ClickBlocks\Core\Template

General information

Inheritance	no
Child classes	no
Interfaces	ArrayAccess
Source	Framework/core/template.php

The class is the main Framework's template engine that used php as a template language. The class provides a simple and convenient way to manipulate the variables in the template, and also supports nested templates with the possibility of caching.

Variables templates may be global, i.e. visible in all templates, the work that is carried out through **Template** class and local - visible only in the template for which they are installed.

You can access to global template variables through the interface **ArrayAccess** (as well as a static class methods), and to local variables of a particular pattern through overloading properties. The following example shows how to work with global and local variables:

```
// Sets array of global variables.
Template::setGlobals(['gv1' => 1, 'gv2' => 2]);

// Begins the work with some template.
$tpl = new Template($template);
// Creates the local variable "foo"
$tpl->foo = 'test';
// Redefines the global variable "gv1"
// by defining the local variable with the same name.
$tpl->gv1 = 2;
// Changes value of the global variable "gv2"
$tpl['gv2'] = 1;
```

As can be seen from the example priority of local variables is higher than priority of global variables. This means that if the template has any local variable whose name coincides with a global variable, then when the template is processed value of a local variable will be taken.

Public static methods

getGlobals()

```
public static array getGlobals()
```

Returns array of global variables of templates.

setGlobals()

```
public static void setGlobals(array $globals, boolean $merge = false)
```

$globals	array	array of global variables.
$merge	boolean	if it is TRUE then old global variables is merged with new variables. Otherwise new global variables is completely replaced the old ones.

Allows to set global template variables.

Public non-static properties

cacheID

```
public string $cacheID
```

Unique identifier of the template cache. If it is not set, the template itself will be used as ID of the cache.

cacheExpire

```
public integer $cacheExpire = 0
```

Expiration time of the template cache, in seconds. If its value less or equal zero, a template is not cached.

cacheGroup

```
public string $cacheGroup = 'templates'
```

Name of the cache group of templates.

Public non-static methods

__construct()

```
public void __construct(string $template = null, integer $expire = 0, string $cacheID = null, ClickBlocks\Cache\Cache $cache = nul
```

$template	string	template string or path to the template file.
$expire	integer	template cache lifetime in seconds.
$cacheID	string	unique identifier of the template cache.
$cache	ClickBlocks\Cache\Cache	cache object.

Class constructor which allows set a template and turns on caching of it.

getCache()

```
public ClickBlocks\Cache\Cache getCache()
```

Returns a cache object. If no cache object was set then the cache object returned by method `CB::cache()` will be used.

setCache()

```
public void setCache(ClickBlocks\Cache\Cache $cache)
```

$cache	ClickBlocks\Cache\Cache	cache object.

Sets a cache object.

isExpired()

```
public boolean isExpired()
```

Returns TRUE if cache lifetime is expired or not set (property "expire" is 0). In other cases the method returns FALSE.

getVars()

```
public array getVars()
```

Returns array of local template variables.

setVars()

```
public void setVars(array $variables, boolean $merge = false)
```

$vars	array	array of local template variables.
$merge	boolean	if it is TRUE then new variables will be merged with the old variables. Otherwise new variables is completely replaced the old ones.

Sets local template variables.

getTemplate()

```
public string getTemplate()
```

Returns template.

setTemplate()

```
public void setTemplate(string $template)
```

$template	string	new template.

Sets template to process.

offsetGet()

```
public mixed offsetGet(string $name)
```

$name	string	name of a global variable.

Returns value of a global template variable. If a variable with the given name does not exist the method returns NULL.

offsetSet()

```
public void offsetSet(string $name, mixed $value)
```

$name	string	name of a global variable.
$value	mixed	value of a global variable.

Sets value of a global template variable.

offsetExists()

```
public boolean offsetExists(string $name)
```

$name	string	name of a global variable.

Returns TRUE if global variable with such name already exist and its value is not NULL. Otherwise the method returns FALSE.

offsetUnset()

```
public void offsetUnset(string $name)
```

$name	string	name of global template variable.

Removes global template variable by its name.

__get()

```
public mixed __get(string $name)
```

$name	string	name of local template variable.

Returns value of a local template variable. If variable with such name does not exist the method returns NULL.

__set()

```
public void __set(string $name, mixed $value)
```

$name	string	name of local template variable
$value	mixed	value of local template variable.

Sets value of a local template variable.

__isset()

```
public boolean __isset(string $name)
```

$name	string	name of local template variable.

Returns TRUE if a local variable with such name already exists and its value i not NULL. Otherwise the method returns FALSE.

__unset()

```
public void __unset(string $name)
```

$name	string	name of local template variable.

Removes a local variable by its name.

render()

```
public string render()
```

Processes the template and returns the result. If template caching is used and the cache unique identifier is not set, md5 hash of template will be taken as the cache identifier.

> If you often use the same template, be sure to clearly define the cache ID. Otherwise, the template cache conflict may occur.

show()

```
public void show()
```

Processes the template and shows result in browser.

__toString()

```
public string __toString()
```

Converts object of class **ClickBlocks\Core\Template** to string. Value of this string is the processed template.

Protected static properties

globals

```
protected static array $globals = []
```

Stores array of global template variables.

Protected non-static properties

cache

```
protected ClickBlocks\Cache\Cache $cache
```

Cache object.

vars

```
protected array $vars = []
```

Array of local template variables.

template

```
protected string $template
```

Template string or path to a template file.

Protected non-static methods

getCacheID()

```
protected mixed getCacheID()
```

Returns template cache ID.