

一 概述

层叠样式表，可以美化 html 页面

二 注释

`/* 注释内容 */`

不支持类似 java 中的 `//` 单行注释 和 `/**` 文档注释 `*/`

三 div、p、span

这三个标签都可以看做是容器，可以包裹其它 html 内容，然后统一设置样式

例：有 10 个 font 需要设置 color，那么不需要每个设置，可以用一个容器，比如 `<div style="color:red">` 多个 font 标签 `</div>` 包裹起来，同时设置 color

元素的类型：

块级元素：默认情况下独占一行的元素，例如：div、p、h1-h6、br

行内元素：默认情况下多个元素处在同一行，例如：span、font

1 div

独占一行

2 Span

多个 span 会在一行并排排列，直到一行排满才会换行

3 P

段落标签，独占一行，但是在行的上下会留一些空白

四 在 html 中引入 css

- 1 **style 属性**：直接在标签里面添加 **style** 属性，不推荐使用，会造成页面结构混乱、不利于维护和扩展

```
<span style="border: 2px solid #00FF00;font-size: 30px;font-style: italic;">span111</span>
```

- 2 **style 标签**：在 **head** 标签里面通过 **<style>**css 内容 **</style>** 书写 css 内容。实现了 css 与 html 初步分离。

```
<style >
    td{
        text-align: center;
        color:#fff00;
    }
</style>
```

- 3 **link 标签**：引入外部 css

3.A 引入方式：body 中也可以

```
<head>
    <link rel="stylesheet" href="1.css"/>
</head>
```

3.B **rel** 属性和 **href** 不可省略，**href** 可相对路径、绝对路径

五 CSS 选择器

1 选择器用来选择需要修饰的标签

2 基本选择器

2.A 标签名选择器

2.A.a 例：选中所有 **span** 标签

```
span{  
    background-color: #DDA0DD;  
    font-size: 16PX;  
}
```

2.B 类选择器

2.B.a css 中通过.类名选择 给标签添加 **class="类名"** 属性

```
<span class="c1"> span111</span>  
<span class="c1 c2"> span222</span>
```

注意：此时 **span222** 既属于 **c1** 类也属于 **c2**

与标签中类名的先后顺序无关，而是与样式中类名的先后顺序有关，后面的类名的样式会覆盖前面类名的样式。

```
.c1{  
    background-color: #DDA0DD;  
    font-size:16PX;  
}  
.c2{  
    background-color: #FFF000;  
    font-size:16PX;
```

}

2.B.b 可以多个标签用同一个类名，同时选择多个

2.C id 选择器

2.C.a #id 名选择

```
<span id="c1"> span111</span>
#c1{
    font-weight:bold;
    //首行缩进
    text-indent:20px;
}
```

2.C.b id 不能重复，唯一的,注意与 class 的区别

3 扩展选择器

例: <div id="d1">
 div222
 div 下的 span

 fontxxx
 font 下的 span
 fontxxx

 </div>

3.A 后代选择器：在父选择器中，再选择指定的后代元素

注意：包括子孙

格式：父选择器 空格 后代选择器

选择id 为d1 下的所有span 元素，包括 font 下的，也即子孙

```
#d1 span{
    background-color: #DDA0DD;
```

```
font-size:32px;
}
```

3.B 子元素选择器

选择父元素下的子元素，不包括子孙

注意：

包括子孙

格式：父选择器 > 后代选择器

选择id 为d1 下的span 元素，不包括 font 下的，也即不包括子孙

```
#d1 > span{
background-color: #DDA0DD;
font-size:32px;
}
```

3.C 分组选择器

格式：多个选择器用逗号分隔

例：同时选择类名为 c1、c2、id 为 p1 下的 span 子孙的元素

```
.C1,.C2,#p1 span{

}
```

3.D 属性选择器

选择有某个属性的元素

例：选择所有设置了 class 属性的元素 *代表所有

```
*[class]{
border:2px solid grey;
}
```

例：选择所有设置了 type="text" 属性的 input 元素 input 代表所有 input 元素

```
input[type="text"]{
border:2px solid grey;
}
```

3.E 相邻兄弟选择器

如果二个元素有相同的父元素，并且是挨着的，这二个元素就是相邻兄弟，可以根据前面的A元素选中后面的B元素

格式：大哥+小弟{} 只能通过A选择B，注意方向
例：

```
<div>
  <p id="p1"></p>
  <p>ppp222</p>
</div>
```

选中

```
#p1+p{
  background-color: #DDD;
}
```

3.F 伪元素选择器

伪元素选择器不仅选中元素，还包括它的状态

3.F.a 元素状态

:link 表示元素未被点击时的状态

:hover 表示鼠标移入时状态

['hʌvər] 英文含义:盘旋

:active 表示元素被点击时状态

:visited 表示元素被点击后的状态

例：设置a标签 在未被点击时的 字体 无下划线

```
a:link{
  font-size: 24px;
  /*无下划线*/
  text-decoration: none;
}
```

六 属性

1 边框、边距

1.A 边框:

`border`-可以分别设置上下左右的边框，也可一起设置

1.B 外边距:

元素和父元素之间的距离

`Margin`-可以设置上下左右的外边距，也可以一起设置，行内元素设置上下外边距无效

1.C 内边距:

元素和子元素之间的距离

`Padding-left` 上下左右

2 常用属性

2.A 元素类型补充,

块级元素: 独占一行, 可以设置任何任何属性, 块级元素可以包含任何元素
行内元素: 同行显示, 不可设置宽、高、上下外边距, 只能包含行内元素

2.B `display`:

`block`: 块内元素默认值

`inline`: 行内元素默认值

`none`: 隐藏 不占据空间

`inline-block`: 结合了行内元素与块级元素的特点, 可以同行显示, 也可以设置宽高

2.C *text-align*:对齐方式

2.D *font-size*:字体大小

2.E *font-weight*:字体加粗

2.F *font-family*:字体样式

2.G *color*:字体颜色

2.H *text-decoration*:下划线

2.I *background-color*:背景颜色

2.J *background-repeat*:背景图片的铺排方式

2.K *background-image*:背景图片

2.L *background-position*;背景图片位置

2.M *width*:宽

2.N *height*:高