

代码与视图模板的分离技巧 PDF转换可能丢失图片或格式，
建议阅读原文

https://www.100test.com/kao_ti2020/295/2021_2022__E4_BB_A3_E7_A0_81_E4_B8_8E_E8_c67_295673.htm 需要什么样的mvc 在基于B/S的应用程序开发中，从基本的技术分工上来说就是两大块，一是软件显示界面，另一个是程序逻辑。在N年前的脚本语言时代，无论是asp、php还是jsp，我们基本是都是把这两者柔和在一起的。尽管我们想方设法做好很多函数或者包含文件来努力达到软件的复用，但仍然无法满足多变的用户需求,这主要是因为以前的纯脚本编码方式无法很好支持及应用面向对象(OO)领域中的强大功能。在常见的B/S软件项目中，界面的设计包括html界面、Wap界面及其它由文本字符协议为基本表示的界面等。以我们接触最多的html页面为例子，在做这些界面的时候往往需要美工先使用photoshop或fireworks等图形界面设计工具进行全局设计，然后再使用进行Dreamweaver等html页面制作工具进行加工制作。而强大的程序逻辑及后台处理都是由服务器端程序完成，这些程序具有较高的稳定性，其开发工具如JBuilder、Eclipse等对View层的界面无法很好的支持(当然那些用记事本写界面的应用除外)，这就使得很多MVC框架的设计都无法两全齐美。随着Struts等MVC框架的陆续出现及应用，我们的视图层变得越来越简单。然而，仍然不时需要在页面中插入Java代码，比如另人讨厌的@。这些MVC框架大多提供了一系列的自定义标签，当自定义标签越来越复杂的时候，对于程序员简单的同时，却把咱们的美工及页面制作人员给吓坏了。页面及程序如何分离 EasyJWeb作为一个快速Java Web MVC框架，其设计

目标不尽是要简化软件开发人员的代码书写工作，更是要能方便界面设计人员的工作。EasyJWeb通过基于模板的方式，建立在强大而简单的模板引擎Velocity基础上，并对Velocity作了一定的扩充。Velocity是一个基于java的模板引擎（template engine），它允许任何人仅仅简单的使用模板语言（template language）来引用由java代码定义的对象。作为一个比较完善的模板引擎，Velocity的功能是比较强大的，但强大的同时也增加了应用复杂性。理论上你可以在EasyJWeb模板使用所有Velocity的脚本及功能，但我们不推荐你在界面模板中使用过多过复杂的脚本表达方式，在万不得已的情况下，不要在界面模板中加入任何复杂的逻辑，更不要在界面模板中加入变量声明、逻辑运算符等等。在EasyJWeb中，我们提供了四条基本的模板脚本语句，基本上就能满足所有应用模板的要求。这四条模板语句很简单，可以直接由界面设计人员来添加。在当前很多EasyJWeb的应用实践中，我们看到，所有界面模板中归纳起来只有下面五种简单模板脚本语句即可实现：

- 1、`!obj` 直接返回对象结果。
- 2、`#if(!obj) #else #end` 判断语句
- 3、`#foreach($info in $list) $info.someList #end` 循环读取集合list中的对象，并作相应的处理。
- 4、`#macro(macroName)#end` 脚本函数(宏)调用，不推荐在界面模板中大量使用。
- 5、包含文件`#inlude("模板文件名")`或`#parse("模板文件名")` EasyJWeb另外提供了很多工具类，通过这些工具类可以大大简化模板的书写。比如CommUtil中的toRowChildList，TagUtil中的options等。通过使用Velocity模板脚本，可以使得视图中除了与数据显示有关的内容以外，不再有其它任何与特定语言相关的标记或符号。由于Velocity

模板对界面不具备任何破坏性，因此就算添加了一系列的模板标记后，美工及页面制作人员不用作任何额外的特殊处理，就能看到纯天然的界面视图。另外还能通过半小时就能学会了解Velocity脚本的用法，根据客户的审美观、习惯乃至心情灵活的更改系统界面。方便他人即是方便自己 EasyJWeb之所以提倡页面模板及程序分离，一个重要原因就是解放程序员。给他人方便就等于给自己方便，EasyJWeb中包含了这样的生活哲学。因此，作为程序员，接触一段时间EasyJWeb后，你会发现除了让视图界面的人员方便的以外，EasyJWeb最主要关照的还是咱们这些审美观经常被批评，最讨厌那些乱七八糟的html代码，css样式以及患有图片恐惧症的程序员。因此，EasyJWeb让程序员非常轻松与视图层的内容打交道。比如要把一个对象发送给视图层，只需要在Action中使用form.addResult(“ name ”,obj)即可，要把一个对象分成各属性发送给视图层，可以直接使用form.addPo(obj)方法即可；要从视图层表单中读取数据到对象中，只需要form.toPo(obj)即可；EasyJWeb为了处理了数据转换、验证、数据保护等诸多细节。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com