
php-private-access Documentation

Release 2.0.0

Jan 30, 2021

CONTENTS

1	Installation	3
1.1	Why <code>--dev</code> ?	3
2	Functions	5
2.1	<code>get_private_field</code>	5
2.2	<code>set_private_field</code>	6
2.3	<code>call_private_method</code>	6
2.4	<code>get_private_const</code>	6
3	License	7

A small simple library to access private properties of the objects. Actually it's more an example of mad skillz than a useful tool. No Reflection API calls!

INSTALLATION

Use composer:

```
composer require sandfoxme/private-access --dev
```

1.1 Why --dev?

This library may be used for debugging or with PHP consoles like PsySH. If you are actually using it in some live system, you're doing something terribly wrong.

FUNCTIONS

These four simple functions can come in handy as helpers for something like [PsySH](#)

Note: All functions also work with protected and public visibilities

Class examples:

```
<?php

class A extends B
{
    private const SECRET_CONST = '';
    private $secret;
    private static $staticSecret;
    private function doStuff() {}
    private static function doStaticStuff() {}
}

class B
{
    private $secret;
    private function doStuff() {}
}

$a = new A();
```

2.1 get_private_field

Get value of a private field

```
<?php

use function SandFox\Debug\get_private_field;

get_private_field($a, 'secret'); // get $a->A::$secret value
get_private_field(A::class, 'staticSecret'); // get A::$staticSecret value
get_private_field([$a, B::class], 'secret'); // get $a->B::$secret value
```

2.2 set_private_field

Set value of a private field

```
<?php

use function SandFox\Debug\set_private_field;

set_private_field($a, 'secret', 'new secret'); // set new $a->A::secret value
set_private_field(A::class, 'staticSecret', 'new secret'); // set new A::
↪ $staticSecret value
set_private_field([$a, B::class], 'secret', 'new secret'); // set new $a->B::$secret_
↪ value
```

2.3 call_private_method

Call private method

```
<?php

use function SandFox\Debug\call_private_method;

call_private_method($a, 'doStuff', 'whatever'); // call $a->A::doStuff('whatever')
call_private_method(A::class, 'doStaticStuff', 'whatever'); // call A::doStaticStuff(
↪ 'whatever')
call_private_method([$a, B::class], 'doStuff', 'whatever'); // call $a->B::doStuff(
↪ 'whatever')
```

2.4 get_private_const

Get value of a private constant

Note: This function works in all supported PHP versions but is useful only with PHP 7.1+ where constants can be private or protected

```
<?php

use function SandFox\Debug\get_private_const;

get_private_const($a, 'SECRET_CONST');
// or
get_private_const(A::class, 'SECRET_CONST');
```

CHAPTER THREE

LICENSE

The library is available as open source under the terms of the [MIT License](#).