# HTML+CSS基础课程

1.

Html和CSS的关系

学习web前端开发基础技术需要掌握：HTML、CSS、JavaScript语言。下面我们就来了解下这三门技术都是用来实现什么的：

1. HTML是网页内容的载体。内容就是网页制作者放在页面上想要让用户浏览的信息，可以包含文字、图片、视频等。

2. CSS样式是表现。就像网页的外衣。比如，标题字体、颜色变化，或为标题加入背景图片、边框等。所有这些用来改变内容外观的东西称之为表现。

3. JavaScript是用来实现网页上的特效效果。如：鼠标滑过弹出下拉菜单。或鼠标滑过表格的背景颜色改变。还有焦点新闻（新闻图片）的轮换。可以这么理解，有动画的，有交互的一般都是用JavaScript来实现的。

Eg：为<h1></h1>中间的内容添加样式

h1{

font-size:12px; //字体大小

color:#930; //颜色

text-align:center; //居中显示

}

2.

HTML文件基本结构

1）

一个HTML文件是有自己固定的结构的。

<html>

<head>...</head>

<body>...</body>

</html>

代码讲解：

1. <html></html>称为根标签，所有的网页标签都在<html></html>中。

2. <head> 标签用于定义文档的头部，它是所有头部元素的容器。头部元素有<title>、<script>、 <style>、<link>、 <meta>等标签，头部标签在下一小节中会有详细介绍。

3. 在<body>和</body>标签之间的内容是网页的主要内容，如<h1>、<p>、<a>、<img>等网页内容标签，在这里的标签中的内容会在浏览器中显示出来

2）<head>

<title>...</title>

<meta>

<link>

<style>...</style>

<script>...</script>

</head>

<title>标签：在<title>和</title>标签之间的文字内容是网页的标题信息，它会出现在浏览器的标题栏中。网页的title标签用于告诉用户和搜索引擎这个网页的主要内容是什么，搜索引擎可以通过网页标题，迅速的判断出网页的主题。每个网页的内容都是不同的，每个网页都应该有一个独一无二的title。

3.

其他部分标签含义

1. <img>插入图片

<img src="http://img.mukewang.com/52b4113500018cf102000200.jpg" >

<img src="图片地址" alt="下载失败时的替换文本" title = "提示文本">

1. <!--HTML注释文字 --> /\*CSS注释文字\*/
2. 斜体强调：<em>需要强调的文本</em>

加粗强调：<strong>需要强调的文本</strong>

1. <span>标签是没有语义的，它的作用就是为了设置单独的样式用的。

Eg：span{

color:blue;

}

<span>美国梦</span>

5）<q></q>短文本引用 <blockquote></blockquote>长文本引用

引用内容会被自动加上双引号

1. <br />换行 <hr />添加水平横线
2. &nbsp; 空格，要几个写几个 eg：<p>今天&nbsp;&nbsp;&nbsp;天气不错</p>
3. 加入地址 <address>北京市西城区德外大街10号</address> 会斜体且另起一段
4. 加入代码 作用： 防止浏览器误认为是要执行代码，而没显示代码。加了标签浏览器就

不会执行了，而是像文本一样显示出来

单行<code> 多行<pre>

10）添加新闻信息列表

ul-li ul = unordered list 无序列表，li即list item

ol-li 有序列表，ol即ordered list

11）<div>标签为网页划分独立的版块

为div命名 <div  id="版块名称">…</div>

1. 表格 <table> <tbody> <tr> <th> <td>

<tbody>…</tbody>：当表格内容非常多时，表格会下载一点显示一点，但如果加上 <tbody>标签后，这个表格就要等表格内容全部下载完才会显示。

Eg:

<table>

<tbody>

<tr>

<th>班级</th>

<th>学生数</th>

<th>平均成绩</th>

</tr>

<tr>

<td>一班</td>

<td>30</td>

<td>89</td>

</tr>

<tr>

<td>二班</td>

<td>35</td>

<td>85</td>

</tr>

</tbody>

</table>

Ps: 1、table表格在没有添加css样式之前，在浏览器中显示是没有表格线的

2、表头，也就是th标签中的文本默认为粗体并且居中显示

1. 为表格添加标题和摘要

<table summary="本表格记录2012年到2013年库存记录，记录包括U盘和耳机库存量">

<caption>2012年到2013年库存记录</caption>

Summary是备注，摘要，不显示在网页中。 caption是标题。

1. <a>超链接

<a href="目标网址" title="鼠标滑过显示的文本">链接显示的文本</a>

把超链接在新窗口中打开：  
 <a href="目标网址" target="\_blank">click here!</a>

1. 邮件



注意：如果mailto后面同时有多个参数的话，第一个参数必须以“?”开头，后面的参数每一个都以“&”分隔。

1. 表单：form，与用户交互

<form method="post" action="save.php">

<label for="username">用户名:</label>

<input type="text" name="username" />

<label for="pass">密码:</label>

<input type="password" name="pass" />

</form>

注释： 1.所有表单控件（文本框、文本域、按钮、单选框、复选框等）都必须放在<form></form>

标签之间（否则用户输入的信息可提交不到服务器上哦！）

2. <form method="传送方式" action="服务器文件">

action ：浏览者输入的数据被传送到的地方,比如一个PHP页面(save.php)。

method ： 数据传送的方式（get/post）。

文本输入框、密码输入框

当用户要在表单中键入字母、数字等内容时，就会用到文本输入框。文本框也可以转化为密码输入框。

语法：

<form>

<input type="text/password" name="名称" value="文本" />

</form>

1、type：

   当type="text"时，输入框为文本输入框;

   当type="password"时, 输入框为密码输入框。

2、name：为文本框命名，以备后台程序ASP 、PHP使用。

3、value：为文本输入框设置默认值。(一般起到提示作用)

举例：

<form>

姓名：

<input type="text" name="myName">

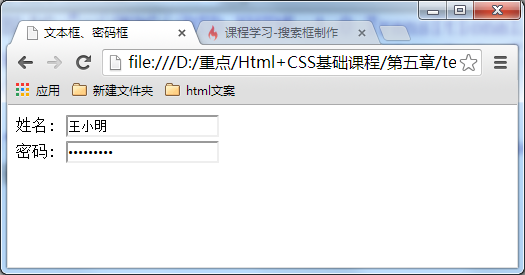
<br/>

密码：

<input type="password" name="pass">

</form>

在浏览器中显示的结果：

[](http://img.mukewang.com/52e4e9be000152ca05250275.jpg)

文本域，支持多行文本输入

当用户需要在表单中输入大段文字时，需要用到文本输入域。

语法：

<textarea rows="行数" cols="列数">文本</textarea>

1、<textarea>标签是成对出现的，以<textarea>开始，以</textarea>结束。

2、cols ：多行输入域的列数。

3、rows ：多行输入域的行数。

4、在<textarea></textarea>标签之间可以输入默认值。

使用单选框、复选框，让用户选择

在使用表单设计调查表时，为了减少用户的操作，使用选择框是一个好主意，html中有两种选择框，即单选框和复选框，两者的区别是单选框中的选项用户只能选择一项，而复选框中用户可以任意选择多项，甚至全选。请看下面的例子:

语法：

<input type="radio/checkbox" value="值" name="名称" checked="checked"/>

1、type:

   当 type="radio" 时，控件为单选框

   当 type="checkbox" 时，控件为复选框

2、value：提交数据到服务器的值（后台程序PHP使用）

3、name：为控件命名，以备后台程序 ASP、PHP 使用

4、checked：当设置 checked="checked" 时，该选项被默认选中

使用下拉列表框，节省空间

下拉列表在网页中也常会用到，它可以有效的节省网页空间。既可以单选、又可以多选。

单选：

<form action="save.php" method="post" >

<label>爱好:</label>

<select>

<option value="看书">看书</option>

<option value="旅游" selected="selected">旅游</option>

<option value="运动">运动</option>

<option value="购物">购物</option>

</select>

</form>

多选：<select multiple="multiple">

提交 <input type="submit" value="提交">

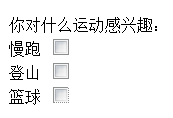
type：只有当type值设置为submit时，按钮才有提交作用

value：按钮上显示的文字

重置 <input type="reset" value="重置">

type：只有当type值设置为reset时，按钮才有重置作用

value：按钮上显示的文字

关于label：

上图实现代码：

你对什么运动感兴趣：<br />

  <label for="jog">慢跑</label>

  <input type="checkbox" name="jog" id="jog" /><br />

  <label for="climb">登山</label>

  <input type="checkbox" name="climb" id="climb" /><br />

  <label for="basketball">篮球</label>

  <input type="checkbox" name="basketball" id="basketball" />

不加label的话鼠标一定要点击小圆点才能激活条目,加了label可以直接点击对应的文字来激活条目。  
label的作用就是获得焦点，在列子中，把鼠标移动到文字上（男、女、输入你的邮箱地址），同样可以激活控件进行选择，输入。但是如果不加label的话，鼠标就必须移动到控件上面。

4.

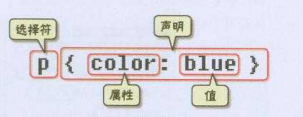
概念

1. 语义化：明白每个标签的用途。

5.

CSS。CSS全称为“层叠样式表 (Cascading Style Sheets)”，它主要是用于定义HTML内容在浏览器内的显示样式，如文字大小、颜色、字体加粗等。

1）css 样式由选择符和声明组成，而声明又由属性和值组成，如下图所示：

[](http://img.mukewang.com/52fde5c30001b0fe03030117.jpg)

当有多条声明时，中间可以英文分号“;”分隔。最后一条声明可以没有分号，但是为了以后修改方便，一般也加上分号。

1. 从CSS 样式代码插入的形式来看基本可以分为以下3种：内联式、嵌入式和外部式三种。

内联式 <p style="color:red;font-size:12px">这里文字是红色。</p>

<span style="color:red;font-size:12px">这里文字是红色。</span>

嵌入式 嵌入式css样式必须写在<style></style>之间，并且一般情况下嵌入式css样

式写在<head></head>之间。

<style type="text/css">

span{

color:red;

}

</style>

外部式

把css代码写一个单独的外部文件中，这个css样式文件以“.css”为扩展名，在<head>内（不是在<style>标签内）使用<link>标签将css样式文件链接到HTML文件内，如下面代码：

<link href="base.css" rel="stylesheet" type="text/css" />

base.css文件：span{

color:red;

font-size:20px;

}

注意：

1、css样式文件名称以有意义的英文字母命名，如 main.css。

2、rel="stylesheet" type="text/css" 是固定写法不可修改。

3、<link>标签位置一般写在<head>标签之内。

1. 三种方式的优先级

就近原则

1. 选择器

标签选择器 最普通

类选择器 <span class="stress">胆小如鼠</span>

.stress{color:red;} /\*类前面要加入一个英文圆点\*/

ID选择器 <span id="setGreen">公开课</span>

#setGreen{ color:green;} /\*加#\*/

Q：什么时候用类选择器，什么时候用ID选择器？

A:在同一个页面内，不允许有相同名字的id对象出现，但是允许相同名字的class。这样，一般网站分为头，体，脚部分，因为考虑到它们在同一个页面只会出现一次，所以用id，其他的，比如说你定义了一个颜色为red的class，在同一个页面也许要多次用到，就用class定义。另外，当页面中用到js或者要动态调用对象的时候，要用到id，所以要根据自己的情况运用。

子选择器 .food>li{border:1px solid red;}

.first>span{border:1px solid red;}

只作用于第一代后代

后代选择器，变“>”为空格，就可以作用于所有后代了

通用选择器，可以匹配html中所有标签元素 \* {color:red;}

伪类选择符 a:hover{color:red;}

一个<a href="http://www.imooc.com">胆小如鼠</a>的小女孩

分组选择符 例子：

h1,span{color:red;}

.first,#second span{color:green;}

1. 继承：继承是一种规则，它允许样式不仅应用于某个特定html标签元素，而且应用于其后代。可继承的比如color，不可继承如border
2. 标签的权值为1，类选择符的权值为10，ID选择符的权值最高为100。

权值相同时，就近原则，使用靠后的样式。

设置最高权值，用!important 格式: p{color:red!important;}

1. CSS格式化排版

字体 body{font-family:"Microsoft Yahei";}

常见的 font-size,color

加粗 {font-weight:bold;}

斜体 {font-style:italic;}

下划线 {text-decoration:underline;}

划掉线 {text-decoration:line-through;}

首行缩进（2）字符 {text-indent:2em;}

行间距 {line-height:2em;}

中文字间隔、字母间隔设置 {letter-spacing:50px;}

单词间距设置 word-spacing

居中（左右同理） text-align:center;

8)

块级元素特点：  
1、每个块级元素都从新的一行开始，并且其后的元素也另起一行。（真霸道，一个块级元素独占一行）  
2、元素的高度、宽度、行高以及顶和底边距都可设置。  
3、元素宽度在不设置的情况下，是它本身父容器的100%（和父元素的宽度一致），除非设定一个宽度。  
  
内联元素特点：  
1、和其他元素都在一行上；  
2、元素的高度、宽度及顶部和底部边距不可设置；  
3、元素的宽度就是它包含的文字或图片的宽度，不可改变。  
  
inline-block 元素特点：  
1、和其他元素都在一行上；  
2、元素的高度、宽度、行高以及顶和底边距都可设置。

盒子模型

内容(content)、填充(padding)、边框(border)、边界(margin)

因此一个元素实际宽度（盒子的宽度）=左边界+左边框+左填充+内容宽度+右填充+右边框+右边界。

边框

div{border:2px solid red;} div{border:2px dotted #ccc;}

上面是 border 代码的缩写形式，可以分开写：

div{

border-width:2px;

border-style:solid;

border-color:red;

}

注意：

1、border-style（边框样式）常见样式有：

dashed（虚线）| dotted（点线）| solid（实线）。

2、border-color（边框颜色）中的颜色可设置为十六进制颜色，如:

border-color:#888;//前面的井号不要忘掉。

3、border-width（边框宽度）中的宽度也可以设置为：

thin | medium | thick（但不是很常用），最常还是用象素（px）。

1. 下边框就是border-bottom：

还有margin-top，padding-right之类的

1. 默认顺序是top-right-bottom-left（上右下左）

比如#box1{margin:0 0 30px 0;}就是为id名称为box1的div元素加入30象素的下边距

1. 别的缩写方式 （无空格表示值相同）

除了 上 右 下 左

还有 上下左右 //一个数值

上下 左右 //两个数值

上 左右 下 //三个数值

1. 布局模型

在网页中，元素有三种布局模型：  
 1、流动模型（Flow）  
 2、浮动模型 (Float)  
 3、层模型（Layer）

流动模型：  
1 块级元素从上到下垂直显示 宽度100%  
2 内联元素从左到右水平分布显示

浮动模型：

任何元素在默认情况下是不能浮动的，但可以用 CSS 定义为浮动

Eg:已知<div id="div1"></div>

<div id="div2"></div>

都靠左 div{float:left;}

一左一右 #div1{float:left;}

#div2{float:right;}

层模型有三种形式：

1、绝对定位(position: absolute)

2、相对定位(position: relative)

3、固定定位(position: fixed)

Ps：

A）1和3的区别：页面可以上下翻滚时，absolute规定的块状元素相对于body是不动的，

fixed则相对于界面显示屏幕不动

B）Relative与Absolute组合使用

举例，box3作为父元素，box4以3为参考定位

在head里：#box3{

width:200px;

height:200px;

**position:relative;**

}

#box4{

width:99%;

**position:absolute;**

bottom:0px;

}

在body里：<div id="box3">

<img src="http://img.mukewang.com/541a7d8a00018cf102000200.jpg">

<div id="box4">当我还是三年级的学生时是一个害羞的小女生。</div>

</div>

11） 字体缩写

网页中的字体css样式代码也有他自己的缩写方式，下面是给网页设置字体的代码：

body{

font-style:italic;

font-variant:small-caps;

font-weight:bold;

font-size:12px;

line-height:1.5em;

font-family:"宋体",sans-serif;

}

这么多行的代码其实可以缩写为一句：

body{

font:italic small-caps bold 12px/1.5em "宋体",sans-serif;

}

注意：

1、使用这一简写方式你至少要指定 font-size 和 font-family 属性，其他的属性(如 font-weight、font-style、font-varient、line-height)如未指定将自动使用默认值。

2、在缩写时 font-size 与 line-height 中间要加入“/”斜扛。

一般情况下因为对于中文网站，英文还是比较少的，所以下面缩写代码比较常用：

body{

font:12px/1.5em "宋体",sans-serif;

}

只是有字号、行间距、中文字体、英文字体设置。

1. 颜色

①英文命令 p{color:red;}

②RGB p{color:rgb(133,45,200);}或p{color:rgb(20%,33%,25%);}

③十六进制 p{color:#00ffff;}

可简写为p{color:#0ff;} 即十六进制颜色AABBCC可简写为ABC

1. 长度单位

px像素 其实是相对于显示器的相对单位

em 就是本元素给定字体的 font-size 值 如：p{font-size:12px;text-indent:2em;}

% 如：p{font-size:12px;line-height:130%}

1. 水平居中

①行内元素（在父容器内居中显示）：text-align:center;

②定宽块状元素 margin:20px auto; 即：{margin-top:20px;

margin-left:auto;

margin-right:auto;}

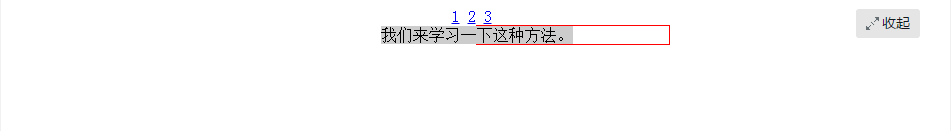
③不定宽块状元素

三种方法

方法一：加入table标签（将不定宽块状转化为定宽块状）

方法二：display:inline; （将不定宽块状转化为[行内元素](http://www.imooc.com/code/2049" \o "忘了的小伙伴，让我们点击链接复习一下吧! )）

方法三：设置浮动和相对定位

咳咳，看个例子。 

这个居中是两个居中

代码：<!DOCTYPE HTML>

<html>

<head>

<meta charset="utf-8">

<title>不定宽块状元素水平居中</title>

<style>

.container{ //第一个居中：.container

float:left;

position:relative;

left:50%

}

.container ul{

list-style:none;

margin:0;

padding:0;

position:relative;

left:-50%;

}

.container li{float:left;display:inline;margin-right:8px;}

/\*下面是代码任务区\*/

.wrap{ //第二个居中：.wrap

clear:both; //消除掉之前float的干扰

border:1px solid red;

float:left;

position:relative;

left:50%;

}

.wrap-center{

background:#ccc;

position:relative;

left:-50%;

}

</style>

</head>

<body>

<div class="container">

<ul>

<li><a href="#">1</a></li>

<li><a href="#">2</a></li>

<li><a href="#">3</a></li>

</ul>

</div>

<!--下面是代码任务区-->

<div class="wrap">

<div class="wrap-center">我们来学习一下这种方法。</div>

</div>

</body>

</html>

15）垂直居中

①父元素高度确定的单行文本

设置父元素的 height 和[line-height](http://www.imooc.com/code/2083" \o "忘了的小伙伴，让我们点击链接复习一下吧!)高度一致

②父元素高度确定的多行文本

<table> eg：<body><table><tbody><tr><td class="wrap">

<div>

<p>看我是否可以居中。</p>

</div></td></tr></tbody></table></body>

16）隐性改变display类型

当为元素（不论之前是什么类型元素，display:none 除外）设置以下 2 个句之一：

 1. [position : absolute](http://www.imooc.com/code/2073" \o "忘了的小伙伴，让我们点击链接复习一下吧! )

 2. float : left 或 [float:right](http://www.imooc.com/code/2071" \o "忘了的小伙伴，让我们点击链接复习一下吧! )

简单来说，只要html代码中出现以上两句之一，元素的display显示类型就会自动变为以 display:inline-block（[块状元素](http://www.imooc.com/code/2048" \o "忘了的小伙伴，让我们点击链接复习一下吧! )）的方式显示，当然就可以设置元素的 width 和 height 了，且默认宽度不占满父元素。

# JS入门

1. 常识
2. 在<script type="text/javascript">与</script>之间写命令；或者将HTML和js分开：<script src="script.js"></script>，在.js文件里写js命令。
3. : javascript作为一种脚本语言可以放在html页面中任何位置，但是浏览器解释html时是按先后顺序的，所以前面的script就先被执行。比如进行页面显示初始化的js必须放在head里面，因为初始化都要求提前进行（如给页面body设置css等）；而如果是通过事件调用执行的function那么对位置没什么要求的。
4. 变量先声明（var）后调用；区分大小写；可以是数字、英文字母、下划线、美元符，但不能数字开头；
5. 语法
6. 定义函数：function 函数名() {函数代码;}
7. Onclick:点击这个button之后的动作

比如<input type="button" value="点击我" onclick="contxt()" /> 点击之后执行函数contxt()

1. document.write()

如：document.write(mystr+"<br>"+”lloveyou”);

# **JS中如何输出空格**

在写JS代码的时候，大家可以会发现这样现象:

****document.write("   1      2                3  ");****

****结果: 1 2 3****

无论在输出的内容中什么位置有多少个空格，显示的结果好像只有一个空格。

这是因为浏览器显示机制，对手动敲入的空格，将连续多个空格显示成1个空格。

****解决方法:****

1. 使用输出html标签&nbsp;来解决

****document.write("&nbsp;&nbsp;"+"1"+"&nbsp;&nbsp;&nbsp;&nbsp;"+"23");****

****结果:****  1    23

2. 使用CSS样式来解决

 document.write("<span style='white-space:pre;'>"+"  1        2    3    "+"</span>");

****结果:**** 1       2     3

 在输出时添加“white-space:pre;”样式属性。这个样式表示"****空白会被浏览器保留****"

1. alert 警告

alert(字符串或变量);

1. 在点击对话框"确定"按钮前，不能进行任何其它操作。

2. 消息对话框通常可以用于调试程序。

3. alert输出内容，可以是字符串或变量，与document.write 相似。

1. confirm 确认

confirm(str);

参数说明:

str：在消息对话框中要显示的文本返回值: Boolean值

返回值：当用户点击"确定"按钮时，返回true

当用户点击"取消"按钮时，返回false

注: 通过返回值可以判断用户点击了什么按钮

eg： var mymessage=confirm("girl?");

if(mymessage==true)

{

document.write("你是女士!");

}

else

{

document.write("你是男士!");

}

1. prompt 提问

prompt(str1, str2);

参数说明：

str1: 要显示在消息对话框中的文本，不可修改

str2：文本框中的内容，可以修改

返回值:

1. 点击确定按钮，文本框中的内容将作为函数返回值

2. 点击取消按钮，将返回null

1. 打开新窗口（window.open）

window.open([URL], [窗口名称], [参数字符串])

参数说明:

URL：可选参数，在窗口中要显示网页的网址或路径。如果省略这个参数，或者它的值是空字符串，那么窗口就不显示任何文档。

窗口名称：可选参数，被打开窗口的名称。

1.该名称由字母、数字和下划线字符组成。

2."\_top"、"\_blank"、"\_selft"具有特殊意义的名称。

  \_blank：在新窗口显示目标网页

\_self：在当前窗口显示目标网页

  \_top：框架网页中在上部窗口中显示目标网页

3.相同 name 的窗口只能创建一个，要想创建多个窗口则 name 不能相同。

  4.name 不能包含有空格。参数字符串：可选参数，设置窗口参数，各参数用逗号隔开。

参数表:

[](http://img.mukewang.com/52e3677900013d6a05020261.jpg)

例如:打开http://www.imooc.com网站，大小为300px \* 200px，无菜单，无工具栏，无状态栏，有滚动条窗口：

<script type="text/javascript"> window.open('http://www.imooc.com','\_blank','width=300,height=200,menubar=no,toolbar=no, status=no,scrollbars=yes')

</script>

1. window.close(); //关闭本窗口

或

<窗口对象>.close(); //关闭指定的窗口

例如:关闭新建的窗口。

<script type="text/javascript">

var mywin=window.open("http://www.imooc.com");

mywin.close();

</script>

1. DOM

1） HTML文档可以说由节点构成的集合，三种常见的DOM节点:

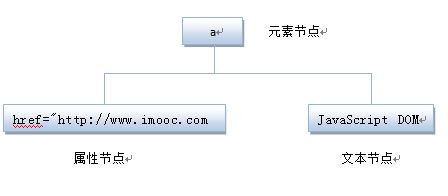
1. 元素节点：上图中<html>、<body>、<p>等都是元素节点，即标签。

2. 文本节点:向用户展示的内容，如<li>...</li>中的JavaScript、DOM、CSS等文本。

3. 属性节点:元素属性，如<a>标签的链接属性href="http://www.imooc.com"。

看下面代码:

<a href="http://www.imooc.com">JavaScript DOM</a>

[](http://img.mukewang.com/52e4bdb80001064c04480196.jpg)

1. 通过ID获取元素

document.getElementById(“id”)

例1： var mychar= document.getElementById("con");

document.write("结果:"+mychar.innerHTML); //输出获取的P标签。

例2：



1. 例题错误集

1）http://www.imooc.com/code/1660

body代码：

<body>

<h2 id="con">JavaScript课程</H2>

<div id="txt">

<h5>JavaScript为网页添加动态效果并实现与用户交互的功能。</h5>

<p>1. JavaScript入门篇，让不懂JS的你，快速了解JS。</p>

<p>2. JavaScript进阶篇，让你掌握JS的基础语法、函数、数组、事件、内置对象、BOM浏览器、DOM操作。</p>

<p>3. 学完以上两门基础课后，在深入学习JavaScript的变量作用域、事件、对象、运动、cookie、正则表达式、ajax等课程。</p>

</div>

<form>

<!--当点击相应按钮，执行相应操作，为按钮添加相应事件-->

**<input type="button" value="改变颜色" onclick="changecolor()">** **//错误：没加括号**

<input type="button" value="改变宽高" onclick="changewandh()">

<input type="button" value="隐藏内容" onclick="hide()">

<input type="button" value="显示内容" onclick="reshow()">

<input type="button" value="取消设置" onclick="cancelall()">

</form>

<script type="text/javascript">

**var txt=document.getElementById("txt"); //错误：没写这句**

//定义"改变颜色"的函数

function changecolor(){

txt.style.color="red";

txt.style.backgroundColor="#CCC";

}

//定义"改变宽高"的函数

function changewandh()

{

txt.style.width="500px";

txt.style.height="500px";

}

//定义"隐藏内容"的函数

function hide(){

txt.style.display="none";

}

//定义"显示内容"的函数

function reshow(){

txt.style.display="block";

}

//定义"取消设置"的函数

function cancelall(){

var mychar=confirm("r u sure?");

if(mychar==true)

{

**txt.removeAttribute('style'); //新的函数，用于取消设置**

}

}

</script>

</body>

1. **JS进阶**

1.

**操作符之间的优先级**（高到低）:

算术操作符 → 比较操作符 → 逻辑操作符 → "="赋值符号

如果同级的运算是按从左到右次序进行,多层括号由里向外。

2.

**a++** ： 把（a++）作为一个整体表达式，a 的值虽然自增1，但是整个表达式的值是取 a自增 之前 的值；  
**++a** : 也把（++a）作为一个整体表达式，a 的值也自增1，但是整个表达式的值是取 a 自增 之后 的值。

3. js**数组长度可变**arr.length=10; 这样重新赋值就好。

1. **break/continue**的区别

前者跳出整个循环，后者只跳出一次循环。

1. **事件**



具体说明：

oninput触发事件是在你修改内容时就会触发，而onchange是在你确定的内容发生修改才会被触发。

onload：加载页面时，触发onload事件，事件写在<body>标签内。

案例：编程出错集合：

<!DOCTYPE html>

<html>

<head>

<title> 事件</title>

<script type="text/javascript">

function count(){

//获取第一个输入框的值

**var a=document.getElementById("txt1").value; //错误：忘写.value**

//获取第二个输入框的值

var b=document.getElementById("txt2").value;

//获取选择框的值

var c=document.getElementById("select").value;

//获取通过下拉框来选择的值来改变加减乘除的运算法则

var r;

switch(c)

{

case "+":

r=a+b;

break;

case "-":

r=a-b;

break;

case "\*":

r=a\*b;

break;

default:

r=a/b;

}

//设置结果输入框的值

**document.getElementById("fruit").value=r; //错误：这整句没写对**

}

</script>

</head>

<body>

<input type='text' id='txt1' />

<select id='select'>

<option value='+'>+</option>

<option value="-">-</option>

<option value="\*">\*</option>

<option value="/">/</option>

</select>

<input type='text' id='txt2' />

**<input type='button' value=' = ' onclick=count() /> //错误：忘了空格（倒数第三位的空格）**

<input type='text' id='fruit' />

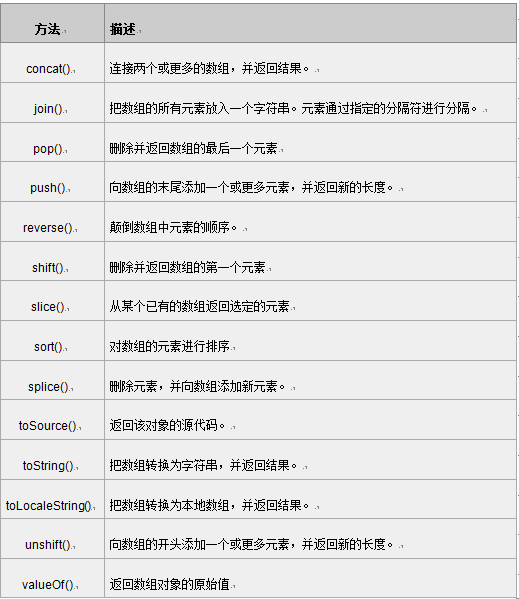
</body>

</html>

1. **对象**

一切事物都是对象，如Array、Date、字符串、Math

**Array:**



slice 作用于数组，[)

\*substring 作用于字符串，（]

**Date ：**



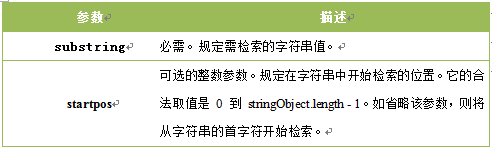
**字符串：**

大小写转换 toUpperCase / toLowerCase

charAt() 方法可返回指定位置的字符。返回的字符是长度为 1 的字符串。

indexOf() 方法可返回某个指定的字符串值在字符串中首次出现的位置。

stringObject.indexOf(substring, startpos)



分割 split() stringObject.split(separator,limit)

提取字符串 substring()/substr()

parseInt() 字符串类型转成整型。

**Math**

注意：Math 对象是一个固有的对象，无需创建它，直接把 Math 作为对象使用就可以调用其所有属性和方法。这是它与Date,String对象的区别。





例题错集：

<!DOCTYPE HTML>

<html >

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<title>系好安全带,准备启航</title>

<script type="text/javascript">

//通过javascript的日期对象来得到当前的日期，并输出

**var my=new Date; //注意**

**var weekday=["日","一","二","三","四","五","六"]; //注意**

document.write(my.getFullYear()+"年"+(my.getMonth()**+1**)+"月"+my.getDate()+"日 星期"+weekday[my.getDay()]);

//成绩是一长窜的字符串不好处理，找规律后分割放到数组里更好操作哦

var scoreStr = "小明:87;小花:81;小红:97;小天:76;小张:74;小小:94;小西:90;小伍:76;小迪:64;小曼:76";

**var scarray=scoreStr.split(";"); //注意区分分割和取出**

//从数组中将成绩撮出来，然后求和取整，并输出。

var sum=0;

for(var i=0;i<scarray.length;i++)

{

**sum+=parseInt(scarray[i].substr(scarray[i].indexOf(":")+1));**

**//注意哪里有双引号哪里没有**

}

document.write("<br>"+"zongfen:"+sum);

document.write("<br>"+"average:"+Math.round(sum/scarray.length));

</script>

</head>

<body>

</body>

</html>

1. **window对象**



1）计时器类型：  
一次性计时器：仅在指定的延迟时间之后触发一次。  
间隔性触发计时器：每隔一定的时间间隔就触发一次。  
2）计时器方法：

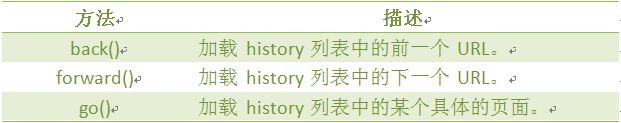
[](http://img.mukewang.com/56976e1700014fc504090143.jpg)

格式：setInterval("clock()",1000)或setInterval(clock,1000)

History对象

window.history.[属性|方法] window可省





ps：back() = go(-1) , forward() = go(1)