# 一、选择器

**1、属性选择器**

声明方法：声明属性和属性值

**CSS2属性选择器：[att=val]**

**CSS3追加三个属性选择器：**

**[att\*=val]：**属性att的属性值包含val

**[att^=val]：**属性att的属性值的开头字符为val

**[att$=val]：**属性att的属性值的结尾字符为val

**2、结构性伪类选择器**

**①伪类选择器和伪元素选择器**

**Ⅰ伪类选择器：**

**例：a:link{}、a:visited{}、a:hover{}、a:active{}**

**Ⅱ伪元素选择器：**

**例： :first-line：**对元素第一行进行样式设置

**:first-letter：**对元素中文字首字母或第一个字进行样式设置

**:before：**用于在某个元素之前插入内容(**<元素>:before{content:}**)

**:after：**用于在某个元素之后插入内容

**②结构性伪类选择器**

**root：**绑定根元素**（html的<html>部分）**

**not:**排除结构元素下的子结构元素 **(body \*:not(h1){}：**对body内除了子元素h1（不是子元素的h1无效）的其他元素进行样式设置**)**

**empty:**元素内容为空时进行样式设置

**target:**点击超链接跳转到target元素生效（**target元素**：元素的id被当作页面超链接）

**first-child、last-child、nth-child、nth-last-child**

父元素单独指定子元素

如：ul的第一个li元素用 first-child

ul的最后一个li元素用 last-child

ul的第n个元素用 nth-child(n)

ul的倒数第n个元素用 nth-last-child(n)

父元素的第奇数个子元素 nth-child(odd)

父元素的第偶数个子元素 nth-child(even)

针对同类型子元素用 nth-of-type()和nth-last-of-type()

循环：nth-child(a\*n+b)

元素只有一个元素进行该样式设置：only-child == nth-child(1):nth-last-child(1)

**3、UI元素状态伪类选择器**

hover、active、focus支持所有浏览器 其他不支持IE(enabled、readonly、checked、invalid、required、in-range等)

**例：**鼠标移动到文本框控件上：**input[type="text"]:hover{}**

光标焦点落在文本框内（鼠标点击到松开这段时间）：**input[type="text"]:active{}**

文本框控件被激活：**input[type="text"]:focus{}**

通用兄弟元素选择器：<子元素> ~<子元素后的兄弟元素>{}

**例：div ~p{}**

**<div>**

**<p>test1</p>**

**<div></div>**

**<p>test2</p>**

**</div>**

**<p>test3</p>**

**test2和test3可以进行样式设置，test1不做指定。**

**二、content**

**1、用选择器插入内容**

**插入文字：**

**:before{content:'massage';}**

**:after{content:'massage';}**

**指定元素不进行插入：(sample:样本)**

**h2.sample:before{content:none;}**

**2、插入图像文件**

**①插入图像文件：**

**：before{content:url(xxx.jpg);}**

**②插入图片另一个方法：把图片作为元素的背景图像追加**

**p{background-image:url(xxx.jpg)}**

（该方法插入的图片不设定打印背景不能够正常打印）

**③将img的alt属性值作为图片标题**

**img:after{content:attr(alt);}**

**3、用content插入项目编号**

**<元素>:before{content:counter(计数器名)}**

**<元素>{counter-increment:计数器名}（before或after选择器指定的计数器名）**

**指定编号样式和种类：**

**h1：before{**

**content:counter(mycounter,uppper-alpha);**

**color:blue;**

**font-size:42px;**

**}**

**嵌套为让内部标题重新计数，需要在上级标题添加属性：counter-reset:下级计数器名；**

**中编号嵌入大编号如‘2-3’：在中级标题添加属性：content:counter(大标题计数器名) '-' counter(中标题计数器名)；**

# 三、字体

**1、给文字添加阴影**

**text-shadow:length length length color;**

4个参数分别为：字体阴影的左右偏移量、阴影的上下偏移量（右和下为正）、阴影的模糊半径、阴影的颜色

**2、文本自动换行**

**work-break:normal/keep-all/break-all;**

**normal:**浏览器默认换行方式（标点符号不能为行首）

**keep-all:**只能在半角空格或者连字符处换行

**break-all:**允许在单词内换行（拆分单个单词）（safari和chrome标点符号可以为行首，IE标点符号仍然不能为行首）

**3、长单词和URL自动换行**

**work-wrap：normal/break-word;**

**break-word:**长单词和URL内部可以换行

**4、使用服务器端字体**

@font-face{    font-family: WebFont;    src: url('font/Fontin\_Sans\_R\_45b.otf') format("opentype");}h1{ font-family:WebFont;}

**WebFont属性声明为服务器端字体、src:url指定字体文件所在的路径、format：字体文件的格式**

文件格式种类：**OpenType**(属性值为opentype)、**TrueType**（属性值为：truetype）;

**显示客户端本地字体：**

@font-face{

font-family:Arial;

src:local('Arial');

}

**font-size-adjust:保持修改字体种类后尺寸大小不变**

**aspect值**：比例值（不同字体aspect值不同）

**font-size-adjust**:0.46(0.46为Times New Roman的aspect值)

**四、盒相关样式**

**1、盒的类型**

**block：**div、p元素(width和height只能用在block)

**inline：**span、a元素

**inline-block**

对齐方式：

**float:需要清除浮动 clear:both/left/right...**

**inline-block:vertical-align:top/bottom/left/right;**

去除元素间的空白：父元素设置font-size：0；（子元素要设置font-size值）

**inline-table：**针对表格

**list-item：**针对列表

**list-style-type**:circle...:列表标记属性

**table:**表格类型（系列table-)

**none:**元素不显示

**2、容纳不下的内容显示方式**

**overflow**: **scroll、hidden、visible、auto**

overflow-x、overflow-y:单独在水平方向、垂直方向指定显示方法

**overflow：hidden + text-overflow:ellipsis; 在盒末尾添加…**

overflow:针对元素内部所有文本；

text-overflow：只能对该元素的文本生效（子元素文本不生效）

**3、box-shadow:和text-shadow类似**

**box-shadow:(inset) length length length color; （）为可选属性**

4个参数分别为：盒阴影的左右偏移量、阴影的上下偏移量（右和下为正）、阴影的模糊半径、阴影的颜色

**inset关键字是在盒内部创建阴影**

**first-letter和first-line伪元素选择器同样适用（首字阴影/首行阴影）**

**4、指定元素的高度和宽度**

**box-sizing:content-box/border-box**

**content-box:内容宽高（默认）**

**border-box:内容+内部补白+边框的宽高（更好控制总宽度）**

**五、背景和边框**

**1、背景**

**background-clip:指定背景的显示范围（从外到里）border-box、padding-box、content-box;**

**background-origin：指定绘制背景图像的起点：border、padding、content;**

**background-size：属性值：content、cover、具体尺寸；**

指定背景图片的尺寸：维持图像纵横比例将其中一个参数设置为auto，如：background-size:20px auto;如何只设置一个参数，另一个参数默认为auto。也可以用百分比表示：图像尺寸/边框区域的尺寸

**background-repeat:** no-repeat、repeat、repeat-x、repeat-y**、round、space**

**space:不裁剪不调整图片尺寸、自动调整间距**

**round:不裁剪但调整图片尺寸。**

**一个元素显示多个背景：用逗号分隔同时指定多个属性。**

**2、渐变**

**线性渐变：background:linear-gradient(to buttom right,color1,color2);**从左上往右下渐变（共8种、上下左右以及对角线）

**放射性渐变：background:radial-gradient(circle/ellipse at left center，color,color...)**

**at left center...共9种 left、center、right... 3x3**

**例：**

**background:linear-gradient(to buttom,blue 30%,red 60%);**蓝色从30%开始线性渐变到60%变成红色

**background:radial-gradient(circle at 100px 50px,orange,black);**从（100，50）开始从橘色到黑色放射性渐变

**3、圆角边框**

**圆角边框：border-radius:左上右下，右上左下；**

**图像边框：border-image:url(xxx.jpg) border-top-image border-left-image border-right-image border-bottom-image/border-width topbottom leftright;**

**其中：**

四个参数表示图像分割时的上左右下边框（九宫格）（截取图片的该参数宽度作为边框）

**border-width:**边框宽度(可设置4个参数)

**topbottom：** 图像上下两边的显示方法

**leftright:**图像左右两边的显示方法

**属性值：**round:平铺 stretch:拉伸 repeat：重复 round stretch:水平方向平铺，垂直方向拉伸

**例：border-image:url(xxx.jpg) 10/5px round stretch;**

**border-image:url(xxx.jpg) 10/5px 10px 5px 10px round round;**

**六、CSS3变形处理**

**transform**

**属性值：rotate(deg1)旋转、scale(num1,num2)缩放、skew(deg1,deg2)倾斜、translate(length1,length2)移动**

**例：transform:rotate(30deg);**

**transform:scale(2,0.5);**

**混合使用:transform:translate(100px,100px) rotate(45deg) scale(1.5);**

**transform:rotate(45deg) scale(1.5) translate(100px,100px);(两个结果不一样)**

**transform-origin:**变形的基准点（如左下角left bottom);

**七、CSS3的动画功能**

**一、Transitions**

**transition:property(属性) duration(过渡时间) timing-function(过渡方式) delay（延迟执行）**

**例：**transition:background-color 1s linear 1s;

**二、Animations**

**1、div{}**

**2、@keyframes mytest关键帧集合名{创建关键帧的代码}**

**代码格式：40%{div属性及属性值} (可以设置多个属性值同时改变的动画)**

**3、div:hover{**

**animation-name: mytest;(关键字集合名)**

**animation-duration:5s;（持续时间）**

**animation-timing-function:linear;(过渡方式)**

**}**

**其他属性：animation-delay（延迟执行时间）、animation-iteration-count（执行次数）、animation-direction（执行方向）**

**也可写在一行：animation：keyframe的名称 动画的执行时长 动画实现方式 延迟时间 执行次数 执行方向**

**实现动画的方法：**

**linear:在动画开始时与结束时以同样速度进行改变**

**ease-in:动画开始速度很慢，然后速度沿曲线值进行加快**

**ease-out:动画开始速度很快，然后速度沿曲线值进行放慢**

**ease:动画开始速度很慢，然后速度沿曲线值进行加快，再速度沿曲线值进行放慢（中间最快的时候比linear快）**

**ease-in-out:动画开始速度很慢，然后速度沿曲线值进行加快，再速度沿曲线值进行放慢（最快的时候比linear慢）**

**八、布局相关样式**

**1、float布局**

**float:**

**clear:**

**2、多栏布局方式column(文章)**

-moz-

-webkit-

column-count栏数、column-width每栏宽度、column-gap多栏之间的间隔距离 column-rule:间隔线（格式和border一样）

**3、盒布局box**

**display:-moz-box;**

**display:-webkit-box;**

**4、弹性盒布局flex**

**父元素添加属性：display:flex;（flex布局下浮动失效）**

**需要自由伸展的子元素添加属性：flex:1;（不需要指定宽度）**

**弹性盒的flex之和为1，如div1的flex=0.5,div2的flex=0.5，则两个盒子的宽度都为可伸缩部分宽度的一半**

**改变元素的显示顺序：**

**order 例：order:2;**

**改变元素的排列方向：**

**flex-direction:row/row-reverse/column/column-reverse:横向排列、横向反向排列、纵向排列、纵向反向排列**

**flex子元素宽度：**

**flex-grow:元素宽度之和小于总宽度时，例：总宽度600px，元素宽度100px，flex-grow分别等于1，3，1。宽度分别为100+(600-100\*3)\*（1/5))=160px，100+(600-100\*3)\*（3/5))=280px，100+(600-100\*3)\*(1/5))=160px**

**flex-shrink:元素宽度之和大于总宽度时，例：总宽度600px，元素宽度300px，flex-grow分别等于1，3，1。宽度分别为300-(300\*3-600)\*(1/5))=240px，300-(300\*3-600)\*(3/5)=120px，300-(300\*3-600)\*(1/5)=240px**

**flex-basis等价于width**

**flex-warp:换行**

**属性值：warp/nowarp**

#container{

flex-direction:row;

flex-wrap:wrap;

}

==

#container{

flex-flow:row wrap;

}

**元素对齐：**

**① justify-content:除子元素以外主轴方向上空白部分的布局（main axis）（左右）**

**属性值：flex-start、flex-end、center、space-between、space-around**

**space-between和space-around区别：前者首尾在边界，后者均匀，前后空白占中间空白的一半。**

**② align-items:子元素的对齐方式(cross axis)（上下）**

**属性值：flex-start、flex-end、center、baseline、stretch**

**align-items是容器元素的样式属性，指定容器内所有元素**

**align-self用于指定单个元素的对齐方式。**

**align-content:指定行对齐（范围大于align-items）**

**属性值：flex-start、flex-end、center、space-between、space-around(上下)**

**5、calc方法**

**width:calc(50% -100px):宽度=父元素宽度的50%-100px**

**height:calc(10em+3px)：可以用来对不同计数单位混合运算**