

How I started contributing to php-src

And how everyone can as well

Adam Sychala

/jorgsowa

Where the repository of PHP code is located?

github.com/php/php-src



git.php.net

php / php-src

Type / to search

<> Code

Issues 587

Pull requests 433

Actions

php php-src Public

Edit Pins

Watch 1397

Fork 7.7k

Starred 37.6k

master 485 Branches 1,344 Tags

Go to file

Add file

<> Code

devnexus

ext/pcntl adding Pcntl namespace for Qos class.

17cf7b3 · 46 minutes ago

136,816 Commits

.circleci	Merge branch 'PHP-8.3'	2 months ago
.github	Merge branch 'PHP-8.3'	last week
TSRM	Autotools: Simplify POSIX Threads check (#14855)	3 days ago
Zend	Replace @deprecated by #[\Deprecated] for internal funct...	4 hours ago
benchmark	Change default method of disabling JIT	6 months ago
build	Autotools: Refactor builtin checks (#14835)	2 days ago
docs-old	Move back mailinglist-rules.md and release-process.md ...	5 months ago
docs	[skip ci] Fix some typos (#14893)	19 hours ago
ext	ext/pcntl adding Pcntl namespace for Qos class.	46 minutes ago
main	Update http links to https and sync www.php.net URLs (#...	4 days ago

The PHP Interpreter

[www.php.net](#)

Readme

View license

Security policy

Activity

Custom properties

37.6k stars

1.4k watching

7.7k forks

1,344 tags

Contributors 961

<https://github.com/php/php-src>

Try Pitch

Opening PHP's [#internals](#) mailing list to the outside.

Search...



▲
0
▼

[low priority] WeakMaps with scalar keys

19 minutes ago by Rob Landers

▲
1
▼

Website php.net updates and coordination

1 hour ago by Roman Pronskiy

▲
-1
▼

[Discussion] Add date_test_set_now() function

9 days ago by Go Kudo



▲
0
▼

ext/gd adding imagecompare

1 day ago by David CARLIER



▲
4
▼

[RFC] Improve language coherence for the behaviour of offsets and containers

6 days ago by Gina P. Banvard



<https://externals.io/>

php / php-src

Type / to search

<> Code

Issues 587

Pull requests 433

Actions

php php-src Public

Edit Pins

Watch 1397

Fork 7.7k

Starred 37.6k

master 485 Branches 1,344 Tags

Go to file

Add file

<> Code

devnexen

ext/pcntl adding Pcntl namespace for Qos class.

17cf7b3 · 46 minutes ago

136,816 Commits

.circleci	Merge branch 'PHP-8.3'	2 months ago
.github	Merge branch 'PHP-8.3'	last week
TSRM	Autotools: Simplify POSIX Threads check (#14855)	3 days ago
Zend	Replace @deprecated by #[\Deprecated] for internal funct...	4 hours ago
benchmark	Change default method of disabling JIT	6 months ago
build	Autotools: Refactor builtin checks (#14835)	2 days ago
docs-old	Move back mailinglist-rules.md and release-process.md ...	5 months ago
docs	[skip ci] Fix some typos (#14893)	19 hours ago
ext	ext/pcntl adding Pcntl namespace for Qos class.	46 minutes ago
main	Update http links to https and sync www.php.net URLs (#...	4 days ago

The PHP Interpreter

[www.php.net](#)

Readme

View license

Security policy

Activity

Custom properties

37.6k stars

1.4k watching

7.7k forks

1,344 tags

Contributors 961

<https://github.com/php/php-src>

User Contributed Notes 32 notes

[+ add a note](#)

▲ 319 ▼ takingsides at gmail dot com

10 years ago

In my opinion this function lacks two flags:

- PHP_ROUND_UP - Always round up.
- PHP_ROUND_DOWN - Always round down.

In accounting, it's often necessary to always round up, or down to a precision of thousandths.

```
<?php
function round_up($number, $precision = 2)
{
    $fig = (int) str_pad('1', $precision, '0');
    return (ceil($number * $fig) / $fig);
}

function round_down($number, $precision = 2)
{
    $fig = (int) str_pad('1', $precision, '0');
    return (floor($number * $fig) / $fig);
}
?>
```

<https://www.php.net/manual/en/function.round.php>

Change language: English

round

(PHP 4, PHP 5, PHP 7, PHP 8)

round — Rounds a float

Description

```
round(int|float $num, int $precision = 0, int $mode = PHP_ROUND_HALF_UP): float
```

Returns the rounded value of **num** to specified **precision** (number of digits after the decimal point). **precision** can also be negative or zero (default).

Parameters

num

The value to round.

precision

The optional number of decimal digits to round to.

If the **precision** is positive, **num** is rounded to **precision** significant digits after the decimal point.

If the **precision** is negative, **num** is rounded to **precision** significant digits before the decimal point, i.e. to the nearest multiple of `pow(10, -$precision)`, e.g. for a **precision** of -1 **num** is rounded to tens, for a **precision** of -2 to hundreds, etc.

mode

Use one of the following constants to specify the mode in which rounding occurs.

Constants	Description
PHP_ROUND_HALF_UP	Rounds num away from zero when it is half way there, making 1.5 into 2 and -1.5 into -2.
PHP_ROUND_HALF_DOWN	Rounds num towards zero when it is half way there, making 1.5 into 1 and -1.5 into -1.
PHP_ROUND_HALF_EVEN	Rounds num towards the nearest even value when it is half way there, making both 1.5 and 2.5 into 2.
PHP_ROUND_HALF_ODD	Rounds num towards the nearest odd value when it is half way there, making 1.5 into 1 and 2.5 into 3.

Return Values

The value rounded to the given **precision** as a [float](#).

<https://www.php.net/manual/en/function.round.php>

[How to contribute?](#)

Stages of adding new features to PHP

Stage 0: Ask for Karma

1 year ago by Jorg Sowa

unread

Hello,
I would like to request for RFC Karma.

I want to create RFC for deprecation of constants ISO_8601.

My wiki account is jorg_sowa.

Kind regards,
Jorg

Stage 1: Write RFC

start › rfc › new_rounding_modes_to_round_function

PHP RFC: Add 4 new rounding modes to round() function

- Date: 2023-08-31
- Target version: PHP 8.4
- Author: Jorg Sowa ✉ jorg.sowa@gmail.com
- Status: Implemented
- Implementation: 🌐 <https://github.com/php/php-src/pull/12056>
- First Published at: 🌐 http://wiki.php.net/rfc/new_rounding_modes_to_round_function

🔍

🕒

🔗

⬆

Introduction

The [RFC](#) proposes to add 4 new modes to the `round()` function.

- 🌐 [PHP_ROUND_CEILING](#),
- 🌐 [PHP_ROUND_FLOOR](#),
- 🌐 [PHP_ROUND_AWAY_FROM_ZERO](#),
- 🌐 [PHP_ROUND_TOWARD_ZERO](#).

Two first comments on the 🌐 [documentation page of the round\(\)](#) function are related to the missing modes of rounding. The first comment has an outstanding number of 309 votes.

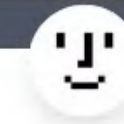
Moreover, the 🌐 [NumberFormatter](#) already implements 4 new proposed modes known as `ROUND_CEILING`, `ROUND_FLOOR`, `ROUND_DOWN`, `ROUND_UP`.

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

Stage 2: Open discussion

◀ [RFC] [Discussion] Add 4 new rounding modes to round() function

🔒 10 months ago by Jorg Sowa — [view source](#) unread



Hello everyone,

I would like to start the RFC process about adding 4 new rounding modes to the `round()` function.

As the topic was previously discussed and the change is quite small, I would like to start the voting on the beginning of next week.

RFC: https://wiki.php.net/rfc/new_rounding_modes_to_round_function

Thanks Niels for spotting incorrect links to the modes.

Kind regards,
Jorg

Stage 3: Discuss

1. Open your RFC for discussion.
2. Answer the questions.
3. Revisit your RFC upon appearing questions.
4. Modify your RFC.
5. Wait a few days for new questions.
6. If new questions appear, go to 2.
7. End the discussion stage.

Stage 4: Voting

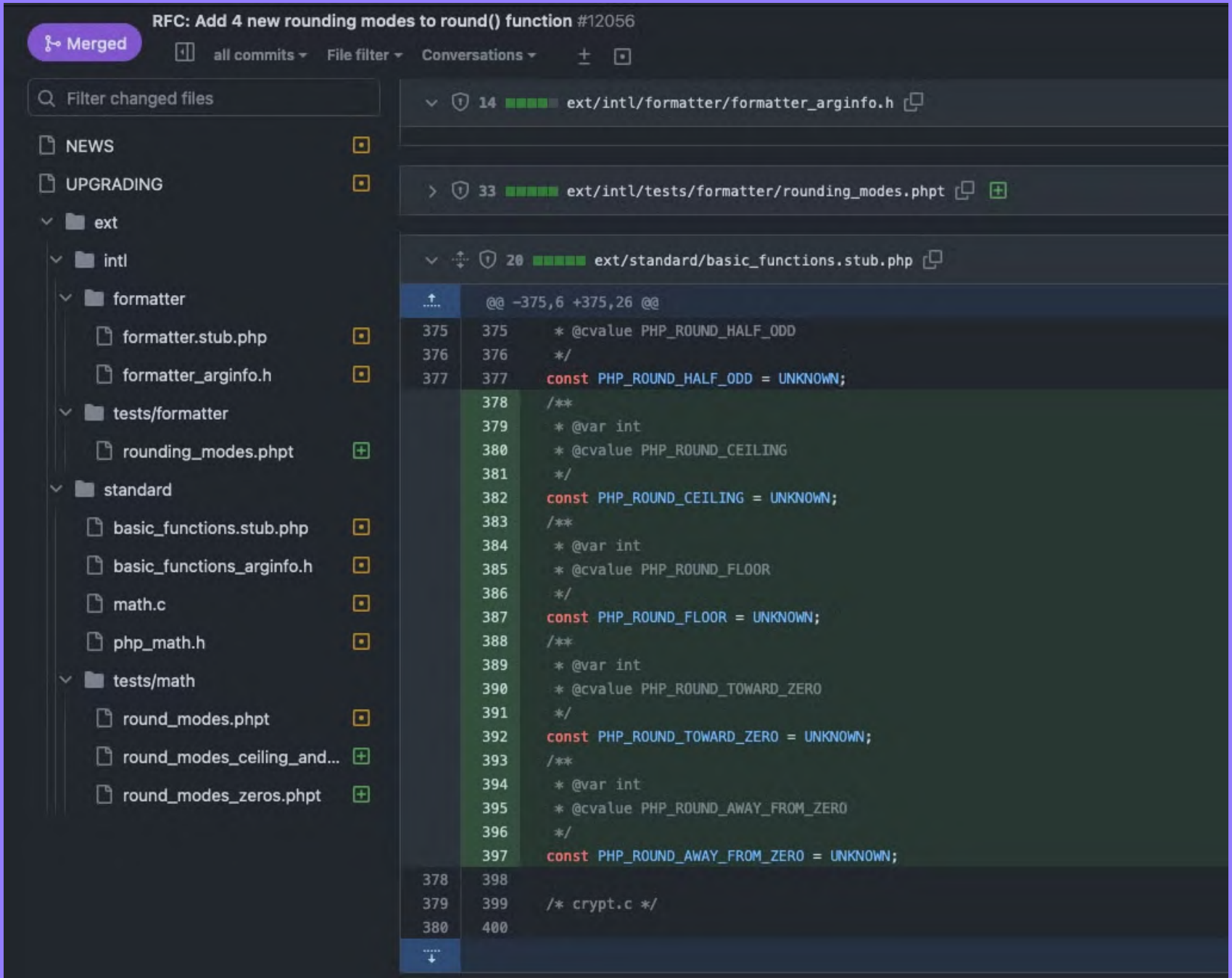
As per the voting RFC a yes/no vote with a 2/3 majority is needed for this proposal to be accepted.

Voting started on 2023-11-15 and will end on 2023-11-30 00:00 GMT.

Implement 4 new rounding modes to `round()` function			
Real name		Yes	No
ashnazg (ashnazg)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
bukka (bukka)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
crell (crell)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
derick (derick)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
galvao (galvao)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
girgias (girgias)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
joey (joey)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
kalle (kalle)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
levim (levim)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
mbeccati (mbeccati)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
nicolasgrekas (nicolasgrekas)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
nielsdos (nielsdos)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
ocramius (ocramius)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
petk (petk)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
ramsey (ramsey)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
santiagolizardo (santiagolizardo)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
sergey (sergey)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
theodorejb (theodorejb)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
timwolla (timwolla)		<input checked="" type="checkbox"/>	<input type="checkbox"/>
Final result:		19	0
This poll has been closed.			

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

Stage 5: Implementation ...or acceptance of the results



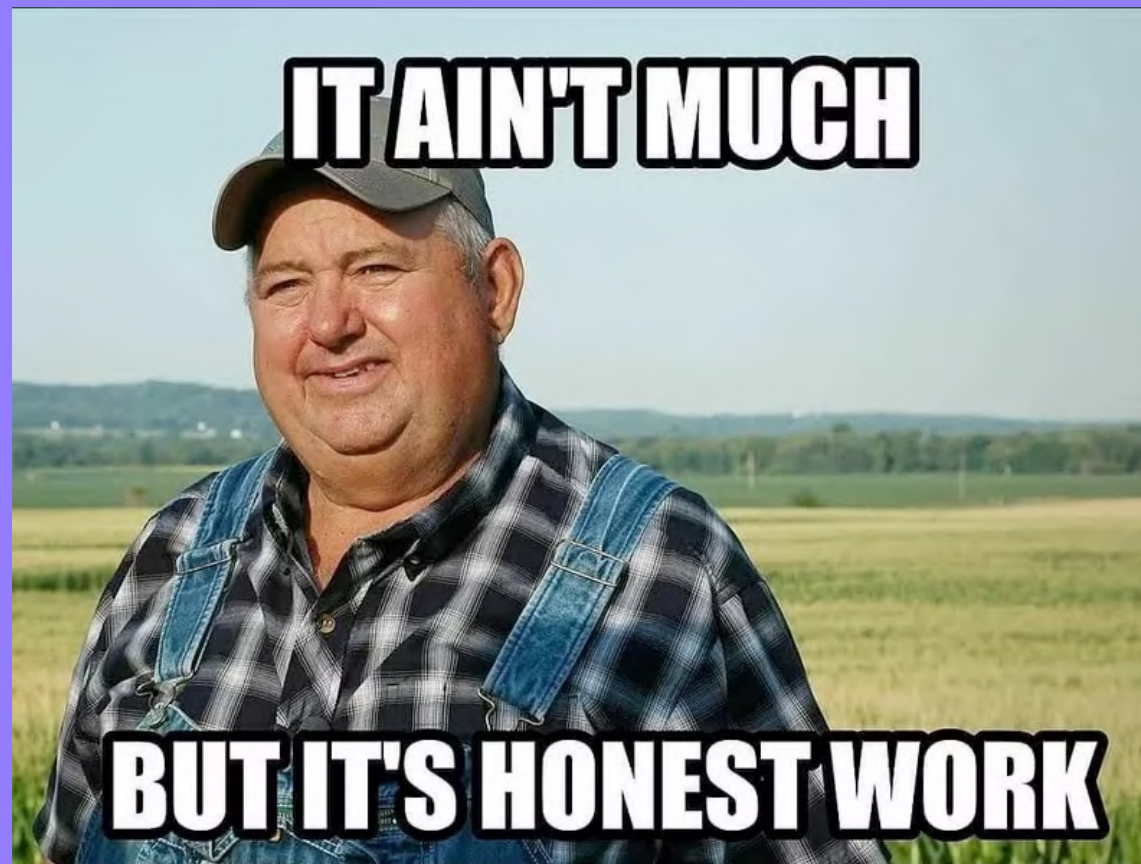
<https://github.com/php/php-src/pull/12056>

Implementation

```
315 341 /* {{{ Returns the number rounded to specified precision */
316 342 PHP_FUNCTION(round)
317 343 {
318 344     zval *value;
319 345     int places = 0;
320 346     zend_long precision = 0;
321 347     zend_long mode = PHP_ROUND_HALF_UP;
322 348
323 349     ZEND_PARSE_PARAMETERS_START(1, 3)
324 350         Z_PARAM_NUMBER(value)
325 351         Z_PARAM_OPTIONAL
326 352         Z_PARAM_LONG(precision)
327 353         Z_PARAM_LONG(mode)
328 354     ZEND_PARSE_PARAMETERS_END();
329 355
330 356     if (ZEND_NUM_ARGS() >= 2) {
331 357         if (precision >= 0) {
332 358             places = ZEND_LONG_INT_OVFL(precision) ? INT_MAX : (int)precision;
333 359         } else {
334 360             places = ZEND_LONG_INT_UDFL(precision) ? INT_MIN : (int)precision;
335 361         }
336 362     }
337 363
338 364     switch (mode) {
339 365         case PHP_ROUND_HALF_UP:
340 366         case PHP_ROUND_HALF_DOWN:
341 367         case PHP_ROUND_HALF_EVEN:
342 368         case PHP_ROUND_HALF_ODD:
343 369         case PHP_ROUND_AWAY_FROM_ZERO:
344 370         case PHP_ROUND_TOWARD_ZERO:
345 371         case PHP_ROUND_CEILING:
346 372         case PHP_ROUND_FLOOR:
347 373         break;
348 374     default:
349 375         zend_argument_value_error(3, "must be a valid rounding mode (PHP_ROUND_*)");
350 376         RETURN_THROWS();
351 377     }
352 378
353 379     switch (Z_TYPE_P(value)) {
354 380     case IS_LONG:
355 381         /* Simple case - long that doesn't need to be rounded. */
356 382         if (places >= 0) {
357 383             RETURN_DOUBLE(zval_get_double(value));
358 384         }
359 385         ZEND_FALLTHROUGH;
360 386
361 387     case IS_DOUBLE:
362 388         RETURN_DOUBLE(_php_math_round(zval_get_double(value), (int)places, (int)mode));
363 389
364 390     EMPTY_SWITCH_DEFAULT_CASE();
365 391     }
366 392 }
367 393 /* }}} */
```

Implementation

```
126 126
127 127     return integral;
128 128
129 129     case PHP_ROUND_CEILING:
130 130         if (value > 0.0 && fractional > 0.0) {
131 131             return integral + 1.0;
132 132         }
133 133
134 134         return integral;
135 135
136 136     case PHP_ROUND_FLOOR:
137 137         if (value < 0.0 && fractional > 0.0) {
138 138             return integral - 1.0;
139 139         }
140 140
141 141         return integral;
142 142
143 143     case PHP_ROUND_TOWARD_ZERO:
144 144         return integral;
145 145
146 146     case PHP_ROUND_AWAY_FROM_ZERO:
147 147         if (fractional > 0.0) {
148 148             return integral + copysign(1.0, integral);
149 149         }
150 150
151 151         return integral;
152 152
129 153     case PHP_ROUND_HALF_EVEN:
130 154         if (fractional > 0.5) {
```

Ways of contributing to PHP

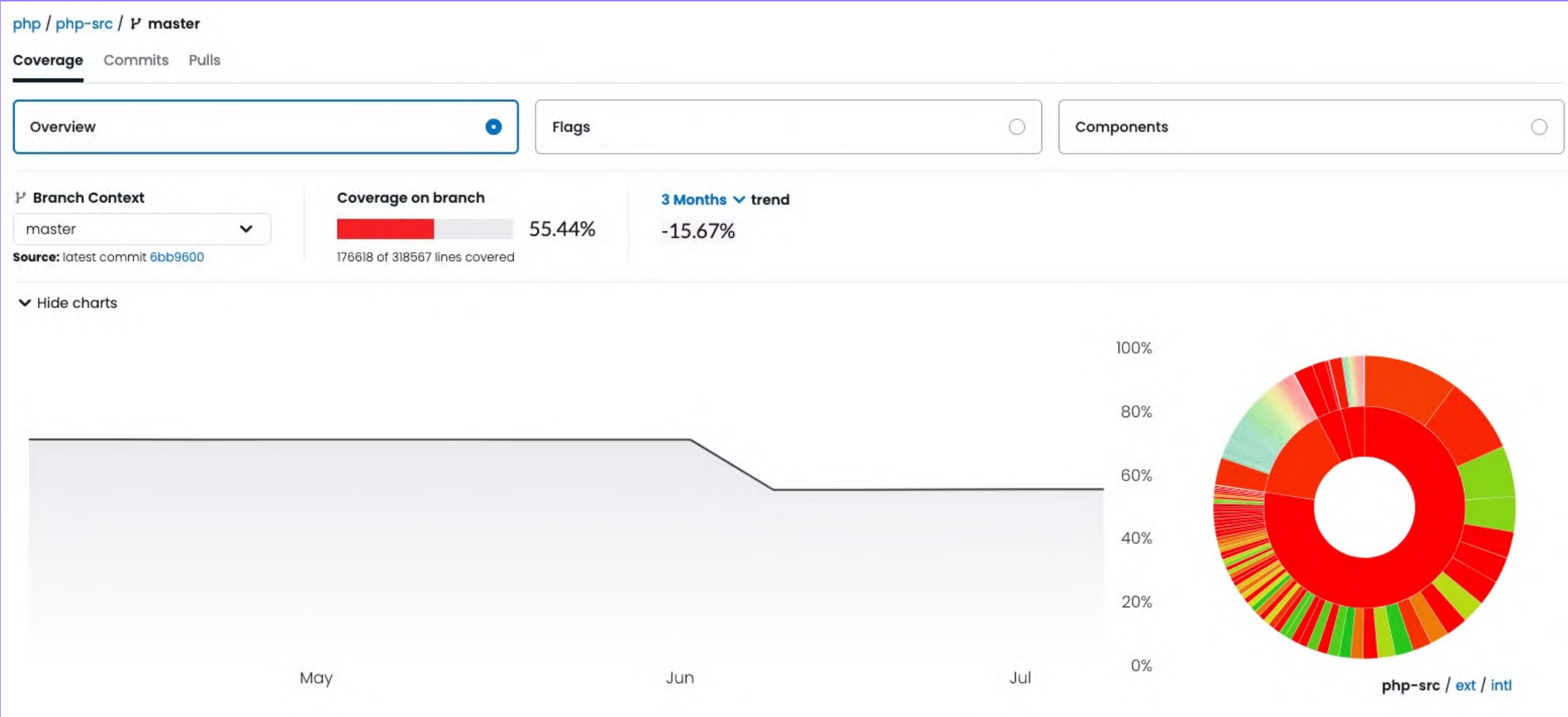
How to contribute to PHP language?

Four Best Ways to Contribute

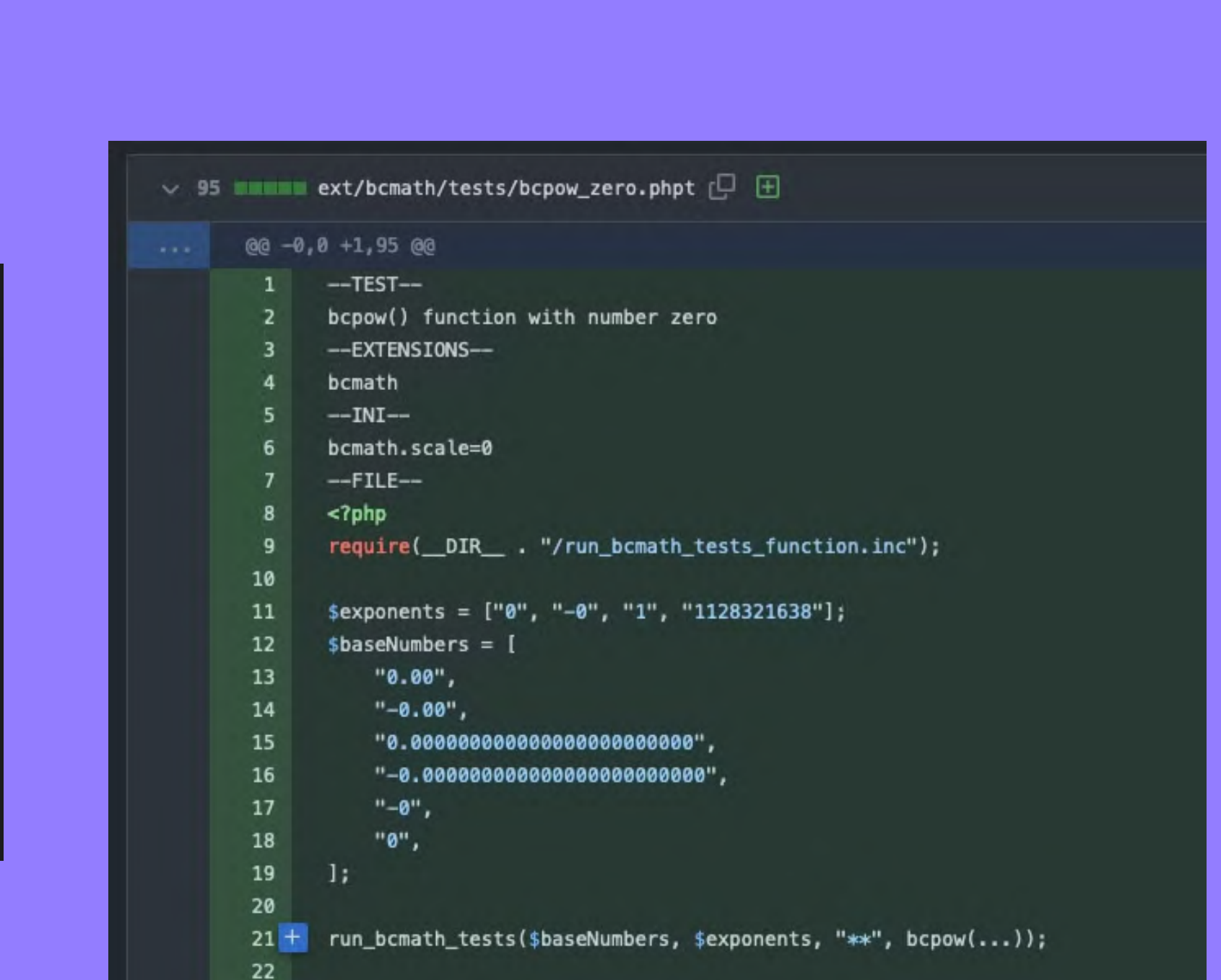
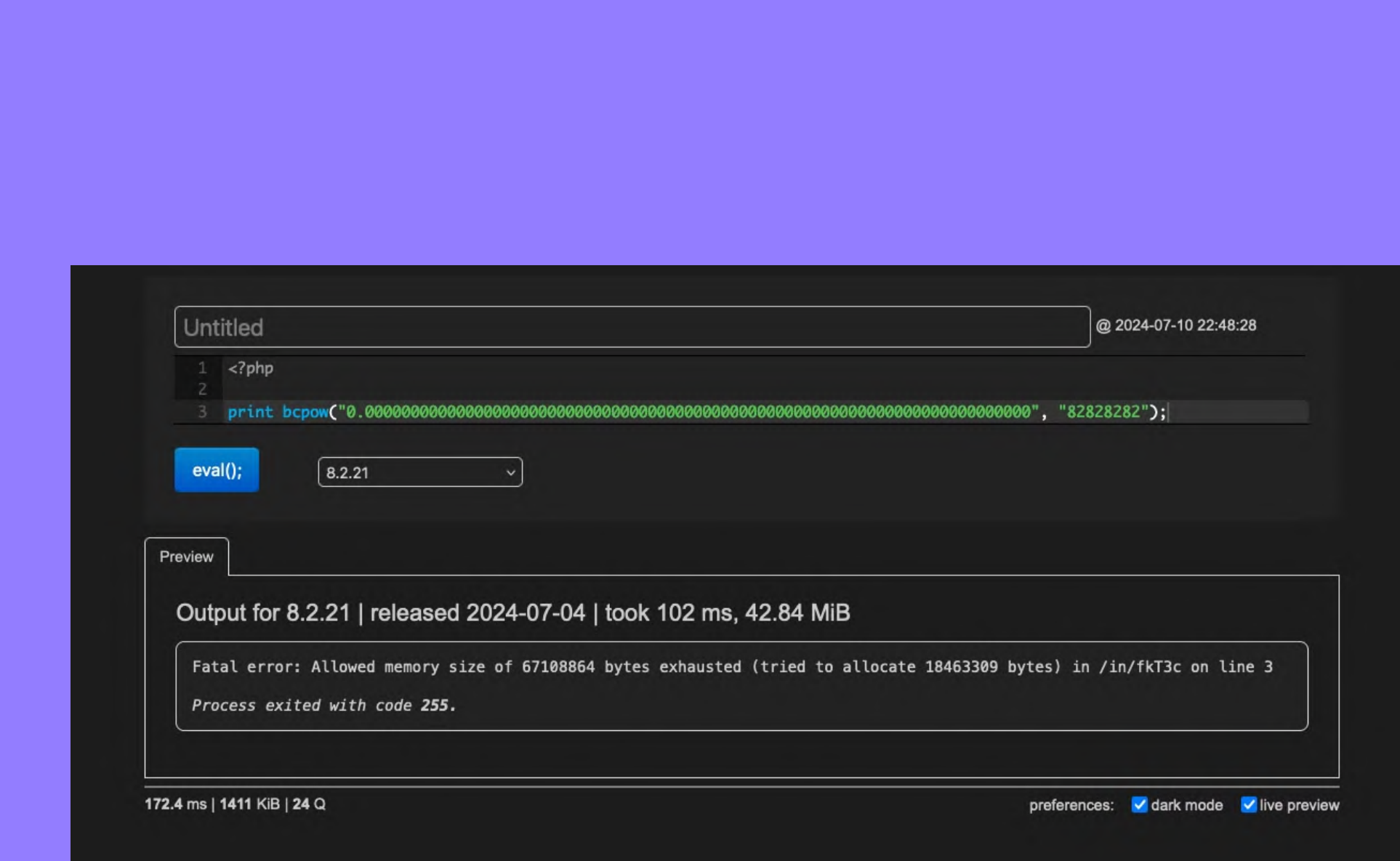
1. Running test suites in [RC](#) and [release](#) distributions of PHP
2. Help finding and diagnosing failed tests, see the [phpt documentation](#)
3. Filing and resolving bug reports on [GitHub Issues](#).
4. Help maintain and or translate documentation files at the doc-* repositories on [github](#).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

<https://www.php.net/get-involved>

Improve code coverage




<https://app.codecov.io/github/php/php-src/tree/master/ext>



[Try Pitch](#)

Improve PHP documentation

 **doc-en** Public

Edit Pins

Watch 39

Fork 709


Star 476

master 7 Branches 0 Tags

Go to file

Add file

Code

 **haszi and haszi** Remove empty configuration pages (#3538) 765b2d6 · yesterday 29,598 Commits

.docker	Create a docker container for a single-line setup and ren...	last year
.github/workflows	Ignore QA section order script for now	last month
appendices	Update incompatible.xml: Move number_format() notice ...	yesterday
chapters	Remove reference to generating "Flash" movies (#2733)	last year
chmonly	Fix other typos outside of php module docs	4 years ago
faq	Update mailinglist FAQ on unsubscribing	4 months ago
features	Remove PHP 5 era changelog and rework markup (#3331)	3 months ago
install	[skip-revcheck] Convert info constant tables to variableli...	2 months ago

English PHP documentation

Readme

Activity


Custom properties

476 stars

39 watching

709 forks

Contributors 519



XML 99.9%

Other 0.1%

<https://app.codecov.io/github/php/php-src/tree/master/ext>


Try Pitch

Improve PHP documentation

PHP 8.2 documentation tracker #1803

Open

12 comments



Girgias commented on Sep 7, 2022 • edited

Member

Pages which need to be added/modified for complete documentation of PHP 8.2:

Based of the migration guide: php.net/manual/en/migration82.php (#1799)

Note

This issue is still a Work In Progress

Core

☒ New features:

☒ Type system ([Rewrite type section](#) #1726)

☒ false null standalone

☒ true

☒ DNF types

☒ Sensitive Parameter attribute

☒ Readonly classes ([Document readonly classes](#) #1996)

☒ error_log_mode INI directive (Part of [Add initial version of the PHP 8.2 migration guide](#) #1799)

☒ Declaring constants in traits ([Update trait docs for PHP-8.2](#) #1995)

☐ Deprecated

☒ Dynamic properties ([Document dynamic properties](#) #1942)

☐ "{\$var}" and "{\$expr}" style string interpolations ([Rewrite and flesh out string interpolation docs](#) #2000)

Classes:

☒ New classes


<https://app.codecov.io/github/php/php-src/tree/master/ext>

Improve PHP documentation

PHP 8.3 documentation tracker #2796

Open

9 comments



Girgias commented on Sep 26, 2023 • edited by iluuu1994

Member

Pages which need to be added/modified for complete documentation of PHP 8.3:

Based of the migration guide: php.net/manual/en/migration83.php (#2780):

Note

This issue is still a Work In Progress

Core

☐ New features:

☐ Readonly Amendments

☐ Anonymous classes can be marked as readonly

☐ Readonly Properties can be reinitialized during cloning

☒ Override attribute ([🔗 Add Override Attribute stub #2996](#))

☒ Typed Class constants (Part [🔗 Update type system documentation #2798](#))

☐ Closures created from magic functions accept named arg

☐ final methods on trait

☐ Fetch class constant syntax

☐ Static Variable Initializer

☐ Fallback value syntax for ini variables

☒ Deprecated

☒ Saner Increment/Decrement operators ([🔗 PHP 8.3: Describe increment/decrement operator changes #2841](#))

☒ Using the [increment](#) operator (++) on empty, non-numeric, or non-alphanumeric strings is now deprecated. Moreover, incrementing non-numeric strings is soft deprecated and the new str_increment() function should be used instead.

<https://app.codecov.io/github/php/php-src/tree/master/ext>

PHP Internals Book

TABLE OF CONTENTS

Contents :: Introduction »

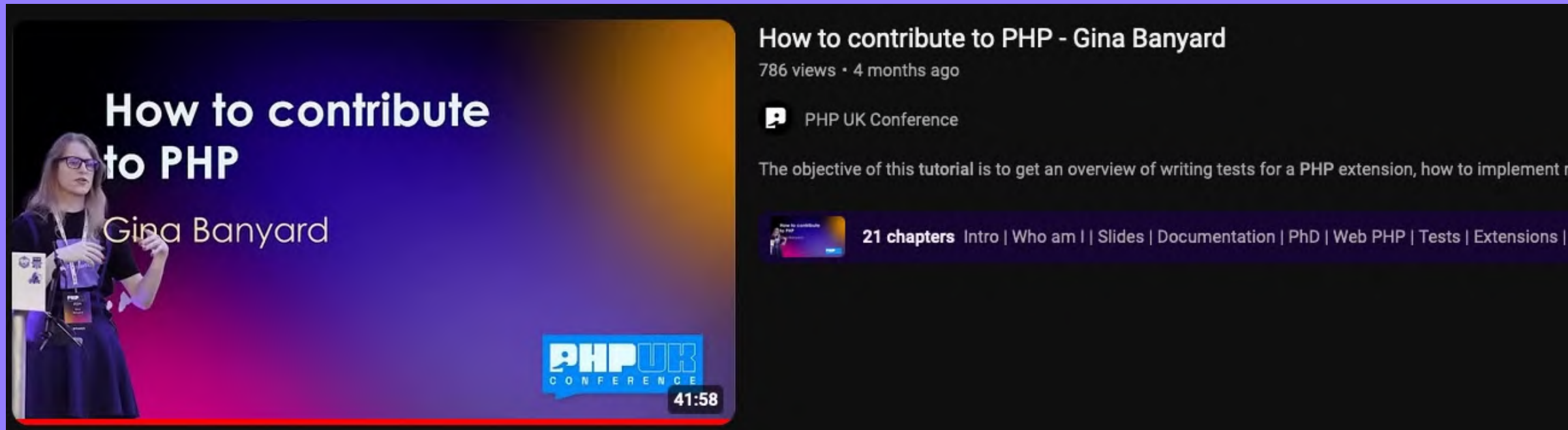
Table Of Contents

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

<https://www.phpinternalsbook.com/>

PHP Internals resources



The image shows a YouTube video player interface. The video title is "How to contribute to PHP" by Gina Banyard. The video has 786 views and was uploaded 4 months ago. The channel is "PHP UK Conference". The video description states: "The objective of this tutorial is to get an overview of writing tests for a PHP extension, how to implement new features, and how to contribute to the PHP project." The video has 21 chapters: Intro | Who am I | Slides | Documentation | PhD | Web PHP | Tests | Extensions | F... The video duration is 41:58. The video player shows a thumbnail of Gina Banyard speaking at a conference.

How to contribute to PHP
Gina Banyard

786 views · 4 months ago

PHP UK Conference

The objective of this tutorial is to get an overview of writing tests for a PHP extension, how to implement new features, and how to contribute to the PHP project.

21 chapters Intro | Who am I | Slides | Documentation | PhD | Web PHP | Tests | Extensions | F...

41:58

<https://www.youtube.com/watch?v=s-0pWt4gpFk>

Thank you for listening

Adam Spychala
/jorgsowa