web标准：结构，表现，行为

结构：html=超文本标记语言

表现：css

行为：Javascript

base:可以设置整体链接的打开状态

表格属性

cellspacing：单元格与单元格之间的距离

cellpadding：内容与单元格之间的距离

caption标签：表格的标题标签

rowsapn：跨行合并

colspan：跨列合并

单元格合并顺序：先上后下，先左后右

css：border-collapse：collapse 合并相邻边框

表单属性

控件input

属性

type： text 单行文本输入框

password 密码输入框

radio 单选输入框

checkbox 复选框

button 普通按钮

submit 提交按钮

reset 重置按钮

image 图片形式提交按钮

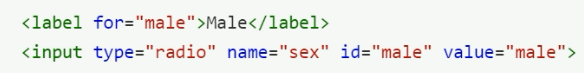
file 文件域

name 控件名称

value 控件默认文本

checked 控件中默认被选中的项

maxlength 控件允许输入的最多字符数

lable 为input元素定义标注=点击文字焦点出现在对应的input （

下拉菜单select

option标签 菜单选项标签

属性

selected 选中菜单

表单域form

属性

action 用于指定接收并处理表单数据的服务器程序url地址

method 用于设置表单数据的提交方式，其值为get和post

name 用于指定表单的名称，

块级元素

总是从新行开始

高度，行高，外边距及内边距都可以控制

宽度默认是容器的100%

可以容纳内联元素和其他块元素

元素：<div>、<p>、<h1>...<h6>、<ol>、<ul>、<dl>、<table>、<address>、<blockquote> 、<form>

行内元素

和相邻行内元素在一行

高，宽无效，但是水平方向的padding和margin可以设置，垂直方向的无效

默认宽度是他内容本身的高度

行内元素只能容纳文本或者其他的行内元素

元素：<a>、<span>、<br>、<i>、<em>、<strong>、<label>、<q>、<var>、<cite>、<code>

内联块状元素：<img>、<input> 相当于inline-block，默认两个内联块元素之间有1px的间距解决办法，让元素浮动

text-align：center；可以让盒子内容（文字、行内元素、行内块元素）居中对齐

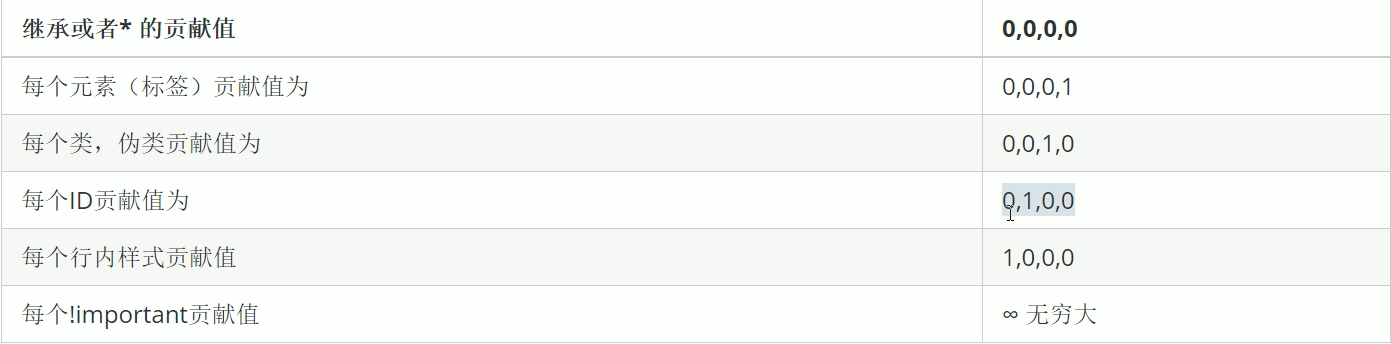
选择器的嵌套层级应不大于3级，位置靠后的限定条件尽量精确

css三大特性

css层叠性：冲突的样式以最后书写的样式为准

css 继承性：子标签会继承父标签的某些样式：文本样式 text，font，line开头的 color

css 优先级：



1.权重相同 就近原则

2.权重会叠加

3.继承的权重为0

背景：background

backgound-position:如果只有一个方位名词，另一个方位则默认居中

background-attachment:决定背景是在视口中固定的还是随包含它的区块滚动的。

fixed：此关键字表示背景相对于视口固定。即使一个元素拥有滚动机制，背景也不会随着元素的内容滚动。

scroll：此关键字表示背景相对于元素本身固定， 而不是随着它的内容滚动（对元素边框是有效的）

边框：border

border-width

border-color

border-style:none

实线:solid

虚线:dashed

点线:dotted

border:1px solid red;

内边距：padding

padding:20px 上下左右

padding：10px 20 上下 左右

padding：10px 20px 30px 上 左右 下

padding：10px 20px 30px 40px 上 右 下 左

外边距：margin

margin：0 auto 可以是块级元素水平居中

margin浏览器通病：相邻的两个垂直元素的外边距会发生合并，并且以边距最大的为准

解决方法：尽量避免这种情况

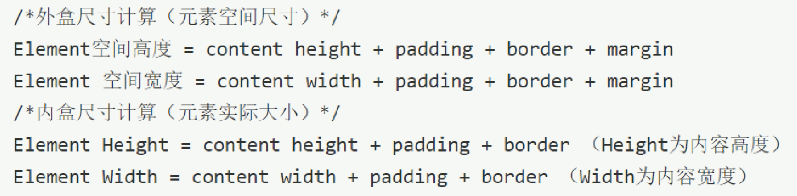
对于两个嵌套关系的块元素，如果父元素没有上边距及边框，则父元素的上边距会与子元素的上边距发生合并，合并之后的外边距为两者中的最大者，即使父元素的外边距为零也是会发生合并

解决方案：1.可以为父元素定义1px上边框或者1px内边距

2.可以为父元素添加overflow：hidden

插入图片，我们用的最多的 比如产品展示类

背景图片我们一般用于小图标背景或者超大背景图片



当一个元素的宽度是继承父元素的宽度，则使用padding不会撑开盒子，给定值使用padding则会被撑开

优先使用宽度其次使用内边距再次使用外边距

box-shadow盒子阴影:水平（正值往右 负值往左） 垂直（正值往下 负值网上） 模糊距离（虚实） 尺寸（影子大小） 颜色 | 内外阴影

float：可以让元素默认转换成行内块元素

position:relative 相对定位最重要的一点是，它可以通过边偏移移动位置，但是原来所占有的位置继续占有。每次移动的位置，是以自己左上角为基点移动

position:absolute 若所有父元素都没有定位，则以浏览器当前的屏幕为准对齐（document文档）

绝对定位是将元素依据最近的已经定位（绝对，固定，相对）的父元素进行定位，使用了绝对定位的元素水平方向的margin和float会失效

position:fixed 固定定位脱离文档流，不占用位置只相对浏览器对齐

元素添加了绝对定位和固定定位之后，元素模式也会发生转换，都转换成行内块模式

vertical-align:垂直对齐，控制图片和文字垂直方向的对齐，只作用于行内块和行内元素（通常用来控制图片/表单的文字对齐），当文字是基于基线对齐的时候会让图片底部增加小bug，解决方法是让文字垂直居中或者顶部对齐

基线对齐：baseline

垂直居中：middle

顶部对齐：top

white-space：设置或者检索对象内文本显示方式，通常用于强制一行显示内容

normal:默认处理方式

nowrap:强制在同一行显示所用文本，直到文本结束或者<br/>

text-overflow:文本溢出隐藏，通常用于文本溢出部分省略号显示

ellipsis:溢出部分省略号显示

网页logo默认写法，a元素里面要留字便于搜索引擎查找，想要隐藏字使用

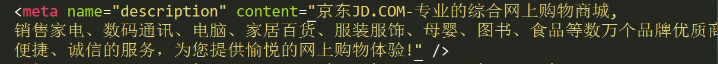
text-indent：-9999px



网站优化三大标签：title、description、keywords

title：网站名（产品名）-网站介绍

description:网站说明：不允许超过120个汉字，使用英文逗号隔开



keywords：关键字



input：默认有内边距

outline：取消蓝色边框

buttom：默认有边框

img元素：默认底部有小间隙，解决办法vertical-align:top，图片随着浏览器大小自由缩放，设置为widht：100% height：auto

## CSS W3C 统一验证工具

CssStats 是一个在线的 CSS 代码分析工具

```

网址是： http://www.cssstats.com/

```

如果你想要更全面的，这个神奇，你值得拥有：

W3C 统一验证工具： http://validator.w3.org/unicorn/ ☆☆☆☆☆

因为它可以检测本地文件哦！！

http://tool.chinaz.com/Tools/CssFormat.aspx css 代码压缩

HTML5新标签

header：头部

nav：导航链接

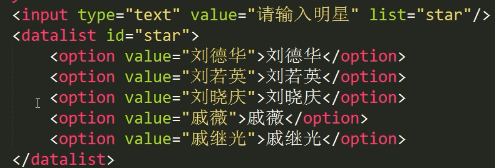
footer：底部

artice:文章

seciton:文档中的节

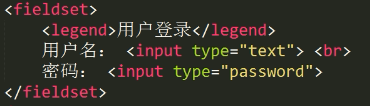
aside：侧边

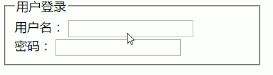
datalist:下拉菜单，利用id配合input元素使用



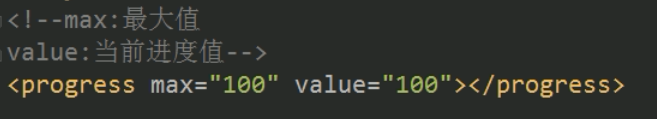
fieldset：自定义控件

legend





progress:进度条



input新类型

email：邮箱

提供了默认的电子邮箱完整验证

tel：电话

本质是为了移动端打开数字键盘

url：网址

验证只能输入合法的网址，必须包含http://

number：输入数字格式

只能输入数字（包含小数点）不能输入其他字符

max：最大值

min：最小值

value：默认值

search：搜索框

提供跟人性化的输入体验

range：拖动滑块

time：小时分钟

date：年月日

datetime-local：日期时间

month：年月

week：星期

input 新属性

placeholder：占位符

autofocus：自动获取焦点

autocomplete:自动弹出历史输入记录：on：打开 off：关闭

1、必须成功提交过，提交过才会有记录

2、当前添加autocomplete的元素必须name属性

multiple:多文件上传

required：必填项

pattern：正则表达式验证

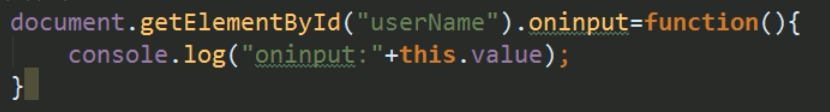
\*：代表任意个

？：代表0个或者1个

+：代表1个或者多个

input:新增事件

oninput:监听当前元素内容的改变，只要内容改变（添加内容，删除内容），就会触发这个事件



oninvalid:当验证不通过时触发



多媒体标签

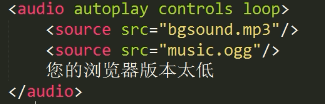
audio：音频，目前有mp3，ogg，wav格式

autoplay：自动播放

controls:控制

loop：循环

source：来源 配合audio标签使用，兼容多个浏览器



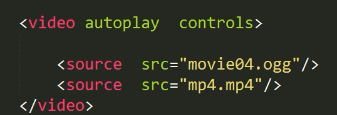
video：视频，目前有ogg，MP4，webm格式

autoplay：自动播放

controls:控制

loop：循环

source：来源 配合video标签使用，兼容多个浏览器



注意事项：当设置宽高的时候，一般情况下只会设置宽度或者高度，让其自动的等比例缩放

结构伪类选择器

li：first-child：第一个li

li：last-child：最后一个li

li：nth-child（n）：第n个li

li：nth-child（even）偶数的li

li：nth-child（odd）奇数的li

li：nth-child（2n）2的倍数li

属性选择器

div[class]:选出带有class属性的元素

div[class=demo]:选出class=demo的元素

div[class^=test]:选出已class为test开头的元素

div[class$=test]:选出已class为test结尾的元素

div[class\*=test]:选出已class有test的元素

新增获取doc元素的方法

querySelector:query:查询 Selector:选择器 querySelector(选择器的名称)

获取单个元素，如果获取的元素不止一个，那么只会返回满足条件的第一个元素

如果是类选择器，必须添加. 如果是id选择器，必须添加#，否者会当成标签来处理



querySelectorAll:获取所有满足条件的元素，返回值是数组

新增操作元素类样式

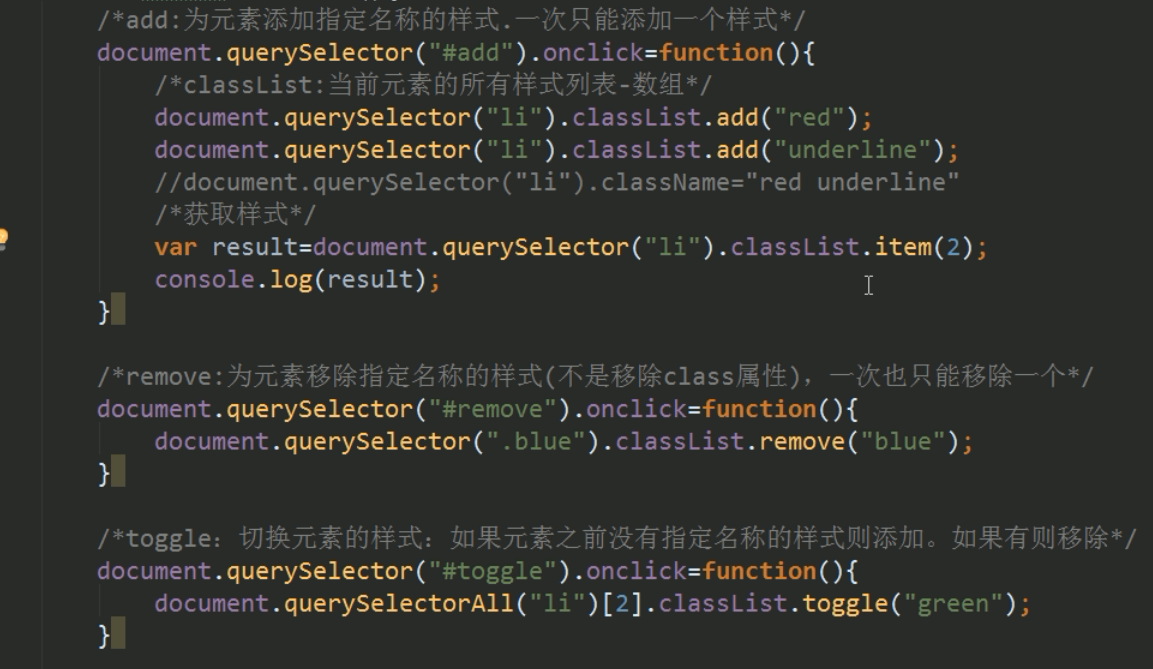
classList:当前元素的所有样式列表-数组

add:为元素添加指定名称的样式，一次只能添加一个样式

item:获取当前元素的样式添加的样式列表-数组

remove:为元素移除指定名称的样式（不是移除class属性），一次只能移除一个

toggle:切换元素的样式，如果元素之前没有指定名称的样式则添加，如果有则移除



元素自定义属性

data-开头

data-后必须至少一有一个字符，多个单词使用-连接

名称应该使用小写--不要包含任何的大写字符

名称中不要有任何的特殊符号

名称中不要有纯数字

js dataset获取自定义属性的是要使用驼峰命名法



网络接口

ononline:网络连通的时候触发这个事件

onoffline:网络断开的时候触发这个事件

使用addEventListener添加事件



全屏接口

requestFullScreen:开启全屏显示（相对元素本身）

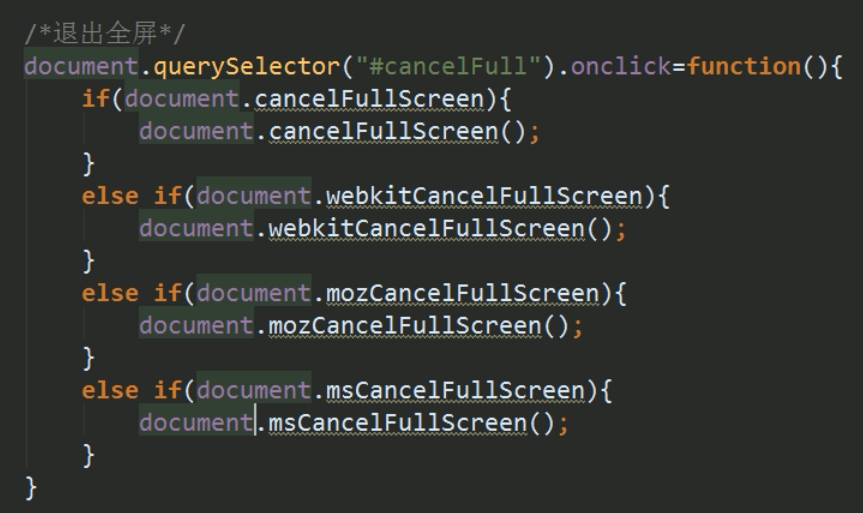
cancelFullScreen:退出全屏显示（相对文档）

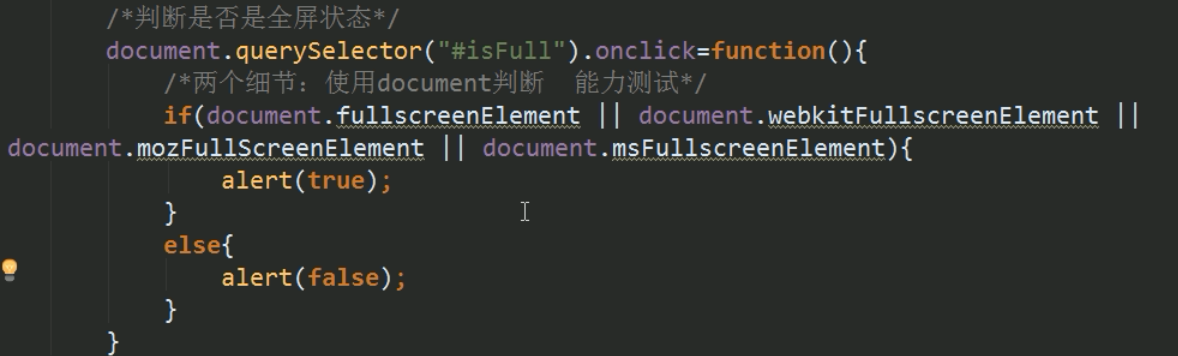
FullScreenElement:是否是全屏状态（相对文档）

需要兼容浏览器,添加不同的前缀

chrome:webkit firefox:moz ie:ms opera:o



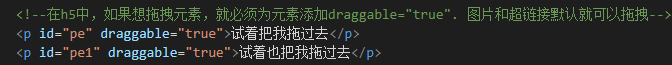




FileReader



draggable拖拽





web储存

1. 需求：随着互联网的快速发展，基于网页的应用越来越普遍，同时也变的越来越复杂，为了满足各种各样的需求，会经常性在本地存储大量的数据，传统方式我们以document.cookie来进行存储的，但是由于其存储大小只有4k左右，并且解析也相当的复杂，给开发带来诸多不便，HTML5规范则提出解决方案
2. HTML5提供的解决方案：
   1. window.sessionStorage
   2. window.localStorage
3. 特点：
4. 设置、读取方便
5. 容量较大，sessionStorage约5M、localStorage约20M
6. 只能存储字符串，可以将对象JSON.stringify() 编码后存储
7. Window.sessionStorage的使用
   1. 特点：
      1. 生命周期为关闭浏览器窗口：相当于存储在当前页面的内内存中
      2. 在同一个窗口下数据可以共享(在当前页面下可以获取到，换另外一个页面下不能获取到)
   2. 方法介绍：(两种存储方式的方法一致)
      1. SetItem(key,value):设置数据，以键值对的方式
      2. getItem(key):通过指定的键获取对应的值内容
      3. removeItem(key):删除指定的key及对应的值内容
      4. clear():清空所有存储内容
   3. 使用说明：

<script>  
 **var** userData=document.getElementById("userName");  
 //存储数据  
 document.getElementById("setData").onclick=**function**(){  
 window.sessionStorage.setItem("userName",userData.value);  
 }  
 //获取数据  
 document.getElementById("getData").onclick=**function**(){  
 **var** value=window.sessionStorage.getItem("userName");  
 alert(value);  
 }  
</script>

1. Window.localStorage的使用
   1. 特点：
      1. 永久生效，除非手动删除：存储在硬盘上
      2. 可以多窗口共享。但是不能跨浏览器
   2. 使用说明：

<script>  
 **var** userData=document.getElementById("userName");  
 //存储数据  
 document.getElementById("setData").onclick=**function**(){  
 window.localStorage.setItem("userName",userData.value);  
 }  
 //获取数据  
 document.getElementById("getData").onclick=**function**(){  
 **var** value=window.localStorage.getItem("userName");  
 alert(value);  
 }  
 //删除数据  
 document.getElementById("removeData").onclick=**function**(){  
 window.localStorage.removeItem("userName");  
 }  
</script>