

CSS属性-官方文档解读

@M了个J

<https://github.com/CoderMJLee>

<https://space.bilibili.com/325538782>



实力IT教育 www.520it.com



CSS属性的描述

- CSS官方文档中对每一个CSS属性都有详细的描述

<i>Name:</i>	<i>font-family</i>
<i>Value:</i>	[<family-name> <generic-family>] #
<i>Initial:</i>	depends on user agent
<i>Applies to:</i>	all elements
<i>Inherited:</i>	yes
<i>Percentages:</i>	N/A
<i>Media:</i>	visual
<i>Computed value:</i>	as specified
<i>Animatable:</i>	no

- **Name**: 属性名称
- **Value**: 可以设置哪些值
- **Initial**: 初始值 (默认值)
- **Applies to**: 适用于哪些元素
 - **all elements**: 所有元素
 - **block-level elements**: 块级元素
- **Inherited**: 是否可以继承
 - **yes**: 可以
 - **no**: 不可以
- **Percentages**: 是否可以使用百分比
 - **N/A**: Not applicable, 不适用, 不支持
- **Media**: 媒体组
 - **visual**: 视觉
- **Computed value**: 如何得出计算值
- **Animatable**: 是否可动画

CSS属性的取值 - 组合 (combinators)

- 每个取值之间的关系，有以下几种情况
 - **并列**：取值之间用空格隔开，代表每个都必须出现，顺序也不能改变（参考font属性）
 - **|**：只能出现一个（参考font-style属性）
 - **||**：出现1个或者多个，顺序任意（参考background属性）
 - **&&**：每个都必须出现，顺序任意
 - **[]**：分组，代表是一个整体（参考text-decoration属性）
- 优先级
 - **并列 > && > || > |**
 - ✓ **a b | c || d && e f** 等价于 **[a b] | [c || [d && [e f]]]**
- 如果是**斜杠/**、**逗号**,两种字符必须以字面形式显示，参考font-family、font

CSS属性的取值 - 出现次数 (multipliers)

- 每个取值或者分组后面，可以跟着以下符号，用来指明取值可以出现的次数
- 星号 (*) : 出现0或多次 (参考font-family属性)
- 加号 (+) : 出现1或多次 (参考content属性)
- 问号 (?) : 出现0或1次 (参考background-position、font属性)
- 花括号 ({A,B}) : 至少出现A次，最多出现B次 (参考margin、border-color属性)
- 井号 (#) : 出现1或多次，每个值之间用逗号,隔开
- 感叹号 (!) : 跟在分组后面，表示这个分组至少会出现1个值 (即使分组内规定所有值都是可选)

CSS属性的取值 - 举例

	in order	any order
zero or more	A? B? C?	A? B? C?
one or more	[A? B? C?]!	A B C
all	A B C	A && B && C

CSS属性的取值 - 举例

- you || me || her
- [you | she]? [study || eat]
- you really{1, 4} [love | hate] [apple | banana | pear]
- [money ,]{2, 3} food

CSS属性的取值 - 类型 (types)

■ CSS属性的取值，有以下几种类型

□ 关键字

✓ 比如red、auto、none、left、right等（没有任何引号、大于小于号包住），以字面形式显示

□ 基本数据类型

✓ 格式为<数据类型>，比如<length>、<number>、<percentage>、<color>、<string>、<time>等

□ 跟特定属性相关的数据类型

✓ 格式为<数据类型>，比如<margin-width>、<border-width>等

□ 引用其他属性的值

✓ 格式为'属性名'，比如'font-size'、'background-image'

CSS属性的取值 - 举例

Property	Value definition field	Example value
<u>'orphans'</u>	<integer>	'3'
<u>'text-align'</u>	left right center justify	<u>'center'</u>
<u>'padding-top'</u>	<length> <percentage>	'5%'
<u>'outline-color'</u>	<color> invert	'#fefefe'
<u>'text-decoration'</u>	none underline overline line-through blink	'overline underline'
<u>'font-family'</u>	[<family-name> <generic-family>]#	'"Gill Sans", Futura, sans-serif'
<u>'border-width'</u>	[<length> thick medium thin]{1,4}	'2px medium 4px'
<u>'text-shadow'</u>	[inset? && [<length>{2,4} && <color>?]]# none	'3px 3px rgba(50%, 50%, 50%, 50%), lemonchiffon 0 0 4px inset'

<number>、<integer>

■ <number>

- 实数，包括整数、小数，比如2.5、90

```
div {  
  line-height: 2.5;  
}
```

■ <integer>

- 整数，比如10、+10、0、-20、968

```
div {  
  z-index: 1;  
}
```

<length>

- <length>代表着长度，一段距离
- <length>的值相当于<number>后面紧跟着一个长度单位

```
h1 {  
  width: 20px;  
  height: 10.5em;  
  margin-top: -30px;  
}
```

- 如果<length>的值为0，可以省略长度单位

```
h1 {  
  width: 0;  
}
```

- 常见长度单位有2种
- 绝对单位：px
- 相对单位：em、rem

<length> - 相对单位em、rem

■ em

□ em的计算值，相对于当前元素的font-size进行计算

□ 例外情况：当em用在font-size属性上时，相对于父元素的font-size进行计算

```
div {  
  font-size: 40px;  
  height: 2em; /* 80px */  
}
```

```
body {  
  font-size: 40px;  
}
```

/ body是div的父元素 */*

```
div {  
  font-size: 2em; /* 80px */  
  height: 2em; /* 160px */  
}
```

■ rem

□ rem的计算值，相对于根元素的font-size进行计算

□ 当rem用在根元素的font-size属性上时，那就相对于根元素font-size的初始值计算

<time>

- <time>表示时间，相当于<number>后面跟着一个时间单位
- s: 秒
- ms: 毫秒, 1秒 = 1000毫秒

<angle>

- <angle>表示角度，，相当于<number>后面跟着一个角度单位
- deg: Degrees, 度, 每个圆周是360 degrees

<percentage>

- <percentage>的写法是<number>后面紧跟着一个百分号%

```
span {  
  font-size: 150%;  
}
```

- 如何获得<percentage>的计算值?

- 可能相对于当前元素的某一属性值计算
- 可能相对于父元素的某一属性值计算
- 可能相对于格式化上下文的某一属性值计算，比如包含块的宽度

```
body {  
  font-size: 20px;  
}
```

/ body是div的父元素 */*

```
div {  
  font-size: 200%;  
  /* 40px, 相对于父元素的font-size计算 */  
}
```

```
div {  
  font-size: 40px;  
  line-height: 200%;  
  /* 80px, 相对于当前元素的font-size计算 */  
}
```

```
div {  
  width: 200px;  
}
```

```
span {  
  display: inline-block;  
  width: 50%; /* 100px, div的内容框是span的包含块 */  
  height: 30px;  
}
```

<string>

- <string>表示字符串
- 可以用双引号或者单引号括住

```
div::after {  
    content: "我是字符串";  
}  
  
div::before {  
    content: '我是字符串';  
}
```

<url>

- <url>的格式是url(资源路径)

```
div {  
    background-image: url("http://www.test.com/1.png");  
}
```

```
div {  
    background-image: url('http://www.test.com/1.png');  
}
```

```
div {  
    background-image: url(http://www.test.com/1.png);  
}
```

- 在某些CSS环境下，例如@import，允许<url>用裸体字符串表示，不用url()来包装

/ 下面2个@import的含义完全相同 */*

```
@import "test.css";  
@import url(test.css);
```