Wiki](#).

To create an RFC, discuss it with the extension maintainer, and discuss it on the development mailing list internals@lists.php.net. RFC Wiki accounts can be requested on <https://wiki.php.net/start?do=register>. PHP extension maintainers can be found in the [EXTENSIONS](#) file in the PHP source code repository. Mailing list subscription is explained on the [mailing lists page](#).

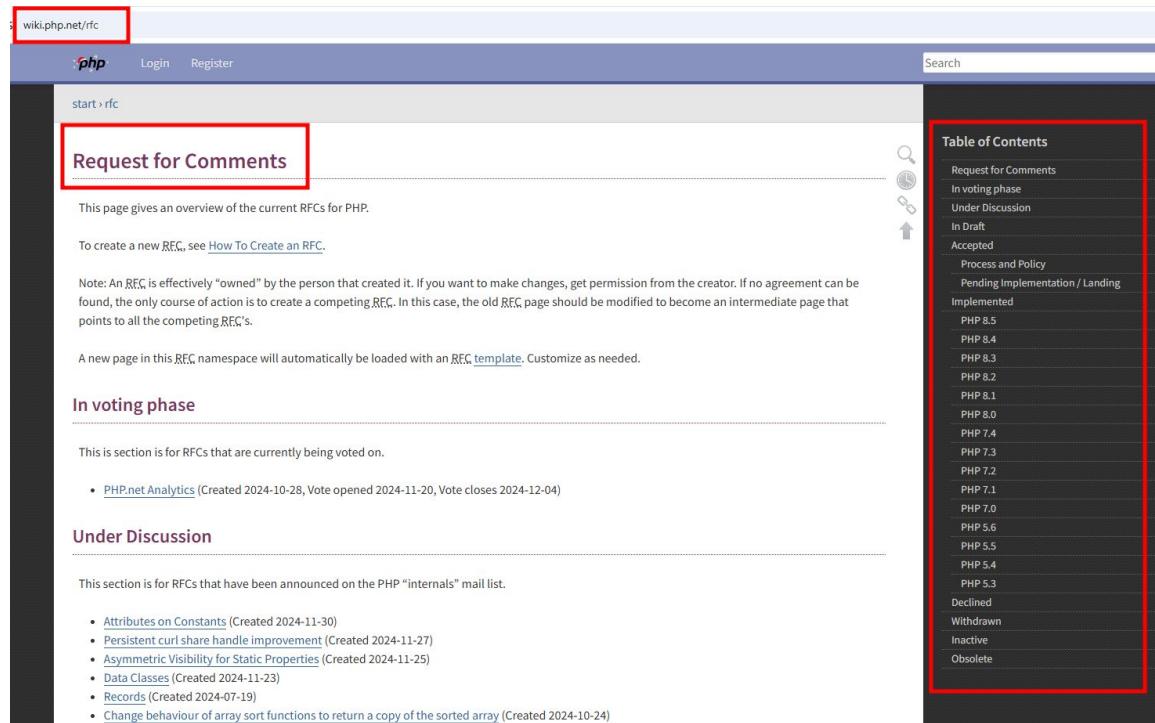
You may also want to read [The Mysterious PHP RFC Process](#) for additional notes on the best way to approach submitting an RFC.

Roadmap: 6 PHP Internals - RFCs

RFCs - the process, the key

Check current RFCs and their status at:

<https://wiki.php.net/rfc>



The screenshot shows the 'Request for Comments' page on the PHP wiki. The URL 'wiki.php.net/rfc' is highlighted in a red box at the top. The page content is as follows:

- Request for Comments** (highlighted in a red box): This page gives an overview of the current RFCs for PHP. To create a new RFC, see [How To Create an RFC](#). Note: An RFC is effectively "owned" by the person that created it. If you want to make changes, get permission from the creator. If no agreement can be found, the only course of action is to create a competing RFC. In this case, the old RFC page should be modified to become an intermediate page that points to all the competing RFC's. A new page in this RFC namespace will automatically be loaded with an RFC template. Customize as needed.
- In voting phase**: This section is for RFCs that are currently being voted on.
 - [PHP.net Analytics](#) (Created 2024-10-28, Vote opened 2024-11-20, Vote closes 2024-12-04)
- Under Discussion**: This section is for RFCs that have been announced on the PHP "internals" mail list.
 - [Attributes on Constants](#) (Created 2024-11-30)
 - [Persistent curl share handle improvement](#) (Created 2024-11-27)
 - [Asymmetric Visibility for Static Properties](#) (Created 2024-11-25)
 - [Data Classes](#) (Created 2024-11-23)
 - [Records](#) (Created 2024-07-19)
 - [Change behaviour of array sort functions to return a copy of the sorted array](#) (Created 2024-10-24)

On the right side, there is a 'Table of Contents' sidebar with a red border, listing various RFC status categories and PHP versions:

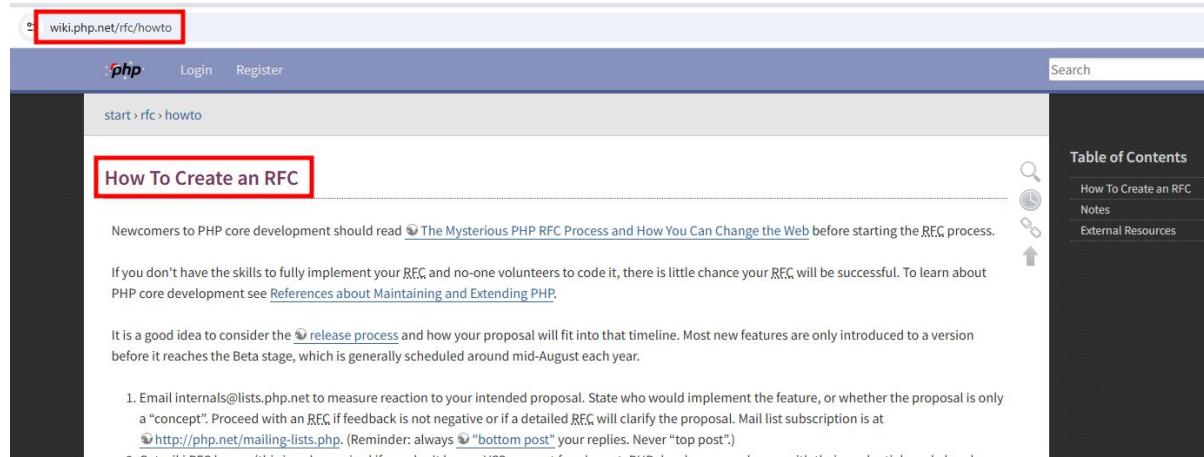
- Request for Comments
- In voting phase
- Under Discussion
- In Draft
- Accepted
- Process and Policy
- Pending Implementation / Landing
- Implemented
- PHP 8.5
- PHP 8.4
- PHP 8.3
- PHP 8.2
- PHP 8.1
- PHP 8.0
- PHP 7.4
- PHP 7.3
- PHP 7.2
- PHP 7.1
- PHP 7.0
- PHP 5.6
- PHP 5.5
- PHP 5.4
- PHP 5.3
- Declined
- Withdrawn
- Inactive
- Obsolete

Roadmap: 6 PHP Internals - RFCs

RFCS - the process, the key

Read the full procedure
to create an RFC at:

<https://wiki.php.net/rfc/howto>



The screenshot shows a web browser displaying the 'How To Create an RFC' page from the PHP documentation. The URL in the address bar is 'wiki.php.net/rfc/howto'. The page has a blue header with the PHP logo, 'Login', and 'Register' buttons. The main content area has a dark background with white text. The title 'How To Create an RFC' is in a red box. The page content includes a note for newcomers, a warning about skills and volunteers, a timeline note, and a numbered list of steps. A sidebar on the right contains a 'Table of Contents' with links to 'How To Create an RFC', 'Notes', and 'External Resources'. There are also icons for search, print, and link.

wiki.php.net/rfc/howto

start > rfc > howto

How To Create an RFC

Newcomers to PHP core development should read [The Mysterious PHP RFC Process and How You Can Change the Web](#) before starting the **REC** process.

If you don't have the skills to fully implement your **REC**, and no-one volunteers to code it, there is little chance your **REC** will be successful. To learn about PHP core development see [References about Maintaining and Extending PHP](#).

It is a good idea to consider the [release process](#) and how your proposal will fit into that timeline. Most new features are only introduced to a version before it reaches the Beta stage, which is generally scheduled around mid-August each year.

1. Email internals@lists.php.net to measure reaction to your intended proposal. State who would implement the feature, or whether the proposal is only a "concept". Proceed with an **REC**, if feedback is not negative or if a detailed **REC** will clarify the proposal. Mail list subscription is at <http://php.net/mailing-lists.php>. (Reminder: always ["bottom post"](#) your replies. Never "top post".)

Roadmap: 6 PHP Internals - RFCs

RFCs - Simple Example



wiki.php.net/rfc

- [Support Closures in constant expressions](#) (Created 2024-10-20, Vote ended 2024-11-01)
- [PHP 8.4](#)
- [Add bcdivmod to BCMath](#) (Created 2024-06-30)
- [Fix up BCMath Number Class / Change GMP bool cast behavior](#) (Created 2024-06-30)
- [Support object type in BCMath](#) (Created and discussion started: 2024-03-24, Voting ended: 2024-04-15)
- [Asymmetric visibility v2](#) (Created 2024-05-09, Voting started: 2024-07-26, Voting ended: 2024-08-13)
- [Property hook improvements](#) (Created 2024-06-29)
- [Correctly name the rounding mode and make it an Enum](#) (Created 2024-04-21)
- [Property access hooks](#) (Created 2022-12-01, Voting started 2024-04-15, Voting ended: 2024-05-24)
- [#\[Deprecated\] Attribute](#)
- This RFC proposes an attribute to mark declarations as deprecated (Created 2020-03-13)
- [Deprecate implicitly nullable parameter types](#) (Created 2023-12-20, Discussion started: 2023-12-29)
- [Lazy objects](#) (Created 2024-06-03, Voting started: 2024-07-26, Voting ended: 2024-08-13)
- [Adding bccround, bccfloor and bccceil to BCMath](#) (Created 2023-10-01)
- [Transform exit\(\) from a language construct into a standard function](#) (Discussion started: 2024-08-13)
- [Grapheme cluster for str_split function: grapheme_str_split](#) (Created 2024-03-04, Discussion started: 2024-02-06, Voting closed: 2024-03-07)
- [Add http_\(get|clear\)_last_response_headers\(\) function](#) (Created: 2024-01-03, Discussion closed: 2024-02-14)
- [RFC1867 for non-POST HTTP verbs](#) (Created 2023-10-06)
- [Change how JIT is disabled by default](#) (Created 2023-11-15)
- [DOM HTML5 parsing and serialization](#) (Created 2023-08-12, Discussion started: 2023-08-13)
- [A new JIT implementation based on IR Framework](#) (Created 2023-09-20, Discussion started: 2023-09-21)
- [XML_OPTION_PARSE_HUGE](#) (Created 2023-09-21, Discussion started: 2023-09-21, Voting ended: 2023-10-19)
- [Increasing the default BCrypt cost](#) (Created 2023-09-07)
- [Multibyte for trim function mb_trim, mb_ltrim and mb_rtrim](#) (Created 2023-10-19)
- [New 4 rounding modes to round\(\) function](#) (Created: 2023-09-31)

wiki.php.net/rfc/bcrypt_cost_2023

start > rfc > bcrypt_cost_2023

PHP RFC: Increasing the default BCrypt cost

- Version: 1.0
- Date: 2023-09-07
- Author: Tim Düsterhus, timwolla@php.net
- Status: Implemented
- Target Version: PHP 8.4
- Implementation: <https://github.com/php/php-src/commit/42a85fc5d95c0c5e867cafcca8ebfca9d90e6c88>
- First Published at: https://wiki.php.net/rfc/bcrypt_cost_2023

Introduction

PHP's default BCrypt cost for `password_hash` is unchanged since the addition of the `password hashing API` in PHP 5.5 which was 11 years ago.

BCrypt's variable cost factor is intended to provide adaptable security against increases in processing power and thus increased cracking speed. It appears appropriate to reconsider the default value after 11 years have passed.

Roadmap: 6 PHP Internals - RFCs

RFCs - Simple Example

Filter files...  

5 files changed +13 -4 lines changed  

NEWS   

UPGRADING   

ext/standard   

php_password.h   

tests/password   

password_hash.php   

password_removed_salt_option.php   

ext/standard/php_password.h   

... @@ -22,7 +22,7 @@ PHP_MINIT_FUNCTION(password);
22 PHP_MSHUTDOWN_FUNCTION(password);
23
24 #define PHP_PASSWORD_DEFAULT PHP_PASSWORD_BCRYPT
25 - #define PHP_PASSWORD_BCRYPT_COST 10
26
27 #ifdef HAVE_ARGON2LIB
28 /**
...   
22 PHP_MSHUTDOWN_FUNCTION(password);
23
24 #define PHP_PASSWORD_DEFAULT PHP_PASSWORD_BCRYPT
25 + #define PHP_PASSWORD_BCRYPT_COST 12
26
27 #ifdef HAVE_ARGON2LIB
28 /**
...   

ext/standard/tests/password/password_hash.php   

ext/standard/tests/password/password_removed_salt_option.php   

Search within code    

+1   

+4   

+1 -1   

+5 -3   

+2   

Roadmap: 6 PHP Internals - Extensions

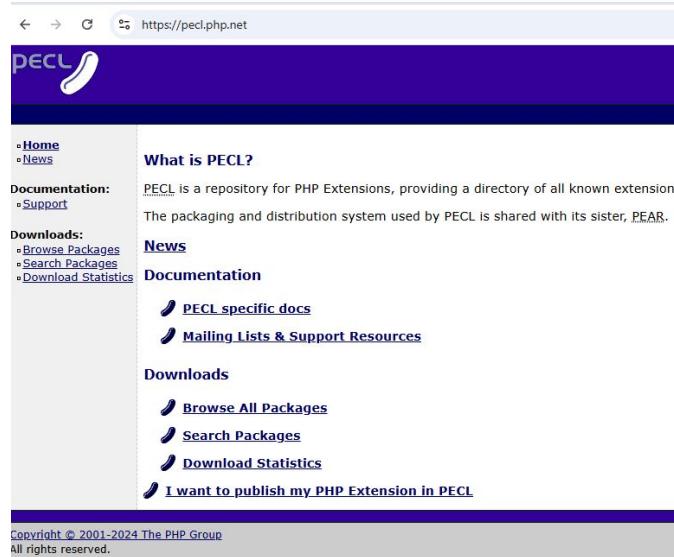
Extensions

- PHP has its core base extensions (extensions so fundamental that they are part of the core)
- Most of them are written in C, but it is also possible to use another programming languages such as C++, Rust, Go, Zig, etc.
- Reasons to write are many
- There are plenty of PHP's extension on the web, mainly listed on <https://pecl.php.net/>

Roadmap: 6 PHP Internals - Extensions

Extensions

- There are plenty of PHP's extension on the web, mainly listed on <https://pecl.php.net/>



Roadmap: 6 PHP Internals - Extensions

Other resources

- **Zephir Programming Language**
 - Zephir, an open source, high-level language designed to ease the creation and maintainability of extensions for PHP with a focus on **type** and **memory safety**.
 - <https://zephir-lang.com>



Roadmap: 6 PHP Internals - Extensions

Other resources - Zephir Programming Language

```
#ifdef HAVE_CONFIG_H
#include "config.h"
#endif

#include "php.h"
#include "php_test.h"
#include "test.h"

#include "kernel/main.h"

ZEPHIR_INIT_CLASS(Test_Hello) {
    ZEPHIR_REGISTER_CLASS(Test, Hello, hello, test_hello_method_entry, 0);
    return SUCCESS;
}

PHP_METHOD(Test_Hello, say) {
    php_printf("%s", "Hello World!");
}
```

Roadmap: 6 PHP Internals - Extensions

Other resources - Zephir Programming Language

```
namespace Test;

/**
 * MyTest (test/mytest.zep)
 */
class MyTest
{
    public function someMethod()
    {
        /* Variables must be declared */
        var myArray;
        int i = 0, length;

        /* Create an array */
        let myArray = ["hello", 0, 100.25, false, null];

        /* Count the array into a 'int' variable */
        let length = count(myArray);

        /* Print value types */
        while i < length {
            echo typeof myArray[i], "\n";
            let i++;
        }

        return myArray;
    }
}
```

```
#ifdef HAVE_CONFIG_H
#include "config.h"
#endif

#include "php.h"
#include "php_test.h"
#include "zend/zend_exceptions.h"
#include "zend/zend_interfaces.h"

ZEPHIR_INIT_CLASS(Test_MyTest)
{
    ZEND_REGISTER_CLASS(Test, MyTest, test, mytest, test_mytest_method_entry, 0);

    return SUCCESS;
}

PHP_METHOD(Test_MyTest, someMethod)
{
    zval myArray, tmp;
    zend_long i = 0, length;
    ZEND_PARSE_PARAMETERS_NONE();

    /* Initialize variables */
    ZVAL_UNDEF(&myArray);
    ZVAL_UNDEF(&tmp);

    /* Create the array */
    array_init_size(&myArray, 5);

    /* Add values to the array */
    add_next_index_string(&myArray, "hello");
    add_next_index_long(&myArray, 0);
    add_next_index_double(&myArray, 100.25);
    add_next_index_bool(&myArray, 0); // false
    add_next_index_null(&myArray);

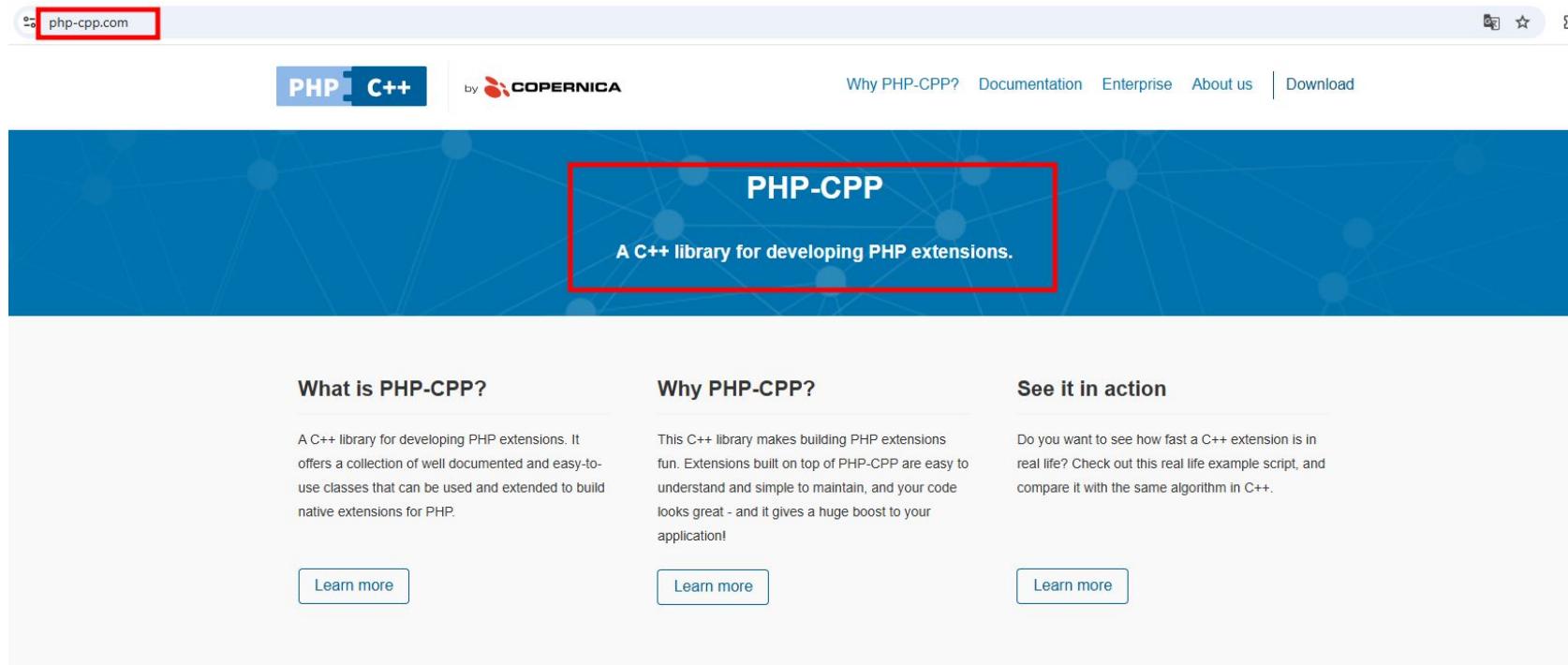
    /* Get the array count */
    length = zend_array_count(Z_ARRVAL(myArray));

    /* Loop through the array */
    while (i < length) {
        /* Get the value type */
        zval *value = zend_hash_index_find(Z_ARRVAL(myArray), i);
        if (value) {
            php_printf("%s\n", zend_get_type_by_const(Z_TYPE_P(value)));
        }
        i++;
    }

    /* Return the array */
    RETVAL_ZVAL(&myArray, 1, 0); // Return a copy
}
```

Roadmap: 6 PHP Internals - Extensions

Other resources - PHP-CPP - Extensions in C++ made easy



The screenshot shows the homepage of the PHP-CPP website. The URL 'php-cpp.com' is highlighted with a red box in the browser's address bar. The page features a blue header with the 'PHP C++' logo and 'by COPERNICA'. The main title 'PHP-CPP' is centered in a white box with a red border, with the subtitle 'A C++ library for developing PHP extensions.' below it. The background is a blue network-like graphic. Below the header, there are three sections: 'What is PHP-CPP?', 'Why PHP-CPP?', and 'See it in action', each with a 'Learn more' button.

php-cpp.com

PHP C++ by COPERNICA

Why PHP-CPP? Documentation Enterprise About us Download

PHP-CPP

A C++ library for developing PHP extensions.

What is PHP-CPP?

A C++ library for developing PHP extensions. It offers a collection of well documented and easy-to-use classes that can be used and extended to build native extensions for PHP.

[Learn more](#)

Why PHP-CPP?

This C++ library makes building PHP extensions fun. Extensions built on top of PHP-CPP are easy to understand and simple to maintain, and your code looks great - and it gives a huge boost to your application!

[Learn more](#)

See it in action

Do you want to see how fast a C++ extension is in real life? Check out this real life example script, and compare it with the same algorithm in C++.

[Learn more](#)

Roadmap: Internals Information

Resources:

Internals

- PHP Internals Book: <https://www.phpinternalsbook.com/>
- Book: Extending and Embedding PHP by Sara Golemon

Official

- Github discussions and pull requests
- Internals mailing list
- Check the already created RFCs and their implementation
- <https://wiki.php.net/internals/extensions>
- <https://qa.php.net/write-test.php>
- <https://www.php.net/get-involved.php>
- <https://wiki.php.net/rfc/howto>
- <https://qa.php.net/write-test.php>
- <http://doc.php.net>

Extra

- Use AI tools like ChatGPT
- Nikita Popov <https://www.npopov.com/>
- Extensions:
 - <https://github.com/phplang/extension-tutorial>
 - <https://speakerdeck.com/carusoqabriel>

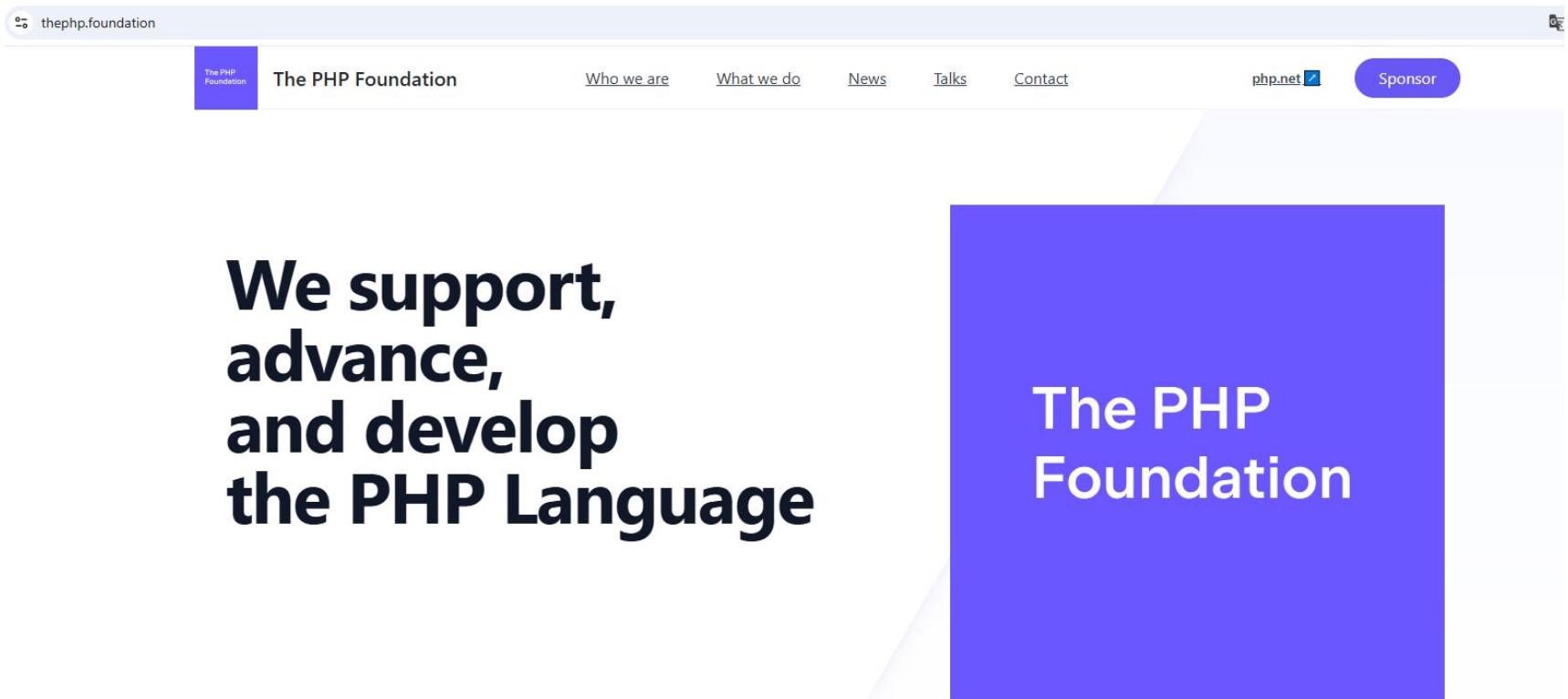
Roadmap: Internals Information

Resources:

Youtube:

- [Gina Banyard - How to contribute to PHP](#)
- [Gabriel Caruso - Contribute to PHP Core](#)
- [PHPKonf 2020 - Ekin Bayar: We want YOU to contribute to PHP](#)
- [Gabriel Caruso: Contributing to PHP's Core – phpCE 2018](#)
- [Sammy Kaye : Writing tests for PHP source](#)
- [PHP Round Table](#)

Last but not least! ... donate! :)



The screenshot shows the homepage of The PHP Foundation. At the top, there is a navigation bar with the following items: a logo for 'thephp.foundation', the text 'The PHP Foundation', and links for 'Who we are', 'What we do', 'News', 'Talks', and 'Contact'. To the right of these are links for 'php.net' and 'Sponsor'. The main content area features a large, bold, dark text block on the left that reads: 'We support, advance, and develop the PHP Language'. To the right of this text is a large, solid blue rectangular area with white text that reads: 'The PHP Foundation'.

thephp.foundation

The PHP Foundation

[Who we are](#) [What we do](#) [News](#) [Talks](#) [Contact](#)

[php.net](#) [Sponsor](#)

We support,
advance,
and develop
the PHP Language

The PHP Foundation