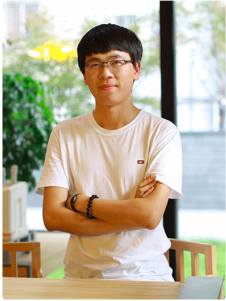


前端开发模板引擎 -- Jade系列之神奇的 mixins 和模板继承

根号三 (/user/1117) 发布在前端开发模板引擎 -- Jade语法浅谈 (/article/column/809) 2015年1月27日 view: 8259 RalphKab



推荐导师：芋头君

大搜车前端负责人，前端乱炖站长，多个知名 app 的开发者

资深全栈，八年前端经验，5 年带团队经验，面人无数

面对面、一对一 针对性剖析

有关成长、面试、入门、学习方法

点此约见（杭州线下或者远程语音）

https://s.souche.com/_S5nA3

在文章任何区域双击即可给文章添加【评注】！浮到评注点上可以查看详情。

☐ 隐藏标注



微信扫码关注公众号

每天为您推送精品文章

1、神奇的 mixins

想必有 less & sass 经验的同学对 mixins 一定都不陌生了，mixins 呢，其实就是方便对一堆代码块的重复利用。在 Jade 里面当然也少不了代码块的复用咯~~

mixins 的定义很简单，就只需要 mixin 代码块名字 即可

```
mixin demo
  p this is a demo
```

调用方式就更简单了，我们需要这样 +代码块名字

```
+demo
```

输出结果非常理想

```
<p>this is a demo</p>
```

什么？你说这都是只能用于内容固定的代码块？没关系，我们来看看内容不固定的时候应该怎么办

```
mixin info(name, skills)
  p my name is #{name}
  ul.skills-box
    each skill in skills
      li #{skill}

+info('sqrththree', ['html', 'css', 'javascript', 'nodejs'])
```

编译结果如下

WRITTEN BY

根号三 (/user/1117)

<http://blog.sqrththree.com/>

TA的新浪微博

<http://weibo.cn/2402130601>

PUBLISHED IN

前端开发模板引擎 -- Jade语法浅谈 (/article/column/809)

简单的介绍一下我对 jade 语法的了解和使用过程中的学到的一些方法。

 订阅 (/article/column/1/rss)

本专栏其他文章

- 前端开发模板引擎 -- Jade系列之神时间:2015-01-17 浏览:8259 赞:2
- 前端开发模板引擎 -- Jade系列之数时间:2015-01-14 浏览:6637 赞:2
- 前端开发模板引擎 -- Jade之标签的时间:2015-01-10 浏览:5447 赞:4
- 前端开发模板引擎 -- Jade之开篇时间:2015-01-08 浏览:8325 赞:1

 赞 2

 (/user/34)

 (/user/1673)

收藏 1

```
<p>my name is sqrtthree</p>
<ul class="skills-box">
  <li>html</li>
  <li>css</li>
  <li>javascript</li>
  <li>nodejs</li>
</ul>
```

什么？你还想要嵌套的 `mixins`，٩ (̄ ̅ ̇ ̈́) 怎么这么多事。。

```
mixin info(name, skills)
  p my name is #{name}
  ul.skills-box
    each skill in skills
      li #{skill}

mixin developersInfo(developer)
  h2 #{developer.name}
  +info(developer.name, developer.skills)

+developersInfo({"name": "sqrtthree", "skills": ['html','css','javascript','nodejs']})
```

编译结果是这个样子的

```
<h2>sqrtthree</h2>
<p>my name is sqrtthree</p>
<ul class="skills-box">
  <li>html</li>
  <li>css</li>
  <li>javascript</li>
  <li>nodejs</li>
</ul>
```

那如果我们遇到不定参的问题呢？

```
mixin info(className, ...skills)
  ul(class=#{className})
    each skill in skills
      li #{skill}

+info('list','html','css','javascript','nodejs')
```

我们来看编译结果

```
<ul class="list">
  <li>html</li>
  <li>css</li>
  <li>javascript</li>
  <li>nodejs</li>
</ul>
```

变量都能传递了，那属性能传递么？当然能咯~

```
mixin attr(text)
  p(class=attributes.class) #{text}

+attr('text')(class="tips")
```

编译结果如下：

```
<p class="tips">text</p>
```

属性传递的时候还有一个语法糖呢。

```
mixin attr(text)
  p&attributes(attributes) #{text}

+attr('demo')(class="tips", id="j-tips", data-id="18")
```

我们来看编译结果是什么样的

```
<p id="j-tips" data-id="18" class="tips">text</p>
```

`mixins` 还有一个高级用法，那就是内联代码块。先上例子

```
mixin code(text)
  p #{text}
  if block
    block
  else
    p no block.

+code('this is a demo')
  p this is a block.
```

编译结果如下：

```
<p>this is a demo</p>
<p>this is a block.</p>
```

从上面的代码中，我们可以看出，我们在调用`+code`的时候给他设置一个代码块 `p this is a block`，那么在 `mixins` 里面，程序会自动把这个代码块的内容传递给 `block` 变量。然后我们可以通过判断进行相应的操作。

2、模板的继承

我们在做整站开发时，通常每一个页面的头部和底部都是一样的，之前我们的做法可能是写完一个之后然后不断的复制粘贴，但是现在我们有了 `Jade` 之后，来看看我们可以怎么做？

在 `Jade` 里面，我们通过 `和` 这两个关键字实现模板之间的继承，一个代码块可以看成是 `Jade` 中的一个 `block`，我们将在子模板中实现

我们先看一个小例子，定义一个简单的代码块

```
block demo
  p this is a demo.
```

默认情况下，代码块的内容是会被输出的

```
<p>this is a demo.</p>
```

在同一个文件中，我们可以直接通过这种方式进行调用

```
block demo
  p this is a demo.
block demo
block demo
block demo
```

结果是这样的

```
<p>this is a demo.</p>
<p>this is a demo.</p>
<p>this is a demo.</p>
<p>this is a demo.</p>
```

那么如果是不同文件之间，如何进行调用呢？

比如我们有一个 `layout.jade` 文件，内容是这样的。

```
doctype html
html
  head
    title this is a demo.
  body
    h1 demo.

    block cont
```

然后呢，我们有一个 `demo.jade` 文件是这样的

```
extends layout // 这里的 layout 指的是需要继承的模板文件

block cont
  h2 Look, this is a demo.
  p this is a paragraph.
```

然后通过命令行工具编译 `demo.jade` 文件，结果是神奇的

```
<!DOCTYPE html>
<html>
  <head>
    <title>this is a demo.</title>
  </head>
  <body>
    <h1>demo.</h1>
    <h2>Look, this is a demo.</h2>
    <p>this is a paragraph.</p>
  </body>
</html>
```

怎么样？是不是觉得很方便呢？

模板的继承解决了多个文件之间代码复用的问题，其实上面我们提到的页头和页脚我们用模板包含的方式可能更合适一些，下面我们来看一看模板的包含。

`includes` 允许你静态包含一段 `Jade`，或者别的存放在单个文件中的东西比如 `css` `html`。

比如我们有一个 `footer.jade` 文件，他里面有着这样的内容

```
div#footer
  p copyright.
```

然后我们在我们的 `demo.jade` 文件中这样使用

```
doctype html
html
  head
    title this is a demo.
  body
    h1 Hello, World.

    include footer
```

编译结果是这个样子的

```
CTYPE html>
<html>
  <head>
    <title>this is a demo.</title>
  </head>
  <body>
    <h1>Hello, World.</h1>
    <div id="footer">
      <p>copyright.</p>
    </div>
  </body>
</html>
```

当然，通过 `includes` 引入 `css` 代码也是一样的。大家可以尝试一下，我这里就不举例子了。

本站专栏文章皆为原创，转载请注明出处（带有 前端乱炖 字样）和本文的显式链接(<http://www.html-js.com/article/2596> (<http://www.html-js.com/article/2596>))，本站和作者保留随时要求删除文章的权利！

评论

参与讨论。支持markdown语法

发表评论

LoveHall201314 (/user/6600)

3年前

回复

博主，那我该怎么在js中去获取一个mixin呢？

前端乱炖 (/user/34)

3年前

回复

赞了此文章！