Szablony tematyczne i Smarty

Spis treci

Szablony tematyczne i Smarty

Smarty jest silnikiem szablonu PHP który jest wytrzymay,szybki, atwy w nauce i uyciu, i ma czyst i kompaktow skadnie do projektowania dedykowanego. Pomaga zbudowa znacznie prostszy kod HTML i dziaa poprzez gromadzenie, a nastpnie buforowania kadej strony.

Moesz si dowiedzie wicej wchodzc na oficjaln stron: http://www.smarty.net/

Podstawowe zasady

Ograniczniki

Ograniczniki umozliwiaj silnikowi szablonu "rozpozna" Smarty zwraca si wewntrz szablonu. Ograniczniki wykorzystywane przez Smarty s w nawiasach klamrowych The delimiters used by Smarty: `{smarty_call}`

> ⚠ Jeli trzeba uy rzeczywistych nawiasów klamrowych w kodzie szablonu Smarty posiada specjalne poczenie dla Ciebie: `{redelim}` na lewej nawias klamrowy ({), i `{rdelim}` dla prawej nawias klamrowy (}).

Komentarze

Jak w przypadku kadego jzyka programowania lub skryptów, mona umieci komentarze w kodzie szablonu, a silnik szablonu nie bdzie ich analizowac.

Komentarze uywaj regularnych ograniczników, wraz z otwieraniem i zamykaniem * znaków.

```
{* One-line comment. *}

{*
Multiple
lines
comment.
*}

{* Comment for Smarty {$code} *}
```

Zmienne

Tak jak w PHP, zmienne s reprezentowane przez znak dolara, po którym nastpuje nazwa zmiennej. Umieszczenie zmiennej bezporednio w ograniczniku Smarty oznacza, e chcesz wywietli zmienn tak jaka jest.
Na przykad, `{$foo}` Smarty jest odpowiednikiem PHP `echo $foo;`.

Mona równie stosowa moyfikatory do zmiennych:

```
{* Displaying the content of the variable in lowercase. *}
{$foo|lower}

{* Displaying the content of the variable after replacing one word with another. *}
{$foo|replace:'bar':'baz'}

{* Example of "chaining" modifiers.         *}
{$foo|lower|capitalize|truncate:10:'...'}
```

Moesz uzyska pen list dostpnych modyfikatorów na oficjalnej stronie internetowej: http://www.smarty.net/docs/en/language.modifiers.tpl

Tryby warunkowe

Smarty ma warunkowy system `if` / `else` / `elseif` :

```
{if $coffee == 'good'}
  {* Happy coder *}
{elseif $coffee == 'very good'}
  {* Very happy coder *}
{else}
  {* Grumpy coder *}
{/if}
```

Moesz si dowiedzie wicej o rónych trybach warunkowych na oficjalnej stronie: http://www.smarty.net/docsv2/en/language.function.if.tpl

Ptle

Smarty obsuguje dwie ptle: `section` `i foreach`. Mona uy iteratorów, a nawet trybu warunkowego `fo reachelse` :

```
<?php
  $items_list = array( 23 => array('no' => 2456, 'label' => 'Salad'), 96 => array('no' => 4889, 'label' =>
'Cream') );
  $smarty->assign('items', $items_list);
?>
<ul>
{foreach from=$items key=myId item=i}
  <li><a href="item.php?id={$myId}">{$i.no}: {$i.label}</li>
{/foreach}
</ul>
```

Moesz dowiedzie si wicej o kadej ptli na oficjalnych stronach internetowych:

- http://www.smarty.net/docs/en/language.function.section.tpl
- http://www.smarty.net/docs/en/language.function.foreach.tpl

Wczanie pliku

Mona atwo doczy plik szablonu do innego uywajc funkcji `include`, `extends` or `block`. Dziki dziedziczeniu, wczenie pliku moe mie wpyw na wiele szablonów naraz.

Metoda przypisania uywa dwóch obowizkowych argumentów:

- `file`: Uwgldnia nazw plikiu szablonu.
- `assign`: Nazwa zmiennej wyjcia zostanie przypisana.

Oto kilka przykadów:

```
{* Including a file. *}
{include file='steps.tpl'}

{* Placing the included content in a variable, then display the variable.         *}
{include file='text.tpl' assign='MyText'}
{$MyText}
```

Demonstracja dziedziczenia z `extends` i `block`:

```
{* Filename: parent.tpl *}
<html>
<head>
<title>{block name="title"}Default Title{/block}</title>
...
{block name="title"}New Title{/block}
</head>
</html>

{* Filename: child.tpl *}
{extends file='parent.tpl'}
{block name='title'}New Page Title{/block}
```

Ten ostatni przykad uywa funkcji `block`, który jest przeznaczony do okrelenia nazwy powierzchni róda matrycy do matrycy dziedziczenia.

Mona si dowiedzie wiecej na temat dziedziczenia szablonów i kadej funkcji umieszczenia na swojej stronie na oficjalnej stronie:

- http://www.smarty.net/docs/en/advanced.features.template.inheritance.tpl
- http://www.smarty.net/docs/en/language.function.include.tpl
- http://www.smarty.net/docs/en/language.function.extends.tpl
- http://www.smarty.net/docs/en/plugins.block.functions.tpl

Funkcja debugowania

Kompletny zestaw procesów wewntrznych z szablonu Smarty mog by wywietlane, gdy wywietlana jest strona.

Podczas rozwoju tematycznego, mona to zrobi dla kadego adowania strony poprzez edycje pliku `/config/smarty.config.inc.php` i edycje `$smarty->debugging` wartoci:

```
$smarty->debugging = true;
```

Gdy motyw jest na stronie produkcyjnej, mona wczy funkcje debugowania poprzez dodanie dyrektywy`{DEBUG}` w pliku szablonu.

Mona take zarzdza funkcj debugowania bezporednio z PrestaShop w menu "Paramenty Zaawansowane" na sronie "Konserwacja" zmie opcje "Konsola Debugownia" do swoich upodoba,

Moesz dowiedzie si wicej o tej funkcji `debug` na oficjalnej stronie: http://www.smarty.net/docs/en/language.function.debug.tpl

Zaawansowane zastosowania

Zrzuty

Funkcja `capture` umoliwia pobieranie danych wyjciowych szablonu bez wywietlania go. Na przykad: `{capture name="myCapture'} ... {/capture}`

W celu korzystania z takich materiaów naley przywoa `$smarty` super zmienna: `$smarty.capture.myCapture`

Nie zapomnij sprawdzi chwytu przed uyciem go:

> ⓘ `{if $smarty.capture.<name>}` or `{if isset($smarty.capture.<name>}`

Funkcja przechwytywania ma moliwo automatycznego przypisania zmiennej:

```
{capture name='myCapture' assign='myVar'}
   ...
{/capture}
{* Then, in order to use the content, call the $myVar variable. *}
```

Moesz dowiedzie si wicej o funkcji `capture` na oficjalnej stronie: http://www.smarty.net/docs/en/language.function.capture.tpl

Przypisanie zmiennych

Moliwe jest przypisanie zmiennej do pliku szablonu (widok) uywajc funkcj `assign` :

```
{assign var='myVar2' value='My value'}

{* Will display "My value". *}
{$myVar2}
```

Moesz dowiedzie si wicej o funkcji `assign` na oficjalnej stronie: http://www.smarty.net/docs/en/language.function.assign.tpl

Zmienna $smarty

`$smarty` jest to tak zwana super zmienna. To sprawia, e mona pobra kilka przydatnych informacji:

- `capture` wartoci: `$smarty.capture.myVariable`
- wartoci GET: `{$smarty.get.<name>}`
- wartoci POST: `{$smarty.post.<name>}`
- aktualny timestamp: `{$smarty.now}`, lub niestandardowego formatowania `{$smarty.now|date_format:'%d-%m-%Y %H:%M:%S'}`
- stae PHP: `{$smarty.const.<constant name>}`

⚠ w Smarty 2, `$smarty` mog by stosowane `foreach`:

Gdy ptla jest zdefiniowana jako: `{foreach from=$myarray key="mykey" item="myitem"}`
...mona wykona wezwanie `$smarty.foreach.name.property`

Od PrestaShop 1.5 zaleca si polega tylko na skadni Smarty 3 z szablonu PrestaShop, W zwizku z tym `$smarty.foreach.varName.property` wezwanie call musi by zastapiony przez `$varName@property` równowane wezwanie.

Moesz dowiedzie wicej o zmiennej `$smarty` na oficjalnej stronie: http://www.smarty.net/docs/en/language.variables.smarty.tpl

Literówki

Etykieta `literal` umoliwia blokowi danych do wykorzystanie jej dosownie, bez Smarty próbuje je interpretowa:

```
{literal}
<script type="text/javascript">
  function myFonction()
  {
    ...
  }
</script>
{/literal}
```

Moesz dowiedzie si wicej na temat funkcji dosownej na oficjalnej stronie: http://www.smarty.net/docs/en/language.function.literal.tpl

Funkcje

Funkcje Smarty nie uywa $ prefix, podobnie jak zmienne: `{debug}`, `{rdelim}`, `{ldelim}`, ...

Mog one zaakceptowa argumenty: `{include file='<name of the file>'}`

Struktura wywoania funkcji jest wic: `{nameOfTheFunction arg1='...' arg2='...'}`

Nie mona uy modyfikatorów na funkcjach, na przykad `{nameOfTheFunction arg1='...' |lower}` nie bdzie dziaa zgodnie z oczekiwaniami. .

Moesz dowiedzie sie wicej o funkcjach Smarty na oficjalnej stronie:

- http://www.smarty.net/docs/en/language.syntax.functions.tpl
- http://www.smarty.net/docs/en/language.builtin.functions.tpl
- http://www.smarty.net/docs/en/language.custom.functions.tpl

Wtyczki

Wtyczki Smarty umoliwia atwe rozszerzenie standardowego zachowania. S one zapisane w PHP. Na przykad, PrestaShop ma wtyczki Smarty, które tworz specyficzn funkcje obsugiwania tumacze: `{l}`

Po pierwsze w temacie:

```
{l s='Hello dear viewer'}
```

i w module (nawet jeli jest nadpisany!) And in a module (even if overridden!):

```
{l s='Hello dear view' mod='myModule'}
```

Moesz dowiedzie si wicej o wtyczkach Smarty na oficjalnej stronie: http://www.smarty.net/docs/en/plugins.tpl

Smarty niemile widziane z PrestaShop

Kilka rzeczy które trzeba wyranie unika przy uyciu Smarty:

- Nie rób staego bezporedniego poczenia PrestaShop. Bardziej szczegóowo nawet nie uywaj `{$smarty.const._DB_PASSWD_}`, z oczywistych powodów..
- Nie zastpuj przypisanych zmiennych PrestaShop.
- Nie twórz niepotrzebnie kodu trudnego do odczytania. Na przykad powstrzymujc si od jakichkolwiek zawoa `include` od doczonego ju od wewntrz pliku.
- Nie rób bezporedniego poczenia PHP. Na przykad nie naley uywa `{php} // PHP code {/php}`