# HTML

## HTML基础

### 纯文本

常见的.txt文件，即文本文档

对比常用的word文件

.doc ：在传输过程中，同时传递内容和样式，文件的大小包含文字和其样式

.txt ：在传输过程中，只能传递内容，不能传递样式，文件大小只含文字

纯文本文件特点：

1. 只有文本，没有样式
2. 用记事本等纯文本编辑器可以读写
3. Html/CSS/js等等都是纯文本文件

### 什么是HTML

Hyper text Markup Language:超文本标记语言，是用来描述网页的一种语言。

作用：负责描述文档语义的语言

HTML就是用文本来描述文本的语义，所用到的文本即超文本，超文本在浏览器渲染时不显示的。

HTML不负责描述修改文本的样式，只是通过标签给文本增加语义

例如：

h1标签的作用是什么？

给文本增加主标题的语义。

我们写文本是会区分语义的：比如主题是什么，哪里是段落。计算机不会自动区分语义的，那么就需要我们去给普通文本添加语义，那我们用到的这些文本就是超文本。

超文本在浏览其中是不会显示的，但是在编辑器中就是普通文本。

超文本就是用来添加语义的

例如：

<h1></h1> 将两个标签内部的文本添加了一个一级标题的语义

<h2></h2> 将两个标签内部的文本添加了一个二级标题的语义

<h3></h3> 将两个标签内部的文本添加了一个三级标题的语义

<h4></h4> 将两个标签内部的文本添加了一个四级标题的语义

<h5></h5> 将两个标签内部的文本添加了一个五级标题的语义

<h6></h6> 将两个标签内部的文本添加了一个六级标题的语义

<p></p> 将两个标签内部的文本添加了一个段落的语义

常见问题

h1标签的作用是什么？

并不是将文字加粗增大，自己独占一行，他真正的作用就是添加一个一级标题，真正的样式是由css来控制的。（css我们后面会讲到）

### Html结构

### 基本骨架



<html>双标签他有一个属性

lang = “en” 规定我们html文件标签都必须使用英文。全部小写

字符集：

meta 单标签 charset设置编码格式 UTF-8包含全世界各种语言的国际标准字库

国家标准字库：gb2312/ gbk

<head>双标签

<body>双标签



## Html语法

#### html标签

Html语言对换行和空格不敏感 （注意书写规范）

Html中所有文字内容之间的空格、换行、缩进都会被折叠成一个空格

HTML 标签是由*尖括号*包围的关键词，比如 <html>

HTML 标签通常是*成对出现*的，比如 <b> 和 </b>

标签对中的第一个标签是*开始标签*，第二个标签是*结束标签*

开始和结束标签也被称为*开放标签*和*闭合标签*

<标签>内容</标签>



标签一定要写完，写正确，否则可能会影响其他标签



### 基础标签

#### H标签

<h1> 一级标题标签

<h2> 二级标题标签

<h3> 三级标题标签

<h4> 四级标题标签

<h5> 五级标题标签

<h6> 六级标题标签

#### P标签

<p> 段落标签

<br/> 换行

<hr/> 分隔线标签

实例:

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <title>我的第一个标题</title>  </head>  <body>  <h1>这是标题 1</h1>  <h2>这是标题 2</h2>  <h3>这是标题 3</h3>  <h4>这是标题 4</h4>  <h5>这是标题 5</h5>  <h6>这是标题 6</h6>  </body>  </html> |

### 特殊符号

&gt; 大于号

&lt; 小于号

&quot; 引号

&nbsp; 空格

&copy; 版权

&reg; 注册

### 插入图片

img: 图像标签

src: 图像地址（必须写的属性）

路径：

相对路径：（不可以跨盘符）

同级查找

子集查找

上级查找

绝对路径：

alt: 图像的代替文字

title: 图像的鼠标提示文字

width: 图像宽度

height: 图像高度

示例:

### a标签

超级链接也叫超链接，可以跳转到相应的网站。

属性：

href：属性值是我们要跳转的网页的路径

target:目标的意思。设置我们的跳转后的网站是否在新窗口打开，默认在原来的窗 口打开。如果想要在新的窗口中打开，需要添加属性值：“\_blank” blank是空白的意思

title：鼠标悬停文本

#### 页面内锚点的跳转

添加锚点：

1. 给对应的标签位置添加id属性 属性值就是id名
2. 在需要添加锚点的位置，加入一个a标签，不用书写href属性，添加一个name，命名和id一样

跳转锚点：

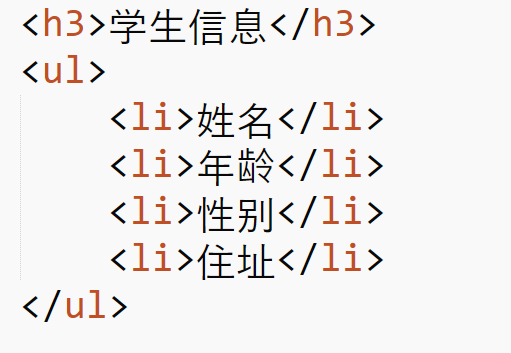
将href属性值书写为#id名，#name名

#### 跨页面锚点跳转

## 列表

### 无序列表

使用ul标签结合li标签一起使用



##### li中可以存放其他标签，ul，li也可以

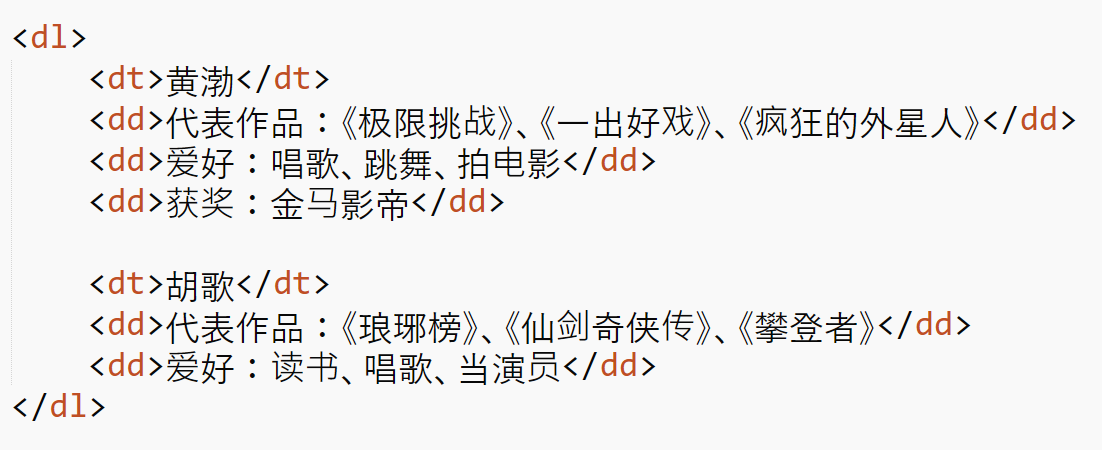


### 有序列表

有序列表和无序列表很相似。

有序列表标签是：ol和li一起使用

### 定义列表



## 表格：

### 表格基础

table tr td

表格的标签之间是嵌套关系。table>tr>td

table : 表格

tr : 表格的行

th: 表头

td：表格的单元格

#### 合并单元格

rowspan:合并行单元格。属性值是数字，是几表示合并几个单元格。

colspan :合并列单元格。属性值是数字，是几就表示合并几个单元格。

在网页中做出此表格

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 |
| 6 | 7 | | 8 | |
| 9 | 10 | 11 | 12 |
| 13 | | 14 | 15 |

## audio音乐标签



## video视频标签



## 表单

form表单所有的表单元素都需要写在form标签内部，不是结构性的标签，而是功能性标签。

功能：规定我们提交的数据发送到哪里，发送的方法是哪一种

method：方法，提交数据的方法，属性值get、post

action：提交的位置

#### 1、文本框

允许用户输入文本

input输入

<input />

<input type=”text” value=”默认文本”/>

#### 2、密码框

type=“password”

#### 3、按钮

type属性的值为：

按钮：button

提交按钮：submit

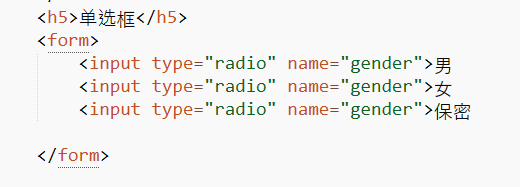
重置按钮：reset



### 4、单选框

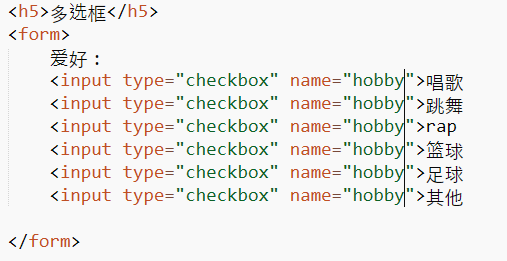
type=“radio”，同组单选框只能选一个

性别：



#### 5、多选框

type=“checkbox”



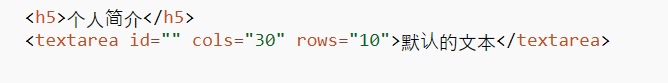
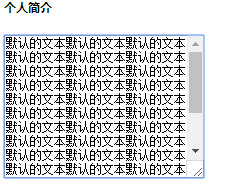
#### 6、文本域

标签：textarea。可以输入多少行

两个属性值：规定显示多少行多少列文字

rows：属性值是数字，表示多少行字

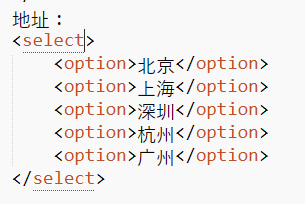
cols：属性值是数字，表示每一行有多少个字节

#### 7、下拉菜单

是一组标签，必须同时出现

select>option



#### 8、文件域

<input type=”file”>

## h5表单类型



## h5表单元素



**练习实例：**



|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>  <meta charset="UTF-8">  <title>注册页面</title>  </head>  <body>  <table border="0" align="center">  <tr>  <td><img src="images/head.png"></td>  </tr>  <tr>  <td>  <form>  <table border="0" cellpadding="10" cellspacing="0">  <tr>  <td width="200" align="right">电子邮箱</td>  <td><input type="text" size="35" /></td>  </tr>    <tr>  <td align="right">会员登陆名</td>  <td><input type="text" size="35" /></td>  </tr>  <tr>  <td align="right">密码</td>  <td><input type="password" size="35" /></td>  </tr>  <tr>  <td align="right">再次输入密码</td>  <td><input type="password" size="35" /></td>  </tr>    <tr>  <td align="right">会员身份</td>  <td>  <input type="radio" name="userType" checked="checked" />买家  <input type="radio" name="userType" />卖家  <input type="radio" name="userType" />两者都是  </td>  </tr>  <tr>  <td align="right">验证码</td>  <td>  <input type="text" size="15">  <img src="images/verycode.gif" align="absMiddle">  &nbsp;&nbsp;&nbsp;&nbsp;<a href="#">看不清?换一张</a>  </td>  </tr>  <tr>  <td></td>  <td><input type="image" src="images/btn\_reg.gif"></td>  </tr>    <tr>  <td></td>  <td><textarea cols="50" rows="10">注册协议  【审慎阅读】您在申请注册流程中点击同意前，应当认真阅读以下协议。请您务必审慎阅读、充分理解协议中相关条款内容，其中包括：  1、与您约定免除或限制责任的条款；  2、与您约定法律适用和管辖的条款；  3、其他以粗体下划线标识的重要条款。  如您对协议有任何疑问，可向平台客服咨询。  【特别提示】当您按照注册页面提示填写信息、阅读并同意协议且完成全部注册程序后，即表示您已充分阅读、理解并接受协议的全部内容。如您因平台服务与淘宝发生争议的，适用《淘宝平台服务协议》处理。如您在使用平台服务过程中与其他用户发生争议的，依您与其他用户达成的协议处理。  阅读协议的过程中，如果您不同意相关协议或其中任何条款约定，您应立即停止注册程序。  淘宝平台服务协议  隐私权政策  法律声明  支付宝及客户端服务协议</textarea></td>  </tr>  </table>  </form>  </td>  </tr>  <tr>  <td><img src="images/footer.png"></td>  </tr>  </table>  </body>  </html> |

# CSS

### 样式引入方式

可以将css样式编写到元素的style属性当中

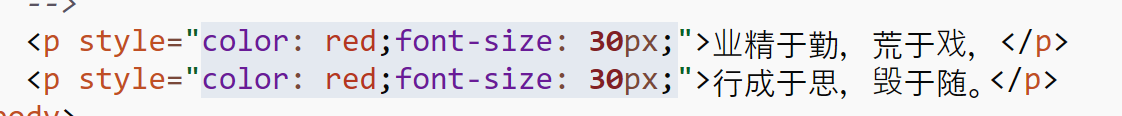
颜色：color 分号结尾

字体：font-size

将样式直接写在style属性中，我们称之为内联样式

内联样式只对当前的元素中的内容起作用，不方便复用

内联样式属于结构和表现耦合，不方便以后维护，不推荐使用



所以：

我们将css写在head标签中的style标签中 内部样式

外部样式

1. 新建一个后缀名为.css的文件，把所有的css代码都放入此文件中。
2. 在html页面中，使用link标签引入此文件

<link href=”css文件的路径” ref=”stylesheet” >



### css语法：

<style type="text/css">

选择器{

颜色;

字体;

...;

}

</style>



复用性增强，其他p标签都可以使用此样式。此方式推荐适合用。

### 选择器：

#### 标签（元素）选择器

语法：

<style type="text/css">

标签(元素){

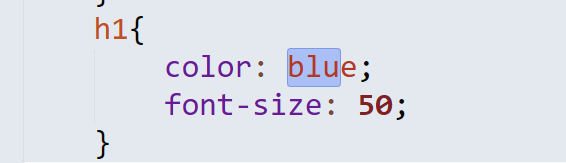
颜色;

字体;

...;

}

</style>



#### id选择器

语法：

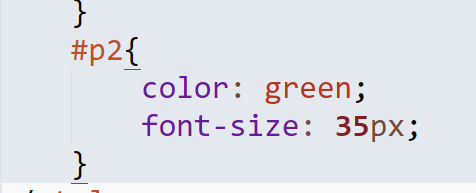
#id属性值{

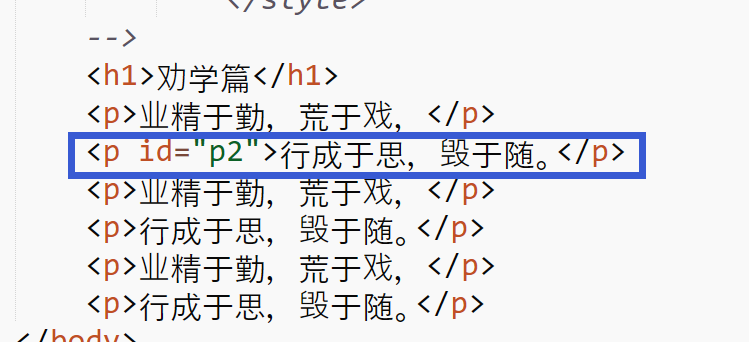
color: green;

font-size: 35px;

。。。；

}





注意：

id属性值是不可以重复的。

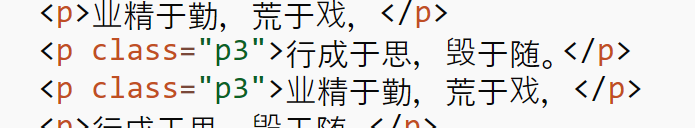
#### 类选择器

语法：

.class属性值{

样式设置

}



#### 交集（复合）选择器

语法：

选择器1选择器2......选择器N{

样式设置

}

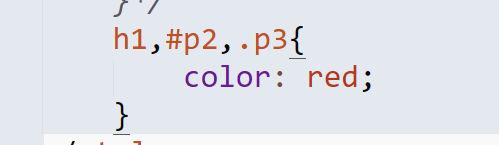
#### 并集（分组）选择器

语法：

选择器1，选择器2......选择器N{

样式设置

}



#### 后代选择器

元素之间的关系

父元素

子元素

兄弟元素

祖先元素

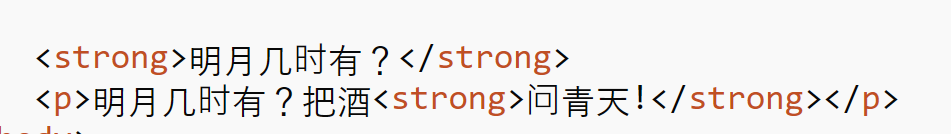
后代元素

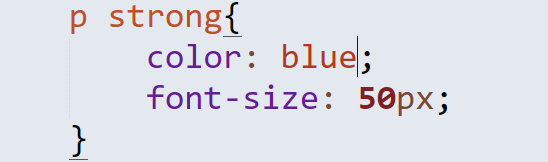
语法：

父元素/选择器 后代标签/选择器{

样式设置；

}



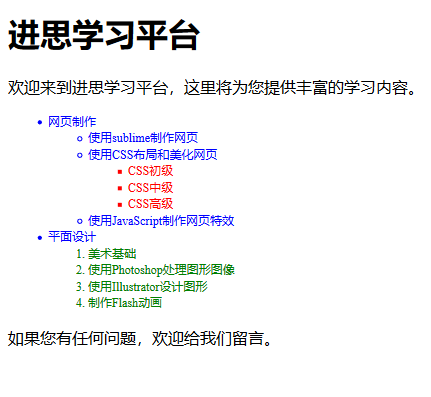


### 继承

继承就像儿子继承父亲的遗产一样，在css中，祖先元素上的样式，也会被他的后代元素所继承，利用继承，可以将一些基本的样式给祖先元素，这样所有后代元素将会自动继承这些样式。

但是不是所有的样式都会被继承。当我们讲到此类标签时，再讨论。

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>  <meta charset="UTF-8">  <title>继承的CSS应用</title>  <style type="text/css">  ul {  color:blue;  font-size:12px;  }  ul li ul li ul li{color:red;}  ul li ol li{color:green;}  </style>  </head>  <body>  <h1>进思学习平台</h1>  <p>欢迎来到进思学习平台，这里将为您提供丰富的学习内容。</p>  <ul>  <li>  <dl>  <dt>网页制作</dt>  </dl>  <ul>  <li>使用sublime制作网页</li>  <li>使用CSS布局和美化网页  <ul>  <li>CSS初级</li>  <li>CSS中级</li>  <li>CSS高级</li>  </ul>  </li>  <li>使用JavaScript制作网页特效</li>  </ul>  </li>  </li>  <li>  <dl>  <dt>平面设计  </dt>  </dl>  <ol>  <li>美术基础</li>  <li>使用Photoshop处理图形图像</li>  <li>使用Illustrator设计图形</li>  <li>制作Flash动画</li>  </ol>  </li>  </ul>  <p>如果您有任何问题，欢迎给我们留言。</p>  </body>  </html> |
|  |



### 文本标签

em标签用于一段内容中的着重点（语气上的强调）。默认斜体

strong标签用于表示一个内容的重要性（表示强调的内容，比em更强烈，默认为粗体）

这两个标签可以单独使用，也可以一起使用。

b标签加粗字体

i标签斜体

h5规范中规定，对于不需要着重的内容而是单纯的加粗或者是倾斜就使用b和i

small标签中的元素会比他的父元素中的文字小：如网站的版权声明

### 单位

长度单位：

计算机常用为：px像素使用最多

一个像素就是屏幕上的小点。

颜色单位：

在css中可以直接使用颜色单词来表示不同的颜色。

red红

blue蓝

green绿

也可使用：RGB值来表示不同的颜色

RGB指的是通过Red Green Blue三原色

通过这三种颜色的不同浓度，来表示出不同的颜色

实例：rgb（红色的浓度,绿色的浓度,蓝色的浓度）;

颜色的浓度需要一个0~255之间的值。

还可以使用十六进制的rgb值来表示颜色，原理和上边RGB原理一样，只不过使用十六进制数来代替，使用三组两位十六进制数组来表示一个颜色。

三组每组代表一个颜色，分别是：red，green, blue

十六进制：

0 ，1，2，3，4，5，6，7，8，9，a，b，c，d，e，f

00表示没有，相当于rgb中的0

ff表示最大值，相当于rgb中的255

红色：

#ff0000 简写：#f00

#aabbcc 简写：#abc

em:他是相对于当前元素的字体大小来计算的。

1em=1font-size

### 字体样式

字体颜色：

color设置文字颜色

字体的大小：

font-size:16px;

字体的设置

font-family：微软雅黑;

注意：当采用某种文字时，如果浏览器支持则使用该字体，如果字体不支持，则使用默认字体。该样式可以同时设置多个字体。

实例：

font-family：华文彩云, 隶书，微软雅黑;

注意：如果客户计算机中没有此字体那么就不可用，回去第二个字体，直到最后默认的，或者找到计算机中含有的字体

字体分类：

serif:衬线字体

sans-serif:非衬线字体

monospace等宽字体

cursive草书 字体

fantasy虚幻字体

我们可以将字体设置为一种大的字体类型，这样浏览器会自动为我们选择其中的某一种字体



字体样式

font-style此属性可以用来设置文字的斜体

属性值：

normal默认样式：正常显示

italic文字斜体

oblique文字倾斜

提示:大部分浏览器不会对倾斜和斜体进行区分。

font-weight:此属性设置文字的粗细

属性值：

normal：默认正常显示

bold：定义粗体字符

bolder：定义更粗体字符

lighter:定义更细的字体

### 文本的样式

#### 设置文本颜色

color:red;

#### 设置文本大小写

可以用来设置文本的大小写

可选值

None默认值，该怎么显示就怎么显示，不做任何处理了

Capitalize 单词的首字母大写，通过空格来识别单词

Uppercase 所有的字母都大写

Lowercase 所有的字母都小写

统一文本格式时使用，对中文没有作用

#### 设置水平对齐方式

text-align: 用于设置文本的对齐方式

可选值：left默认值：文本靠左对齐

right：文本靠右对齐

center:文本居中对齐(两边都不齐)

justify：两端对齐(通过调整文本之间的空格大小，来达到一个两端对齐的目的)

#### 设置首行文本缩进

text-indent

设置首行缩进

32px缩进是不准确

(设置只缩进两个字体大小2em)

当给它指定一个正值时，会自动向右侧缩进指定的像素

如果为它指定一个负值，则会向左移动指定的像素

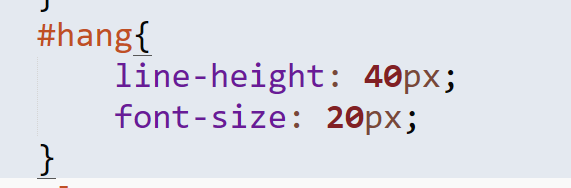
通过这种方式可以将一些不想显示的文字隐藏起来

搜索引擎会看见，用户看不见的东西

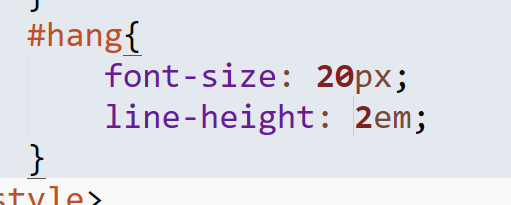
Text-indent：-99999px;）

#### 行高的设置

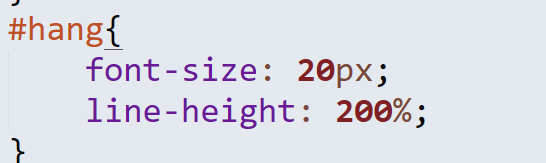
第一种写法



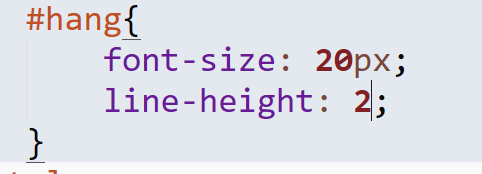
第二种写法使用em



第三种写法：



第四种写法



注意：行高在设置的时候，必须在font的下面，守则font会将行的的值设置为默认值

#### 设置文本的装饰

text-decoration

可以用来设置文本的修饰

可选值

None默认值，不添加任何修饰，正常显示

Underline为文本添加下划线

Overline 为文本添加上划线

Line-through 为文本添加删除线

#### 文本间距

Letter-spacing：可以指定字符间距

实例：letter-spacing:10px(对中英文都有效)

word-spacing：设置单词之间的间距

word-spacing：10px(对中文没什么影响)

#### 垂直对齐方式

vertical-align属性：middle、top、bottom

### 伪类选择器

|  |  |  |
| --- | --- | --- |
| **伪类名称** | **含义** | **示例** |
| **a:link** | **未单击访问时超链接样式** | **a:link{color:#9ef5f9;}** |
| **a:visited** | **单击访问后超链接样式** | **a:visited {color:#333;}** |
| **a:hover** | **鼠标悬浮其上的超链接样式** | **a:hover{color:#ff7300;** |
| **a:active** | **鼠标单击未释放的超链接样式** | **a:active {color:#999;}** |
| **:focus** | **用于选取获得焦点的表单元素**  **焦点就是光标，主要针对表单元素** | **input:focus{ color:#999;}** |

#### 鼠标样式

cursor的可选值：

|  |  |  |
| --- | --- | --- |
| **值** | **说明** | **图例** |
| **default** | **默认光标** | 图1.BMP |
| **pointer** | **超链接的指针** | 图2.BMP |
| **wait** | **指示程序正在忙** | 图3.BMP |
| **help** | **指示可用的帮助** | 图4.BMP |
| **text** | **指示文本** | 图5.BMP |
| **crosshair** | **鼠标呈现十字状** | 图6.BMP |

### 背景

background-color：背景颜色

background-image:url（图片路径） 背景图片

background-repeat属性 背景重复方式

可选值：repeat

背景定位

background-position属性

#### 列表样式：

无序

list-style-type可以不写type

可选值：

none；没有样式

cirle;圆

square方块

disc：圆

有序

list-style-type：

可选值：

upper-roman

lower-roman

lower-alpha

|  |
| --- |
| list-style-type: decimal;  list-style: lower-alpha;  list-style: upper-alpha;  list-style: lower-armenian;  list-style: upper-armenian;  list-style: lower-roman;  list-style: upper-roman; |

list-style-image:

url(图片地址)

### css的两种盒模型

#### W3C的标准盒模型

**在标准的盒子模型中，width指content部分的宽度**

#### IE的盒模型

**在IE盒子模型中，width表示content+padding+border这三个部分的宽度**

#### box-sizing的使用

如果想要切换盒模型也很简单，这里需要借助css3的box-sizing属性

box-sizing: content-box 是W3C盒子模型

box-sizing: border-box 是IE盒子模型

**box-sizing的默认属性是content-box**

#### 边框

##### 设置边框得宽度

|  |
| --- |
| 第一种方式： border-width: 10px;影响了四个边框的宽度  border-width: 10px 20px 30px 40px;顺时针设置四个边框的宽度  border-width: 10px 30px 40px;顺时针设置四个边框的宽度：分别是上 左右 下  border-width: 10px 40px;顺时针设置四个边框的宽度：分别是上下 左右 第二种方式： border-top-width: 10px;  border-right-width: 20px;  border-bottom-width: 30px;  border-left-width: |

##### 设置边框的颜色

第一种方式：

border-color: blue; 设置边框颜色

border-color: blue yellow red green;顺时针设置边框颜色

border-color: yellow red green;顺时针设置：上黄 左右红 下绿

border-color: yellow green;顺时针设置：上下黄 左右红

第二种方式：

border-top-color: red;

border-right-color: yellow;

border-bottom-color: blue;

border-left-color: green;

##### 设置边框样式

border-style

可选值：

none 默认值 没有边框

solid 实线

dotted 点状边框

dashed 虚线

double 双线

##### 边框的简写

大部的浏览器中，边框颜色和宽度是有默认值的。而边框的默认样式都是none。

边框的简写样式，可以同时设置四个边框的样式、颜色、宽度，没有顺序要求

注意：同时指定四个不能分别指定。

实例：

border: red 10px solid；

#### 内边距

分别设置上、右、下、左内边距：

padding属性

[padding-top](mk:@MSITStore:C:\Users\52911\Desktop\W3School离线手册(2015.07.17).chm::/www.w3school.com.cn/cssref/pr_padding-top.asp)

[padding-right](mk:@MSITStore:C:\Users\52911\Desktop\W3School离线手册(2015.07.17).chm::/www.w3school.com.cn/cssref/pr_padding-right.asp)

[padding-bottom](mk:@MSITStore:C:\Users\52911\Desktop\W3School离线手册(2015.07.17).chm::/www.w3school.com.cn/cssref/pr_padding-bottom.asp)

[padding-left](mk:@MSITStore:C:\Users\52911\Desktop\W3School离线手册(2015.07.17).chm::/www.w3school.com.cn/cssref/pr_padding-left.asp)

设置内边距简写：

padding: 100px;同时设置四个内边距的宽度

padding: 100px 200px;设置上下 左右 内边距的宽度

padding: 100px 200px 300px;设置上 左右 下 内边距的宽度

padding: 100px 200px 300px 400px;顺时针设置内边距的宽度

#### 外边距：

外边距指的是：当前盒子与其他盒子之间的距离，他不会影响可见框的大小，而是会影响到盒子的位置。盒子有四个方向的外边距。

属性：margin

[margin-top](mk:@MSITStore:C:\Users\52911\Desktop\W3School离线手册(2015.07.17).chm::/www.w3school.com.cn/cssref/pr_margin-top.asp)

[margin-right](mk:@MSITStore:C:\Users\52911\Desktop\W3School离线手册(2015.07.17).chm::/www.w3school.com.cn/cssref/pr_margin-right.asp)

[margin-bottom](mk:@MSITStore:C:\Users\52911\Desktop\W3School离线手册(2015.07.17).chm::/www.w3school.com.cn/cssref/pr_margin-bottom.asp)

[margin-left](mk:@MSITStore:C:\Users\52911\Desktop\W3School离线手册(2015.07.17).chm::/www.w3school.com.cn/cssref/pr_margin-left.asp)

值也可以指定负值

上面四个的值都可设置为：auto表示自动

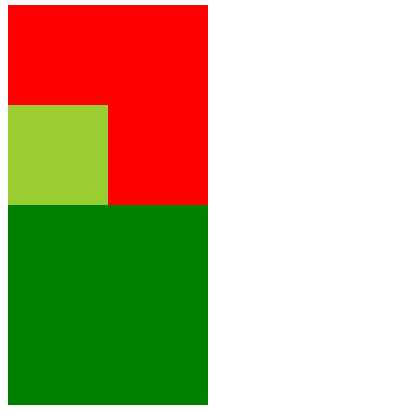
如果只设置左为auto会变成最大，如果是垂直则为0

那么通常我们将 左和右同时设置为auto

简写：

margin：2px 3px 4px 5px;

通常写法：margin：0px auto;设置水平居中





垂直外边距的重叠

在网页中垂直外边距的会发生外边距的重叠。

所谓的外边距重叠指的是：兄弟间的相邻外边距会取最大值而不是求和

如果父子元素边距相邻。那子元素会带着父元素一起移动

#### 内联元素的盒模型

内容区：不能设置宽和高

内边距：可以设置水平方向的内边距，垂直方向的边距也可以设置，但是不会影响页面布局

边框： 可以设置边框，垂直边框不会影响页面布局，

外边距：水平方向的相邻外边距不会重叠，而是求和，不支持垂直外边距。

### display

内联元素不支持宽和高，我想将内联元素变成块元素怎么办？

使用display属性

属性可选值:

none 不显示元素，并且元素不会在页面中继续占有位置

inline可以将一个元素作为内联元素显示行内元素/内联元素

block可以将一个元素设置为块元素

inline-block 将一个元素转换为行内块元素

它可以使一个元素既有行内元素的特点又有块元素的特点

既可以设置宽高又不独占一行。

### visibility

可选值：

visible 默认值，元素默认会在页面显示

hidden 元素会隐藏不显示，但是位置还在

### overflow

子元素是默认是存在与父元素的内容区中的，

理论上讲元素的最大值可以等于父元素内容区大小

如果子元素的大小超过了父元素的内容区，则超过的大小会在父元素以外的位置显示

超出的内容我们称为:溢出内容

父元素默认是将溢出内容，在父元素外边显示的

通过overflow可以设置父元素如何和处理溢出内容。

可选值：

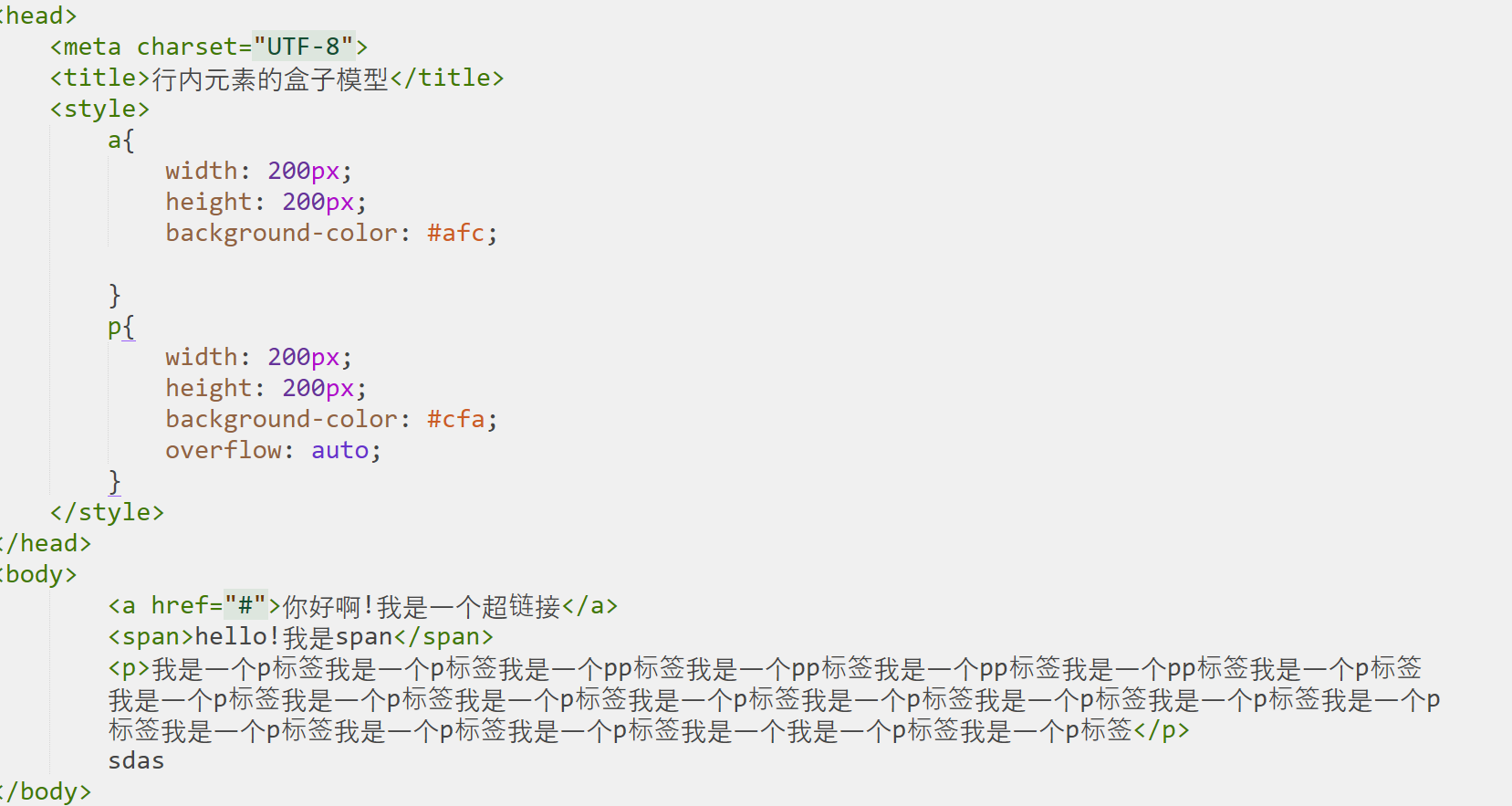
visible 默认值，不会处理溢出内容

hidden 溢出的内容会被修剪

scroll 会为父元素添加滚动条，通过拖动滚动条来查看完整内容

注意：它不会管你有没有内容，有没有内容溢出都会加上滚动条

auto：自动调配

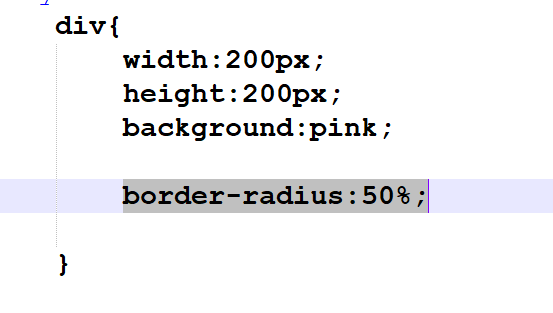


### 圆角边框：

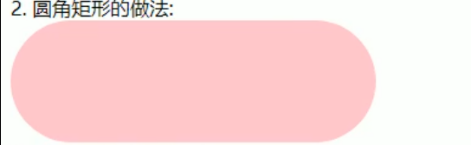
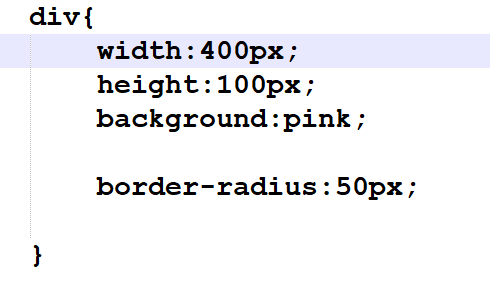
border-radius:length;

圆形的做法：

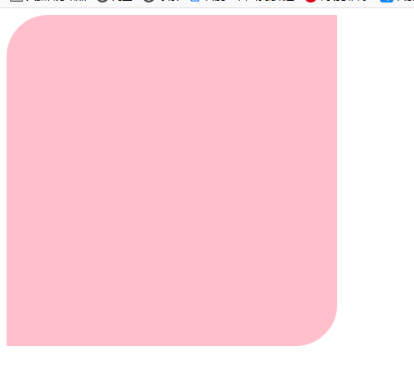
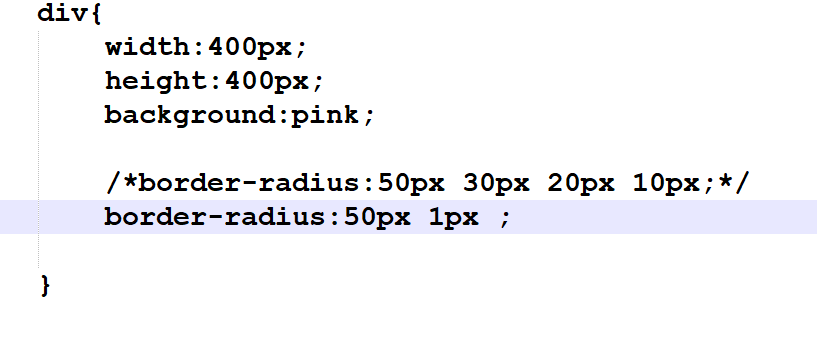
下面的示例中：50%取的是宽度和长度的一半 所以页面显示的是一个圆

圆角矩形的做法：设置长度为高度的一半即可

设置四个角任意弧度：border-radius:左上角 右上角 右下角 左下角;

 分开写：border-top-left-radius、border-top-right-radius、border-bottom-left-radius、border-bottom-right-radius

### 盒子阴影

css3新增：盒子阴影

属性：box-shadow



注意：outset是默认的，但是它不能写在代码中，写了就会失效。

盒子阴影不占用空间，不会影响其他盒子的排列

### 文字阴影

css3新增属性：可以给文字添加阴影，使用：text-shadow



### 传统网页布局的三种方式：文档流、浮动、定位

#### 标准流（文档流/普通流）

所谓的文档流：就是标签按照规定好的方式排列。

### 浮动

块元素

在文档流中默认垂直排列，如果想要块元素在页面中水平排列，可以使块元素脱离文档流，使用float来使元素浮动，从而脱离文档流（脱标），

可选值：

none 默认值，元素默认在文档流中排列，

left：元素会立即脱离文档流，在页面的左侧浮动。

right：元素会立即脱离文档流，向页面的右侧浮动。

当为一个元素设置浮动以后，元素会立即脱离文档流，元素脱离文档流以后，他下边的元素会立即向上移动。如果浮动元素上边是一个没有浮动的元素，则浮动元素不会超过块元素。浮动的元素不会超过它上面的兄弟元素，最多一边齐。遇到父类的边框和浮动的元素就会停止浮动

注意：

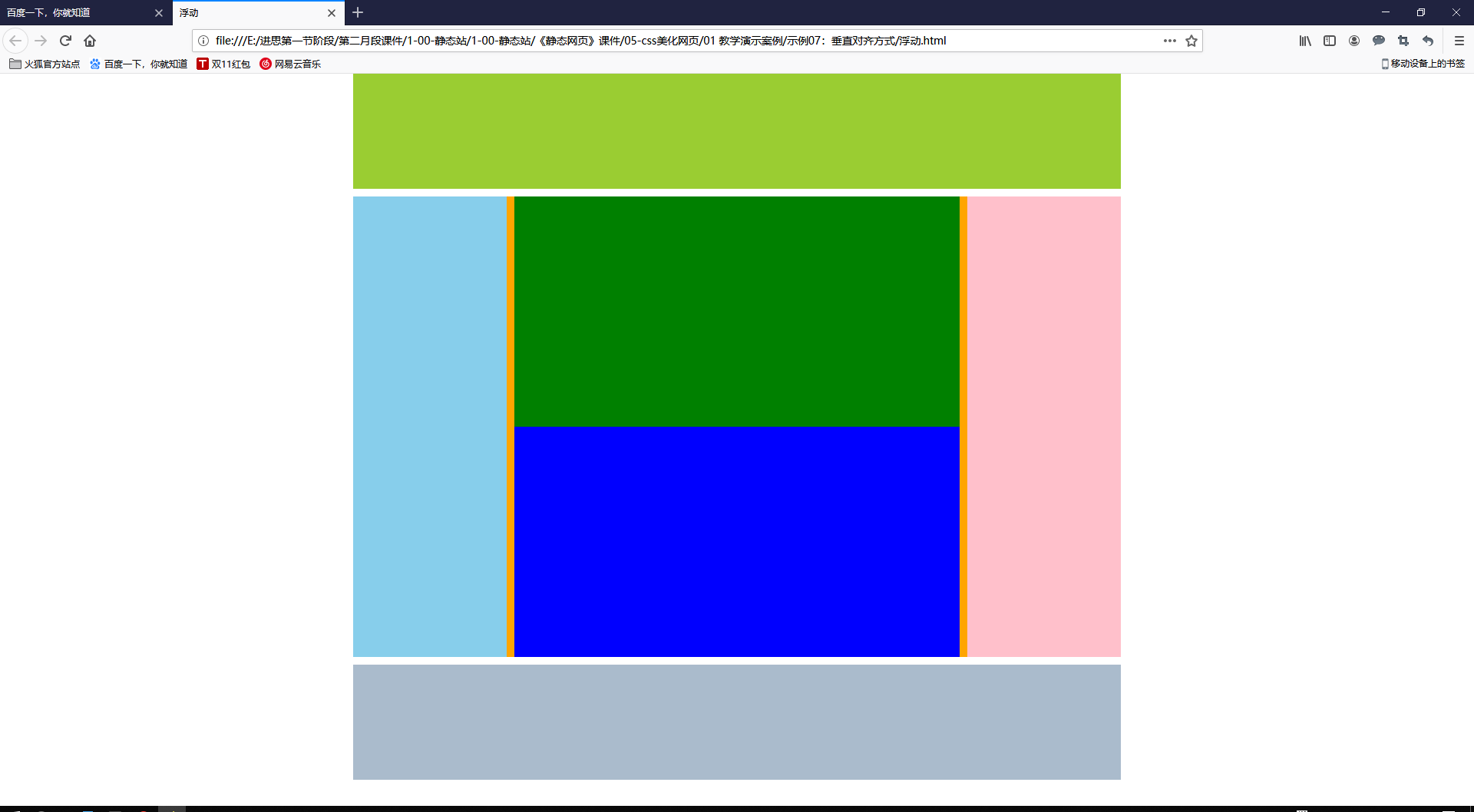
1、浮动的元素不会盖住文字。

2、在文档流中，子元素的宽度默认为父元素的全部，当元素设置浮动之后，会完全脱离文档流，宽高会完全被内容撑开。

内联元素

当内联元素开启浮动之后，也会脱离文档流，会变成块元素。

#### 页面设计



### 常见的网页布局

### 清除浮动（高度塌陷问题）

#### 第一种解决方式:

当父元素没有设置高度，而子元素设置了浮动，那么就会出现高度塌陷问题。

关于高度塌陷问题：

在文档流中，父元素的高度默认是被子元素撑开的。也就是子元素多高，父元素就多高

但是当为子元素设置浮动以后，子元素会完全脱离文档流。此时有于父元素的高度塌陷了，则父元素下的所有元素都会向上移动，这样会导致页面布局混乱。

所以在开发中一定要避免出现高度塌陷问题。

我们可以将父元素的高度写死，这样可以避免塌陷问题的出现，但是一但高度写死，父元素的高度将不能自动适应子元素的高度，所以这种方案是不推荐使用的

解决高度塌陷：

原理：

根据w3c的标准，在页面中元素都有一个隐含的属性叫做Block Foramtting Context

块格式化上下文简称BFC,该属性可以设置打开或者关闭，默认是关闭的。

当开启元素的BFC 以后元素将会具有一下特性：

1. 父元素的垂直外边距不会和子元素重叠
2. 开启BFC的元素不会浮动元素所覆盖。
3. 开启BFC的元素可以包含浮动的子元素。

如何开启元素的BFC

1. 设置元素浮动

——此方式开启，虽然可以撑开父元素，但是会导致父元素的宽度丢失

而且也会导致下面的元素上移

1. 设置元素的绝对定位
2. 设置元素为inline-block

——可以解决问题，但是也会导致宽度的丢失，不推荐使用这种方式

1. 设置父元素overflow

将值设置为：hidden或者auto 一般设置为hidden

#### 第二种解决方式:（官方推荐）

有时候我们希望清除掉其他元素浮动对当前元素产生的影响，这时可以使用clear来完成功能。

clear可以用来清除其他浮动元素对当前元素的影响

可选值：

none:默认值，不清除浮动

left清除左侧浮动对当前元素的影响

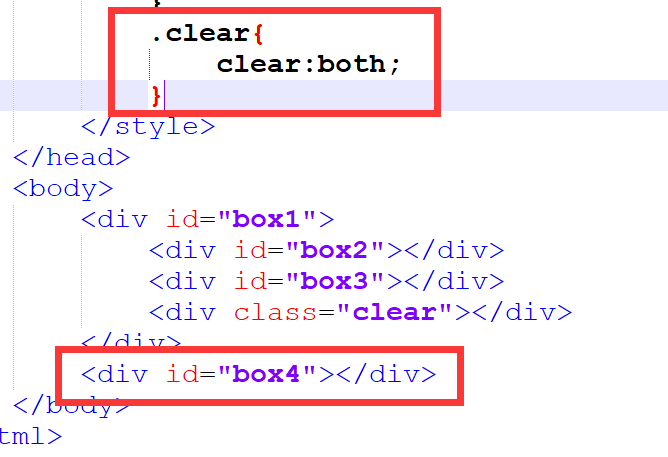
right清除右侧浮动对当前元素的影响

both:清除两侧浮动对当前元素的影响

实际上是清除当前元素两侧对其影响最大的浮动

方法：额外标签法（隔墙法）

就是在最后一个浮动的子元素后面添加一个空的块元素,清除此空的块元素两侧浮动的影响



这种写法：通俗易懂，书写简单，缺点是添加许多无意义的标签，结构化较差。

#### 第三种解决方式

结合伪元素一起使用

伪元素选择器可以帮助我们利用css创建一个新的标签元素，而不需要html标签，从而简化html结构

伪元素右那些：

::before 在元素内部的前面插入内容

::after 在元素内部的后面插入内容

注意：

before 和 after创建的是一个行内元素

创建的元素在页面中是找不到的，所有称为伪元素

语法：

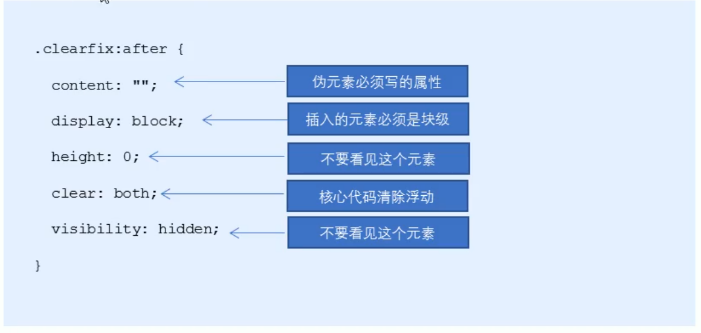
element::after/before{ }

before和after中必须右content属性

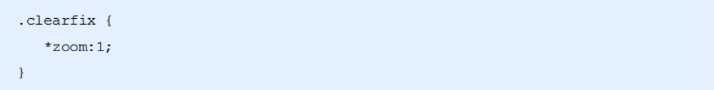
伪元素选择器和标签选择器的权重一样都是1；

css2中伪元素选择器的写法为：一个冒号 css3中改为了一个冒号

如何使用伪选择器解决高度塌陷：

 为开启浮动的元素的父元素添加class=“clearfix” 属性 在css中输入一下代码，类似于

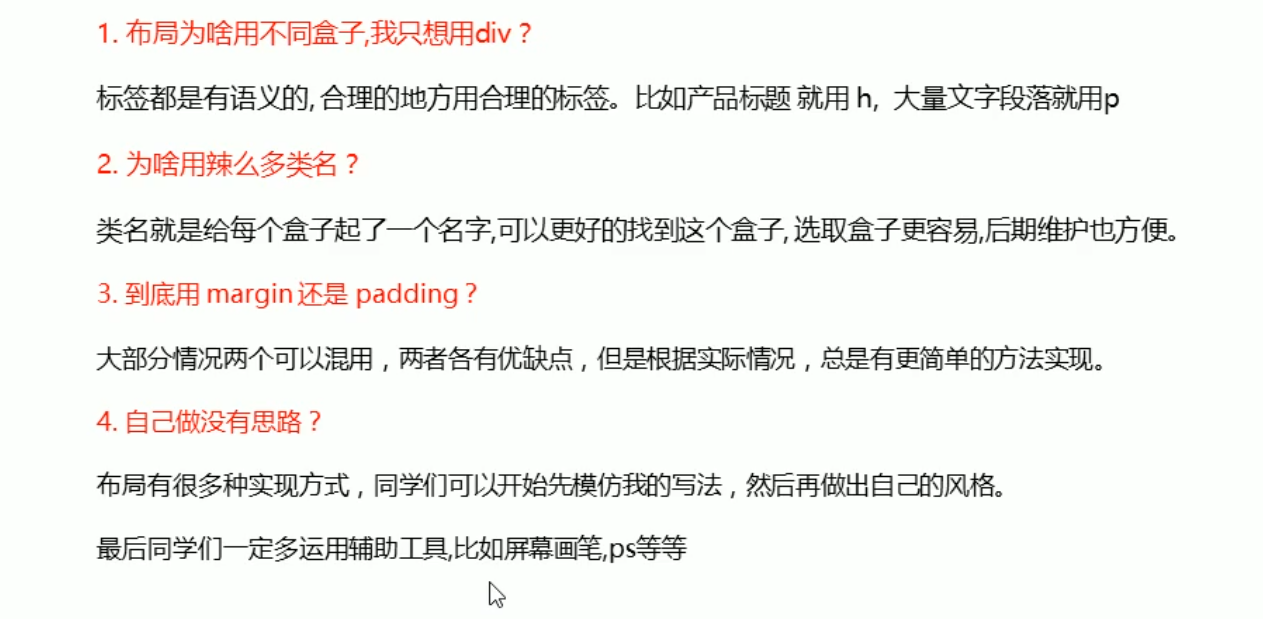
第二种方法

 兼容IE6

第四种方法：双伪元素清除浮动法



### 解惑以及经验



### 定位

什么是定位？为什么要开启定位？

定位指的是将指定的元素移动到任意位置

通过定位就可以任意摆放元素

通过position属性来设置元素的定位

可选值：

static：默认值，元素没有开启定位

relative：开启元素的相对定位

absolute：开启元素的绝对定位

fixed：开启元素的固定定位（也是绝对定位的一种）

sticky:粘性定位（了解）IE不支持

#### 相对定位

当开启相对定位时：

可以通过：left right top bottom四个属性来设置元素的偏移量

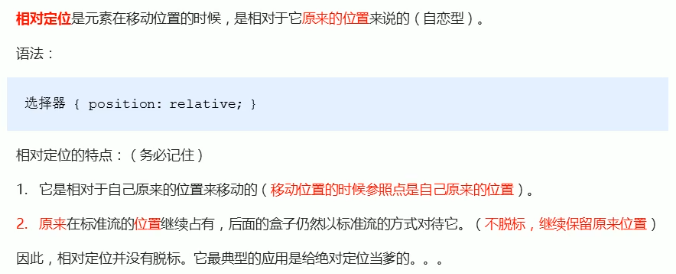
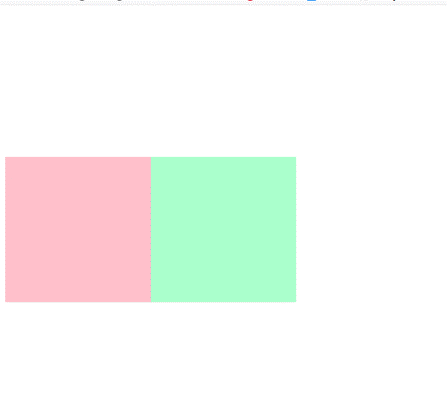
left：元素相对于其定位位置的右侧偏移量

right:元素相对于其定位位置的右侧偏移量

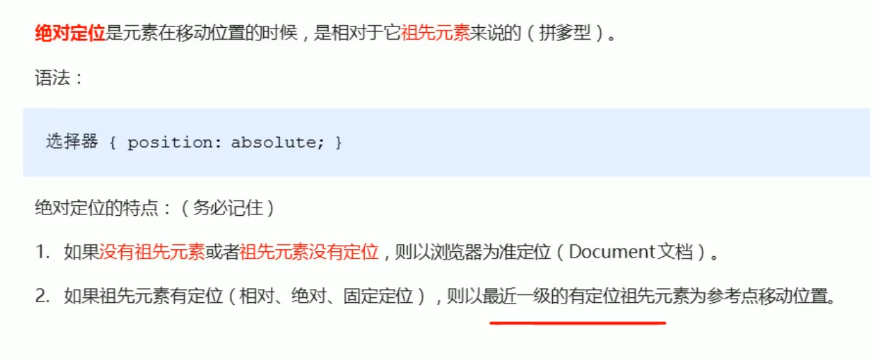
top：元素相对于其定位位置的上边的偏移量

bottom：元素相对于其定位位置下边的偏移量

特点：

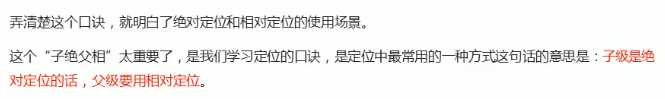
 相对定位的元素不会脱离文档流，而且提升一个层级。

#### 绝对定位

开启绝对定位：

当position属性的值为absolute时，则开启元素的绝对定位

绝对定位：

1. 开启绝对定位，会使元素脱离文档流
2. 开启绝对定位以后如果不设置偏移量元素的位置不会发生变化
3. 绝对定位是相对于离他**最近的**开启了定位的祖先元素进行定位的（一般开启了子元素的绝对定位，都会同时开启父元素的相对定位 ）——子绝父相

——如果所有的祖先元素都没有开启定位，则会相对于浏览器窗口进行定位

1. 提升一个层级
2. 绝对定位会改变元素的性质

-内联元素变成块元素

-块元素的宽度和高度会被内容撑开。

#### 固定定位

 position的值为：fixed。就开启了元素的固定定位

固定定位也是绝对定位的一种，他的特点基本和绝对定位一样

不同的是：

固定定位永远都相对于浏览器窗口进行定位

固定定位会固定在浏览器窗口某个位置，不会随滚动条滚动。

##### 元素的层级：

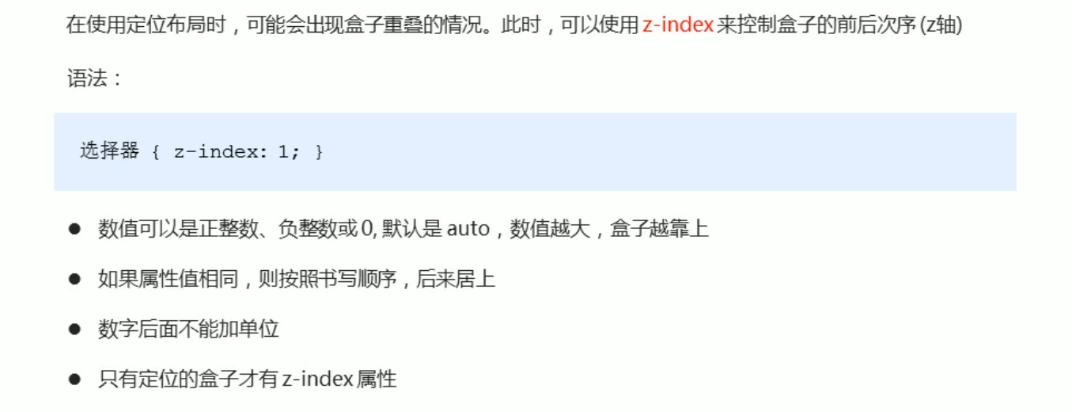
如果定位元素的层级是一样的，则下边元素会盖住上边的

通过z-index属性可以用来设置元素的层级

可以为z-index指定一个正整数作为值，该值将会作为当前元素的层级

层级越高，越优先显示。

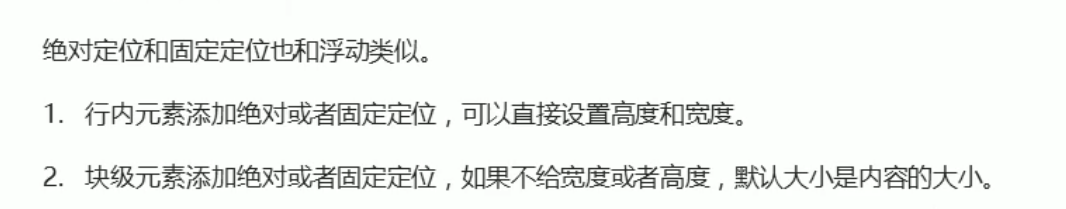
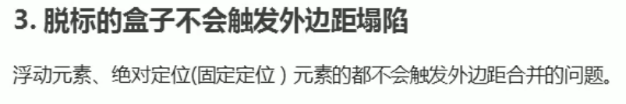
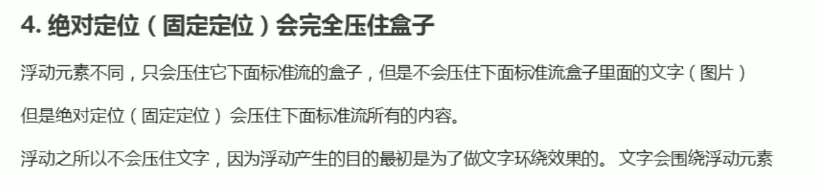
注意：对于没有开始定位的元素不能使用z-index

 父元素的层级再高，也不会盖住子元素。

#### 粘性定位



#### 定位总结

注意：

### 背景透明

设置背景透明使用属性：opacity

值为0到1之间

0代表完全透明

1代表不透明

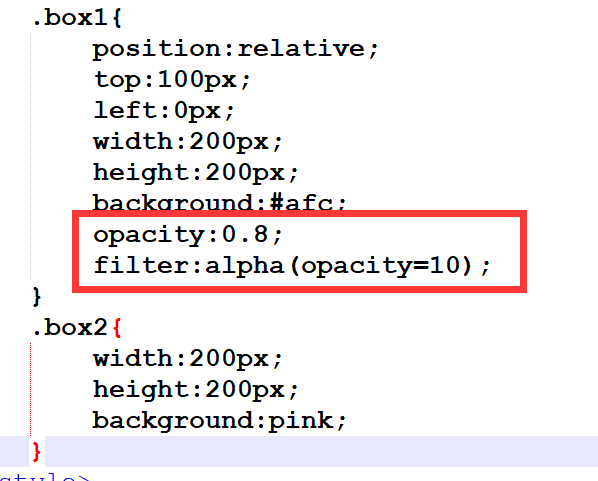
IE8以下：

透明度设置是：

alpha(opacity=透明度)

透明度，需要一个0-100之间的值0代表完全透明、100代表完全不透明

实例：

 filter:alpha(opacity=50)

### Chrome调试工具

## 精灵技术

### 为什么要用精灵技术

### 精灵技术的使用（原理）

精灵技术主要针对于背景图片使用。就是把多个小背景图片整合到一张大图片中。

sprites精灵图或者雪碧图

### 精灵图的使用

利用background-position

### 字体图标的产生和优点

#### 字体图标的产生

字体图标使用场景:主要用于显示网页中通用、常用的一些小图标

精灵图是有诸多优点的，但是缺点很明显

1. 图片文件还是比较大的
2. 图片本身放大和缩小会失真
3. 一旦图片制作完毕想要更换非常复杂

此时，有一种技术的出现很好的解决了以上问题，就是字体图标iconfont

字体图标可以为前端工程师提供一种方便高效的图标使用方式，展示的是图标，本质属

于字体

#### 字体图标的优点

轻量级：一个图标字体要比一系列的图像要小，一旦字体加载了，图标就会马上渲染出

来，减少了服务的请求

灵活性：本质是文字，可以随意改变文字的颜色、产生阴影、透明效果、旋转等

兼容性：几乎支持所有浏览器

注意：字体图标不能替代精灵技术，只是对工作中的图标部分技术的提升和优化。

总结：

1、遇到一些结构小和样式比较简单的小图标，就用字体图标。

2、如果遇到一些结构和样式复杂一点的小图片，就是精灵图。

#### 字体图标下载

1、icomoon字库<http://icomoon.io>

icomoon成立于2011年，推出了第一个自定义图标字体生成器，它允许用户选择所需

要的图标，使他们成一字型。该字库内容种类繁多，非常全面，唯一的遗憾是国外服务  
器，打开速度慢。

2、阿里iconfont字库 <http://www.iconfont.cn>/

这个是阿里妈妈M2UX的一个iconfont字体图标字库，包含了淘宝图标和阿里妈妈图标库。可以使用AI制作上传生成。

#### 字体图标的使用



#### 字体图标的追加和加载原理

上传json文件——选择新的重新下载，替换原来文件

## css三角的做法

### css三角的应用:透明边框线 transparent

### css三角强化运用



## 表单元素的轮廓线

outline: none

## 文本域的拖拽问题

resize: none

## 溢出文字省略号

### 单行文本溢出显示省略号—必须满足三个条件

1. 先强制一行内显示文本

white-space:nowrap(默认normal自动换行)

1. 超出的部分隐藏

overfloat: hidden;

1. 文字用省略号代替超出的部分

text-overflow: ellipsis;

### 多行文本内容溢出显示省略号

多行文本溢出显示省略号，有较大兼容性问题，适用于webkit浏览器或移动端(移

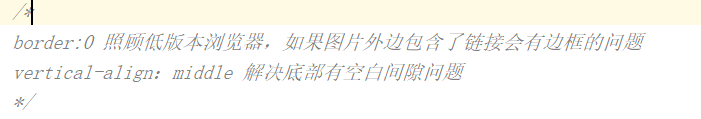
动端大部分是webkit内核)

1. overflow: hidden;
2. text-overflow: ellipsis;
3. 弹性伸缩盒模型显示  
   display:-webkit-box;
4. 限制在一个块元素显示的文本的行数  
   -webkit-line-clamp:2;
5. 设置或者检索伸缩盒对象的子元素的排列方式  
   -webkit-box-orient: vertical;

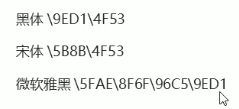
## 图片底侧空白间隙的解决方案：因为默认机械对齐

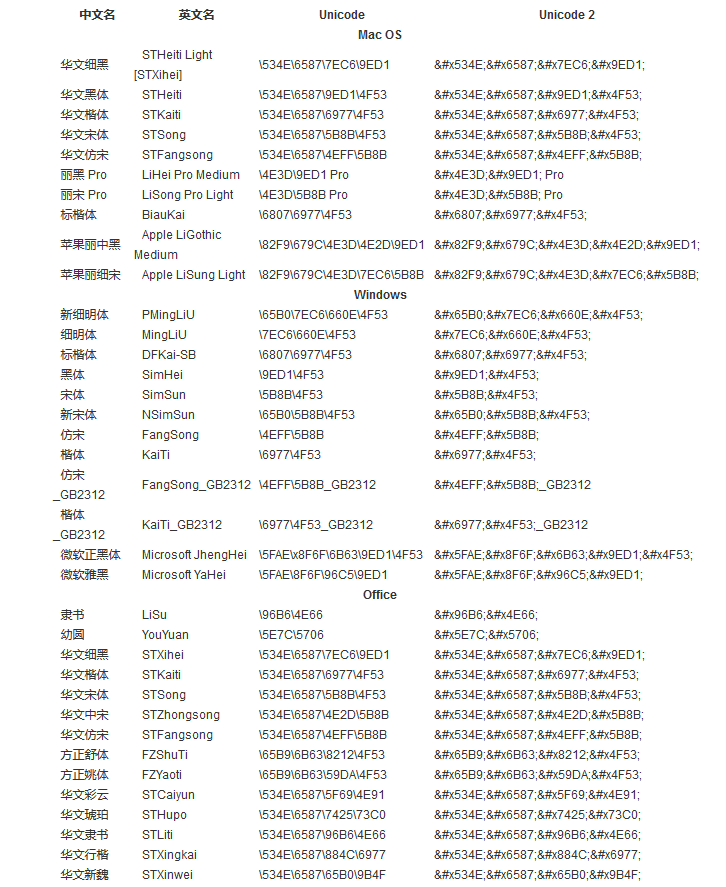
vertical-align:bottom;

## css的初始化









# H5

## 新增语义化的标签

<header> 头部标签

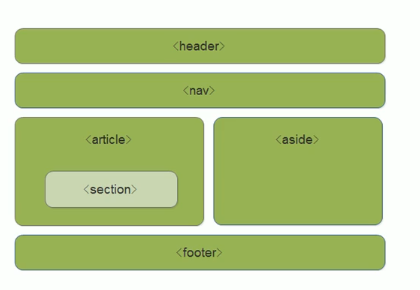
<nav> 导航栏

<article> 内容标签

<section> 定义文档某个区域

<aside> 侧边栏标签

<footer> 尾部标签



注意：

这种语义化标准主要针对搜索引擎的

这些标签页面中可以使用多次

再IE9中，需要把这些元素转换为块级元素

其实，我们移动端更喜欢使用这些标签

HTML5还增加了很多其他标签，一会慢慢学

## 视频音频

## 新增input表单类型

email url…

tel 手机号码

search 搜索框

color 生成一个颜色选择表单

## 新增表单属性

|  |  |  |
| --- | --- | --- |
| required [rɪˈkwaɪərd] | required | 拥有此属性之后的元素不能为空，必填 |
| placeholder | 提示文本 | 表单的提示内容，有默认值就不会显示 |
| autofocus[ˈɔːtoʊfoʊkəs] | autofocus | 自动聚焦，页面加载完毕，自动聚焦到指定表单 |
| autocomplete [kəmˈp liː t] | off/on | 当用户开始键入时，浏览器基于之前键入过的值，应该显示出在字段中填写的选项，默认打开值为on，  关闭为off，必须结合name此属性才能生效 |
| multiple/ mʌltɪpl / | multiple | 可以多选文件提交 |

# C3

c3特性具有兼容性问题，ie9+支持

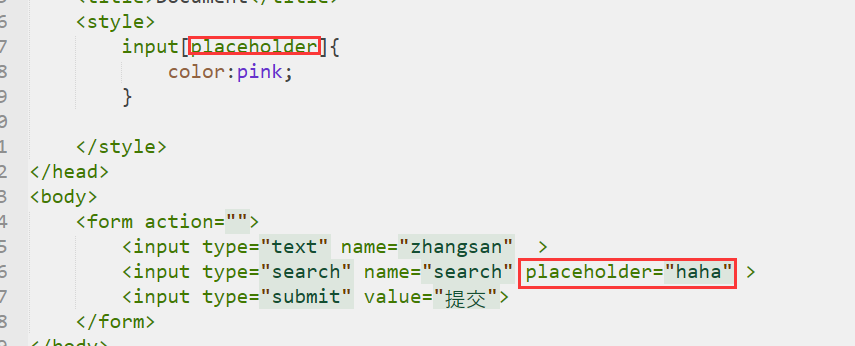
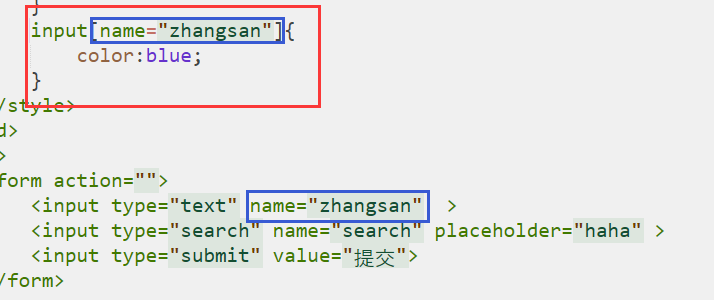
移动端支持优于pc端

不断改进中

应用广泛

现在主要学习：新增属性选择器、盒子模型以及其他特性

## 新增属性选择器

1. 属性选择器
   1. E[att] 根据特定的属性来选择元素 [ 属性 ]   
      示例：
   2. E[att=”val”] 根据 属性=属性值 来选择元素  
      示例：  
      
   3. E[att^=”val”] 选择属性值以val开头的元素  
      
   4. E[att$=”val”] 选择属性值以val结尾的元素
   5. E[att\*=”val”] 选择属性值中包含val的元素

注意：类选择器、属性选择器、伪类选择器 权重都为10

1. 结构伪类选择器  
   根据父元素选择其中的子元素
   1. E :first-child选择第一个子元素
   2. E :last-child选择最后一个子元素
   3. E :nth-child(n)选择第n个子元素

n可以是数字可以是公式或者关键字  
数字：代表第几个子元素

关键字：even偶数 odd奇数

公式：n代表从0开始计算，但是第0个和超出的忽略

2n表示偶数

2n+1表示奇数

6n表示 6，12，18......

n+6表示从第六个开始

 -n+6表示前六个包含6

* 1. E :first-of-type指定类型E的第一个
  2. E :last-of-type指定类型E的最后一个
  3. E :nth-of-type(n)指定类型E的第n个

总结：nth-child和nth-of-type的区别

E:nth-child：对父元素里面所有子元素进行排序选择，先找到第n个子元素，然后

匹配是否符合（是否是E元素）。（使用较多）

E:nth-of-type：对父元素里面指定的子元素（E）进行排序，再匹配是否是E元素

1. 伪元素选择器

:: before

:: after

## css3盒模型

为了不再设置padding和border时，使我们的盒子变大，我们将盒模型切换为

box-sizing : border-box;

box-sizing的默认值为content-box;

## css的其他特性

### css3滤镜filter: blur(5px); 模糊处理,数值越大，越模糊

### 计算盒子的宽度：calc()函数

calc()此css函数可以让我们再声明css属性时执行一些计算

 width: calc(100% - 30px); (+ - \* /)

京东就是用的此技术

### 过渡

IE9+支持，不支持没关系 不影响页面布局

transition:要过渡的属性 花费时间 运动曲线 何时开始；

1. 属性：想要变化的css属性，高度过度、背景颜色过度 内外边距都可以过度。想要  
    所有属性都变化过渡，就写all。
2. 花费时间：单位时秒比如1s(1秒)
3. 运动曲线：默认时ease(可以省略)
4. 何时开始：单位是 秒 可以设置延迟出发时间，默认是0s（可以忽略）

## favicon的制作

使用ps将logo图标提取出来，打开比特虫将图片制作成图标ico。

如何将ico图标引入页面：

把

**<link rel="shortcut icon" href=" favicon.ico" />**

**放入<head>标签中**

## 网站TDK三大标签SEO

### SEO搜索引擎优化：

是一种利用搜索引擎规则，提高网站在有关搜索引擎内的自然排名

搜索引擎：百度、谷歌........

SEO就是对网站进行深度优化，从而获得免费流量，就是提高网站在某个搜索引擎中的  
 排名

### TDK三大标签

#### T:title标签

title使页面内的第一个标签，使搜索引擎了解网页的入口和对网站主题归属的最佳判

建议：网站名、产品名

#### D:description

网站说明：介绍网站是做什么的

<meta name=”description” content=”我们的网站是，，，提供.......”>

#### K：keywords

网站关键词。是搜索引擎关注的重点之一

用法：<meta name=”keywords” content=”留给SEO去写，最好的产品写在前面：手机、电脑.......”>

## 模块化开发

头部尾部

版心

### 命名：

|  |  |
| --- | --- |
| 名称 | 说明 |
| 快捷导航栏 | shortcut |
| 头部 | header |
| 标志 | logo |
| 购物车 | shopcar |
| 搜索 | search |
| 热点词 | hotwords |
| 导航 | nav |
| 导航左侧 | dropdown包含.dd、.dt |
| 导航右侧 | navitems |
| 页面底部 | footer |
| 页面底部服务模块 | mod\_service |
| 页面底部帮助模块 | mod\_help |
| 页面底部版权模块 | mod\_copyright |

### logo SEO优化

1. logo里面首先放一个h1标签，提权，告诉搜索引擎，这个地方很重要
2. h1里面放一个超链接，可以返回首页，把logo的背景图片给链接即可
3. 为了搜索引擎能够收录我们，我们链接里面要放文字（网站的名字），但是文字不要显示出来

文字不显示的两种做法：font-size:0;京东的做法

text-indent:-99999px;然后 overflow:hidden;淘宝的做法

## 2D转换

### 2D转换之位移translate：不会影响其他元素的位置

移动盒子的位置：

定位、盒子的外边距、2d转换移动

transform: translate(x,y)

x为x轴坐标

y为y轴坐标

transform: translatex(x轴坐标) 只移动x轴

transform: translatey(y轴坐标) 只移动y轴

重点：

不会影响其他元素的位置，结合定位实现垂直水平居中

translate中的百分比单位是相对与自身元素

对行内元素无效

### 2D转换之旋转 rotate

transform:rotate(度数)；

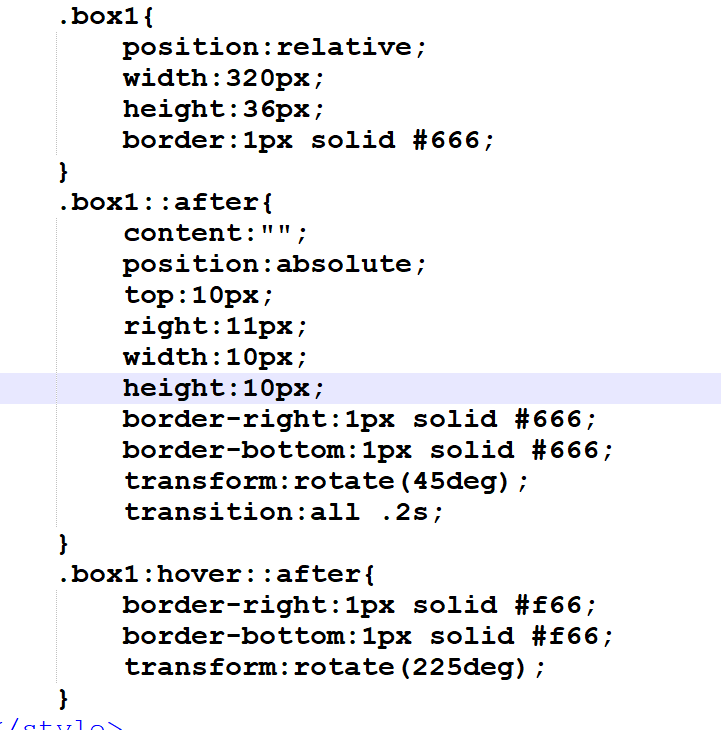
重点：

度数必须加单位deg

度数可以是正直表示顺时针旋转，负值表示逆时针旋转

旋转的中心点是元素的中心点

### 箭头的做法：定位加旋转



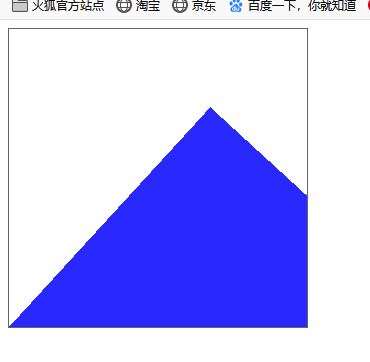
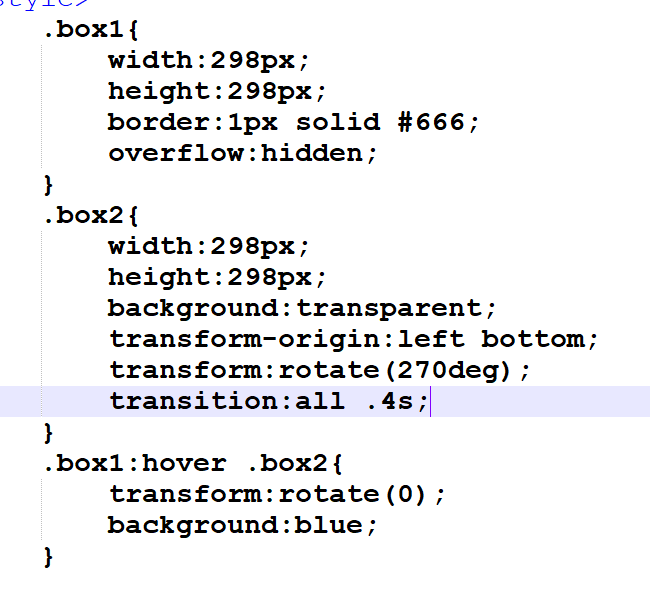
### 设置旋转中心

语法：transform-origin:

重点：

注意后面的参数x和y用空格隔开

x y默认转换的中心点是元素的中心点（50% 50%）

 还可以将x y的值设置为像素或者方位名词（left/right/center/top/bottom）

### 2D转换之缩放：

缩放：顾名思义，可以缩小和放大。只要给元素添加上了这个属性就能控制它把他放大和缩小。

语法：

transform:scale(x,y);

重点：

注意其中的x和y之间使用都好隔开

transform：scale(1,1) 表示宽和高都放大一倍，相当于没有放大

transform：scale(2,2) 表示宽和高都放大了两倍

transform：scale(2) 只写一个参数，第二个参数省略，表示两个参数都一样（2,2）

scale缩放最大的优点是，可以设置转换点缩放，默认是中心点，不影响其他盒子

### 2D转换的简写

transform: translate(x,y) rotate(deg) scale(x,y);

注意：

1. 同时使用多个转换，其格式为：transform:translate(x,y) rotate(度数) scale(x,y)....等

其顺序会影响转换的效果，（先旋转会改变坐标）

### 总结：

转换transform我们简单理解就是变形有2D和3D之分

我们暂时学习了三个部分:位移、旋转、缩放

2D位移最大的优势是不影响其他盒子，里面参数用%,是相对于自身的宽度和高度来计 算的

可以分开写比如translateX(x)和translateY(y)

可以改变旋转/缩放中心：transform-origin可选值可以是百分比、像素、方位词

2D旋转rotate(度数) 度数的单位是deg

2D缩放scale(x,y) 数值可以写一个表示x y相同，没有单位。

当我们进行综合方法，同时有位移和其他属性的时候，记得要将位移放到最前

## 动画

### 动画的基本使用

动画animation是css3中具有颠覆性的特性之一，可以设置多个节点精确控制一个或一组动画，常用来实现复杂的动画效果。

相较于过渡，动画可以实现更多变化，更多控制，连续自动播放等效果。

动画的使用：

1、动画的设置

@keyframes 动画名称{

form{

transform:translatex(0px);

}

to{

transform:translatex(1090px);

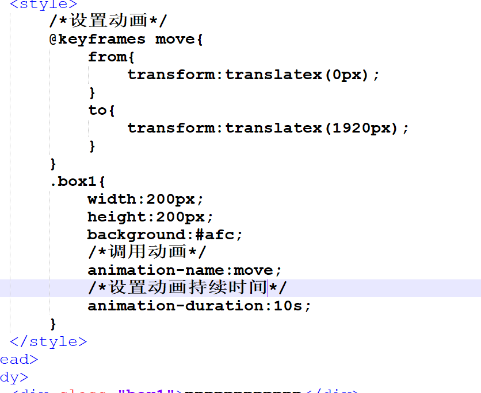
}

}

1. 调用动画

animation-name:动画名称；

1. 设置动画时间

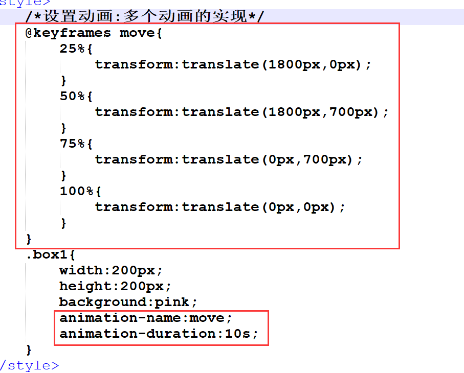
animation-duration:2s;

动画序列：

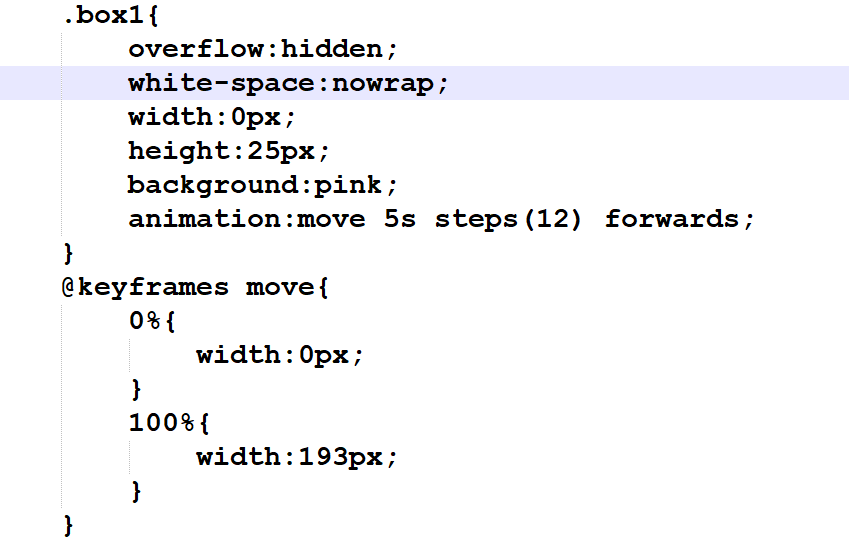
0%是动画的开始，100%是动画的完成，这样的规则就是动画序列

在@keyframes中的规定某项css样式，就能创建由当前样式逐渐改变为新样式的 动画效果。

动画是使元素从一种样式逐渐变化为另一种样式的效果，可以改变任意多的样式任 意多的次数

 请使用百分比来规定变化的发生时间，或用关键from和to 等同于0%和100%

animation-timing-function:ease/ease-in/ease-in-out/ease-out/linear;



简写：

animation:动画名称 持续时间 运动曲线 何时开始 播放次数 是否反方向 动画结束的状态；

name 2s ease-in-out linear steps(步长) infinite alternate forwards；

## 3D

什么是3D？

我们就生活再3D空间中。

最多看到一个立方体的三个面

三维坐标系

三维坐标系其实就是指立体空间，立体空间是由3个轴共同组成的。

x轴是水平向右的 正值向右 负值向左

y轴是垂直方向的 正值向下 负值向上

z轴是垂直屏幕的 正值向外 负值向内

### 3D位移

3D位移再2D位移的基础上多加了一个可以移动的方向，就是Z轴的方向

第一种写法

transform: translateX(x);设置x轴方向的位移

transform: translateY(y);设置Y轴方向的位移

transform: translateZ(z);设置z轴方向的位移

第二种写法：

transform:translateX(x) translateY(y) translateZ(z)；

第三种写法

transform: translate3d( x , y, z)；设置x轴y轴z轴的位移

注意:

第三种写法：x y z 不可以省略，不用的话可以写0

#### 透视perspective

再2D平面产生近大远小视觉立体，但是效果是平面的

如果想要再网页产生3D效果需要透视（理解成3D物体投影在2D平面上）

默认人类的视觉位置，可认为安排一只眼睛去看

透视我们也称为视距：视距就是人的眼睛到屏幕的距离

距离视觉点越近的再电脑平面成像越大，越远成像越小。

透视的单位为像素

注意：透视写在被观察元素的父盒子上

### 3D旋转rotate3d

3D旋转可以让元素在三维平面内沿着x轴，y轴，z轴或者自定义轴进行旋转

transform: rotateX(度数);单位deg表示 沿着x轴方向旋转度数

transform: rotateY(度数);单位deg表示 沿着y轴方向旋转度数

transform: rotateZ(度数);单位deg表示 沿着z轴方向旋转度数

transform:rotate3d(x,y,z,deg);

xyz表示旋转轴的矢量值，xyz决定一个矢量，deg决定旋转的度数

#### 3D呈现：transform-style

控制子元素是否开启三维立体环境

transform-style: flat;子元素不开启3d立体空间 默认的

transform-style: preserve-3d; 子元素开启立体空间

代码交给父级，但是影响的是子元素

很重要 必用

## 浏览器私有前缀

-moz-:代表firefox浏览器私有属性

-ms-:代表ie浏览器私有属性

-webkit:代表safari、chrome私有属性

-o-：代表Opear私有属性

提倡的写法：

-moz-border-radius:10px;

-webkit-border-radius:10px;

-o-border-radius:10px;

border-radius:10px;

## 圣杯布局

## 双飞翼布局