CSS

# 基础概念

Html主要的侧重点是显示内容。Html的标签(元素)自己带有自己的样式。

CSS是主要侧重内容的显示方式。修饰内容，使得html更美观。主要的作用就是定义html标签(元素)的样式。

CSS 指层叠样式表 (Cascading Style Sheets)

是为了解决内容与表现分离的问题

# CSS

## 案例：

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>css的第一个案例</**title**>  *<!--  css有3种定义方式：  行级样式：定义在html标签的style属性上  内部样式：定义在<style>标签内部  外部样式：定义在css文件中 通过link标签来引入  -->* </**head**> <**body**>  <**p style="color**: **red"**>内容显示为红色</**p**>  <**p style="color**: **blue"**>内容显示为蓝色</**p**> </**body**> </**html**> |

行级样式会让内容和样式合并在一起，更新麻烦。

内部样式：

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>内部样式</**title**>  *<!--定义内部样式-->  <!--所有的p标签的内容都为红色-->* <**style type="text/css"**>  **p**{  **color**: **red**;  }  </**style**> </**head**> <**body**>  <**p**>内容显示为红色</**p**>  <**p**>内容显示为蓝色</**p**> </**body**>  </**html**> |

外部样式

定义一个css文件：outerstyle.css

|  |
| --- |
| **p**{  **color**: **red**; } |

定义html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>外部样式</**title**>  *<!--引入外部样式-->* <**link rel="stylesheet" type="text/css" href="css/outerstyle.css"**/> </**head**> <**body**>  <**p**>内容显示为红色</**p**>  <**p**>内容显示为蓝色</**p**> </**body**>  </**html**> |

## 基础语法

CSS 语法由三部分构成：选择器、属性和值：

|  |
| --- |
| selector {property: value} |

选择器 (selector) 通常是你希望定义的 HTML 元素或标签，属性 (property) 是你希望改变的属性，并且每个属性都有一个值。属性和值被冒号分开，并由花括号包围，这样就组成了一个完整的样式声明（declaration）：

body {color: blue}

选择器是用于说明，html中有哪个或者哪些元素被选中，并且修改其样式。可以理解为被修改样式的范围。

选择器的分类：

标签选择器：selector本身就是一个html标签，那么表示html文档所有该标签都会被修改。

|  |
| --- |
| **p**{  **color**: **red**; } |

类选择器：selector的名称 以.开头,跟上一个自定义的名称 ，那么html文档中如果有元素要想被修改为这个类样式，那么直接在标签的class属性上，指明类名即可。

Css

|  |
| --- |
| .**red**{  **color**: **red**; } .**blue**{  **color**: **blue**; } |

Html

|  |
| --- |
| <**body**>  <**p class="red"**>内容显示为红色</**p**>  <**p class="blue"**>内容显示为蓝色</**p**> </**body**> |

Id选择器：selector的名称 以#开头，跟上的是要修饰的html元素的id属性值。通常一个html文档中的元素id是唯一的。Id选择器是用于修饰某一个元素的。

Css

|  |
| --- |
| **#p1**{  **color**: **green**; } |

Html

|  |
| --- |
| <**body**>  <**p class="red"**>内容显示为红色</**p**>  <**p class="blue"**>内容显示为蓝色</**p**>  <**p id="p1"**>内容显示为绿色</**p**> </**body**> |

总结：

标签选择器范围最广，类选择器次之，id选择器范围最小

标签选择器优先级最低，类选择器次之，id选择器优先级最高

### 选择器的分组

所谓的选择器的分组就是不同选择器，具有相同的样式声明，那么可以合并。只需将选择器名称用逗号分隔即可。

|  |
| --- |
| h1,h2,h3,h4,h5,h6 {  color: green;  } |

### 选择器的继承

子元素默认情况下会继承父元素的样式。如果要考虑浏览器兼容问题，那么需要进行测试，有的继承会出问题。

### 派生选择器

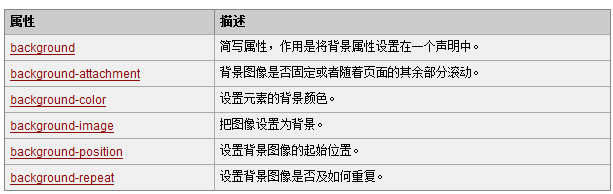
派生选择器用于当父级元素不一样的时候 修饰这个父级元素的某个子集元素。

|  |
| --- |
| <**style type="text/css"**>  **div**,**p**{  **font-size**: 20**px**;  **color**: **#000**;  }  **div span**{  **color**: **blue**;  }  **p span**{  **color**: **red**;  } </**style**> |

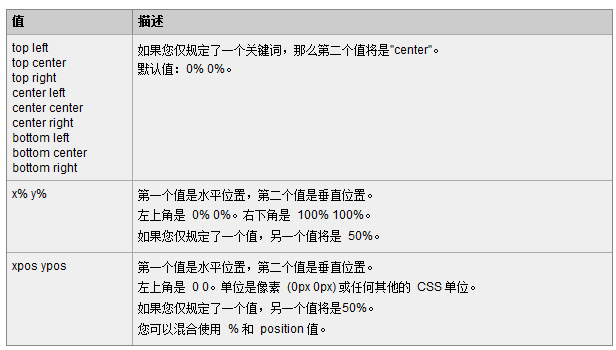
Html

|  |
| --- |
| <**body**>  <**p**>内容显示为<**span**>红色</**span**></**p**>  <**div**>内容显示为<**span**>蓝色</**span**></**div**> </**body**> |

## 背景属性



背景位置的值：



background 简写属性在一个声明中设置所有的背景属性。

可以按顺序设置如下属性：

* background-color
* background-image
* background-repeat
* background-attachment
* background-position

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>背景属性</**title**>  *<!--当有背景颜色和背景图片时，背景图片将背景颜色盖住-->* <**style type="text/css"**>  **body**{  **width**: 1366**px**;  **height**: 700**px**;  **border**: 1**px solid black**;  **background-image**: **url**(**"image/bg.jpg"**);  **background-repeat**:**no-repeat**;  **background-position**: **center center**;  }  .**bg**{  **background**: **cyan url**(**"image/bg.jpg"**) **no-repeat fixed center center**;  }  </**style**> </**head**> <**body**> </**body**> </**html**> |

## 文本属性

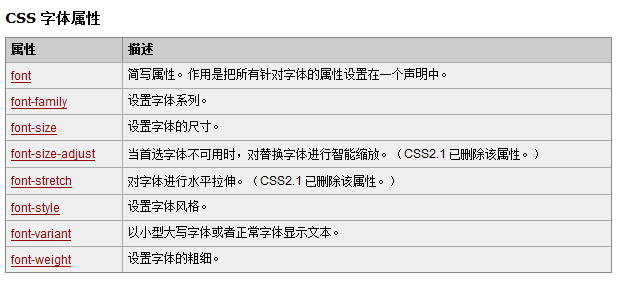
通过文本属性，您可以改变文本的颜色、字符间距，对齐文本，装饰文本，对文本进行缩进，等等。



|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>文本属性</**title**>  <**style type="text/css"**>  **p**{  **color**: **red**;  **line-height**: 40**px**;  **letter-spacing**: 21**px**;  **text-align**: **left**;  **text-decoration**: **none**;  **white-space**: **nowrap**;  **border**: 1**px solid green**;  **text-indent**: 100**px**;  }  **a**{  **text-decoration**: **none**;  }  </**style**> </**head**> <**body**>  <**p**>文本属性</**p**>  <**p**>那年冬天，祖母死了父亲的差使1也交卸了，正是祸不单行的日子。  我从北京</**p**>  <**a href="#"**>sss</**a**>  ttt </**body**> </**html**> |

## 字体属性

字体属性允许您设置字体系列 (font-family) 和字体加粗 (font-weight)，您还可以设置字体的大小、字体风格（如斜体）和字体变形（如小型大写字母）。



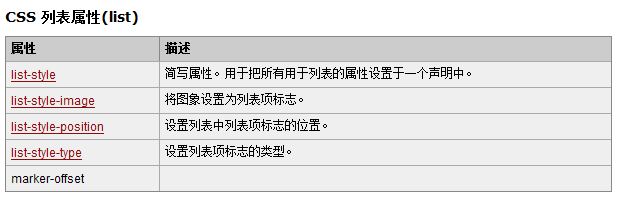
font 简写属性在一个声明中设置所有字体属性。

按顺序设置如下属性：

* font-style
* font-variant
* font-weight
* font-size/line-height
* font-family

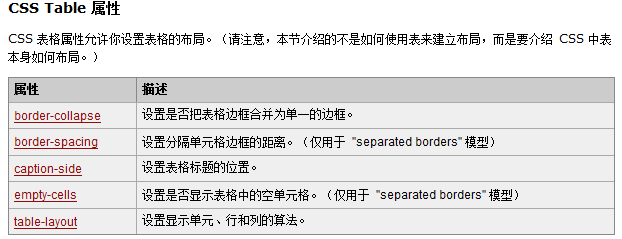
|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>font属性</**title**>  <**style type="text/css"**>  **p**{  **font-family**: **黑体**;  **font-size**: 20**px**;  **font-style**: **italic**;  **font-variant**:**small-caps**;  **font-weight**: **bold**;  }  </**style**> </**head**> <**body**>  <**p**>那年冬天，祖母死了父亲的差使1也交卸了，正是祸不单行的日子。ABCadsf  我从北京</**p**> </**body**> </**html**> |

## 列表属性

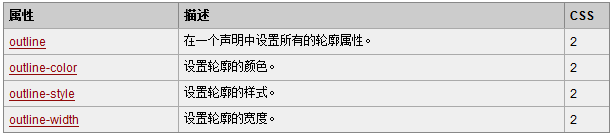


|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>列表属性</**title**>  <**style type="text/css"**>  **ul**{  **list-style-type**: **none**;  **list-style-image**: **url**(**"image/logo.jpg"**);  **border**: 1**px solid blue**;  }  </**style**> </**head**> <**body**>  <**ul**>  <**li**>python</**li**>  <**li**>html</**li**>  <**li**>css</**li**>  <**li**>mysql</**li**>  </**ul**> </**body**> </**html**> |

## 表格属性



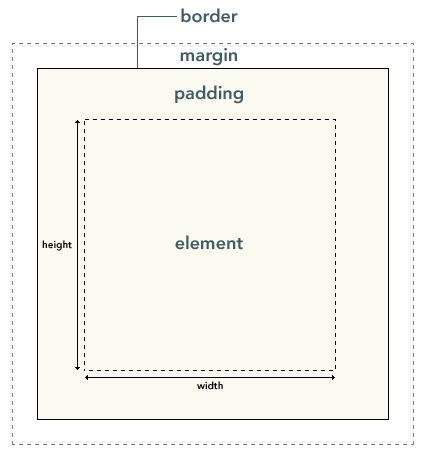
## 轮廓属性



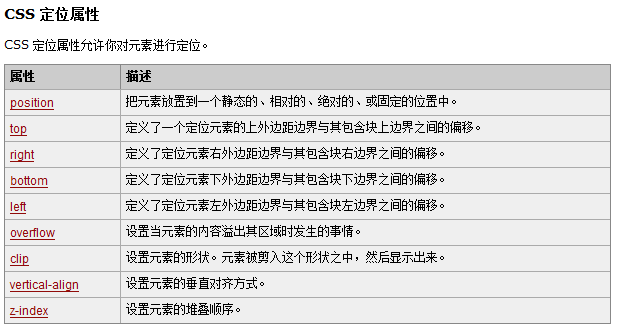
|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>outline</**title**>  <**style type="text/css"**>  **p**{  **outline**: **red dotted** 1**px**;  }  </**style**> </**head**> <**body**>  <**p**>ttt</**p**> </**body**> </**html**> |

## 盒子模型

CSS 框模型 (Box Model) 规定了元素框处理元素内容、内边距、边框 和 外边距 的方式。



## 定位



|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>position</**title**>  <**style type="text/css"**>  **div**{  **position**:**absolute**;  **top**: 50**px**;  **left**: 100**px**;  **width**: 1000**px**;  **height**: 700**px**;  **margin**: **auto**;  **border**: 1**px solid black**;  }  **span**{  **width**: 300**px**;  **height**: 200**px**;  **display**: **block**;  **border**: 1**px solid blue**;  }  **#box2**{  **position**: **absolute**;  **background-color**: **red**;  **left**: 30**px**;  **top**: 20**px**;  **z-index**: 10;  }  **#box1**{  **position**: **relative**;  **background-color**: **white**;  **z-index**: 18;  }  </**style**> </**head**> <**body**> <**div**>  <**span id="box1"**></**span**>  <**span id="box2"**></**span**>  <**span id="box3"**></**span**> </**div**> </**body**> </**html**> |

## 浮动

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>浮动</**title**>  <**style type="text/css"**>  **#box1**{  **width**: 100**px**;  **height**: 100**px**;  **border**: 1**px solid rebeccapurple**;  }  **#box2**{  **width**: 100**px**;  **height**: 100**px**;  **border**: 1**px solid blue**;  }  **#box3**{  **width**: 100**px**;  **height**: 100**px**;  **border**: 1**px solid green**;  }  **#box4**{  **width**: 100**px**;  **height**: 100**px**;  **border**: 1**px solid red**;  **clear**: **both**;  **float**: **right**;  }  **#box2**,**#box3**{  **float**: **right**;  }  </**style**> </**head**> <**body**> <**span id="box1"**> **&nbsp;&nbsp;&nbsp;&nbsp;**</**span**> <**div id="box2"**></**div**> <**div id="box3"**></**div**> <**div id="box4"**></**div**>  </**body**> </**html**> |