

34. 模版的一些杂项

学习要点：

1. 原样输出
2. 模版注释
3. 标签扩展

本节课我们来学习一下模版中最后的一些杂项知识点。

一. 原样输出

1. 有时，我们需要输出类似模版标签或语法的数据，这时会被模版解析；
2. 此时，我们就使用模版的原样输出标签`{literal}`；

```
{literal}
变量标签形式：{$name}
{/literal}
```

二. 模版注释

1. 对于在 HTML 页面中的标签，用 HTML 注释是无效的，需要模版定义的注释；

```
{//$name}
{/*$name*/}
{/*
```

多行注释

```
*/}
```

2. 注释和`{`符号之间不能有空格，否则无法实现注释隐藏；
3. 生成编译文件后，注释的内容会自动被删除，不会显示；

三. 标签扩展

1. 标签库分为内置和扩展标签，内置标签库是 Cx 标签库，就是我们一直用的；
2. 标签库源文件在：根目录下 `thinkphp/library/think/template/taglib`；
3. 其中 `TagLib.php` 是标签解析基类，`Cx.php` 是标签库解析类，继承自 `TagLib`；
4. 通读源代码，我们发现，基本都是我们之前所学习的标签，用法也很简单；

```
{eq name='xxx'}yyy{/eq}
```

5. 因为 Cx 是内置标签，使用的时候是非常简洁的形式，如果是扩展标签则如下格式：

```
{cx:eq name='xxx'}yyy{/cx:eq}
```

6. 如果自己定义一个扩展的标签库，可以按照 `Cx.php`，在它旁边创建 `Html.php`：

```
class Html extends TagLib
{
    //定义标签列表
```

```
protected $tags = [  
    //标签定义： attr 属性列表 close 是否闭合(0 或者 1，默认 1)  
    'font'      =>    ['attr' => 'color,size', 'close'=>1]  
];  
  
//闭合标签  
public function tagFont($tag, $content)  
{  
    $parseStr = '<span style="color: ' . $tag['color'] .  
';font-size:' . ($tag['size'] * 10) . 'px">' . $content . '</span>';  
    return $parseStr;  
}  
  
}
```

7. 对于，扩展标签，我们需要在模版中引入这个标签，并使用扩展标签；

```
{taglib name="html"}  
{html:font color='red' size='10'}  
我是 ThinkPHP  
{/html:font}
```

8. 如果想把这个扩展标签库移到应用目录区，比如：application\taglib；

9. 我们这个时候，需要在 template.php 配置一下预加载标签；

```
// 预先加载的标签库  
'taglib_pre_load'      =>    'app\taglib\Html',
```

10. 最后，需要更改一下 Html.php 的命名空间；

```
namespace app\taglib;
```