

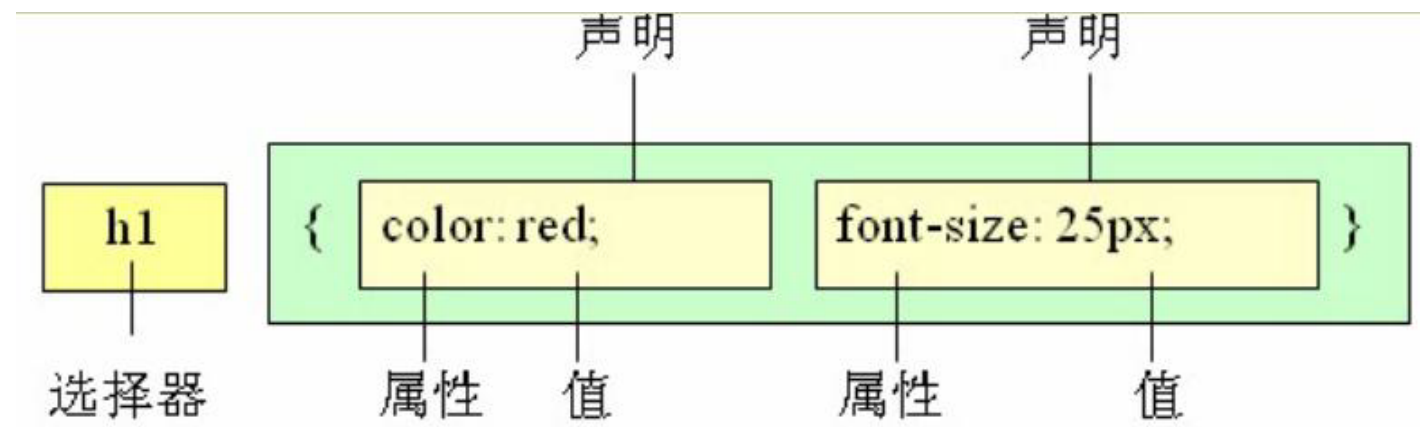
# 第二章：CSS基础-陈泓锦

## CSS简介

### CSS是什么

- Cascading Style Sheets
- 用于定义页面元素的样式
  - 字体和颜色
  - 位置和大小
  - 动画效果

CSS 规则由两个主要的部分构成：选择器以及一条或多条声明。



### CSS的使用

1. 外链
2. 嵌入
3. 内联

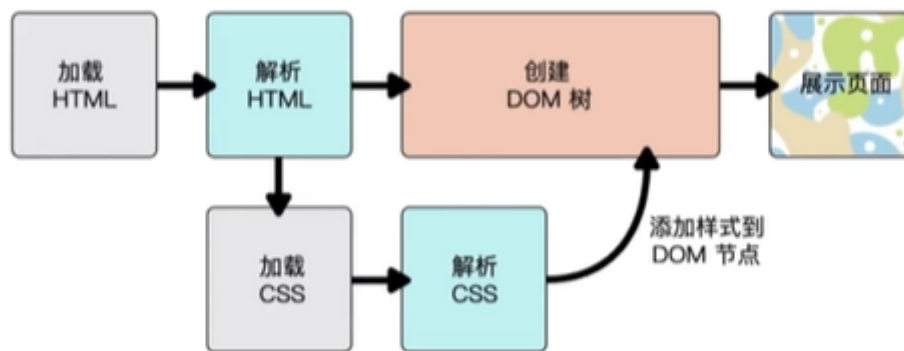
## HTML

```
1 <link rel="stylesheet" href="index.css"> 外
2
3 <style>
4     li {
5         list-style: none
6     }
7 </style>
8
9 <p style="color="red">Example</p>
```

内联

嵌入

## CSS是如何工作的



## CSS调试

```
<div id="mainBox" class="flex layout-column">
  <div id="mainContainer" class="app layout-column flex suite-doc">
    <div class="navigation-bar-wrapper"></div>
    <span id="modal-manager" style="display: none;"></span>
    <div class="progress-viewer-wrapper"></div>
    <div class="NPS-questionnaire"></div>
  </div>
  <div class="app-main-container flex layout-row explorer-v3 is-suite">
    <div class="app-main main_content">
      <div class="global-back" style="right: 97px;"></div>
      <div class="suite-body flex layout-column" style="height: 88.0026px; background: url('https://sf3-cn-feishucdn.com/obj/easecscv/421e71e6b739402bb2e8d74da746c77_3.png') repeat;">
        <div class="scrollbar-container etherpad-container-wrapper has-comment-toggle cle arfix ps ps-active-y">
          ::before
          <div class="section-nav-container" style="background-color: #f0f0f0;"></div>
          <div class="etherpad-container-sub-wrapper">
            <div class="etherpad-container flex layout-row suspension-comment">
              <div class="doc-position"></div>
              <div class="etherpad-client-container flex layout-column">
                <div id="editor-container-box" class="flex layout-column contenteditable="false">
                  <div id="doc-container">
                    <div id="outerdocbody" class="outerdocbody">
                      <div id="innerdocbodywrap">
                        <div contenteditable="true" class="adit-container innerdocbody main docbody syntax flex doc-new-layout chrome window notranslate chrome8 8 docbody-write reminder_enabled" id="innerdocbody" data-placeholder="style="min-height: 80px; font-family: LarkMacSafariFont, LarkEmojiFont, LarkChineseQuote, -apple-system, BlinkMacSystemFont, 'Helvetica Neue', Arial, 'Segoe UI', 'PingFang SC', 'Microsoft Yahei', 'Hiragino Sans GB', 'sans-serif, 'Apple Color Emoji', 'Segoe UI Symbol', 'Noto Color Emoji'; white-space: pre-wrap; overflow-anchor: none; data-gramm="false" spellcheck="false" autocorrect="on" data-last-height="1566" data-last-width="730">...
                        </div>
                      </div>
                    </div>
                  </div>
                </div>
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
```

Styles Computed Layout Event Listeners DOM Breakpoints Properties Accessibility

Filter :hov .cls +

```
element.style {
  min-height: 88px;
  font-family: LarkMacSafariFont, LarkEmojiFont, LarkChineseQuote, -apple-system, BlinkMacSystemFont, 'Helvetica Neue', Arial, 'Segoe UI', 'PingFang SC', 'Microsoft Yahei', 'Hiragino Sans GB', sans-serif, 'Apple Color Emoji', 'Segoe UI Symbol', 'Noto Color Emoji';
  white-space: pre-wrap;
  overflow-anchor: none;
}

#mainBox #outerdocbody .maindocbody, #mainBox .maindocbody, #mainContainer #outerdocbody .maindocbody, doc_index_c_b5ad6.css:1
#mainContainer .maindocbody {
  min-height: 100px;
  overflow-wrap: break-word;
  word-wrap: break-word;
  word-break: break-word;
  padding-left: 20px;
  padding-right: 40px;
  padding-bottom: 18px;
}

.window #outerdocbody .maindocbody, .window .print .maindocbody {
  line-height: 1.65;
  letter-spacing: .02em;
  color: #000;
}

.window #outerdocbody .maindocbody, .window .print .maindocbody {
  line-height: 1.65;
  letter-spacing: .02em;
  color: #000;
}

#outerdocbody .maindocbody, .print .maindocbody {
  min-height: 400px;
  font-size: 16px;
  line-height: 1.65;
  color: #000;
  outline: none;
  webkit-box-sizing: border-box;
  box-sizing: border-box;
}
```

# CSS基础选择器

## 通配选择器\*

使用 \* 可为所有元素设置样式。

CSS

```
1 * {  
2     text-decoration: none;  
3     color: #6c757d;  
4 }
```

## 标签选择器

标签选择器是指用 HTML 标签名称作为选择器，按标签名称分类，为页面中某一类标签指定统一的 CSS 样式。

CSS

```
1 h1 {  
2     color: red;  
3 }
```

## id选择器 必须唯一

id 选择器可以为标有特定 id 的 HTML 元素指定特定的样式。

HTML 元素以 id 属性来设置 id 选择器，CSS 中 id 选择器以 # 来定义。

HTML

```
1 #container {  
2     background: red;  
3 }  
4  
5 ...  
6 <h1 id="container">houdunren.com</h1>
```

## 类选择器 适合一类有相同特征的元素

如果想要差异化选择不同的标签，单独选一个或者某几个标签，可以使用类选择器。

类选择器是为一类状态声明样式规则，下面是把文本居中定义为类样式。

```
HTML

1 <style>
2 .text-center {
3     text-align: center;
4 }
5 </style>
6 <h1 class="text-center">HJ</h1>
7 <h2 class="text-center">hj.com</h2>
```

## 属性选择器

	A	B	C
1	选择器	示例	描述
2	[attribute]	[target]	带有 target 属性所有元素
3	[attribute=value]	[target=_blank]	targe 属性 等于"_blank" 的所有元素
4	[attribute~=value]	[title~=houdunren]	title 属性包含单词 "houdunren" 的元素
5	[attribute =value]	[title =hd]	title 属性值为 "hd"的单词，或hd-cr以-连接的的独立单词
6	[attribute*=value]	a[src*="hdcms"]	src 属性中包含 "hdcms" 字符的每个素
7	[attribute^=value]	a[src^="https"]	src 属性值以 "https" 开头的每个元
8	[attribute\$=value]	a[src\$=".jpeg"]	src 属性以 ".jpeg" 结尾的所有 元素

不通过标签 / ID  
而是 状态 / DOM 结构

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。

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

## 动态伪类

	A	B	C
1	状态	示例	说明
2	:link	a:link	选择所有未被访问的链接
3	:visited	a:visited	选择所有已被访问的链接
4	:hover	a:hover	鼠标移动到元素上时
5	:active	a:active	点击正在发生时
6	:focus	input::focus	选择获得焦点的 input 元素

## 结构性伪类

	A	B	C
1	状态	示例	说明
2	:first-child	li:first-child	选择li的第一个元素
3	:last-child	li:last-child	选择li的最后一个元素

## 组合选择器

### 直接组合

满足A且满足B

input.err ⇒ 类型 input, 同时包含"err"

CSS

```
1 input:focus {
2     border: none
3 }
```

### 后代组合

后代选择器选择父元素里面子元素。

CSS

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

### 亲子组合

只能选择作为某元素的最近一级子元素。直接子元素

CSS

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

## 选择器组

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。

CSS

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

## CSS字体

CSS Fonts (字体)属性用于定义字体系列、大小、粗细、和文字样式(如斜体)。

### 字体font-family

CSS 使用 font-family 属性定义文本的字体系列。

如:

CSS

```
1 p { font-family:"微软雅黑";}
2 div {font-family: Arial,"Microsoft Yahei", "微软雅黑";}
```

- 各种字体之间必须使用英文状态下的逗号隔开
- 一般情况下,如果有空格隔开的多个单词组成的字体,加引号.
- 尽量使用系统默认自带字体, 保证在任何用户的浏览器中都能正确显示

· 设置多种字体应付各刁民的电脑  
优先级逐渐↓

· 列表最后务必带上通用字体族.

# 通用字体族

Serif 衬线体

Georgia、宋体

Sans-Serif 无衬线体

Arial、Helvetica、黑体、微软雅黑

Cursive 手写体

Caflisch Script、楷体

Fantasy

Comic Sans MS, Papyrus,



Monospace 等宽字体

Consolas、Courier、中文字体

使用 Web Fonts

特殊的  
一般正常人不会安装  
的字体!

```
<h1>Web fonts are awesome!</h1>
<p>Web fonts are a CSS feature... </p>

<style>
  @font-face {
    font-family: 'Megrim';
    src: url('/megrim.woff2') format('woff2'),
        url('/megrim.ttf') format('truetype')
  }

  @font-face {
    font-family: 'Buda';
    src: url('/buda.woff2') format('woff2'),
        url('/buda.ttf') format('truetype');
  }

  h1 {
    font-family: Megrim, Cursive;
    font-size: 40px;
  }
</style>
```



使用代码：

```
@font-face {
  font-family: '...';
  src: url(...)
      format(...);
}
```

## 字体大小font-size

CSS 使用 font-size 属性定义字体大小。

关键字

- small、medium、large

长度

- px、em

百分数

- 相对于父元素字体大小

谷歌浏览器默认的文字大小为16px

## 字体粗细font-wight

CSS 使用 font-weight 属性设置文本字体的粗细。

属性值	描述
normal	默认值（不加粗的）
bold	定义粗体（加粗的）
100~900	400 等同于 normal，而 700 等同于 bold 注意这个数字后面不跟单位

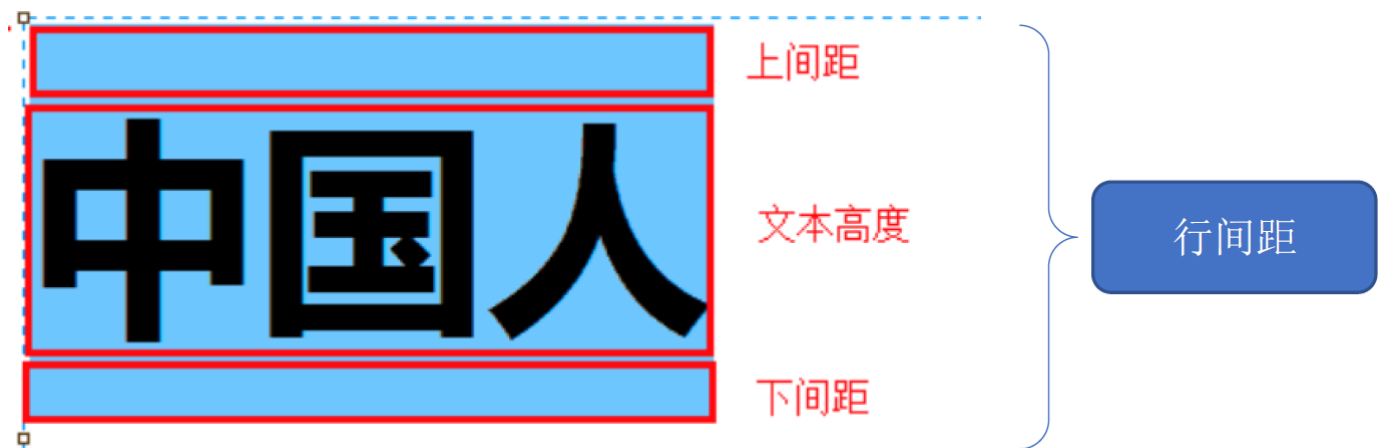
## 行间距line-height

line-height 属性用于设置行间的距离（行高）。可以控制文字行与行之间的距离。

CSS

```
1 p {  
2     line-height: 26px;  
3 }
```

若不带单位，  
则解释为 font-size 的一倍



<https://blog.csdn.net/Hengistchan>

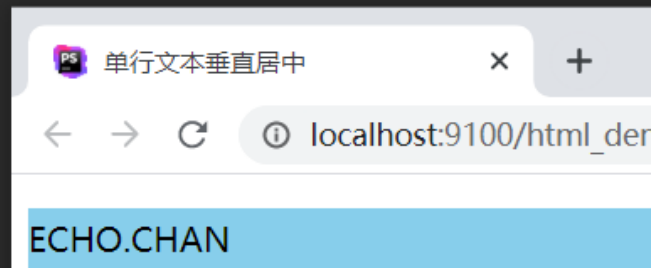


- 在开发中，可以设置行间距等于字段高度使得一段文字居中显示的效果。

## CSS

```
1 p {  
2     height: 30px;  
3     line-height: 30px;  
4 }
```

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <title>单行文本垂直居中</title>  
  <style>  
    p {  
      background-color: skyblue;  
      height: 30px;  
      line-height: 30px;  
    }  
  </style>  
</head>  
<body>  
  <p>  
    ECHO.CHAN  
  </p>  
</body>  
</html>
```



<https://blog.csdn.net/Hengistchan>

## 字体样式font-style

CSS 使用 font-style 属性设置文本的风格。

```
p{
  font-style: normal;
}
```

属性值	作用
normal	默认值，浏览器会显示标准的字体样式 font-style: normal;
italic	浏览器会显示斜体的字体样式。

## 字体复合属性

字体属性可以把以上文字样式综合来写, 这样可以更节约代码:

CSS

```
1 body {
2   font: font-style font-weight font-size/line-height font-family;
3 }
```

- 使用 font 属性时，必须按上面语法格式中的顺序书写，**不能更换顺序**，并且各个属性间以空格隔开
- 不需要设置的属性可以省略（取默认值），但**必须保留 font-size 和 font-family 属性**，否则 font 属性将不起作用

## 文字样式

### 对齐文本text-align

text-align 属性用于设置元素内文本内容的水平对齐方式。

属性值	解释
left	左对齐 (默认值)
right	右对齐
center	居中对齐

justify

分散/两端对齐.

## 文本间距spacing

属性值	解释
letter-spacing	字母间距
word-spacing	单词间距

## 文本缩进text-indent

text-indent 属性用来指定文本的第一行的缩进，通常是将段落的首行缩进。

## 装饰文本text-decoration

text-decoration 属性规定添加到文本的修饰。可以给文本添加下划线、删除线、上划线等。

属性值	描述
none	默认。没有装饰线（最常用）
underline	下划线。链接 a 自带下划线（常用）
overline	上划线。（几乎不用）
line-through	删除线。（不常用）

<https://blog.csdn.net/Hengistchan>

先写这么多！😁