The Big “Why equal doesn’t equal” Quiz

Presented by:

Juliette Reinders Folmer

"More people would learn from their mistakes if they weren't so busy denying them."

~Harold J. Smith~

www.InspirationBoost.com

Some best practices (1)

- Know your **variable types** and how **type juggling** works ;-)
- Always use **strict checking** unless loose checking will avoid code duplication
- Auto-document your code: show what you are testing for

DON'T:

```
if ( strpos( $x, $y ) ) {}
```

DO:

```
if ( strpos( $x, $y ) !== false ) {}
```

Advies en zo
Meedenken en -doen

Some best practices (2)

- Make it a habit to test for type AND for what you want to know
- Non-empty string:

```
if ( is_string( $x ) && $x !== '' ) {}
```

- Non-empty array:

```
if ( is_array( $x ) && count( $x ) > 0 ) {}
```

or

```
if ( is_array( $x ) && $x !== array() ) {}
```

Hint: unless you are 100% sure that what you have is a non-empty array, always wrap foreach statements within this!

Advies en zo
Meedenken en -doen

Some best practices (3)

- Use function_exists(), method_exists() and extension_loaded() to write cross-version code
- Also be aware that across PHP versions, the output of numerous functions when used inappropriately has changed.
- Never be stingy with parentheses ()
- unset() variables once you've used them – especially after loops – to avoid contamination
- isset() is your friend

And remember....

- ... that what you get from **$_POST**/**$_GET**/database are strings
- ... that ctype_ functions are locale() dependent
- ... that switch() does a loose type check
- ... that calculations with floats can only be done reliably by casting to string and using bcmath (but mind the scale!)

Want to know more ?

Extensive cheatsheets are available!

http://adviesenzo.nl/php-cheatsheet/

General typing | is_...() | Null | Boolean | Integers | Floats | Numeric tests | String casting | String tests | Array casting

Array testing | Objects | Resources | Arithmetic | ctype extension | filter extension - bool/int/float | filter extension - string

Numeric tests	is_numeric()	> 0	>= 0	== 0	=== 0	!= 0	!== 0	< 0	<= 0	is_nan()	is_finite()	is_infinite()	floor()	ceil()	round()	Numeric tests
$x = null : (= NULL)	false	false	true	true	false	false	true	false	true	false	true	false	0	0	0	$x = null : (= NULL)
$x = null : (= NULL)	false	false	true	true	false	false	true	false	true	false	true	false	0	0	0	$x = null : (= NULL)
$x = bool : (= false)	false	false	true	true	false	false	true	false	true	false	true	false	0	0	0	$x = bool : (= false)
$x = bool : 1 (= true)	false	true	true	false	false	true	true	false	false	false	true	false	1	1	1	$x = bool : 1 (= true)
$x = int : 1	true	true	true	false	false	true	true	false	false	false	true	false	1	1	1	$x = int : 1
$x = int : 0	true	false	true	true	true	false	false	false	true	false	true	false	0	0	0	$x = int : 0
$x = int : -1	true	false	false	false	false	true	true	true	true	false	true	false	-1	-1	-1	$x = int : -1
$x = int : 42	true	true	true	false	false	true	true	false	false	false	true	false	42	42	42	$x = int : 42
$x = int : 13369593 †i8	true	true	true	false	false	true	true	false	false	false	true	false	13369593	13369593	13369593	$x = int : 13369593 †i8
$x = int : 83 †i9	true	true	true	false	false	true	true	false	false	false	true	false	83	83	83	$x = int : 83 †i9
$x = int : 57	true	true	true	false	false	true	true	false	false	false	true	false	57	57	57	$x = int : 57
$x = float : 1.3	true	true	true	false	false	true	true	false	false	false	true	false	1	2	1	$x = float : 1.3
$x = float : 0.005	true	true	true	false	false	true	true	false	false	false	true	false	0	1	0	$x = float : 0.005
$x = float : 0	true	false	true	true	false	false	true	false	true	false	true	false	0	0	0	$x = float : 0
$x = float : -1.3	true	false	false	false	false	true	true	true	true	false	true	false	-2	-1	-1	$x = float : -1.3
$x = float : NAN †f5	true	false	false	false	false	true	true	false	false	true	false	false	NAN	NAN	NAN	$x = float : NAN †f5
$x = float : -INF †f6	true	false	false	false	false	true	true	true	true	false	false	true	-INF	-INF	-INF	$x = float : -INF †f6

(Source: https://github.com/jrfnl/PHP-cheat-sheet-extended)

Advies en zo
Meedenken en -doen

Keep in touch!

(I'm self-employed, you can hire me ;-))

Email: juliette@adviesenzo.nl

Web: http://www.adviesenzo.nl/

LinkedIn: http://nl.linkedin.com/in/julietterf

Twitter: http://twitter.com/jrf_nl

GitHub: http://github.com/jrfnl/

Please rate this talk on joind.in/10291

Endorsements and recommendations on LinkedIn are much appreciated too!