

HTML 事半功倍之 Emmet

HTML 事半功倍之 Emmet

概述

特别说明

基本用法

快速生成HTML结构语法

生成标签

生成多个相同标签

父子级关系标签（子元素）

兄弟关系标签

返回上层

带有类名或者ID名字的标签

自动计数的或者说有顺序的标签

生成标签的内容

自定义属性

分组操作

CSS 缩写

只需输入 CSS 标签与属性值的首字母

一些用 `-` 连接的 CSS 样式和属性值，都可以取首字母

输出默认的 CSS 样式，用 `+` 操作符

参数别名

使用别名代替完整的单位

颜色值

`#` 是一个值分离器

在编写 Emmet 代码时需要添加 `!important`

有多个属性值的 CSS 属性

CSS3 渐变

进阶用法

自动计数

模拟文本/随机文本

复杂嵌套例子

参考资料

概述

Emmet是什么?

Emmet is a plugin for many popular text editors which greatly improves HTML & CSS workflow.

Emmet 是一个可用在许多流行文本编辑器上的极大简化 HTML 和 CSS 工作流程的插件。

前身是 Zen coding，他使用仿 CSS 选择器的语法来生成代码，极大提高了编写 HTML/CSS 的效率，之后改名为 Emmet，但是随之而来的改变不仅限于名字，还增加了许多新的特性。

特别说明

以下所有的语法，都不能出现空格，除非是在特定的括号中。因为 Emmet 在遇到空格时，就认为已经结束，会停止解析。

所以在生成目标结果前，一定不能格式化缩写。

基本用法

快速生成HTML结构语法

生成标签

!! 输入语法后名按 **tab** 键即可。

html:5	=> 将生成html5标准的包含body为空基本dom
div	=> <code><div> </div></code>
foo	=> <code><foo> </foo></code>
a:mail	=> <code></code>
a:link	=> <code></code>
base	=> <code><base href=""></code>
br	=> <code>
</code>

link	=> <link rel="stylesheet" href="">
script:src	=> <script src=""></script>
form:get	=> <form action="" method="get"></form>
label	=> <label for=""></label>
input	=> <input type="text">
inp	=> <input type="text" name="" id="">
input:hidden	=> <input type="hidden" name=""> input:h即可
input:email	=> <input type="email" name="" id="">
input:password	=> <input type="password" name="" id="">
input:checkbox	=> <input type="checkbox" name="" id="">
input:radio	=> <input type="radio" name="" id="">
select	=> <select name="" id=""></select>
option	=> <option value=""></option>
bq	=> <blockquote></blockquote>
btn	=> <button></button>
btn:s	=> <button type="submit"></button>
btn:r	=> <button type="reset"></button>

生成多个相同标签

加上 `*N`，N 表示重复次数。

```
div*4

<!-- => -->
<div></div>
<div></div>
<div></div>
<div></div>
```

父子级关系标签（子元素）

可以使用 `>` 运算符将元素嵌套在彼此内部，比如 `ul>li`。

```
ul>li

<!-- => -->

<ul>
  <li></li>
</ul>
```

兄弟关系标签

使用 `+` 运算符将元素以相同层级放在同一父元素上。

```
div+p+btn

<!-- => -->

<div></div>
<p></p>
<button></button>
```

返回上层

使用 `^` 运算符，您可以爬上树的一个层次，并更改上下文。

```
div+div>p>span+em^bq

<!-- => -->

<div></div>
<div>
  <p><span></span><em></em></p>
  <blockquote></blockquote>
</div>
<!-- blockquote 在和 p 标签同级下插入 -->
```

当然了，`^` 也可以多个并用，有几个 `^` 就返回几层。

```
div+div>p>span+em^^bq

<!-- => -->
<div></div>
<div>
    <p><span></span><em></em></p>
</div>
<blockquote></blockquote>
<!-- 有两个 ^^, blockquote 会插入到和两个 div 同级 -->
```

特别说明：`^` 返回上层因为比较复杂，所以不太建议使用，因为有这个思考的时间，不如多写几次。

带有类名或者ID名字的标签

直接写标签名，如果是类则用 `.`，如果是 ID 则用 `#` 连接。比如：`li.demo` 或者 `div#box`。

```
div#demo

<!-- => -->
<div id="demo"></div>

div.class

<!-- => -->
<div class="class"></div>
```

自动计数的或者说有顺序的标签

可以用自增符合 `$`。

```
div.demo$*3
```

```
<!-- => -->
```

```
<div class="demo1"></div>
```

```
<div class="demo2"></div>
```

```
<div class="demo3"></div>
```

生成标签的内容

可以用 `{}` 表示。可以与 `$`、`lorem` 搭配使用。

```
div{demo$}*3
```

```
<!-- => -->
```

```
<div>demo1</div>
```

```
<div>demo2</div>
```

```
<div>demo3</div>
```

```
div>lorem3
```

```
<!-- => -->
```

```
<div>Lorem, ipsum dolor.</div>
```

自定义属性

使用 `[attr1='' attr2='']`。

```
input[type=radio]
```

```
<!-- => -->
```

```
<input type="radio">
```

需要说明的是：

1. 方括号内的属性数量不限，如果你喜欢，可以一直加；

2. 如果不是默认属性值的话会生成插入占位，比如：`div[title colspan]` 会变成 `<div title="" colspan="">` 前提是编辑器支持这样的写法；
3. 属性值可以使用单引号或者双引号都可以；
4. 属性值如果不包含空格可以省去引号。

分组操作

使用 `()` 包裹使之成为一个整体。

```
div>(ul>li>span)+strong

<!-- => -->
<div>
  <ul>
    <li><span></span></li>
  </ul>
  <strong></strong>
</div>
```

CSS 缩写

Emmet 有一个特殊的 CSS 解析器，可以将这些缩写扩展成一个完整的 CSS 属性。

只需输入 **CSS 标签与属性值的首字母**

```
/* fl → */
float: left;

/* poa → */
position: absolute;
```

一些用 `-` 连接的 **CSS 样式和属性值**，都可以取首字母

```
/* whscbs → */  
whitewhite-space-collapse:break-strict;
```

输出默认的 **CSS** 样式，用 **+** 操作符

```
/* bg+ → */  
background: #fff url() 0 0 no-repeat;
```

参数别名

Emmet 为几个常用的单位值提供了别名。

```
/* p → */  
%;  
  
/* e → */  
em;  
  
/* x → */  
px;
```

使用别名代替完整的单位

```
/* m10p30e5x → */  
margin: 10% 30em 5ex;
```

颜色值

Emmet 支持十六进制的颜色值。


```
/* c#3 → */  
color: #333;  
  
/* #1 → */  
#111111;  
  
/* #e0 → */  
#e0e0e0;  
  
/* #fc0 → */  
#ffcc00 ;
```

是一个值分离器

不需要使用连字符分隔它。

```
/* bd5#0s → */  
border: 5px #000 solid;
```

在编写 Emmet 代码时需要添加 `!important`

在缩写的末尾添加 `!`。

```
/* fl! → */  
float: left !important;
```

有多个属性值的 CSS 属性

在编写时只需在属性值之间添加 `-`。

```
/* p4-6-8 → */  
padding: 4px 6px 8px;
```

CSS3 渐变

```

/* lg(left #f90 20% red) → */
background-image: -webkit-gradient(linear, 0 0, 100% 0);
background-image: -webkit-linear-gradient(left #f90 20% red);
background-image: -moz-linear-gradient(left #f90 20% red);
background-image: -o-linear-gradient(left #f90 20% red);
background-image: linear-gradient(left #f90 20% red);

```

进阶用法

自动计数

如果生成两位数，则使用两个连续的使用 `@` 修饰符（`@-` 表示降序，`@+` 表示升序），比如 `$@-`、`@N` 可以改变起始值，需要注意的是如果配合升降序使用的话 N 是放到 `+` / `-` 符号后。

```
ul>li*5.demo$@-5
```

```
<!-- => -->
```

```
<ul>
```

```
<li class="demo9"></li>
```

```
<li class="demo8"></li>
```

```
<li class="demo7"></li>
```

```
<li class="demo6"></li>
```

```
<li class="demo5"></li>
```

```
</ul>
```

```
ul>li.demo$@-5*5
```

```
<!-- => -->
```

```
<ul>
```

```
<li class="demo9"></li>
```

```
<li class="demo8"></li>
```

```
<li class="demo7"></li>
```

```
<li class="demo6"></li>
```

```
<li class="demo5"></li>
```

```
</ul>
```

模拟文本/随机文本

在开发时经常要填充一些文本内容占位，Emmet 内置了 Lorem Ipsum 功能来实现生成随机文本。loremN 或者 lipsumN，N 表示单词数量，可在生成 HTML 结构时搭配使用。

```
(p>lorem6)*3

<!-- => -->
<p>Lorem ipsum dolor sit amet consectetur.</p>
<p>Asperiores laborum voluptates est. Quia, adipisci?</p>
<p>Non fuga modi assumenda ipsam possimus.</p>
```

复杂嵌套例子

```
div#sidebar>(p.header+div.content>ul>(li>a[href='/api/categories/'
data-index='$' ]>lorem3)*3)^div#main

<!-- => -->
<div id="sidebar">
  <p class="header"></p>
  <div class="content">
    <ul>
      <li><a href="/api/categories/1" data-index="1">Lorem, ipsum
dolor.</a></li>
      <li><a href="/api/categories/2" data-index="2">Perspiciatis,
aliquam quaerat!</a></li>
      <li><a href="/api/categories/3" data-index="3">Incidunt,
officia aliquid.</a></li>
    </ul>
  </div>
</div>
<div id="main"></div>
```

参考资料

- [Emmet 官网](#);

- [Emmet 语法总结 - 知乎 \(zhihu.com\)](#)
- [Cheat Sheet \(emmet.io\)](#)