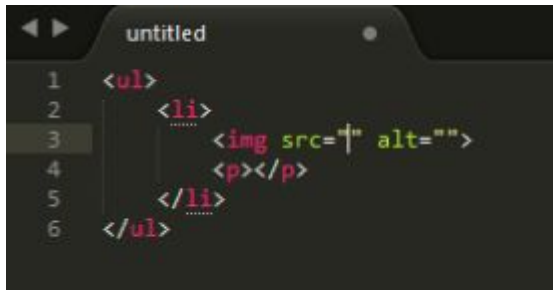


Emmet 前端开发技巧分享

Emmet 和 HTML(emmet 在 sublime 上的效果)

为文本编辑器安装 Emmet 插件，一旦写好所写之后，按一下 **tab** 键就能生成你所请求的代码。以下为几个实例。

1. 输入: `ul>li>img+p`



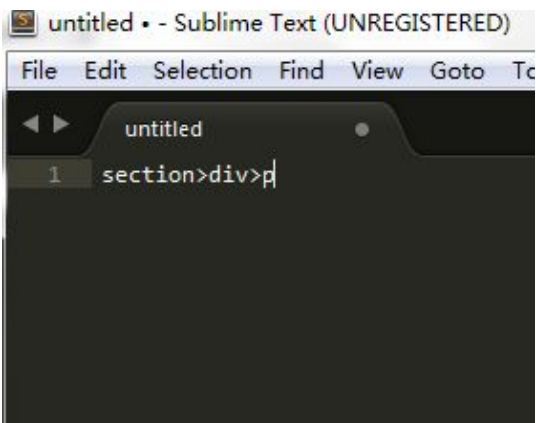
```
1 <ul>
2   <li>
3     <img src="" alt="" />
4     <p></p>
5   </li>
6 </ul>
```

生成:

2. 创建初始文档

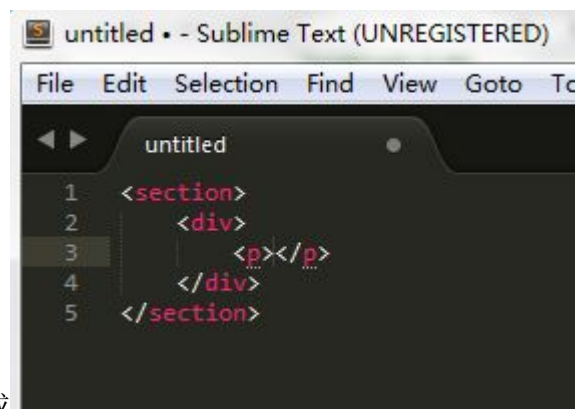
- ① html:5 或!: HTML5 文档类型
- ② html:xt: XHTML 过渡型文档类型
- ③ html:xs: XHTML 严格型文档类型
- ④ html:4t: HTML4 过渡型文档类型
- ⑤ html:4s: HTML4 严格型文档类型

3. 子元素: 使用>运算符生成彼此嵌套的元素



```
1 section>div>p|
```

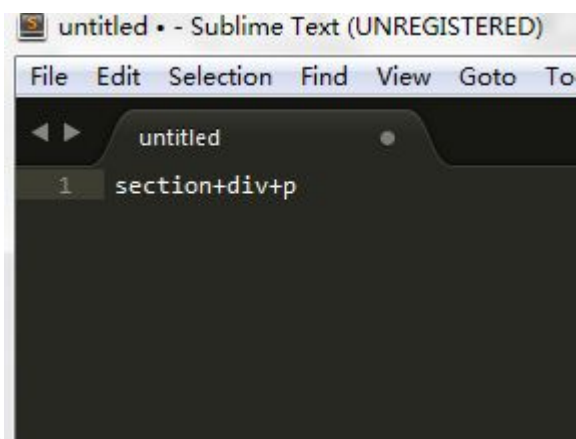
输入



```
1 <section>
2   <div>
3     <p></p>
4   </div>
5 </section>
```

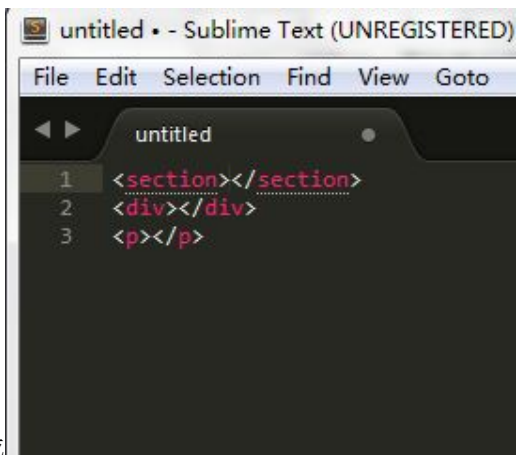
生成

4. 相邻元素+: 使用+运算符可以用来生成彼此相邻的元素:



```
1 section+div+p|
```

输入



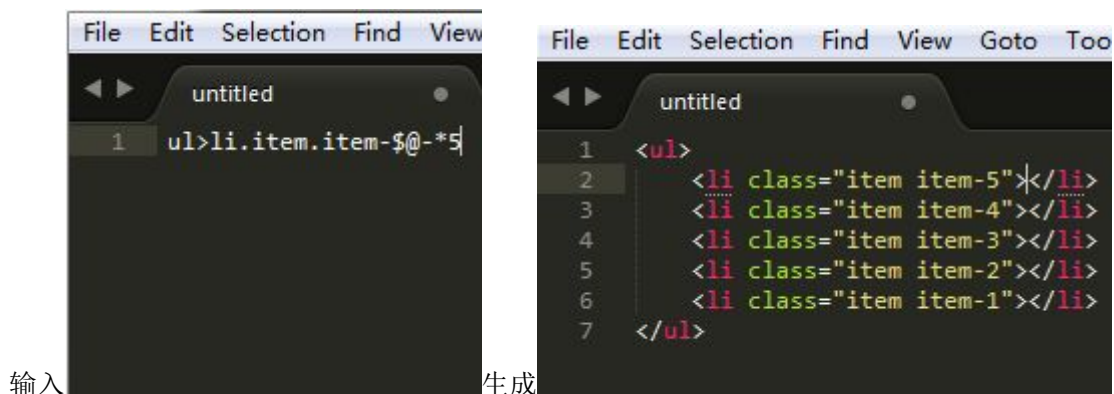
```
1 <section></section>
2 <div></div>
3 <p></p>
```

生成

5. 返回上一层^: 使用^运算符, 可以让你的代码返回上一层。当你使用>嵌套元素时, 想让后面的回到上一层, 那么这个方法很适用。



6. 乘法*: 一次性生成多个相同的元素, 还可以通过\$符号做递增, 通过\$@-符号做递减, 通过\$@3*5 从第三个开始命名



```
File Edit Selection Find View G
untitled
1 ul>li.item.item-$@-3*5
```

输入

```
File Edit Selection Find View Goto To
untitled
1 <ul>
2   <li class="item item-7"></li>
3   <li class="item item-6"></li>
4   <li class="item item-5"></li>
5   <li class="item item-4"></li>
6   <li class="item item-3"></li>
7 </ul>
```

生成

```
File Edit Selection Find View
untitled
1 ul>li.item.item-$@3*5>a
```

输入

```
File Edit Selection Find View Goto Tools Project Prefer
untitled
1 <ul>
2   <li class="item item-3"><a href=""></a></li>
3   <li class="item item-4"><a href=""></a></li>
4   <li class="item item-5"><a href=""></a></li>
5   <li class="item item-6"><a href=""></a></li>
6   <li class="item item-7"><a href=""></a></li>
7 </ul>
```

生成

7. 组合：更有效的利用嵌套。在 Emmet 中可以通过()将一个块组合在一起

```
File Edit Selection Find
untitled
1 ul>(li>a)*3
```

输入

```
File Edit Selection Find View Goto
untitled
1 <ul>
2   <li><a href=""></a></li>
3   <li><a href=""></a></li>
4   <li><a href=""></a></li>
5 </ul>
```

生成

```
untitled - Sublime Text (O
File Edit Selection Find V
untitled
1 ul>li*3>a
```

输入

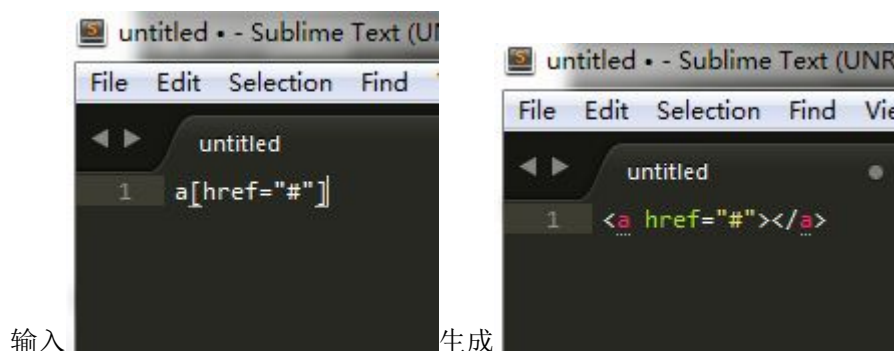
```
File Edit Selection Find View Goto
untitled
1 <ul>
2   <li><a href=""></a></li>
3   <li><a href=""></a></li>
4   <li><a href=""></a></li>
5 </ul>
```

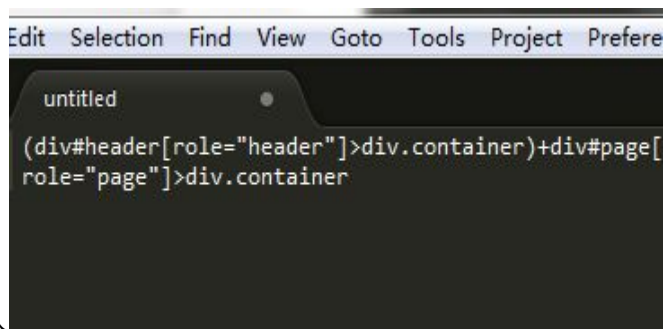
生成



8. 快速添加类名、ID、文本和属性:

- ① 使用 E#ID 添加 ID 名
- ② 使用 E.class 添加类名
- ③ 使用 E[attr] 添加属性
- ④ 使用 E{text} 添加文本



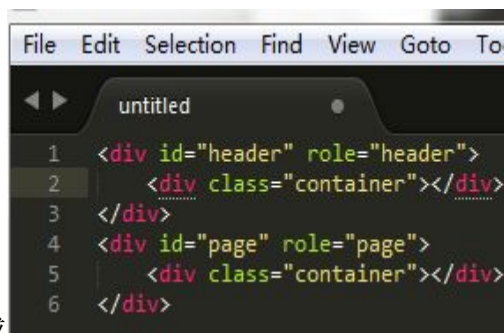


```
untitled

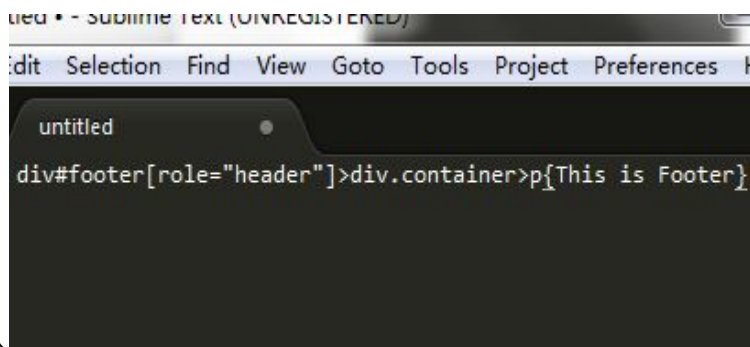
```

输入

生成



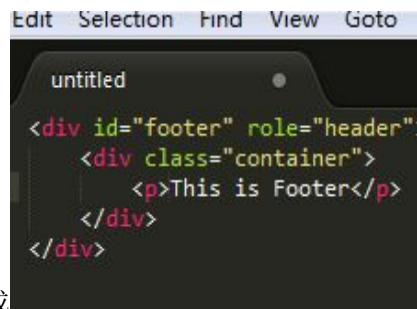
```
untitled
1 <div id="header" role="header">
2   <div class="container"></div>
3 </div>
4 <div id="page" role="page">
5   <div class="container"></div>
6 </div>
```



```
untitled
div#footer[role="header"]>div.container>p[This is Footer]
```

输入

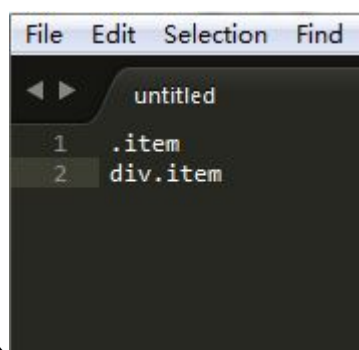
生成



```
untitled
<div id="footer" role="header">
  <div class="container">
    <p>This is Footer</p>
  </div>
</div>
```

9. **省略标签名:** 在 Emmet 中可以省略标签名, 默认情况下, 如.item 和 div.item 起到的作用是一致的
在实际中还有几种情况:

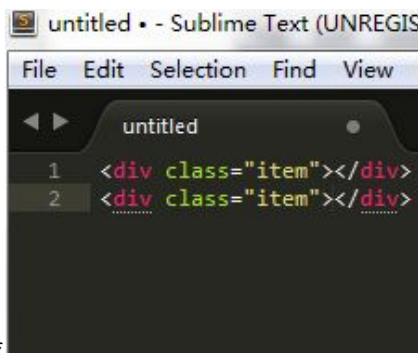
- ① ul 和 ol 中输入指的是 li
- ② table、tbody、thead 和 tfoot 指的是 tr
- ③ tr 中指的是 td
- ④ select 和 optgroup 指的是 option



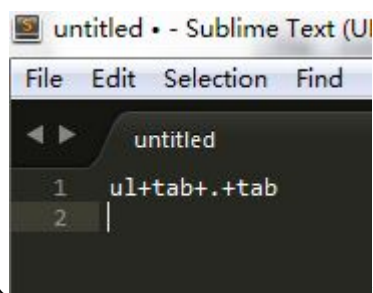
```
untitled
1 .item
2 div.item
```

输入

生成



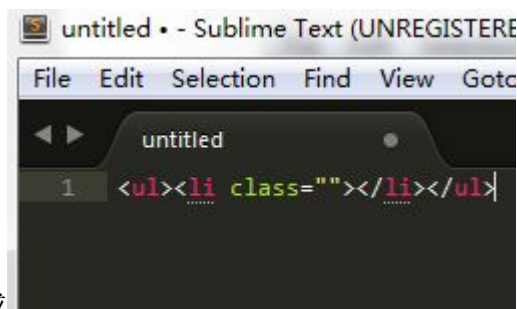
```
untitled
1 <div class="item"></div>
2 <div class="item"></div>
```



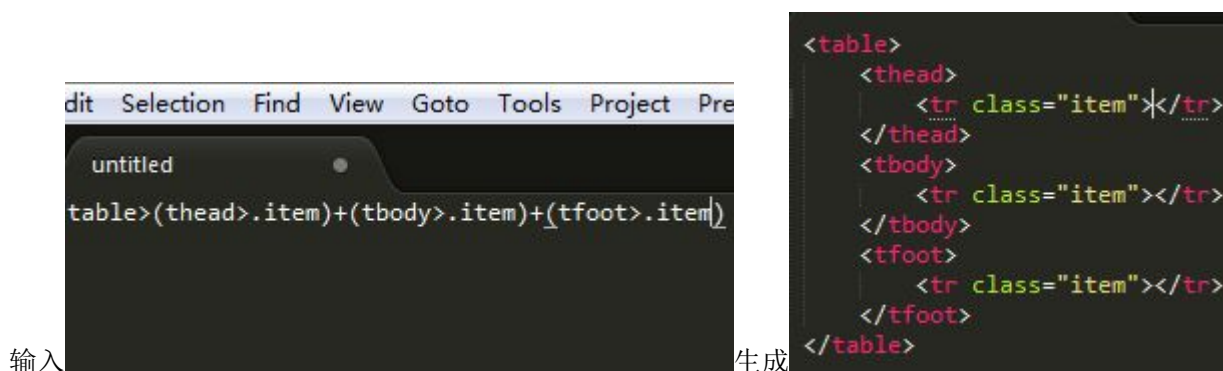
```
untitled
1 ul+tab+.+tab
2 |
```

输入

生成



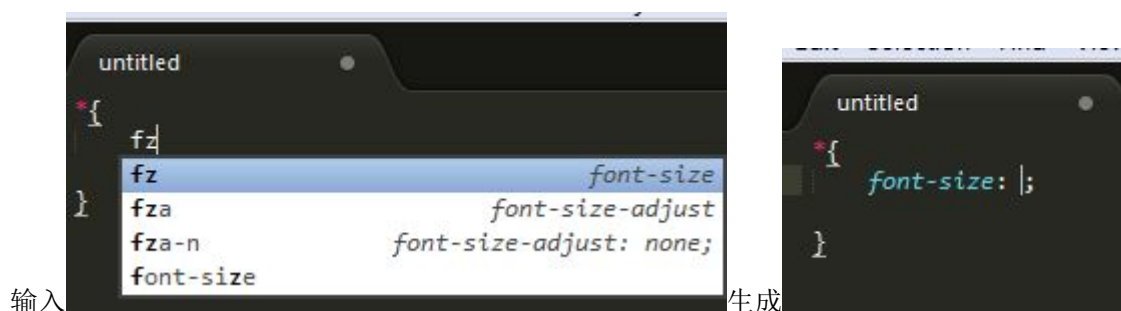
```
untitled
1 <ul><li class=""></li></ul>
```



10. Emmet 和 CSS (emmet 在 sublime 上的效果)

1) 属性: Emmet 定义了所有已知的 CSS 属性和缩写。

所以 border-bottom 缩写是 bdb, border-top 缩写是 bdt。正如下面的示例 font-size 缩写是 fz:



2) 属性值: fz18 将输出 font-size:18px h10p+m5e



3) 单位: CSS 经常使用的单位有 px、em、rem、%、ex。那么这些单位在 Emmet 中都是可以使用的。在 Emmet 中每一个单位都有其缩写形式:

- ① px → 默认
- ② p → %
- ③ e → em
- ④ r → rem
- ⑤ x → ex



4) 多个单位: CSS 中的某些属性, 比如 margin, 允许多个值。在 Emmet 中要做到这一点, 只需要每个值之间使用破折号(-)。



5) 颜色: 在 Emmet 中使用 # 前缀, 后面紧跟颜色值, 但不同的字符数将会输出不同的十六进制代码。

#1 → #111111

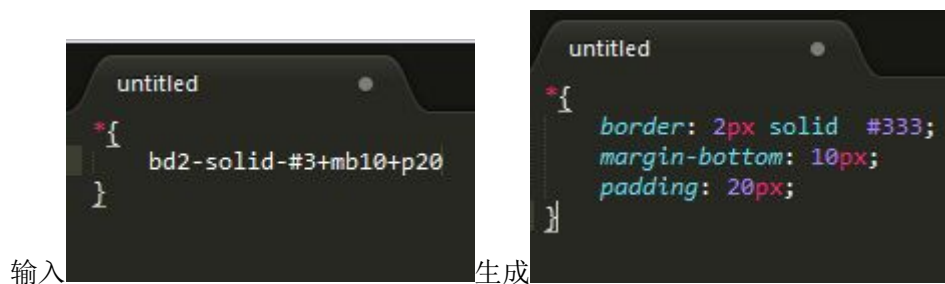
#E0 → #e0e0e0

#FC0 → #FFCC00



!important 尽管在 CSS 中 !important 并不经常使用, 但在 Emmet 也带有一定的缩写。添加 ! 就可以自动生成: p0!

6) **多属性**：可以使用加号+运算符来创建多个属性。



11. 其他

(1).模糊匹配

如果有些缩写你拿不准，Emmet 会根据你的输入内容匹配最接近的语法，比如输入 `ov:h`、`ov-h`、`ovh` 和 `oh`，生成的代码是相同的：

(2).供应商前缀：如果输入非 W3C 标准的 CSS 属性，Emmet 会自动加上供应商前缀，

比如 `trs -super-foo -w-super-foo -wm-super-foo`。

w 表示 -webkit-

m 表示 -moz-

s 表示 -ms-

o 表示 -o-

(3). 渐变：lg(left, #fff 50%, #000)