Smarty 2.6.x Cheatsheet

Smarty - http://www.smarty.net/ - template engine - a free library for PHP.

Setup directories

templates/	$smarty->template_dir	Input - tpl files.
configs/	$smarty->config_dir	Input - conf files.
templates_c/	$smarty->compile_dir	Temporary - compiled templates and configuration files.
cache/	$smarty->cache_dir	Temporary - cached output files.

Template syntax

Comment	{* Blah blah *}
Function call	{insert file='banner_ads.tpl' title=$title}

Scalar variable
Numbered array
Assocciative array
Object property
Object method
	{$scalar}
	{$num_array[3]}
	{$assoc_array.key1}
	{$object->property}
	{$object->method()}

Configuration variable	{#ConfigVarName#} - same as {smarty.config.ConfigVarName}
Embedded variables	{func var="text $var1 $arr1[0] $arr2[key] `$arr2.key` text"}
Modifiers	{$articleTitle|lower|spacify|truncate:30:"..."}
Math	{$var1 * $var2 + 1}

$smarty reserved variable

{$smarty.get.VarName}	$_GET[VarName]
{$smarty.post.VarName}	$_POST[VarName]
{$smarty.cookies.VarName}	$_COOKIE[VarName]
{$smarty.server.VarName}	$_SERVER[VarName]
{$smarty.env.VarName}	$_ENV[VarName]
{$smarty.session.VarName}	$_SESSION[VarName]
{$smarty.request.VarName}	Merged get, post, cookies, server, env
{$smarty.now}	time() - current timestamp
{$smarty.const.CONST_NAME}	Constants defined with define('CONST_NAME', 'ConstValue');
{$smarty.capture.VarName}	Output captured with {capture}...{/capture}
{$smarty.config.Name1}	Configuration variable, equivalent to {#Name1#}

{$smarty.section.Name1.Sth}	Parameter of section Name1 created by {section} Members: index, index_prev, index_next, iteration, first, last, rownum, loop, show, total
{$smarty.foreach.Name1.Sth}	Parameter of loop Name1 create by {foreach} Members: index, iteration, first, last, show, total
{$smarty.template}	Name of the current template, e.g. "FileName1.tpl"
{$smarty.version}	Version of Smarty
{$smarty.ldelim}, {$smarty.rdelim}	Left and right delimiter, also printed by {ldelim}, {rdelim}

Variable modifiers

|capitalize	Capitalize the first letter of all words in a variable.
|cat:'xxx'	Value is concatenated to the given variable.
|count_characters	Count the number of characters in a variable.
|count_paragraphs	Count the number of paragraphs in a variable.
|count_sentences	Count the number of sentences in a variable.
|count_words	Count the number of words in a variable.
|date_format:'%H:%M:%S'	Formats a date and time into the given strftime() format.
|default:'no title'	Set a default value for a variable.
|escape:'html'	Encode or escape a variable to: html, url, single quotes, hex, hexentity, javascript, mail.
|indent:4	Indents a string on each line.
|lower	Lowercase a variable.
|nl2br	"\n" line breaks will be converted to html <br /> tags (equivalent to nl2br function).
|regex_replace:"/[\r\t\n]/":" "	A regular expression search and replace (preg_replace syntax).
|replace:'Garden':'Vineyard'	Simple search and replace (equivalent to str_replace).
|spacify	A way to insert a space between every character.
|string_format:"%.2f"	A way to format strings, such as decimal numbers and such (sprintf syntax).
|strip	Replaces all repeated spaces, newlines and tabs with a single space, or with the supplied string.
|strip_tags	Strips out markup tags, basically anything between < and >.
|truncate:30:"...":true	Truncates a variable to a character length.
|upper	Uppercase a variable (equivalent to strtoupper).
|wordwrap:30:"<br />\n"	Wraps a string to a column width, the default is 80 (equivalent to wordwrap).

Built-in functions

{capture name=banner}...{/capture}	Collect the output of the template between the tags into a variable instead of displaying it.
{config_load file="example.conf"}	Loading config #variables# from a configuration file into the template.
{foreach from=$Arr1 item=Foo key=Key name=Name}... {foreachelse}... {/foreach}	Loop over an associative array as well a numerically-indexed array. Loop parameters in $smarty.foreach.Foo: index, iteration, first, last, show, total.
{if $name eq 'Fred'}... {elseif $name eq 'John'}... {else}... {/if}	Qualifiers: == (eq), != (ne, neq), > (gt), < (lt), >= (gte, ge), <= (lte, le), ===, ! (not), % (mod) is [not] div by, is [not] even, is [not] even by, is [not] odd, is [not] odd by
{include file='page_header.tpl'}	Including other templates in the current template.
{include_php file='load_nav.php'}	(Deprecated) Include a php script in your template.
{insert name="getBanner"}	Work much like {include} tags, except that {insert} tags are NOT cached when template caching is enabled.
{ldelim}, {rdelim}	Used for escaping template delimiters, by default { and }.
{literal}...{/literal}	Allow a block of data to be taken literally.
{php}...{/php}	(Not recommended) Allow PHP code to be embedded directly into the template.
{section name=customer loop=$custid}... {sectionelse}... {/section}	Looping over arrays of data.
{strip}...{/strip}	Tags are stripped of the extra spaces or carriage returns at the beginnings and ends of the lines before they are displayed.

Custom functions

{assign var='name' value='Bob'}	Assigning template variables during the execution of a template.
{counter}	Print out a count.
{cycle values="#eeeeee,#d0d0d0"}	Alternate a set of values.
{debug}	Dumps the debug console to the page.
{eval var=$foo}	Evaluate a variable as a template.
{fetch file='navbar.js'}	Retrieve files from the local file system, http, or ftp and display the contents.

{html_checkboxes}	Creates an html checkbox group with provided data.
{html_image}	Generates an HTML <img> tag.
{html_options}	Creates the html <select><option> group with the assigned data.
{html_radios}	Creates a HTML radio button group.
{html_select_date}	Creates date dropdowns. It can display any or all of year, month, and day.
{html_select_time}	Creates time dropdowns. It can display any or all of hour, minute, second and meridian.
{html_table loop=$data}	Dumps an array of data into an HTML <table>.
{mailto address="me@example.com" encode="javascript"}	Automates the creation of a mailto: anchor links and optionally encodes them.
{math equation="x+y" x=$height y=$width}	(Inefficient) Allows the template designer to do math equations in the template.
{popup text='Hello World!'}	Create Javascript popup layer/windows.
{popup_init src='overlib/overlib.js'}	Integration of overLib, a library used for popup windows.
{textformat}...{/textformat}	Cleans up spaces and special characters, and formats paragraphs by wrapping at a boundary and indenting lines.

Plugins

Functions	{Func1 Param1="Val1", Param2="Val2"}	$smarty->register_function()	function.NAME.php
Modifiers	{$val|Modifier1:ParamVal1:ParamVal2}	$smarty->register_modifier()	modifier.NAME.php
Block functions	{BlockFunc Param1="Val1"}...{/BlockFunc}	$smarty->register_block()	block.NAME.php
Compiler functions	Compiler functions are called only during compilation of the template.	$smarty->register_compiler_function()	compiler.NAME.php
Prefilters	Functions that your templates are ran through before they are compiled.	$smarty->register_prefilter()	prefilter.NAME.php
Postfilters	Functions that your templates are ran through after they are compiled.	$smarty->register_postfilter()	postfilter.NAME.php

Output filters	When the template is invoked via display() or fetch(), its output can be sent through output filters.	$smarty->register_outputfilter()	outputfilter.NAME.php
Resources	Retrieving templates using whatever possible source you can access with PHP: databases, sockets, LDAP, and so on.	$smarty->register_resource()	resource.NAME.php
Inserts	{insert name="Func1"}	-	insert.NAME.php