

后盾网 人人做后盾

www.houdunwang.com

Smarty模版

后盾网 2011-2013 v3.0

模版引擎(Template Engine)

- 模版引擎的作用是将视图图界面与程序代码分离开。使程序员专注于程序代码编写，设计师专注于页面的设计。
- 模版引擎的目的，就是要达到上述提到的逻辑分离的功能。它能让程序开发者专注于资料的控制或是功能的达成；而视觉设计师则可专注于网页排版，让网页看起来更具有专业感！因此模版引擎很适合公司的网站开发团队使用，使每个人都能发挥其专长。

Smarty下载

- <http://www.smarty.net/>

特点

- Smarty拥有高效的执行速度
- Smarty是编译行模版引擎，第一次执行时会将模版进行编译，如果模版没有进行过修改则直接读取编译文件，提高执行效率
- Smarty提供了优秀的缓存特性，将页面缓存为静态文件在缓存周期内直接读取缓存文件，让网站性能发生质的飞跃

模版引擎(Template Engine)

安装Smarty模版引擎

- 下载后解压，其中libs为核心目录，包括Smarty核心文件。也就是说有这一个目录就可以了

配置Smarty模版引擎

- `define("ROOT", $_SERVER['DOCUMENT_ROOT']);` //服务器绝对路径
- `define("SMARTY_PATH", "demo/smarty/smarty3/libs");` //smarty目录
- `include ROOT . '/' . SMARTY_PATH . "/Smarty.class.php";` //载入Smarty类库
- `$smarty = new Smarty();` //实例化Smarty对象
- `$smarty->template_dir = "template";` //模版目录
- `$smarty->compile_dir = "temp/compile";` //编译目录
- `$smarty->config_dir = "temp/config";` //配置文件目录
- `$smarty->cache_dir = "temp/cache";` //缓存目录
- `$smarty->left_delimiter = "{";` //开始定界符
- `$smarty->right_delimiter = "}" ;` //结束定界符

Smarty配置

注释

- `{* *}` 注释不会被解析，只是模版中的注释

Smarty函数

- Smarty拥有内置函数及自定义函数特性，通过相应函数完成适当功能如循环数组等，后面详细介绍

属性

- 属性与我们学习的HTML属性相似，数值型属性不用加引号，字符串要加引号，也可以使用Smarty变量

Smarty语法特性

变量

- {\$变量名} 例：`$result = array("user" => "后盾");` 调用如下
`{ $result.user }`

Smarty变量操作

分配变量

- `assign(变量名,值)`通过这一行简单语句，就可向模版中分配值

显示模板

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

例

```
$arr = array("web"=>array("name"=>"后盾网","age"=>30));  
$smarty->assign("arr",$arr);  
$smarty->display("index.html");
```

分配变量与显示模版

Smarty可以方便取得超全局数组、常量、SESSION等等变量，Smarty称为保留变量

get,post,request,session,cookies的访问方式相似，举例如下：

- `{$_smarty.get.id}`
- `{$_smarty.cookies.uid}`
- `{$_smarty.session.uid}`

获得常量

- `{$_smarty.const.webname}`

获得当前时间

- `{$_smarty.now|date_format:" %y-%m-%d %H:%M" }`

注：注意大小写

保留变量

变量调节器

- 变量调节器用于对变量进行个性化处理或修改
- 使用时加|即可
- 变量调节器可以组合使用
- 如果变量调节器有参数用:进行分隔

capitalize	将变量里的所有单词首字大写。
count_characters	计算变量里的字符数
cat	将cat里的值连接到给定的变量后面
count_paragraphs	计算变量里的段落数量
count_sentences	计算变量里句子的数量
count_words	计算变量里的词数
date_format	格式化日期
default	为空变量设置一个默认值

变量调节器

escape	用于html转码,url转码
indent	在每行缩进字符串，默认是4个字符。使用缩进时如果是在HTML中，则需要使用 (空格)来代替
lower	将变量字符串小写
upper	将变量改为大写
regex_replace	寻找和替换正则表达式 { \$b regex_replace:"/baidu/":"百度" }
replace	简单的搜索和替换字符串

变量调节器

`truncate` 从字符串开始处截取某长度的字符.默认是80个.
你也可以指定第二个参数作为追加在截取字符串后面的文本字符串。

`strip_tags` 去除html标签

注：对变量可以使用多个调节器，执行顺序是从左到右的，使用时用|线
隔开就可以了

```
{ $b|upper|truncate:20: "..."} 
```

变量调节器

```
{literal}
```

```
{if $arr.web.age>30 && $arr.web.name=="后盾网"}
```

```
    这是后盾网
```

```
{else}
```

```
    这不是后盾网
```

```
{/if}
```

```
{/literal}
```

- ⑩ 如果以literal函数包住的内容，那么Smarty引擎对此内容不做任何处理，原样显示出来。

将内容当做文本处理

```
{if $arr.web.name== “后盾网” && address= “北京” }
```

这是后盾网

```
{else}
```

这不是后盾网

```
{/if}
```

- ⑩ if 语句和 php 中的 if 语句一样灵活易用，并增加了几个特性以适宜模板引擎. if 必须于 /if 成对出现. 可以使用 else 和 elseif 子句. 可以使用以下条件修饰词：eq、ne、neq、gt、lt、lte、le、gte、ge、is even、is odd、is not even、is not odd、not、mod、div by、even by、odd by、==、!=、>、<、<=、>=. 使用这些修饰词时必须和变量或常量用空格隔开.

条件函数

```
{foreach from=$arr key=key item=value}
```

```
    {$value}
```

```
{/foreach}
```

- Key 键名
- Item 值
- From 数组名

遍历多维数组

```
    {foreach key=key item=value from=$arr}
```

```
        {foreach key=k2 item=v2 from=$value}
```

```
            {$k2}<br/>{$v2|truncate:2:""}<br/>
```

```
        {/foreach}
```

```
    {/foreach}
```

循环函数

```
{section loop=$rows name=name start=1 max=10 step=3}

    {$rows[name].id} {$rows[name].username}<br/>

{/section}
```

说明

- 模板的 section 用于遍历数组中的数据. section 标签必须成对出现. 必须设置 name 和 loop 属性. 名称可以是包含字母、数字和下划线的任意组合. 可以嵌套但必须保证嵌套的 name 唯一. 变量 loop (通常是数组)决定循环执行的次数. 当需要在 section 循环内输出变量时, 必须在变量后加上中括号包含着的 name 变量. sectionelse 当 loop 变量无值时被执行.

复杂的循环输出函数


```
{section loop=$row name=n}
    {if $smarty.section.n.first}
        <h1>{$row[n].title}</h1>第1个记录
    {else}
        {$row[n].title}
    {/if}
{/section}
```

Section的first使用

```
{section loop=$row name=n}

    {if $smarty.section.n.index<=3}

        <h1>{$row[n].title}</h1>前3个记录

    {else}

        {$row[n].title}

    {/if}

{/section}
```

Section的index使用

```
{include file="top.tpl" assign=b}
```

载入模板文件top.tpl，如果配置assign属性将把的模板内容存到变量中

包含模板

在smarty模板引擎的libs\plugins目录中创建以modifier.开始的变量调节器文件

Modifier.substr.php

```
Function smarty_modifier_substr($string,$len){  
    return mb_substr($string,0,$len," utf8" );  
}
```

自定义变量调节器

在smarty模板引擎的libs\plugins目录创建以function. 开始的函数文件

function.js.php

```
function smarty_function_js ($param,&$smarty){
```

```
$file = $param[ 'file' ];
```

```
return " <script type= 'text/javascript'  
      src= '$file' ></script>" ;
```

```
}
```

```
<js file= "jquery.js" />
```

创建函数

在smarty模板引擎的libs\plugins目录中创建以block. 开始的块文件
block.student.php

```
function smarty_block_student($params, $content, &$smarty)
{
    $cid = $params['cid']; $row = $params['row'];
    $sql = "select * from stu where cid=$cid order by sid limit $row";
    $data = query($sql); $str="";
    foreach($data as $v){
        $c = $content;
        foreach($v as $m=>$n){
            $c=str_replace("[\${field}.${m}]", $n, $c);
        }
        $str.=$c;
    }
    return $str;
}
```

模板文件调用：

```
{student row="2" cid="3"}
```

```
<tr>
```

```
<td>[ ${field.stuname} ]</td>
```

```
<td>[ ${field.sex} ]</td>
```

```
</tr>
```

```
{/student}
```

创建块标签


```
$smarty->cache_dir = "cache" ; //缓存目录
```

```
$smarty->caching = true; //是否缓存
```

```
$smarty->cache_lifetime = 3; //缓存时间
```

```
$smarty->clear_cache(); //清除缓存数据
```

缓存控制

1. `$smarty-> caching = true;`//开启缓存
2. `$smarty->cache_lifetime = 3;`//缓存时间
3. `//如果缓存失效，重新获取数据`
4. `if(!$smarty->is_cached("index.html" ,$_SERVER['REQUEST_URL'])){`
5. `$date = time();`
6. `$smarty->assign("date" , $date);`//分配变量
7. `}`
8. `$this->display("index.html" ,$_SERVER['REQUEST_URI']);`

缓存控制

```
$smarty->assign("time", $time);  
  
$smarty->register_block("nocache", "nocache", false);  
  
function nocache($params, $content, &$smarty) {  
    return $content;  
}
```

局部缓存
