

Be a Bold Coder

Beth Tucker Long

@e3betht

Who am I?

Beth Tucker Long (@e3betht)

- PHP Developer
- Stay-at-home mom
- User group leader
- Mentor & Apprentice



Audience Participation?

- Completely fine. Ask me questions any time.

Why?

Show of Hands

Why?

- Build confidence
- Improve code quality and security
- Easier integration

What this talk is not

A Bit of History

The "S" Word

Standards!

- PHP-FIG
- Zend Framework
- Symfony

Wait...Multiple
Standards?

PHP-FIG

- PSR-1 Basic Coding Standard
<http://www.php-fig.org/psr/psr-1/>
- PSR-2 Coding Style Standard
<http://www.php-fig.org/psr/psr-2/>

PHP Coding Standards Fixer

<http://cs.sensiolabs.org>

- Fixes "most" issues
- PSR-1 and PSR-2 compliant

PHP_CodeSniffer

https://github.com/squizlabs/PHP_CodeSniffer

- Phar, PEAR, or Composer
- Preset and customizable standards
- Will fix items for you automatically

PHP_CodeSniffer: Output

FILE: /myDir/myFile.php

FOUND 3 ERROR(S) AFFECTING 3 LINE(S)

2		ERROR		[]	Missing file doc comment
20		ERROR		[x]	PHP keywords must be lowercase; expected "false" but found "FALSE"
47		ERROR		[x]	Line not indented correctly; expected 4 spaces but found 1

PHP_CodeSniffer: Output

- Sublime Text

<http://www.sublimetext.com>

- PhpStorm

<https://www.jetbrains.com/phpstorm/>

- Vim

<http://www.vim.org>

The "S" Word

Security

- <http://php.net/manual/en/security.php>
- [https://www.owasp.org/index.php/PHP Security Cheat Sheet](https://www.owasp.org/index.php/PHP_Security_Cheat_Sheet)
- <http://websec.io>
- Securing PHP: Core Concepts
<https://leanpub.com/securingphp-coreconcepts>
<https://speakerdeck.com/ccornutt>
<http://securingphp.com>

Security

- <http://php.net/manual/en/security.php>
- [https://www.owasp.org/index.php/PHP Security Cheat Sheet](https://www.owasp.org/index.php/PHP_Security_Cheat_Sheet)
- <http://websec.io>
- Securing PHP: Core Concepts
<https://leanpub.com/securingphp-coreconcepts>
<https://speakerdeck.com/ccornutt>
<http://securingphp.com>

Classes

- <https://www.phparch.com/training/web-security/>
- <http://www.zend.com/en/services/training/security>
- <https://www.pluralsight.com/courses/play-by-play-website-security-review-troy-hunt-lars-klint>
- <https://nomadphp.com/products/secure-password-hashing/>
- Check your local user group

The "T" Word

Testing

A little bit of extra work up front saves
lots and lots of work later on.

Use a Testing Framework

Chris Hartjes

@grumpyprogrammer

<http://grumpy-learning.com>

Use a Testing Framework

PHPUnit

<https://phpunit.de>

The "D" Word

Documenting Your Code

- phpDocumentor: <http://www.phpdoc.org/>
- Automates documentation
- Tutorial:
[http://manual.phpdoc.org/
HTMLSmartyConverter/HandS/phpDocumentor/
tutorial_phpDocumentor.howto.pkg.html](http://manual.phpdoc.org/HTMLSmartyConverter/HandS/phpDocumentor/tutorial_phpDocumentor.howto.pkg.html)

```
/**
 * Put your short description here.
 *
 * Put your long description here.
 * You may use multiple lines.
 * You can even use Markdown.
 *
 * @author Beth Tucker Long <beth@musketeers.me>
 *
 * @since 1.0
 *
 * @param int $exampleA This is a method parameter
 * description.
 * @param string $exampleB This is another example.
 */
```

📁 Functions

Creates a packager object with all basic options set.

```
createPackager(string $original_file, string[] $options) : \PEAR_Error | \PEAR_PackageFileManager2
```

Parameters

\$original_file

`string` Path of the original package.xml.

\$options

`string[]` Set of options to merge in.

Returns

`\PEAR_Error` `\PEAR_PackageFileManager2`

Generate a XHProf Display View given the various URL parameters as arguments.

```
displayXHProfReport(object $xhprof_runs_impl, array $url_params, string $source, string $run, string $wts, string $symbol)
```

Know Your Language!

php.net

trim

Change language: [Edit](#) [Report a Bug](#)

(PHP 4, PHP 5)

trim — Strip whitespace (or other characters) from the beginning and end of a string

Description

```
string trim ( string $str [, string $character_mask = " \t\n\r\0\x0B" ] )
```

This function returns a string with whitespace stripped from the beginning and end of **str**. Without the second parameter, **trim()** will strip these characters:

- " " (ASCII 32 (0x20)), an ordinary space.
- "\t" (ASCII 9 (0x09)), a tab.
- "\n" (ASCII 10 (0x0A)), a new line (line feed).
- "\r" (ASCII 13 (0x0D)), a carriage return.
- "\0" (ASCII 0 (0x00)), the NUL-byte.
- "\x0B" (ASCII 11 (0x0B)), a vertical tab.

String Functions

[addslashes](#)[addslashes](#)[bin2hex](#)[chop](#)[chr](#)[chunk_split](#)[convert_cyr_string](#)[convert_uuencode](#)[convert_uuencode](#)[count_chars](#)[crc32](#)[crypt](#)[echo](#)[explode](#)[fprintf](#)[get_html_translation_table](#)[hebrew](#)[hebrevc](#)[hex2bin](#)[html_entity_decode](#)[htmlentities](#)

Notes

Note: Possible gotcha: removing middle characters

Because `trim()` trims characters from the beginning and end of a string, it may be confusing when characters are (or are not) removed from the middle. `trim('abc', 'bad')` removes both 'a' and 'b' because it trims 'a' thus moving 'b' to the beginning to also be trimmed. So, this is why it "works" whereas `trim('abc', 'b')` seemingly does not.

See Also

- [ltrim\(\)](#) - Strip whitespace (or other characters) from the beginning of a string
- [rtrim\(\)](#) - Strip whitespace (or other characters) from the end of a string
- [str_replace\(\)](#) - Replace all occurrences of the search string with the replacement string

User Contributed Notes 13 notes

[+ add a note](#)

▲ 23 ▼ ludko2 at gmail dot com

4 years ago

Non-breaking spaces can be troublesome with trim:

```
<?php
// turn some HTML with non-breaking spaces into a "normal" string
$myHTML = "&nbsp;abc";
$converted = stripslashes(get_html_translation_table(HTML_ENTITIES, ENT_QUOTES));

// this WILL NOT work as expected
// $converted will still appear as " abc" in view source
// (but not in od -x)
$converted = trim($converted);
```


▲ 23 ▼ ludko2 at gmail dot com

4 years ago

Non-breaking spaces can be troublesome with trim:

```
<?php
// turn some HTML with non-breaking spaces into a "normal" string
$myHTML = "&nbsp;abc";
$converted = strtr($myHTML, array_flip(get_html_translation_table(HTML_ENTITIES, ENT_QUOTES)));

// this WILL NOT work as expected
// $converted will still appear as " abc" in view source
// (but not in od -x)
$converted = trim($converted);

// &nbsp; are translated to 0xA0, so use:
$converted = trim($converted, "\xA0"); // <- THIS DOES NOT WORK

// EDITED>>
// UTF encodes it as chr(0xC2).chr(0xA0)
$converted = trim($converted,chr(0xC2).chr(0xA0)); // should work

// PS: Thanks to John for saving my sanity!
?>
```

Know Your Tools!

<https://github.com/>

Know Your Tools!

<https://github.com/>

Contribute!

Sensio Insight

<https://insight.sensiolabs.com/>

- Free for open source, integrated with Git
- Symfony-focused, but works with any PHP code
- Checks PHP, XML, YAML, Twig templates, and Composer dependencies

Sensio Insight

<https://insight.sensiolabs.com/>

- Free for open source, integrated with Git
- Symfony-focused, but works with any PHP code
- Checks PHP, XML, YAML, Twig templates, and Composer dependencies

Rule #11-013
Database queries should use parameter l

SQL Injection is possible because of code looking like this:

```
1 $query = 'SELECT * FROM User WHERE username = '.$username  
   password = ''.md5($password).'';
```

In such a request, you can assign to `$username` value `toto" OR 1 = 1`. The
SQL request will be:

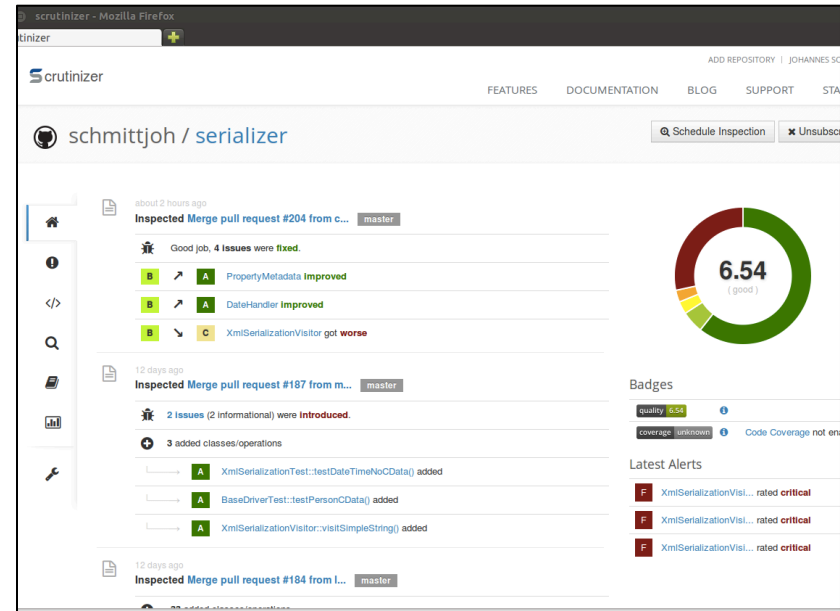
```
1 SELECT * FROM User WHERE username="toto" OR 1 = 1 AND pas  
   71dbe52628a3f83a77ab494817525c6"
```

Severity	Category	Developer
3 Critical	13 Architecture	13 John Smith
27 Major	8 Bugrisk	6 Anne Wood
62 Minor	11 Deadcode	3 Larry Fuller
2 Info	1 Security	2 Nancy Jones

Scrutinizer

<https://scrutinizer-ci.com/>

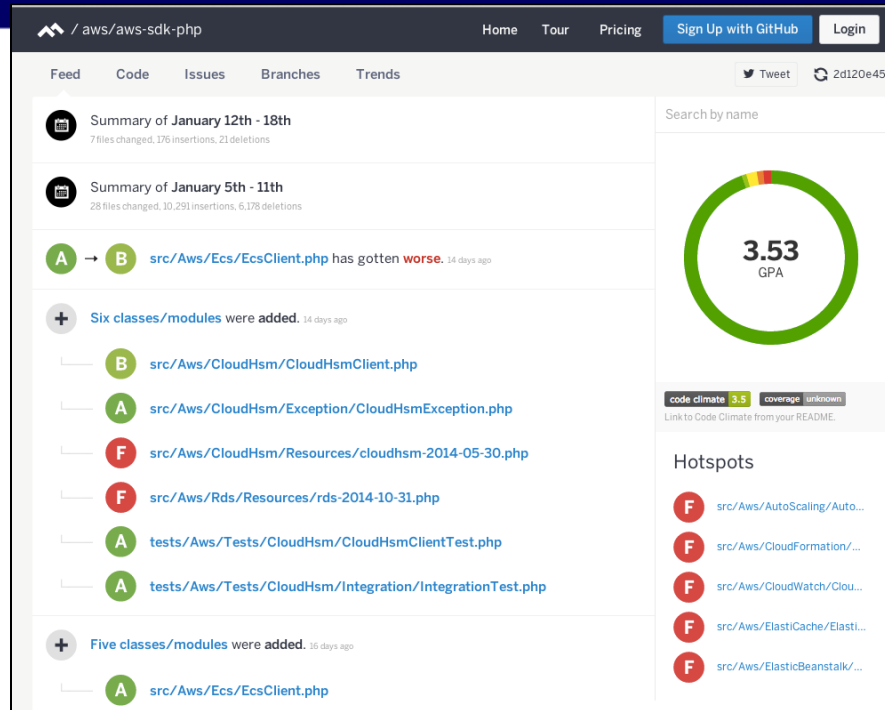
- Free for open source
- Integrated with Git
- Integrates open source checking tools like PHP_CodeSniffer and PHP Mess Detector
- Checks PHP, Python, and Ruby



Code Climate

<https://codeclimate.com/>

- Free for open source
- Integrated with Git
- Checks PHP, JS, Ruby, and coming soon - Python
- Takes security seriously:
codeclimate.com/security



Never Stop
Learning

Learn by Reading

<http://www.phparch.com/magazine/>

<http://www.phptherightway.com>

<http://phpdeveloper.org>

Learn by Reading

<https://leanpub.com>

Amazon

Your local library

Learn by Reading

- Pocket - can pin articles and gives you more recommendations for related articles
- Devdocs.io - documentation for libraries and languages, etc.
- Libraries.io - searches and gives you libraries to use

Learn by Watching

<http://nomadphp.com>

<https://www.youtube.com/user/phpukconference>

<https://www.youtube.com/user/phpacademy/>

<http://www.pluralsight.com/>

<https://www.youtube.com/user/LevelUpTuts>

<https://github.com/phptodayorg/php-must-watch>

Learn by Instruction

<http://www.phparch.com/training/>

<http://www.zend.com/services/training>

<http://www.codecademy.com/en/tracks/php>

<http://www.pluralsight.com/>

Your Local User Group!!!!

Learn by Listening

<http://devhell.info>

<http://looselycoupled.info>

<http://phptownhall.com>

<http://thatpodcast.io>

<http://phproundtable.com>

<https://rungeekradio.com>

<https://voicesoftheelephpant.com>

Learn by Practicing

<https://laracasts.com/series/code-katas-in-php>

<http://codingdojo.org/cgi-bin/index.pl?KataCatalogue>

<http://phpcodepad.com>

Learn by Email

<http://securingphp.com>

<http://www.phpweekly.com>

Learn by Mentoring

<http://phpmentoring.org>

<http://www.slideshare.net>

<http://www.joinind.in>

<http://php.net/conferences/>

<http://nocapes.net>

<http://voicesoftheelephpant.com>

Learn by Mentoring

<http://phpmentoring.org>

IRC Channels

Stack Overflow

Reddit

Learn by Hanging Out

Freenode

<https://webchat.freenode.net>

#phpc

#phpwomen

#phpmentoring

Learn by Attending

<http://php.ug>

<http://php.net/cal.php>

<http://www.meetup.com>

User Groups

If there's not one near you,
start one!

(and ask me to help)

Learn by Speaking

<http://rosstuck.com/tips-on-speaking/>

<http://matthewturland.com/2014/07/15/speaking-resources/>

<http://afilina.com/re-how-to-submit-to-conf/>

<http://www.alittleofboth.com/2014/01/how-to-submit-a-talk-to-a-conference/>

Find a Place to Speak

<http://php.net/conferences>

<http://joind.in/event/callforpapers>

<https://thecfpreport.com>

<https://twitter.com/callbackwomen>

But, I'm Still
Anxious

Resources

- <http://teamtreehouse.com/library/standards-and-best-practices>
- <http://code.tutsplus.com/tutorials/30-php-best-practices-for-beginners--net-6194>
- <https://www.airpair.com/php/posts/best-practices-for-modern-php-development>
- <https://phpbestpractices.org/>
- <https://www.corephp.com/blog/php-best-practices-that-you-must-follow/>
- <http://www.phpbuilder.com/articles/application-architecture/optimization/explore-the-top-11-php-best-practices.html>
- <http://pear.php.net/manual/en/standards.bestpractices.php>
- <http://www.sitepoint.com/php-tips-resources-best-practices-2015/>

Final Thoughts

Bold coding is, of course, about knowledge and skill, but more importantly, it's about being active:

Actively Learning

Actively Participating

Actively Improving

Find Me

- Twitter: e3betht
- Madison PHP User Group (Meetup)
<http://www.madisonphp.com>
- Slides Available on:
<http://www.TreelineDesign.com/slides>

Feedback

<https://joind.in/talk/133b0>

E-mail:

Beth@TreelineDesign.com

@e3betht