

# css第二天

## ☆☆☆一、emmet语法

### 1、简介

Emmet语法的前身是Zen coding,它使用缩写,来提高html/css的编写速度, Vscode内部已经集成该语法。

快速生成HTML结构语法

快速生成CSS样式语法

### 2、☆☆快速生成HTML结构语法

- 生成标签 直接输入标签名 按tab键即可 比如 div 然后tab 键, 就可以生成 `<div></div>`
- 如果想要生成多个相同标签 加上 \* 就可以了 比如 `div*3` 就可以快速生成3个div
- 如果有父子级关系的标签, 可以用 > 比如 `ul > li` 就可以了
- 如果有兄弟关系的标签, 用 + 就可以了 比如 `div+p`
- 如果生成带有类名或者id名字的, 直接写 `.demo` 或者 `#two` tab 键就可以了

```
<!-- .demo -->
| <div class="demo"></div>
•
<!-- #two -->
| <div id="two"></div>
```

- ☆如果生成的div 类名是有顺序的, 可以用 自增符号 \$

15	div{\$}*3
----	-----------

🔑 div{\$}\*3

Emmet 缩写 ×

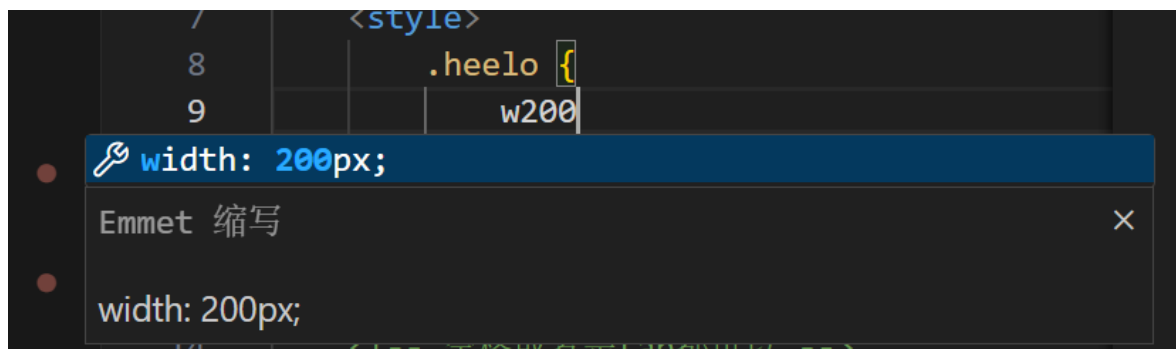
- `<div>1</div>`
- `<div>2</div>`
- `<div>3</div>`

- 如果想要在生成的标签内部写内容可以用 {} 表示

### 3、☆☆快速生成CSS样式语法

CSS 基本采取简写形式即可

比如 `w200` 按tab 可以生成 `width: 200px;`



比如 `lh26px` 按 `tab` 可以生成 `line-height: 26px;`

## 4、快速格式化代码

Vscode 快速格式化代码: `shift+alt+f`

也可以设置 当我们 保存页面的时候自动格式化代码:

- 1) 文件 -----> 【首选项】 -----> 【设置】 ;
- 2) 搜索 `emmet.include`;
- 3) 在 `settings.json` 下的 【工作区设置】 中添加以下语句:

```
"editor.formatOnType": true,  
"editor.formatOnSave": true
```

## 二、css的复合选择器

### 1、什么是复合选择器？

在 CSS 中，可以根据选择器的类型把选择器分为 **基础选择器** 和 **复合选择器**，复合选择器是建立在基础选择器之上，对基本选择器进行组合形成的。

复合选择器是由两个或多个基础选择器，通过不同的方式组合而成的，可以更准确、更高效的选择目标元素（标签）

常用的复合选择器包括：**后代选择器**、**子选择器**、**并集选择器**、**伪类选择器**等等

### 2、后代选择器 (重要)(选择所有后代元素)

定义：

后代选择器又称为包含选择器，可以**选择父元素里面子元素**。其写法就是把外层标签写在前面，内层标签写在后面，中间用空格分隔。当标签发生嵌套时，内层标签就成为外层标签的后代。

语法：

```
元素1 元素2 { 样式声明 }
```

上述语法表示选择元素 1 里面的所有元素 2 (后代元素)。

语法说明：

- 元素1 和 元素2 中间用空格隔开
- 元素1 是父级，元素2 是子级，最终选择的是元素2
- 元素2 可以是儿子，也可以是孙子等，只要是元素1 的后代即可
- 元素1 和 元素2 可以是任意基础选择器

例子：

```
ul li { 样式声明 } /* 选择 ul 里面所有的 li 标签元素 */
```

### 3、子选择器 (重要)(只选择最近一级)

定义:

子元素选择器 (子选择器) 只能选择作为某元素的最近一级子元素。

(简单理解就是选亲儿子元素)

语法:

```
元素1 > 元素2 { 样式声明 }
```

上述语法表示选择元素1 里面的所有直接后代(子元素) 元素2。

语法说明:

- 元素1 和 元素2 中间用 大于号 隔开
- 元素1 是父级, 元素2 是子级, 最终选择的是元素2
- 元素2 必须是亲儿子, 其孙子、重孙之类都不归他管. 你也可以叫他 亲儿子选择器

例子:

```
div > p { 样式声明 } /* 选择 div 里面所有最近一级 p 标签元素 */
```

### 4、并集选择器 (重要)(选择多组标签)

定义:

并集选择器可以选择多组标签, 同时为他们定义相同的样式, 通常用于集体声明。并集选择器是各选择器通过英文逗号 (,) 连接而成, 任何形式的选择器都可以作为并集选择器的一部分。

语法:

```
元素1, 元素2 { 样式声明 }
```

上述语法表示选择元素1 和 元素2。

语法说明:

- 元素1 和 元素2 中间用逗号隔开
- 逗号可以理解为和的意思
- 并集选择器通常用于集体声明

例子:

```
ul, div { 样式声明 } /* 选择 ul 和 div 标签元素 */
```

### 5、☆伪类选择器

定义:

伪类选择器用于向某些选择器添加特殊的效果, 比如给链接添加特殊效果, 或选择第1个, 第n个元素。

语法:

伪类选择器书写最大的特点是用冒号 (:) 表示, 比如 :hover、:first-child。

## 6、☆ 链接伪类选择器

定义:

伪类选择器用于向某些选择器添加特殊的效果, 比如给链接添加特殊效果, 或选择第1个, 第n个元素。

语法:

伪类选择器书写最大的特点是用冒号 (:) 表示, 比如 :hover、:first-child。

**a:link** 没有点击过的(访问过的)链接

**a:visited** 点击过的(访问过的)链接

**a:hover** 鼠标经过的那个链接

**a:active** 鼠标正在按下还没有弹起鼠标的那个链接

链接伪类选择器注意事项

为了确保生效, 请按照 LVHA 的循顺序声明 :link - :visited - :hover - :active。

记忆法: love hate 或者 lv 包包 hao。

因为 a 链接在浏览器中具有默认样式, 所以我们实际工作中都需要给链接单独指定样式。

链接伪类选择器实际工作开发中的写法:

```
/* a 是标签选择器 所有的链接 */
a {
  color: gray;
}
/* :hover 是链接伪类选择器 鼠标经过 */
a:hover {
  color: red; /* 鼠标经过的时候, 由原来的 灰色 变成了红色 */
}
```

## 7、:focus 伪类选择器

定义:

:focus 伪类选择器用于选取获得焦点的表单元素。

焦点就是光标, 一般情况  类表单元素才能获取

例子:

```
input:focus {
  background-color: yellow;
}
```

## 复合选择器总结

选择器	作用	特征	使用情况	隔开符号及用法
后代选择器	用来选择后代元素	可以是子孙后代	较多	符号是 <b>空格</b> .nav a
子代选择器	选择最近一级元素	只选亲儿子	较少	符号是 <b>大于</b> .nav>p
并集选择器	选择某些相同样式的元素	可以用于集体声明	较多	符号是 <b>逗号</b> .nav, .header
链接伪类选择器	选择不同状态的链接	跟链接相关	较多	重点记住 a{} 和 a:hover 实际开发的写法
:focus 选择器	选择获得光标的表单	跟表单相关	较少	input:focus 记住这个写法

## 三、css的显示模式

### 1、什么是元素的显示模式

定义：

元素显示模式就是元素（标签）以什么方式进行显示，比如自己占一行，比如一行可以放多个。

作用：

网页的标签非常多，在不同地方会用到不同类型的标签，了解他们的特点可以更好的布局我们的网页。

### 2、元素显示模式的分类

#### 2.1、块元素

常见的块元素：

```
<h1>~<h6>、<p>、<div>、<ul>、<ol>、<li>
```

标签是最典型的块元素。

块级元素的特点：

- 比较霸道，自己独占一行。
- 高度、宽度、外边距以及内边距都可以控制。
- 宽度默认是容器（父级宽度）的100%。
- 是一个容器及盒子，里面可以放行内或者块级元素。

☆注意：

文字类的元素内不能放块级元素

`<p>` 标签**主要用于存放文字**，因此 `<p>` 里面**不能放块级元素**，特别是不能放 `<div>` 同理，`<h1>~<h6>` 等都是文字类块级标签，里面也不能放其他块级元素

## 2.2、行内元素

常见的行内元素：

```
<a>、<strong>、<b>、<em>、<i>、<del>、<s>、<ins>、<u>、<span>
```

标签是**最典型的行内元素**。有的地方也将行内元素称为内联元素。

行内元素的特点：

- 相邻行内元素在一行上，一行**可以显示多个**。
- **高、宽直接设置是无效的**。
- 默认宽度就是它**本身内容的宽度**。
- 行内元素**只能容纳文本或其他行内元素**。

注意：

**链接里面不能再放链接**  
特殊情况链接 [里面可以放块级元素](#)，但是给 **转换一下块级模式**最安全

## 2.3、行内块元素

常见的行内块标签：

```
<img />、<input />、<td>
```

它们同时具有**块元素和行内元素**的特点。有些资料称它们为行内块元素。

行内块元素的特点：

- 和相邻**行内元素（行内块）在一行上**，但是他们之间会有**空白缝隙**。
- 一行可以**显示多个**（行内元素特点）。
- 默认宽度就是它**本身内容的宽度**（行内元素特点）。
- **高度，行高、外边距以及内边距都可以控制**（块级元素特点）。

## 2.4、元素显示模式总结

元素模式	元素排列	设置样式	默认宽度	包含
块级元素	一行只能放一个块级元素	可以设置宽度高度	容器的100%	容器级可以包含任何标签
行内元素	一行可以放多个行内元素	不可以直接设置宽度高度	它本身内容的宽度	容纳文本或则其他行内元素
行内块元素	一行放多个行内块元素	可以设置宽度和高度	它本身内容的宽度	

学习元素显示模式的主要目的就是分清它们各自的特点，当我们网页布局的时候，在合适的地方用合适的标签元素。

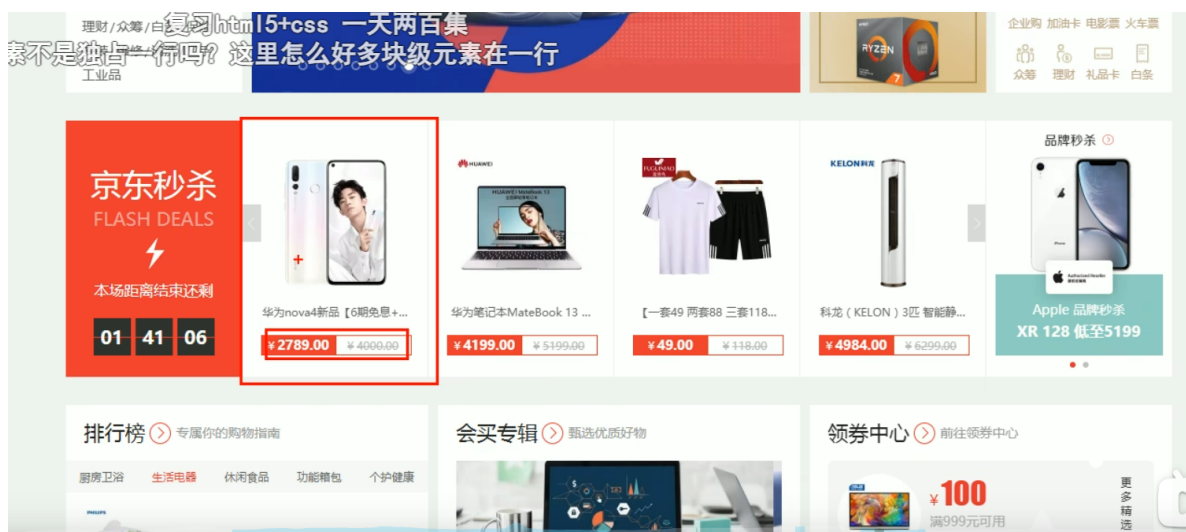
### 3、☆元素显示模式的转换

简单理解:

一个模式的元素需要另外一种模式的特性  
比如想要增加链接 的触发范围。

转换方式

- 转换为块元素: `display: block;`
- 转换为行内元素: `display: inline;`
- 转换为行内块: `display: inline-block;`
- a里面放了一个块级元素: (转换会计模式)



#### 案例：简洁版小米侧边栏

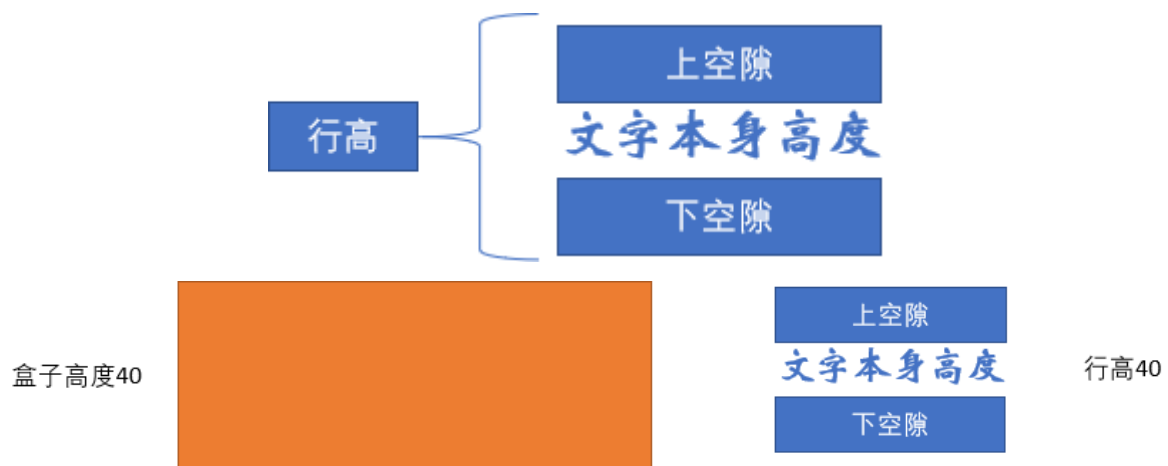
案例的核心思路分为两步:

1. 把链接a 转换为块级元素, 这样链接就可以单独占一行,并且有宽度和高度.
2. 鼠标经过a 给 链接设置背景颜色

### 4、单行文字垂直居中的代码

解决方案:

让文字的行高等于盒子的高度 就可以让文字在当前盒子内垂直居中



#### 简单理解:

行高的上空隙和下空隙把文字挤到中间了,

如果行高小于盒子高度,文字会偏上,

如果行高大于盒子高度,则文字偏下。

ps: 上下空隙等高

## 四、css的背景

通过 CSS 背景属性, 可以给页面元素添加背景样式。

背景属性可以设置背景颜色、背景图片、背景平铺、背景图片位置、背景图像固定等。

### 1、背景颜色

#### 样式名称:

`background-color` 定义元素的背景颜色

#### 使用方式:

```
background-color: 颜色值;
```

#### 其他说明:

元素背景颜色默认值是 `transparent` (透明)

```
background-color: transparent;
```

### 2、背景图片

#### 样式名称:

`background-image` 定义元素的背景图片

#### 使用方式:

```
background-image : none | url (url)
```



参数值	作用
none	无背景图（默认的）
url	使用绝对或相对地址指定背景图像

其他说明：

实际开发常见于 logo 或者一些装饰性的小图片或者是超大的背景图片，**优点是非常便于控制位置**。（精灵图也是一种运用场景）

注意：背景图片后面的地址，**千万不要忘记加 URL**，同时里面的路径**不要加引号**。

### 3、背景平铺

样式名称：

background-repeat 设置元素背景图像的**平铺**

使用方式：

```
background-repeat: repeat | no-repeat | repeat-x | repeat-y
```

参数值	作用
repeat	背景图像在纵向和横向上平铺（默认的）
no-repeat	背景图像不平铺
repeat-x	背景图像在横向上平铺
repeat-y	背景图像在纵向上平铺

### 4、背景图片位置

样式名称：

background-position 属性可以改变**图片在背景中的位置**

使用方式：

```
background-position: x y;
```

参数代表的意思是：**x 坐标和 y 坐标**。可以使用 **方位名词** 或者 **精确单位**

参数值	说明
length	百分数   由浮点数字和单位标识符组成的长度值
position	top   center   bottom   left   center   right 方位名词

其他说明：

#### 1、参数是方位名词

如果指定的两个值都是方位名词，则两个值**前后顺序无关**，比如 left top 和 top left 效果一致

如果只指定了一个方位名词，另一个值省略，则第二个值**默认居中对齐**

## 2、参数是精确单位

如果参数值是精确坐标，那么第一个肯定是 x 坐标，第二个一定是 y 坐标

如果 ☆ 只指定一个数值，则该数值一定是 x 坐标，另一个默认垂直居中

## 3、参数是混合单位

如果指定的两个值是精确单位和方位名词混合使用，则第一个值是 x 坐标，第二个值是 y 坐标

# 5、☆背景图片固定

样式名称：

background-attachment 属性设置背景图像是否固定或者随着页面的其余部分滚动。

使用方式：

```
background-attachment : scroll | fixed
```

参数	作用
scroll	背景图像是随对象内容滚动
fixed	背景图像固定

其他说明：

background-attachment 后期可以制作视差滚动的效果。

# 6、☆背景样式合写(实际开发使用)

背景合写样式：

background: 背景颜色 背景图片地址 背景平铺 背景图像滚动 背景图片位置;

```
div {
    background-image: url(./1.jpg);
    background-position: center center;
    /* 可以直接用f12 选中元素 找到css 选中background 有url 右键打开*/
    width: 1000px;
    height: 1000px;
    /* 背景图片会压住背景元素 */
    /* 默认是不平铺了*/
    /* 控制位置 */

    /* 小图片喜欢用背景弄 */
    /* 因为控制位置很方便 */
    /* 大型图片也喜欢用背景弄 */
    /* 因为直接插入图片很麻烦 当然也可以直接src进去 */

    /* 如果要使用坐标的话 也可以用那个截图小工具来看元素大小 */
    /* 混合单位也是有序的 x横 y竖 */

    /* 视差滚动 */
    /* 图片可以固定scroll/滚动fixed 视差滚动的原理 */

    /* 背景复合属性 实际开发写法 而且顺序没有要求 颜色 地址 平铺 滚动 位置 */
    /* 和font不一样 没有强制要求 */
}
```

使用方式:

```
background: transparent url(image.jpg) repeat-y fixed top ;
```

## 7、背景色半透明

CSS3 提供了背景颜色半透明的效果。

使用方式:

```
background: rgba(0, 0, 0, 0.3);
```

- 最后一个参数是 **alpha 透明度**，取值范围在 0~1之间
- 我们习惯把 0.3 的 0 省略掉，写为 background: rgba(0, 0, 0, .3);

注意:

- 背景半透明是指盒子**背景半透明**，**盒子里面的内容不受影响**
- CSS3 新增属性，是 IE9+ 版本浏览器才支持的，但是现在实际开发,我们不太关注兼容性写法了,可以放心使用

## 8、背景总结

属性	作用	值
background-color	背景颜色	预定义的颜色值/十六进制/RGB代码
background-image	背景图片	url(图片路径)
background-repeat	是否平铺	repeat/no-repeat/repeat-x/repeat-y
background-position	背景位置	length/position 分别是x 和 y坐标
background-attachment	背景附着	scroll (背景滚动) /fixed (背景固定)
背景简写	书写更简单	背景颜色 背景图片地址 背景平铺 背景滚动 背景位置;
背景色半透明	背景颜色半透明	background: rgba(0,0,0,0.3); 后面必须是 4个值

**位置最重要**