

Initializing Smarty

```
include("smarty/Smarty.class.php");
$smarty = new smarty();
```

Assigning Variables

General Format
`$smarty->assign("var", "value");`
Example
`$smarty->assign("who", "world");`

Sample Template

```
Filename : templates\greet.tpl
Hello {$who}
```

Display Output

```
$smarty->display("greet.tpl");
```

Debugging templates

Add this line at the top
`{debug}`

Capturing output in php variable

Use `Fetch()` function
`$output =`
`$smarty->fetch("greet.tpl");`
`echo $output;`

Processing indexed array

```
{section name=id loop=$array}
  Current Item : {$array[id]}<br/>
{/section}
```

Processing associated array

```
{student.name}
{student.roll}
{$student.age}
```

Passing associated array

```
$student=array("name"=>"shumi");
$smarty->assign("student",$std);
```

Loop

```
{section name=id loop=$variable}
  element : {$variable[id]}
{/section}
```

```
{foreach item=curltem from=$items}
  element : {$curltem}
{/foreach}
```

Common Modifiers

capitalize, count_characters, cat, count_paragraphs,
count_sentences, count_words, date_format, default, escape, indent,
lower, nl2br, regex_replace, replace, spacyfy, string_format, strip,
strip_tags, truncate, upper, wordwrap

Builtin Functions

capture, config_load, foreach, foreachelse, include, include_php, insert
if, elseif, else, ldelim, rdelim, literal, php, section, sectionelse, strip

Custom Functions

assign, counter, cycle, debug, eval, fetch, html_checkboxes, html_image
html_options, html_radios, html_select_date, html_select_time
html_table, math, mailto, popup_init, popup, textformat

Smarty reserved variables

<code>\$smarty.get</code>	<code>\$smarty.now</code>	<code>\$smarty.version</code>
<code>\$smarty.post</code>	<code>\$smarty.const</code>	<code>\$smarty.ldelim</code>
<code>\$smarty.session</code>	<code>\$smarty.capture</code>	<code>\$smarty.rdelim</code>
<code>\$smarty.session</code>	<code>\$smarty.config</code>	
<code>\$smarty.env</code>	<code>\$smarty.section</code>	
<code>\$smarty.request</code>	<code>\$smarty.foreach</code>	
<code>\$smarty.env</code>	<code>\$smarty.template</code>	

Logic

```
{if $var == condition}
  something to do
{elseif $var == condition}
  something to do
{else}
  something else to do
{/if}
```

Directory Structure

```
- your_script_folder
| - smarty
| | + libs
| | Smarty.class.php
| - templates
| | greet.tpl
| + templates_c [must be 0777]
| + configs
| my_other_scripts.php
| blah_blah_script.php
```

Processing objects

Access Object Methods
`{object->method`
`param1="val"`
`param2="val"}`

Accessing Object Properties
`{object->property}`

Assign method output to variable
`{object->method`
`param1="val"`
`param2="val"`
`assign="storage_var"}`

Output: `{$storage_var}`

Passing objects to template

```
$my_class=new my_object();
$smarty->register_object(
  "object", $my_class);
```

Using modifiers

General Format
`{$variable | modifier:parameter}`
Example
`{$name|count_characters}`

Execute PHP inside template

```
{php}
for($i=0;$i<10;$i++)
echo "Hello";
{/php}
```

