

Open-Source programování

Ondřej Mirtes



apigen

apigen.org

Namespaces

Nette

[Application](#)

[Caching](#)

[ComponentModel](#)

[Config](#)

[Database](#)

[DI](#)

[Diagnostics](#)

Forms

[Controls](#)

[Rendering](#)

[Http](#)

[Iterators](#)

[Latte](#)

[Loaders](#)

[Localization](#)

[Mail](#)

[Reflection](#)

[Security](#)

[Templating](#)

[Utils](#)

[NetteModule](#)

[None](#)

[PHP](#)

Namespace Nette\Forms

Classes summary

Container	Container for form controls.
ControlGroup	A user group of form controls.
Form	Creates, validates and renders HTML forms.
Rule	Single validation rule or condition represented as value object.
Rules	List of validation & condition rules.

Interfaces summary

IControl	Defines method that must be implemented to allow a component to act like a form control.
IFormRenderer	Defines method that must implement form renderer.
ISubmitterControl	Defines method that must be implemented to allow a control to submit web form.

Namespaces summary

[Nette\Forms\Controls](#)

[Nette\Forms\Rendering](#)

Classes

[Container](#)

[ControlGroup](#)

[Form](#)

[Rule](#)

Namespaces

Nette

Application

Caching

ComponentModel

Config

Database

DI

Diagnostics

Forms

Controls

Rendering

Http

Iterators

Latte

Loaders

Localization

Mail

Reflection

Security

Templating

Utils

NetteModule

None

PHP

Classes

Container

ControlGroup

```
64:     /**
65:      * Fill-in with values.
66:      * @param array|Traversable values used to fill the form
67:      * @param bool    erase other controls?
68:      * @return Container provides a fluent interface
69:      */
70:     public function setValues($values, $erase = FALSE)
71:     {
72:         if ($values instanceof \Traversable) {
73:             $values = iterator_to_array($values);
74:
75:         } elseif (!is_array($values)) {
76:             throw new Nette\InvalidArgumentException("First parameter must be an array");
77:         }
78:
79:         foreach ($this->getComponents() as $name => $control) {
80:             if ($control instanceof IControl) {
81:                 if (array_key_exists($name, $values)) {
82:                     $control->setValue($values[$name]);
83:
84:                 } elseif ($erase) {
85:                     $control->setValue(NULL);
86:                 }
87:
88:             } elseif ($control instanceof Container) {
89:                 if (array_key_exists($name, $values)) {
90:                     $control->setValues($values[$name], $erase);
91:
92:                 } elseif ($erase) {
93:                     $control->setValues(array(), $erase);
94:                 }
95:             }
96:         }
97:     }
98:
99:     return $this;
```



NetBeans

```
1 <?php
2
3 /*
4 * This file is part of the Symfony package.
5 *
6 * (c) Fabien Potencier <fabien@symfony.com>
7 *
8 * For the full copyright and license information, please view the LICENSE
9 * file that was distributed with this source code.
10 */
11
12 namespace Symfony\Component\HttpFoundation\Session\Storage\Handler;
13
14 /**
15 * Adds SessionHandler functionality if available.
16 *
17 * @see http://php.net/sessionhandler
18 */
19
20 if (version_compare(phpversion(), '5.4.0', '>=')) {
21     class NativeSessionHandler extends \SessionHandler {}
22 } else {
23     class NativeSessionHandler {}
24 }
```

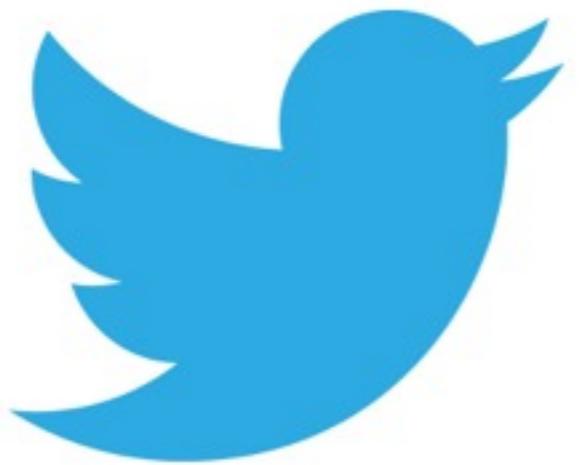
PHP Token Reflection

Emulates the PHP reflection model using the tokenized PHP source.





?



@OndrejMirtes