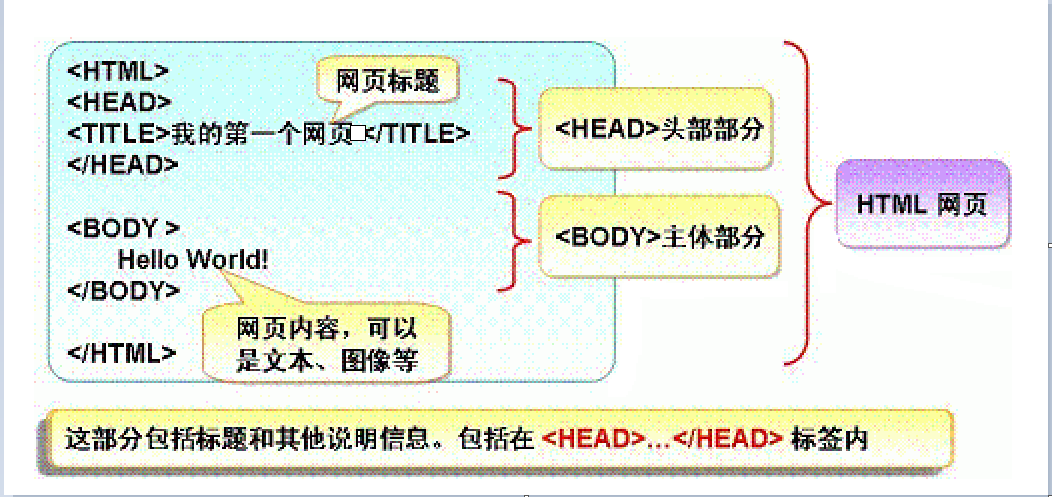
# HTML介绍

HTML的英语意思是：Hypertext Marked Language，即超文本标记语言,是一种用来制作超文本文档的简单标记语言

## 一、HTML基本结构



特点解说：

* 整份文件处于标记<HTML>与</HTML>之间。   
  <HTML>用以声明这是 HTML 文件，让浏览器认出并正确处理此 HTML 文件。
* 文件分两部分，由<HEAD>至</HEAD>称为开头，<BODY>至</BODY>称本文。   
  基本上两者各有适用的标记，如<TITLE>只可出现于开头部分。
* 开头部分用以存载重要资讯，而只有本文部分会被显示。   
  所以大部分标记会运用于本文部分。
* <TITLE>所标示的是文件的标题。   
  会出现于浏览器顶部及为别人 Bookmark 时的名称，所以每页有不同而明确的标题 是需要的。

## 二、标签

HTML 的标签分单标签和成对标签两种。成对标签是由首标签<标签名> 和尾标签</标签名>组成的， 成对标签的作用域只作用于这对标签中的文档。单独标签的格式<标签名>，单独标签在相应的位置插入元素就可以了，大多数标签都有自己的一些属性, 属性要写在始标签内，属性用于进一步改变显示的效果, 各属性之间无先后次序，属性是可选的，属性也可以省略而采用默认值;

其格式如下： <标签名字属性1 属性2 属性3 … >内容</标签名字>

### 1、换行标签<br>

是个单标签，也叫空标签，不包含任和内容，在html文件中的任何位置只要使用了<br>标签，当文件显示在浏览器中时，该标签之后的内容将显示下一行。

### 2、段落标签<p>

由<p>标签所标识的文字，代表同一个段落的文字。不同段落间的间距等于连续加了两个换行符，也就是要隔一行空白行，用以区别文字的不同段落。它可以单独使用，也可以成对使用。单独使用时，下一个<P>的开始就意味着上一个<P>的结束

### 3、原样显示文字标签<pre>

保留原始文字排版的格式

### 4、水平分割线标签<hr>

<hr>标签是单独使用的标签，是水平线标签，用于段落与段落之间的分隔，使文档结构清晰明了，使

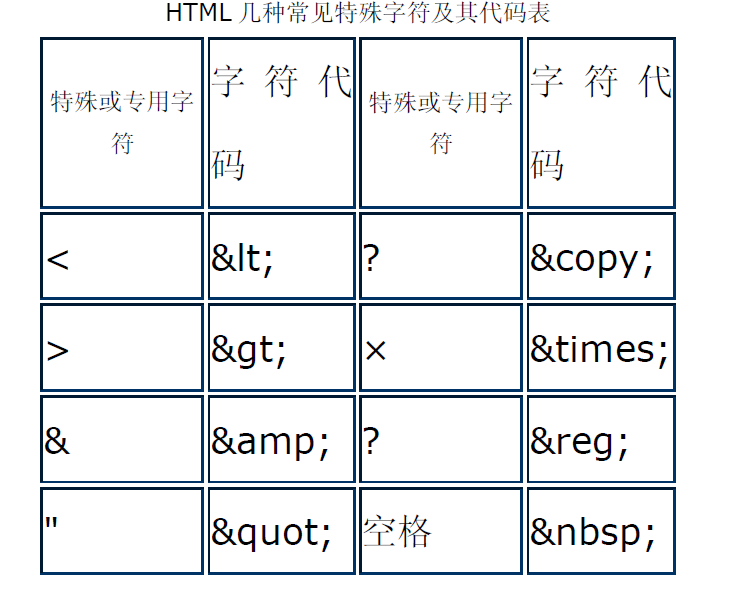
文字的编排更整齐。通过设置<hr>标签的属性值，可以控制水平分隔线的样式。

### 5、署名标签<address>

<address>署名标签一般用于说明这个网页是由谁或是由哪个公司编写的，以及其它相关信息

### 6、特殊字符

在HTML文档中，有些字符没办法直接显示出来，例如?. 使用特殊字符可以将键盘上没有的字符表达出来，而有些HTML文档的特殊字符在键盘上虽然可以得到，但浏览器在解析HTML文当时会报错，例如"<"等，为防止代码混淆，必须用一些代码来表示它们。



### 7、标签注释

在HTML文档中可以加入相关的注释标记，便于查找和记忆有关的文件内容和标识，这些注释内容并不会在浏览器中显示出来。

注释标签的格式有如下：

<!--注释的内容-->

### 8、字体属性

* 标题文字标签<hn> <hn>标签用于设置网页中的标题文字，被设置的文字将以黑体或粗体的方式显示在网页中。

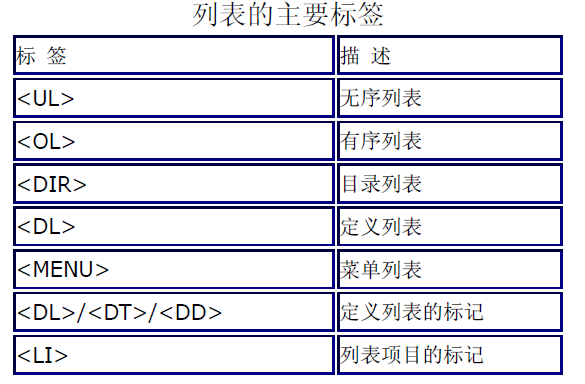
标题标签的格式：<hn align=参数〉标题内容</hn>

* 文字格式控制标签<font> <FONT>标签用于控制文字的字体，大小和颜色。控制方式是利用属性设置得以实现的。

|  |  |  |
| --- | --- | --- |
| FONT标签的属性属性 | 使用功能 | 默认值 |
| face | 设置文字使用的字体 | 宋体 |
| size | 设置文字的大小 | 3 |
| color | 设置文字的颜色 | 黑色 |

格式：<font face=值1 size=值2 color=值3〉文字 </font>

### 9、列表属性

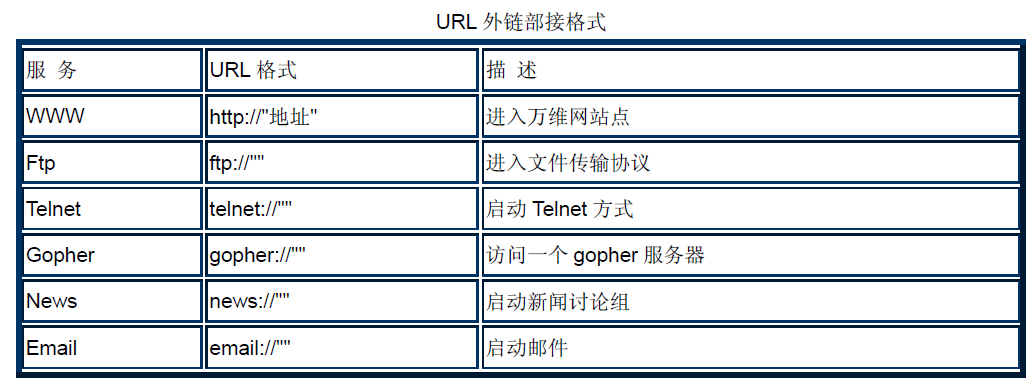


### 10、图像处理

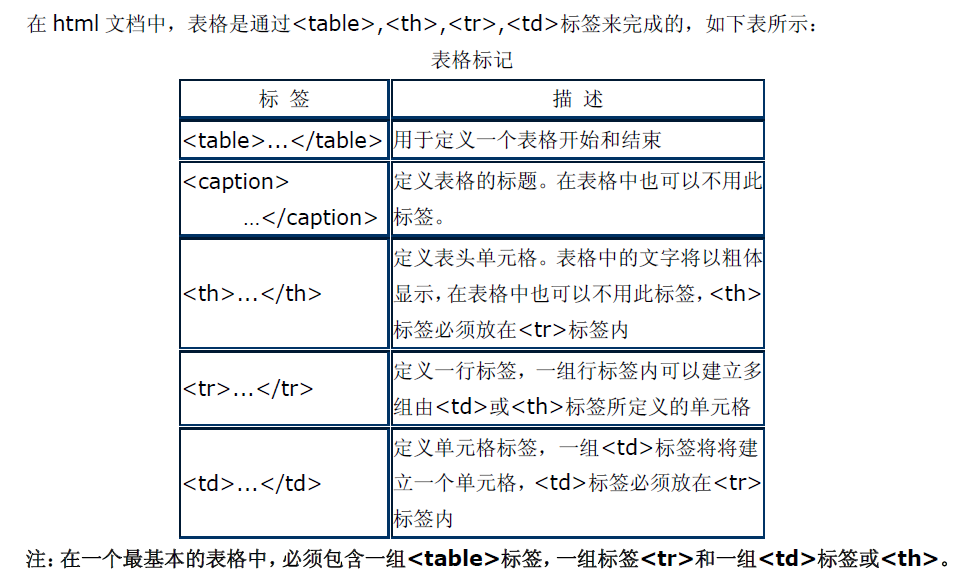
* 背景图像设定：<body background= "image-url">
* 网页中插入图片标签<img>
* 插入普通图片<img src="">
* 用image标签插入avi文件：格式：<img dynsrc="avi文件地址"> 

### 11、建立超链接

格式为：**<**A HREF="资源地址" TARGET="窗口名称" TITLE="指向连接显示的文字">超链接名称</A>



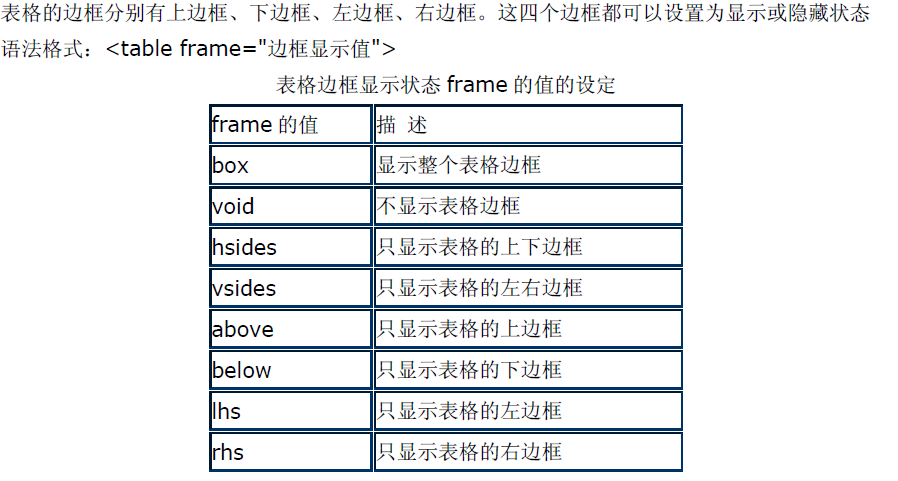
### 12、Table表格



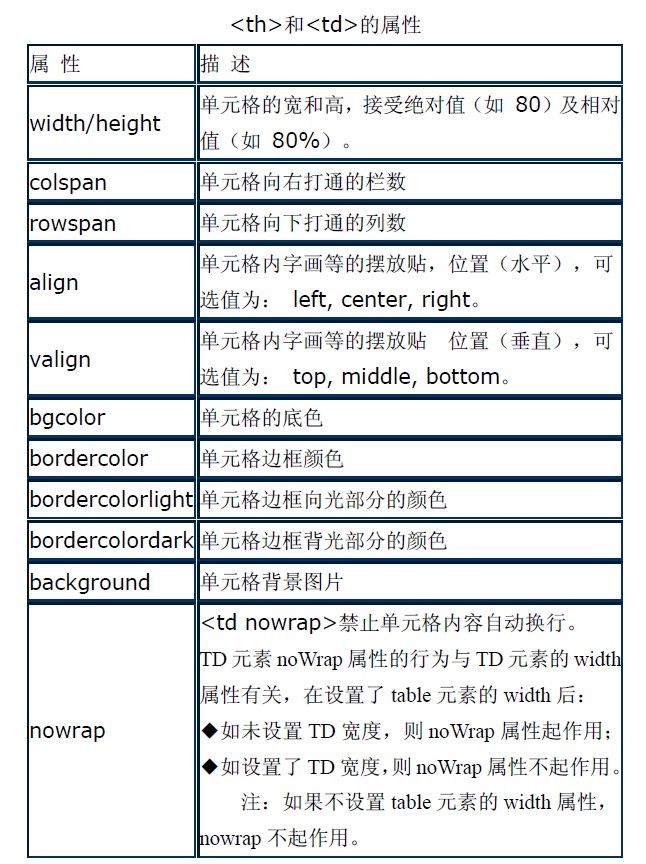
* 表格属性：
* 分割线的显示状态rules：



* 表格的边框显示状态：



* 单元格的设定：



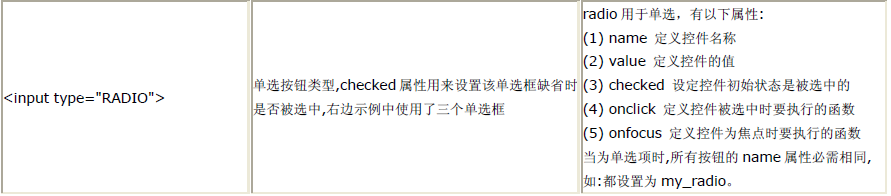
### 13、表单的设计

表单标记<form> 表单在Web网页中用来给访问者填写信息，从而能采集客户端信息，使网页具有交互的功能。一般是将表单设计在一个Html文档中，当用户填写完信息后做提交(submit)操作，于是表单的内容就从客户端的浏览器传送到服务器上，经过服务器上的 ASP 或 CGI 等处理程序处理后，再将用户所需信息传送回客户端的浏览器上，这样网页就具有了交互性

表单标签的格式: <FORM action＝"url" method=get|post name="myform" target="\_blank">... </FORM>

1. 写入标记<input>：<input type="">





常用属性：

1. name 控件名称
2. type控件类型如：button 普通按钮，text 文本框等
3. align 指定对齐方式,可取top, bottom, middle
4. size 指定控件的宽度
5. value 用于设定输入默认值
6. maxlength在单行文本的时候允许输入的最大字符数
7. src 插入图像的地址
8. event 指定激发的事件

# CSS介绍

## 一、CSS语法：



## 二、基本特性：分组，继承，层叠

**分组：**同时改变几种样式的属性，CSS允许声明通过使用分号分隔其内容的方式来进行分组。例如：body,td,p {color:#000000;text-align:left;}

**继承：**通过继承，CSS设置可以被应用到多个标识中。绝大部分的CSS声明可以通过封闭CSS选择器中的HTML标识来被继承。例如：通过一个CSS设置来改变整个页面的字体：body { font-family: "Arial"; font-size: 14px; line-height: 18px; color: #000000}

**层叠：**Cascading描述了覆盖通用样式的局部样式的能力。在相同的方式中，一个普遍地应用到某一块文本上的CSS规则可以被其他应用到相同文本中某个更为特殊的部分的规则所覆盖。

## 三、样式表的放置

总共有4种方法，最常用为前3种

1. 内部（或嵌入式）CSS样式表，即直接把CSS代码粘贴到html中
2. 外部导入CSS样式表，即将外部的\*.css文件导入到所需的网页面HTML文件上，放置于<head><style>…</style></head>之间。
3. 外部链接CSS样式表，即先编写一个\*.css的文件(并非HTML文件)，然后将这个css文件链接到所需的页面HTML文件上。是最常用的方法
4. 行间写入式CSS样式表

## 四、创建class：

创建类（CLASS）可以创建同一个HTML标志的多种风格。

语法：标签.类名{属性：值；属性：值；……属性：值}

引用方法：< 标签CLASS="类名">

## 五、CSS样式的优先权

优先级顺序：内联CSS样式表 > 内部样式表 > 外部样式表 > 浏览器默认样式

参考文档：http://jingyan.baidu.com/article/49ad8bce771c365834d8fa29.html

<html>  
<head>  
    <!--外部样式表-->  
    <link      href="style.css"  rel="stylesheet"  type="text/css">  
    <!--内部样式表,写在head标签内部，优先级第二-->  
    <style type="text/css">  
        body  
        {  
            background-color:green;  
        }  
    </style>  
</head>  
<!--内联css样式表优先级最高，写在html标签里，即style属性-->  
<body style="background-color:blue;">  
验证优先级顺序  
</body>  
</html>

# JS介绍

JavaScript 是一种轻量级的编程语言

## 一、JS实现原理（执行原理）

1、客户端请求某个网页，即我们在上网时在地址栏中输入某个网址，浏览器接收到网址之后，向远程web服务器提出请求

2、web服务器响应请求，web服务器找到请求的页面，并将整个页面包含javascript的脚本代码作为相应内容，发送回客户端机器

3、客户端浏览器解释并执行带脚本的代码，客户端浏览器打开回应的网页文件内容，从上往下逐行读取并显示其中的html或者脚本代码，脚本是从服务器端下载到客户端，然后在客户端进行的，即不占用服务器端的资源，因此通过客户端脚本，客户端分担了服务器的的任务，大大的减轻了服务器的压力，从而间接地提升了服务器的性能

## 二、JS实现方式

<html>

<body>

<script type=”text/javascript”>

Document.write(“HELLO WORLD”);

</script>

</body>

</html>

## 三、执行顺序

自上而下顺序执行

## 四、基本语法

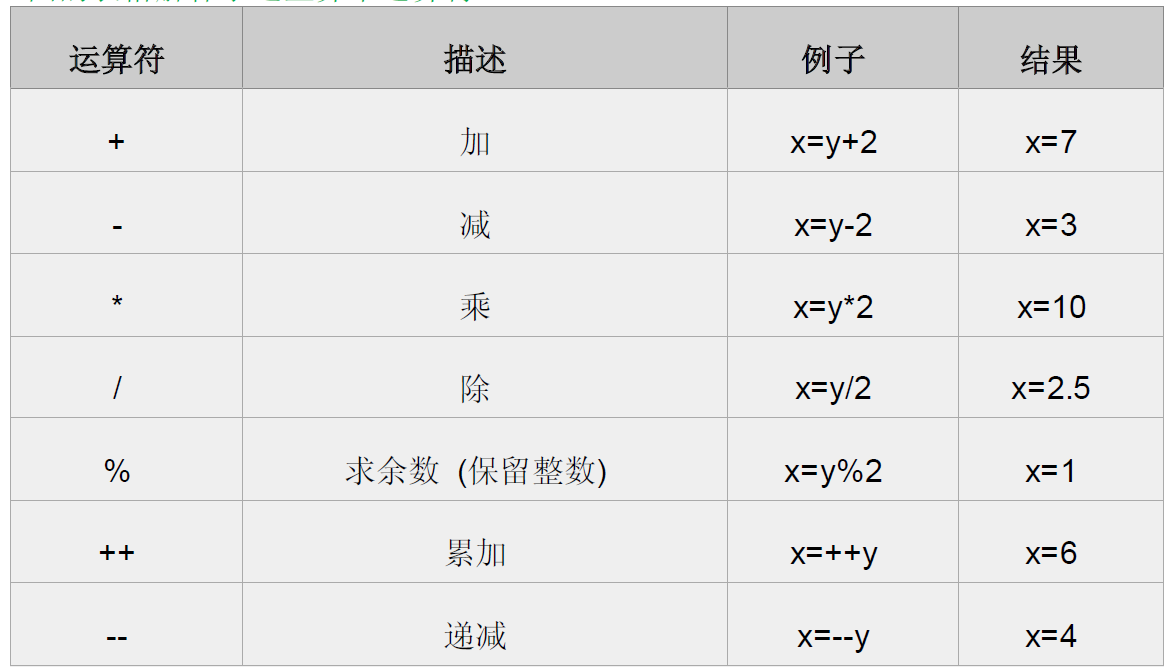
### 1、变量

变量是存储信息的容器，在js 中我们使用var 声明变量，声明变量的同时我们可以为变量赋一个默认值，例如var a; var b=3;(变量的声明区分大小写)

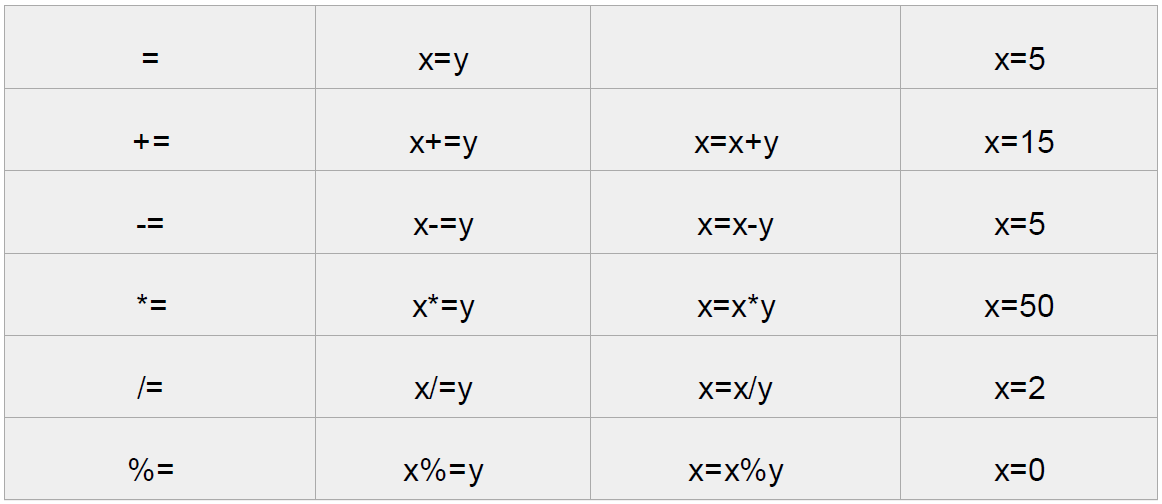
### 2、运算符

包括算术运算符，赋值运算符，比较运算符，逻辑运算符，条件运算符

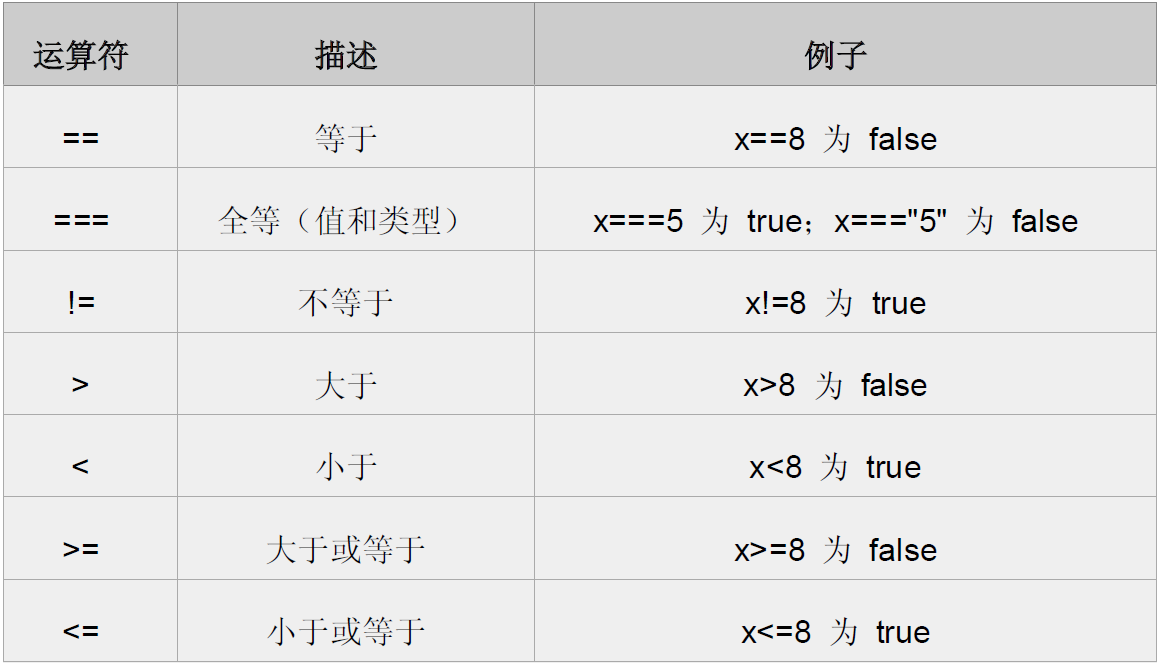
算术运算符，例如：y=5



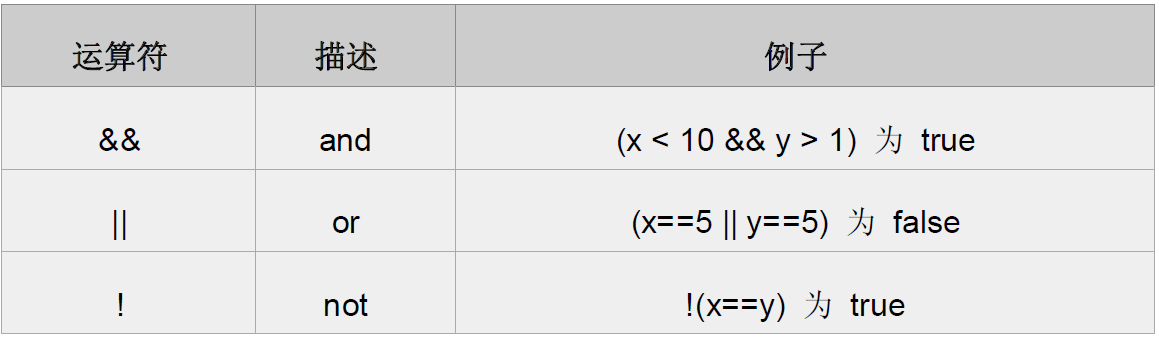
赋值运算符：x=10,y=5



比较运算符，x=5



逻辑运算符，x=6,y=3



条件运算符：

Var x=1;

Var y=2；

X=(y)?value1:value2

声明变量x,y；x 等于y 吗？如果是true 返回x 的值value1，否则返

回x 的值value2.

### 3、逻辑判断

If语句：在一个指定的条件成立时执行代码。

if...else 语句：在指定的条件成立时执行代码，当条件不成立时执行另外的代码。

if...else if....else 语句：使用这个语句可以选择执行若干块代码中的一个。

switch 语句：使用这个语句可以选择执行若干块代码中的一个。

### 4、消息框

警告框：alert("文本")

确认框：confirm("文本")

提示框：prompt("文本","默认值")

### 5、函数

将脚本编写为函数，就可以避免页面载入时执行该脚本。

函数包含着一些代码，这些代码只能被事件激活，或者在函数被调用时才会执行。

语法：function test(var a,var b){

要执行的程序

}

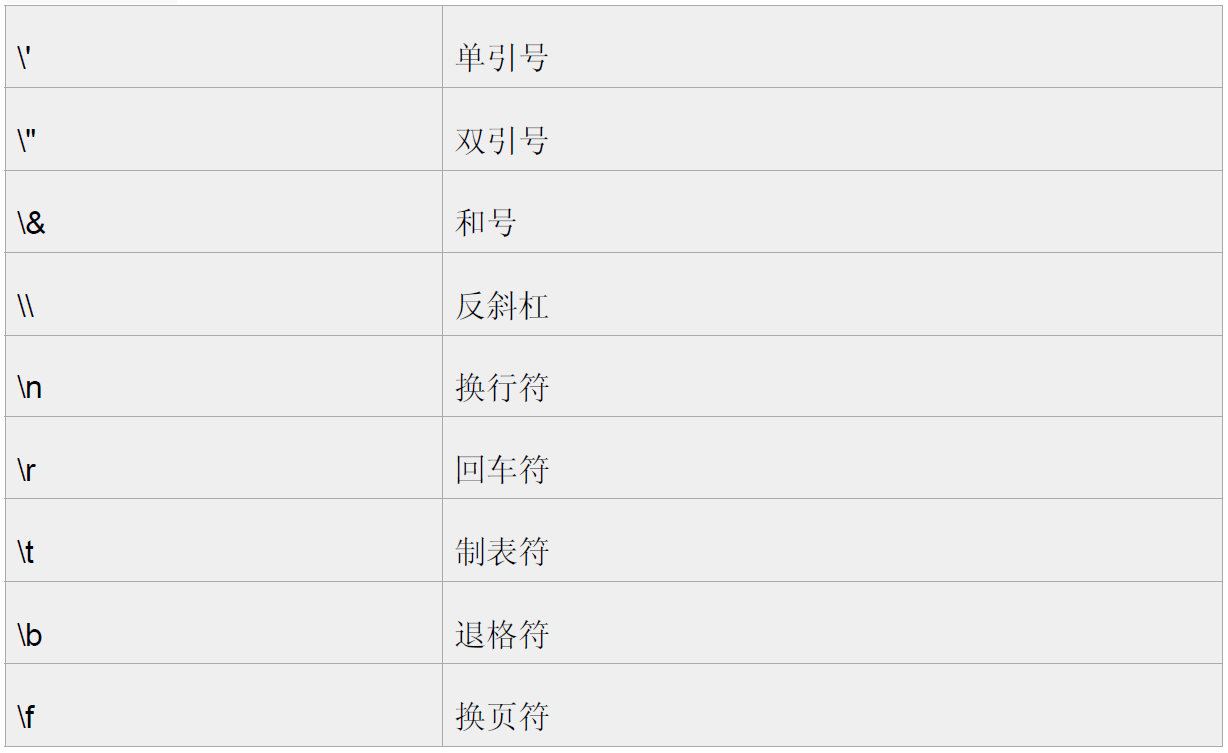
Var a 和var b 是函数定义的要转入的值或者变量，大括号定义了函数的开始和结束

### 6、事件

事件是可以被javascript 侦测到的行为。网页中的每个元素都可以产生某些可以触发JavaScript 函数的事件。比方说，我们可以在用户点击某按钮时产生一个onClick 事件来触发某个函数。

### 7、特殊符号

在javascript 中我们经常使用反斜杠来插入一些特殊字符，比如在文本字符串中插入省略号、换行符、引号和其他特殊字符。



### 8、HTML页面加载解析过程

html解析过程是从开头一行一行执行。  
如果遇到CSS的加载以及JS的加载的话就会阻塞等待这些东西加载完毕甚至是JS执行完毕之后才进行继续加载。所以页面中如果JS全放在头部并且开始就执行的话，如果网速跟不上，可能出现短暂的空白！所以JS一般放在页面末尾。  
css放在头部！  
加载完毕后，会进行结构渲染。