# HTML快速入 门

## 1，什么是HTML

超文本标记语言（英语：HyperText Markup Language，简称：HTML）是一种用于创建网页的标准标记语言。

超文本：文字、图片、动画、音频和视频、超链接和程序等。

纯文本：只有文字，没有样式，可以用记事本打开，没有乱码

标记：也称也标签，HTML是由一堆标签组成的，每个标签都有不同的功能。

网页：凡是用浏览器打开的，都叫做网页。

HTML主要是用来给网页提供内容， CSS主要是用来给网页提供样式

WEB三层结构：

1，结构层-->HTML

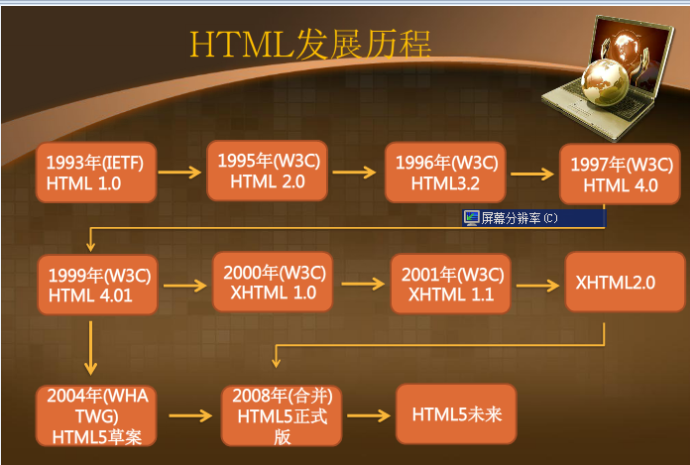
2，样式层-->CSS

3，行为层-->JS

**特点**

* 简单
* 通用性
* 平台无关

## 2，HTML的发展历程



## 3，浏览器



IE: trident内核

Firefox：gecko内核

Safari:webkit内核

Opera:以前是presto内核，Opera现已改用Google Chrome的Blink内核

Chrome:Blink(基于webkit，[Google与Opera Software共同开发](http://baike.baidu.com/view/1369399.htm#1_5))

## 4，编写第一个html网页

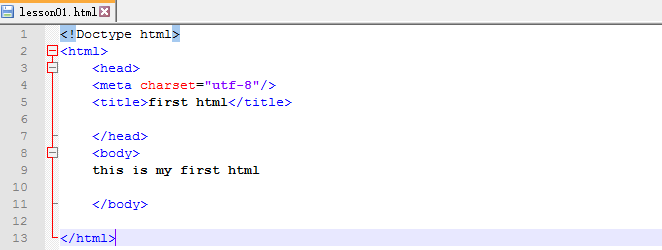
准备工作

浏览器： 谷歌， 火狐， IE， Safri， 欧朋

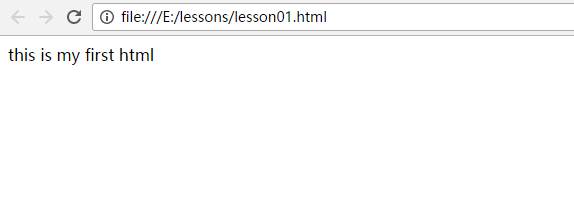
开发工具：Editplus，webstorm,hbuilder,sublimetext

开始编写

1. 新建html文档
2. 在文档输入以下内容



1. 使用浏览器打开



# HTML基本结构及语法

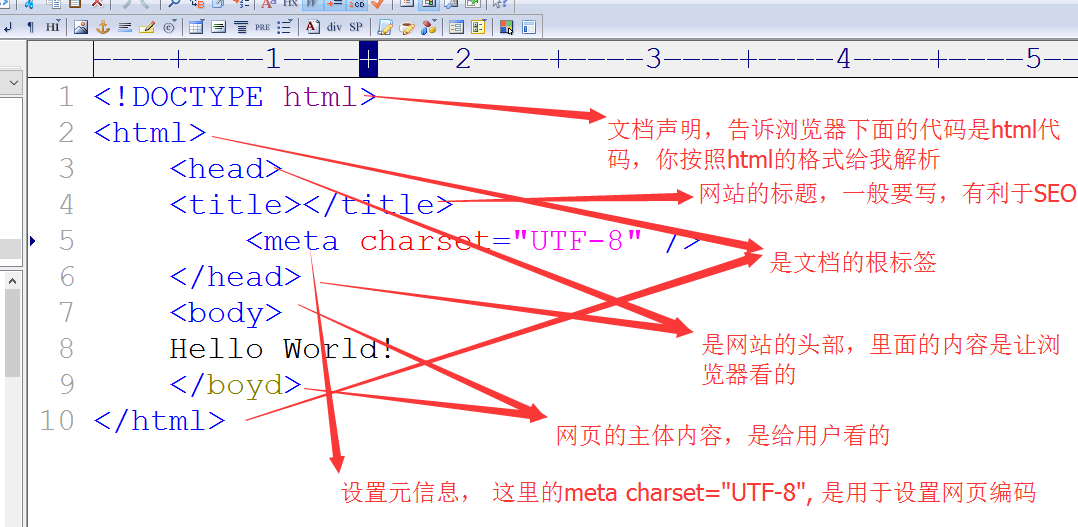
http://www.w3school.com.cn/html/index.asp

## html结构

基本构成（文档声明、头部和主体）

* 文档声明
* 头部
* 主体

基本结构：



<!DOCTYPE> 声明位于文档中的最前面的位置，处于 <html> 标签之前。此标签可告知浏览器文档使用哪种 HTML 或 XHTML 规范。（重点：告诉浏览器按照何种规范解析页面）

## 2，html标签一览表

结构：html、head、body、title、meta

文章：h1~h6、p、br、em、strong

超链接：a

图片：img

布局：div、span 配合CSS完成布局，美化网页

列表：ul、ol、li、dl、dt、dd

表格：table、tr、td、th

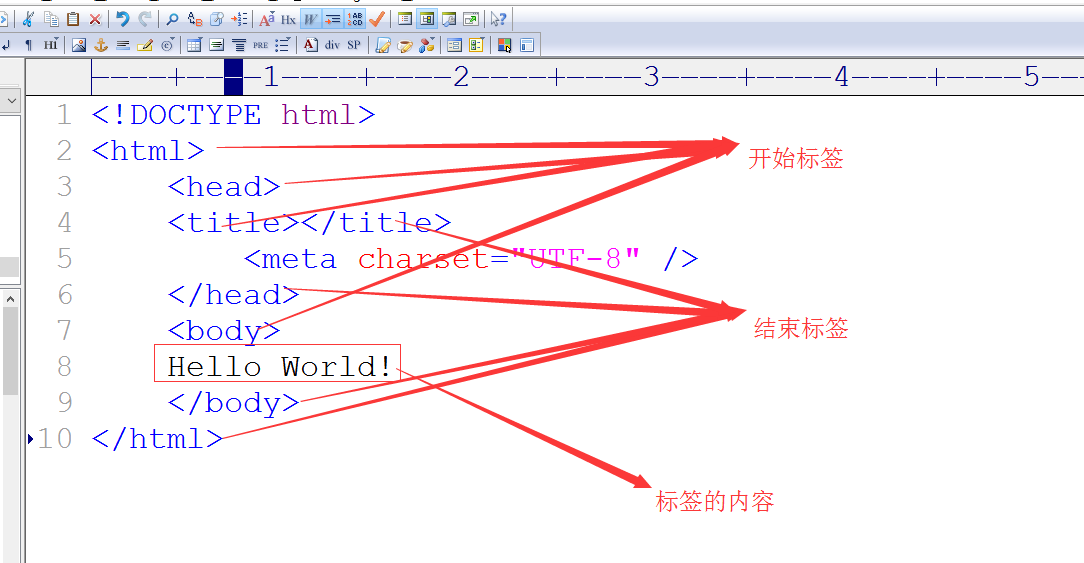
表单：form、input、select、option、textarea

大概有8类，从结构学习，最终学习到表单。

## 3，html标签构成

标签构成：

* 开始标签
* 内容
* 结束标签



元素：

开始标签+结束标签+标签之间的内容 = 元素

标签分类：

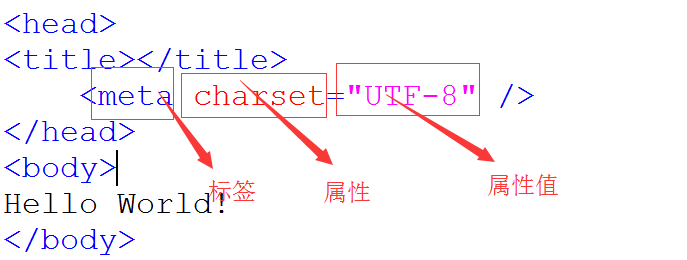
单标签：只有开始标签，没有结束标签，最后的/可加可不加。

双标签：有开始标签，也有结束标签，结束标签最后有一个/。

注意：

并非所有的元素都有结束标签，比如，<img>、<br>和<hr>都可以看作是自结束元素。

属性，可以使用单/双引号、也可以不加



## 4，html标签语法

**语法：**

HTML的语法比较松散

* 不区分大小写
* 对于自结束元素，可以加/，也可以不加/
* 属性可以不加引号

**好习惯：**

* 必须有文档声明
* 标签全部使用小写
* 为属性值加上引号
* 标签可以嵌套，但不能交叉。
* 标签要正确的关闭

# html基本标签

1，标题标签

**作用**：用来定义标题， <h1> 定义最大的标题。<h6> 定义最小的标题。

**格式**：<hn>内容</hn> (n=1-6)

h是headline的缩写

几乎每个网站都有h标签， 尤其是h1和h2

基本使用：



浏览器运行效果：



每一个浏览器都有一个小工具，叫做开发者工具。F12或鼠标右键选择检查元素就可以打开开发者工具。

经典用法：

1，新闻/文章/博客的标题通常使用h1标签：



2，网站的Logo



## 2，段落标签

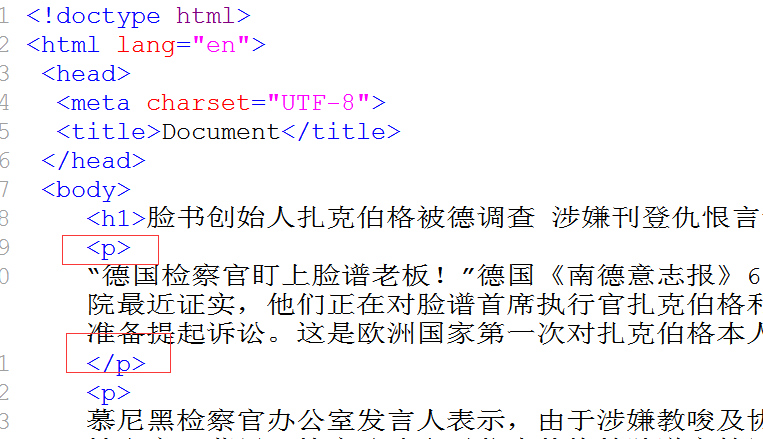
作用：用来分隔内容，使内容调理清晰。

格式：<p>内容</p>

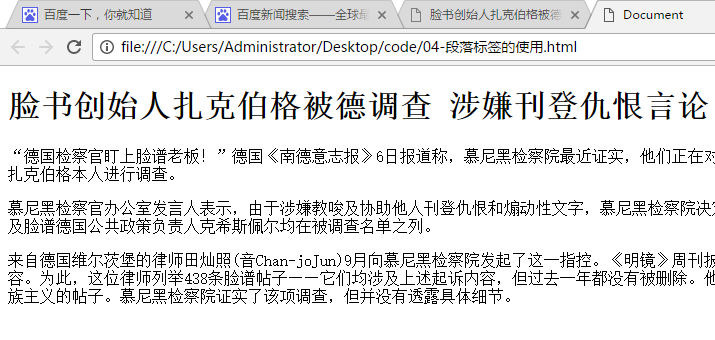
p是paragraph（段落）的缩写

<br/>W3C是建议配合p使用。

基本使用：

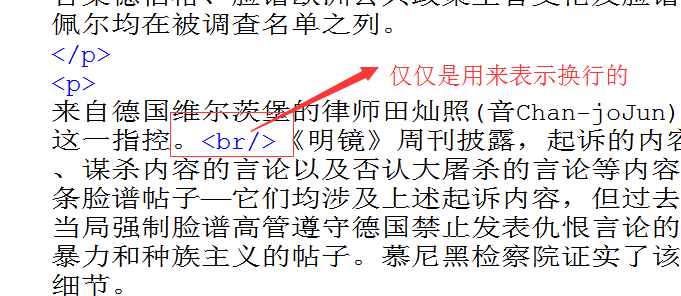


浏览器效果：

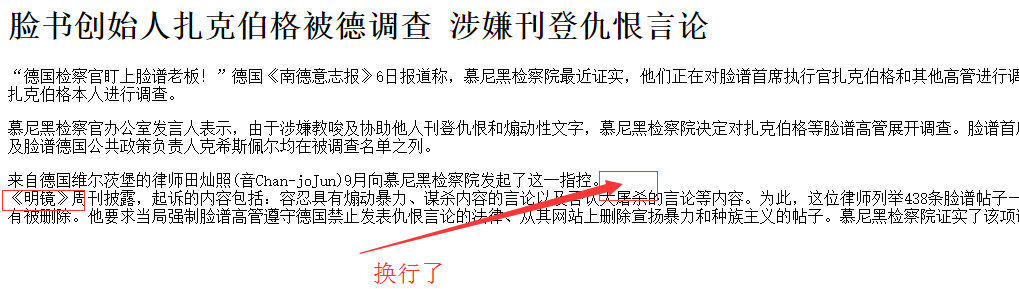


## 3.换行标签<br/>

br的使用：



浏览器效果：



p标签和br标签的区别：

1，p标签是双标签，br标签是单标签

2，它们表示的含义不同，p表示一个段落，br仅仅是表示换行

3，段落标签的经典应用

用于文章/新闻/博客的页面中。



## 3，超链接标签及其属性

作用：用于链接其它资源，从而形成万维网。

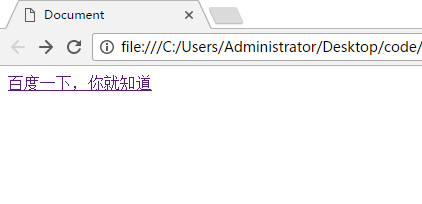
格式：<a href='' target='' name='' title=''>内容</a>

a是anchor（超链接）的缩写

基本用法：



效果：



超链接标签常见的属性：

1，href: 用于链接某个资源

2，target属性：表示目标，我们在打开一个页面时，是在新窗口打开，还是在当前窗口打开，或者在个框架中打开。它有几个属性值，我们只需要知道下面两个就行：

\_blank 表示在新窗口打开

\_self 表示在当前窗口打开， 默认就是\_self

3，title：当鼠标放上去的时候，显示title对应的内容

注意的问题：

1，对于超链接，如果不写href属性，是没有任何意义。

2，如何想访问web上的一个资源，需要加上http://

3，任何标签都有title属性，只在在超链接标签中用的比较多。

**A,站外链接**

所谓的站外链接，就是链接到web上的某个页面或网站



网址：一般描述为URL， 统一资源定位符（Uniform Resource Location）,它是专门用来描述某个资源在web上的地址。

**B,站内链接**

站内是指在一个网站内部的链接

网站是由一系列的见面构成的

对于站内链接，一般不使用绝对路径，会使用相对路径 ，相对路径是相对于本文件的

./ 当前文件 可以省略。

../ 上一层 不能省略。

没有.../

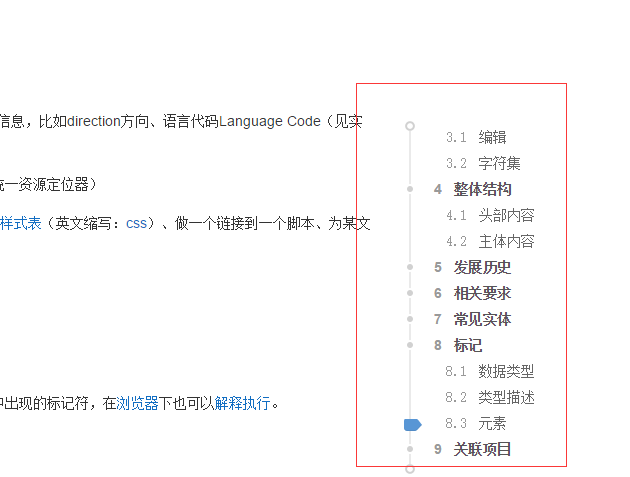
**C,锚点链接**

常见效果：

1，hao123

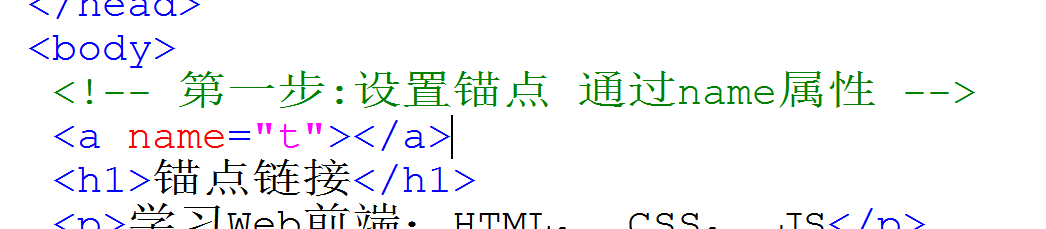


2,百度百科：

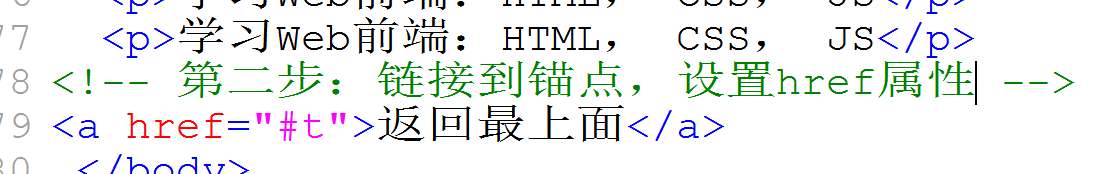


设置锚点链接的步骤：

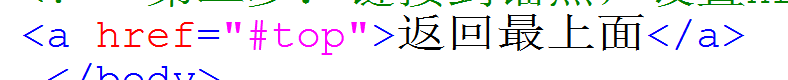
1，建立锚点，需要设置name属性



2，链接到锚点，需要设置href属性



在实际使用时，有一个默认的锚点#top, 表示顶部：（不用再顶部打锚点了）





面试题(以下两者有什么区别) ：

<a href=””></a>

<a href=”#”></a>

不带#表示重新加载，带#仅仅是返回顶部。

**D，其它资源链接**

实际上，在HTML中，a可以链接的资源有很多，包括HTML文件，图片，word文档，压缩文件，视频等等。

如果浏览器可以解析， 那么就直接解析并显示。

如果浏览器不能解析，那么就会以下载的方式来处理。

链接到张图片：



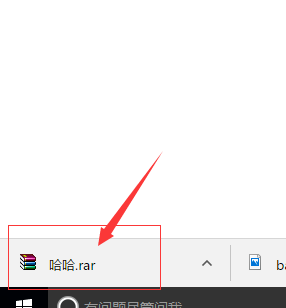
图片浏览器可以解析，结果：



如果链接到一个压缩文件：



由于浏览器不能解析压缩文件，将以下载的方式来处理：



**E,email链接**

**通常是发送邮件的时候，会用到**



如果我们的电脑上安装了outlook或foxmail的由箱工具，会启动这些工具，开始写邮。

关于email链接，用的不多，了解不行。

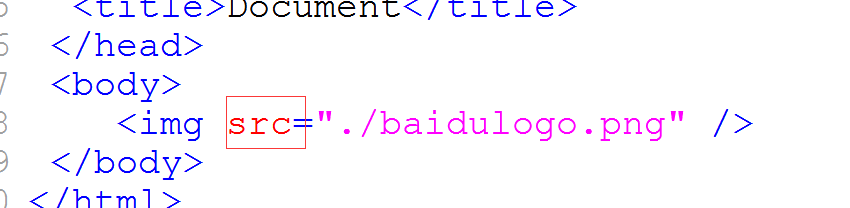
## 4，图像标签及其属性

作用：用于显示图片

格式：<img src='' width='' height='' alt='' title=''>

Img是一个单标签

基本使用：



效果 ：



Img 是image（图片）的缩写

Src 是source（源）的缩写

默认情况下，图片是按照实际大小来显示的

在实际开发中，如何你想改变图片的大小，可以通过以下两个属性：

Width和Height



效果：



注意：

1，如果只设置宽度， 或只设置高度，那么对应的宽高成比例变化。

2，如果不写单位，那么默认的单位是px, 是像素的意思

**alt属性**

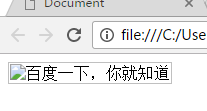
1, 当图片显示不出来的时候，显示alt对应的内容

2, 有利于SEO（搜索引擎优化）， alt是给搜索引擎看的， 是alternate(替代)

Alt的作用1：



效果：

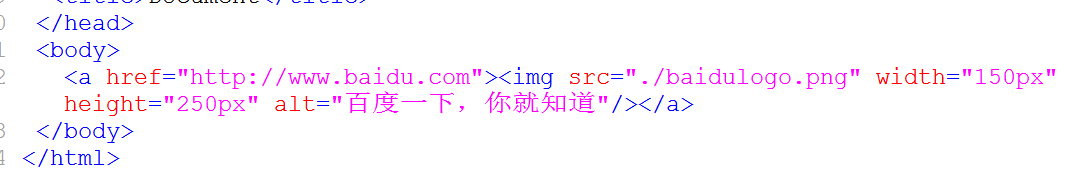


图片作为超链接使用：

1，单个图片作为超链接

2，图片映射

单个图片作用超链接：



效果：

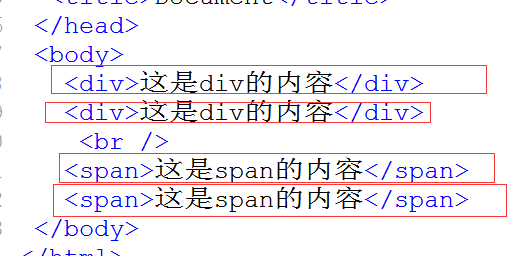


## 5，div和span

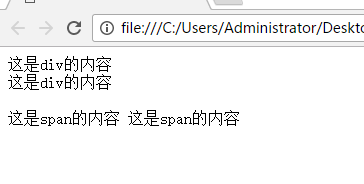
作用：用来组织划分结构， 没有任何语义，是配合CSS完成页面布局，美化网页

格式：<div></div> <span></span>

基本用法：



效果图：



说明：

1, DIV单独占一行，它是一个块级元素， 没有语义

2，span可以实现并列，它是一个行内元素，也没有语义

DIV在做网页的时候用的最多：



span是负责分隔一个模块中非常小的一个内容

## 6，语义化

**什么是语义化**

HTML的每个标签都有其特定含义，Web语义化是指使用语义恰当的标签，使页面有良好的结构，页面元素有含义，能够让人和搜索引擎都容易理解。

**为什么需要语义化**

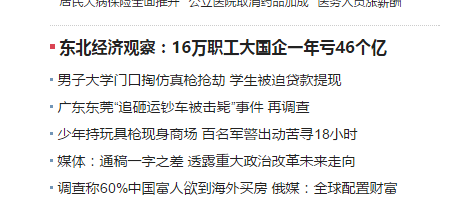
* 为了在没有CSS的情况下，页面也能呈现出很好地内容结构、代码结构
* 方便其他设备解析（移动设备、如屏幕阅读器）
* 有利于SEO
* 便于团队开发和维护

# 列表标签

## 1，基本介绍

先看以下几个效果：

百度新闻：



优酷：



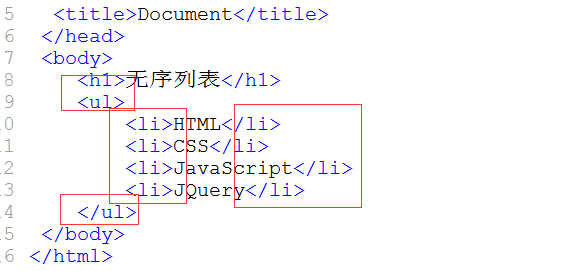
列表是一组标签，分为有序列表，元序列表和自定义列表。

无序列表：没有固定顺序，将一组相关的列表项目排列在一起，列表项之间没有先后顺序。

无序列表：

<ul></ul> 每一个列表项用<li></li>

基本用法：



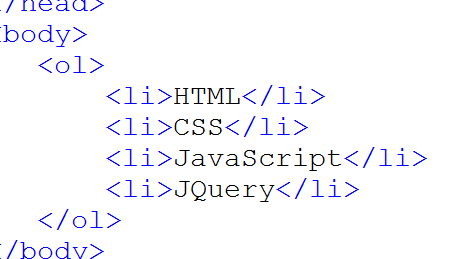
效果：



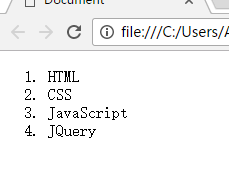
有序列表：

<ol></ol> <li></li>

基本用法：



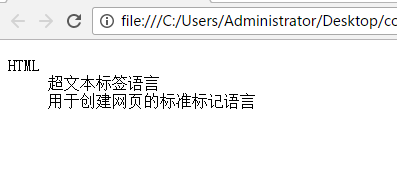
效果：



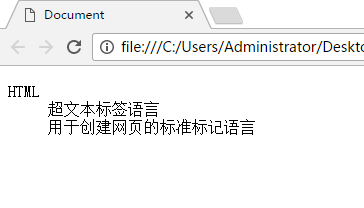
自定义列表 ：

<dl></dl> <dt></dt> <dd><dd>

基本用法：



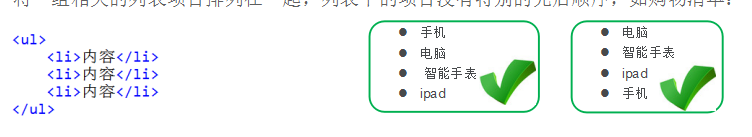
效果：



## 2，无序列表

作用：将一组相关的列表项目排列在一起，列表中的项目没有特别的先后顺序，如购物清单：

格式：



默认情况下，列表项前有一个实心圆点符号，这个符号可以修改

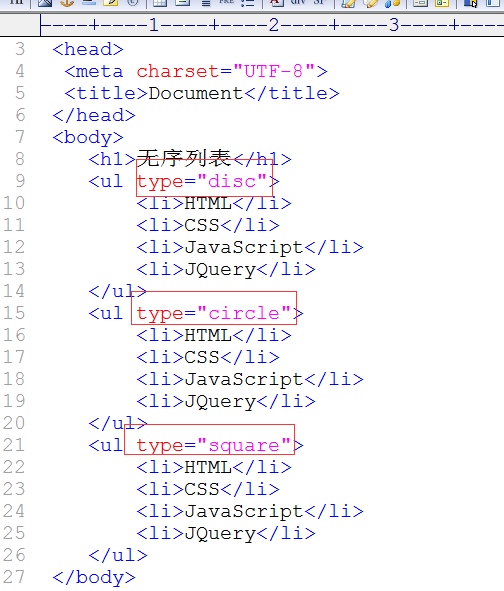
可以指定type=”属性值”进行修改。

type=”disc” 实心圆点，是默认值，可以就写

type=”circle” 空心圆点

type=”square” 实心方块

用法：



效果：



无序列表的经典用法：

1，新闻列表，文字列表，博客列表，商品列表，评论列表中。

2，导航， 需要配合CSS内容才可以实现

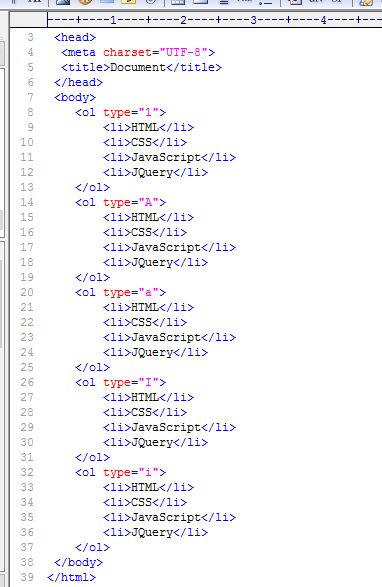
## 3，有序列表

作用：用于将一组相关的列表项目排列在一起，列表中的项目有特别的先后顺序

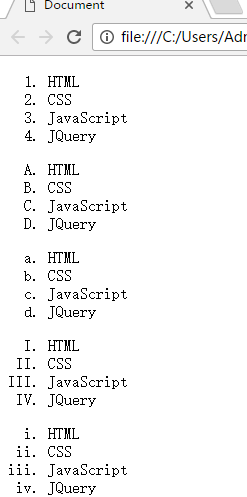
格式：



Ol里面也有一个type属性：



效果：



经典应用：

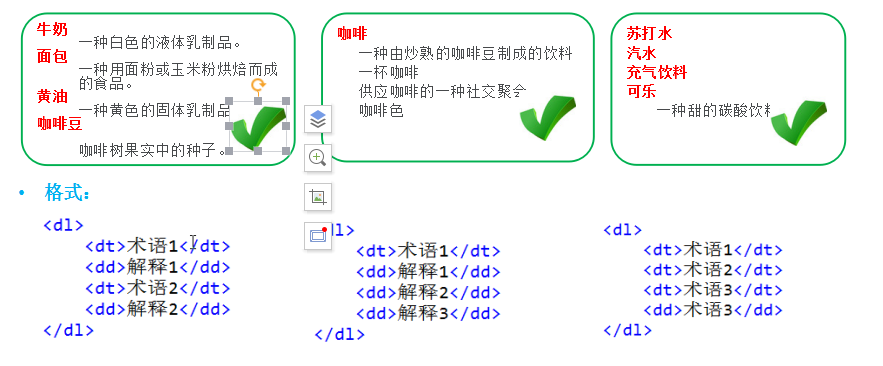
一般情况下，有严格的先后顺序的多修关联项之间，应该使用ol+li

典型就是应用在排行榜

但在实际开发中，有序列表很少用，即使列表项之前有先后顺序，有的也有无序列表，前面用图片来表示顺序。

## 4，自定义列表

作用：用于显示名称及其对应的“值”，如术语及其定义，或时间及相对应的事件。



下面这定义列表的三种用法：



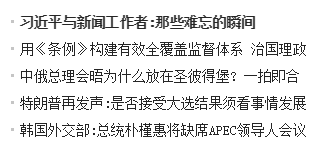


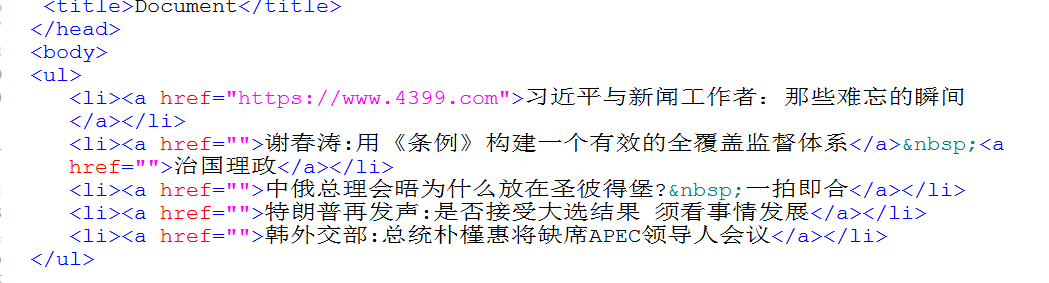
注意：

在实际开发中，dl使用的比较少，但可以用在特殊的场合。

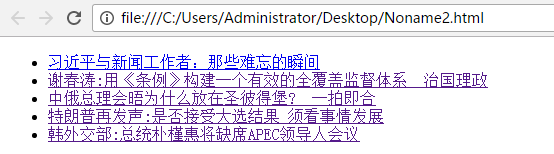
案例练习：

1，实现网易新闻的列表内容：

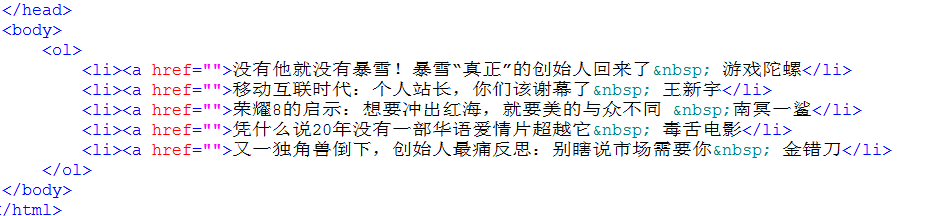




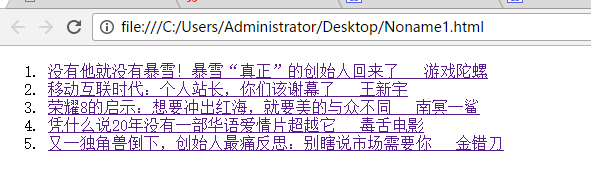
效果：



2，实现百度新闻的热门文章：

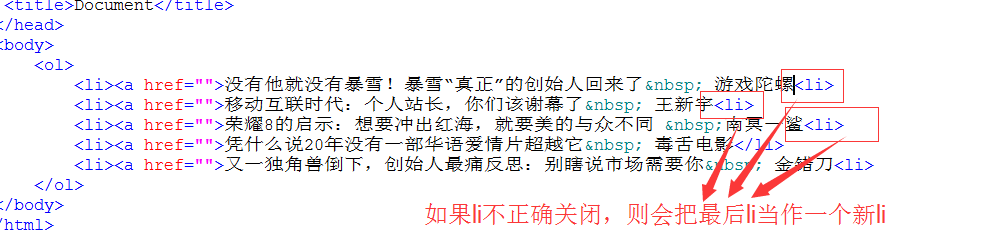


效果 ：

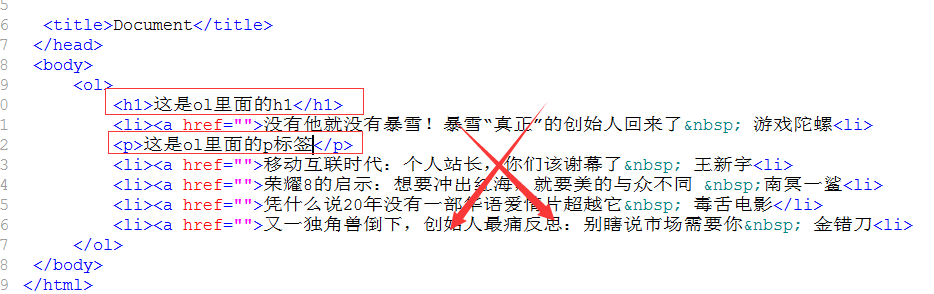


常出现的问题

1，标签的不正确关闭



2，列表标签的包含



列表标签里面只能写列表项，意思就是ol, ul里面只能li, 不能有其它标签， 但是li标签里面可以有其它标签。

# 表格

## 1，表格赏析

基金信息，http://fund.eastmoney.com/dingtou/syph\_yndt.html

体育比赛的统计，http://data.sports.sohu.com/nba/nba\_players\_rank.html

尺码表，http://www.zolsky.com/read/best/6/297.htm

列车时刻表，https://kyfw.12306.cn/otn/leftTicket/init

## 2，创建基本表格

在html中创建表格， 也是一组标签， 最核心的以下几个：

table 表格

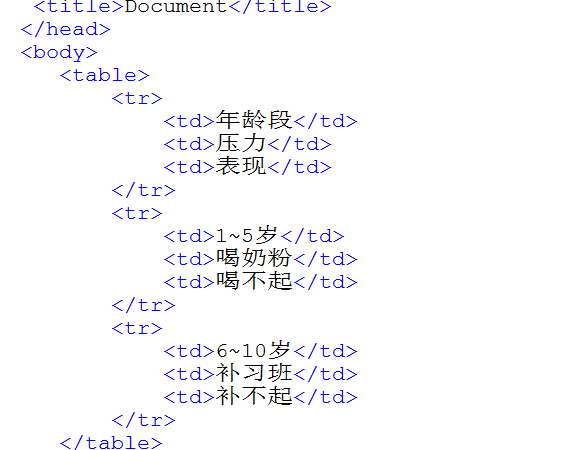
tr 行

td 单元格

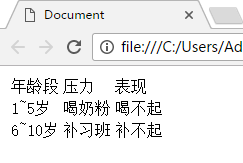
th 单元格

表格是由行和列构成的一个二维结构，其中行在html中用tr实现，列由多个td构成，其中td/th是对应到具体有行号和列号确定的唯一单元格。

基本用法：



效果：



## 3，表格标签相关属性

table的属性：

widht：宽度

height: 高度

align:表格的对齐方式

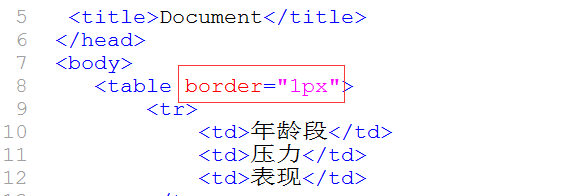
bgcolor: 背景颜色

border: 边框

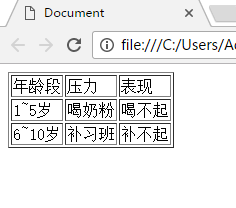
cellspacing: 单元格之间的距离

cellpadding: 单元格中的内容距离单元格边框的距离

border: 表示边框：

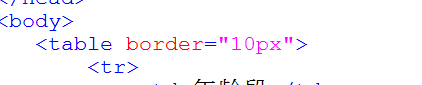


效果：

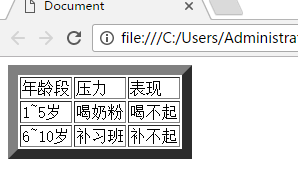


注意：  
 1，border作用于table，但是它对里面的单元格也会起作用。

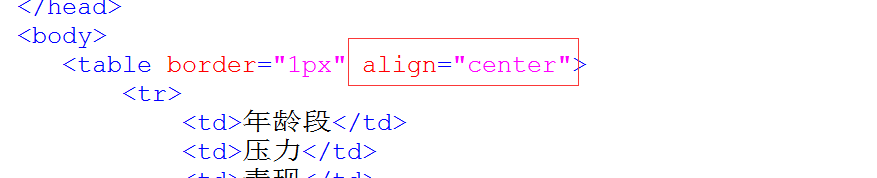
2，如果把border边框的宽度调大，只是对最外层边框起作用，里面的单元格的边框不变。不管border设置多大，单元格的边框都为1， 真正变化的是table的边框：



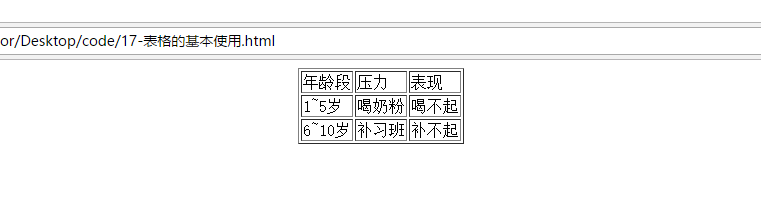
效果：



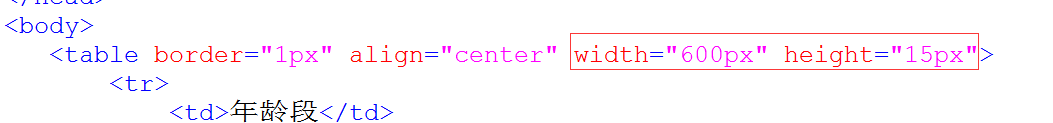
Align: 可以设置表格相对于父元素的对齐方式，在当前页面中，父元素是body.



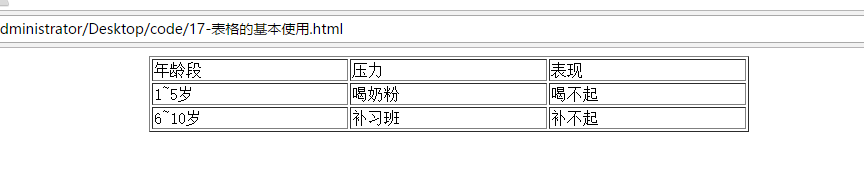
效果是在body中居中对齐：



width和height: 用来设置表格的大小



效果：

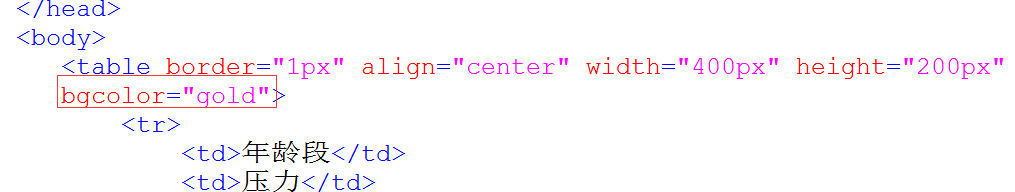


注意点：

对于width和height的理解，一般理解成min-width和min-height。

bgcolor属性：

用来设置背景颜色：

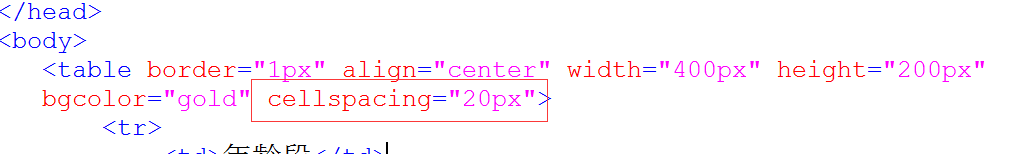


效果：



Cellspacing和cellpadding:

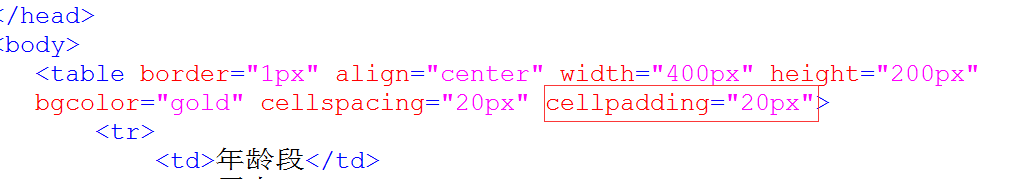
Cellspacing:



效果：



Cellpadding:



效果：



**Tr的属性：**

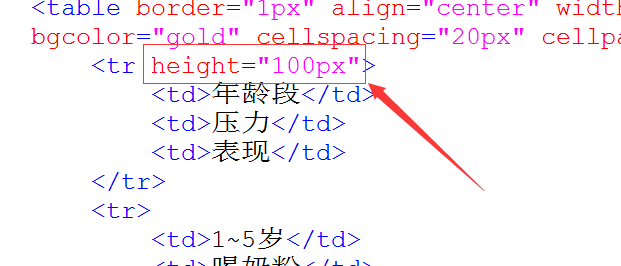
height: 设置高度

bgcolor: 设置背景颜色

align: 设置水平对齐方式

valign: 设置垂直对齐方式

height: 设置行的高度



效果：

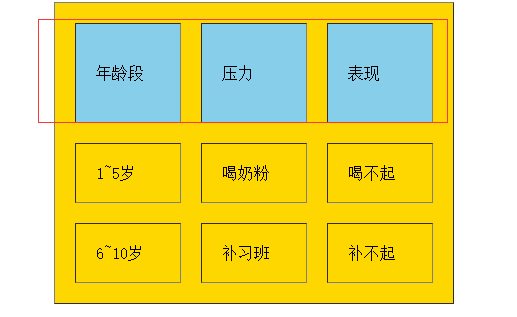


Bgcolor:

设置一行的背景颜色



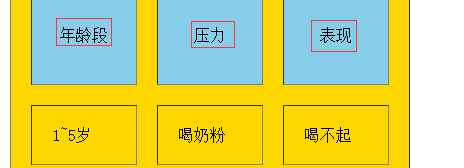
效果：



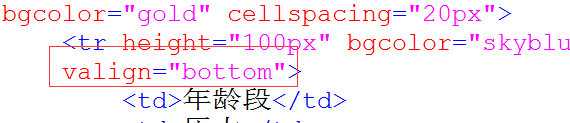
Align: 对齐方式：



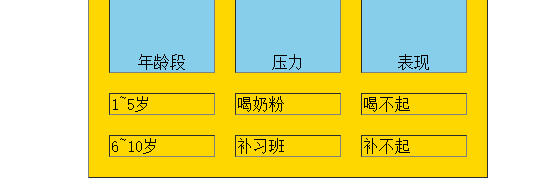
效果：



Valign: 垂直对齐方式



效果：



注意：

设置align的时，属性值：left center right

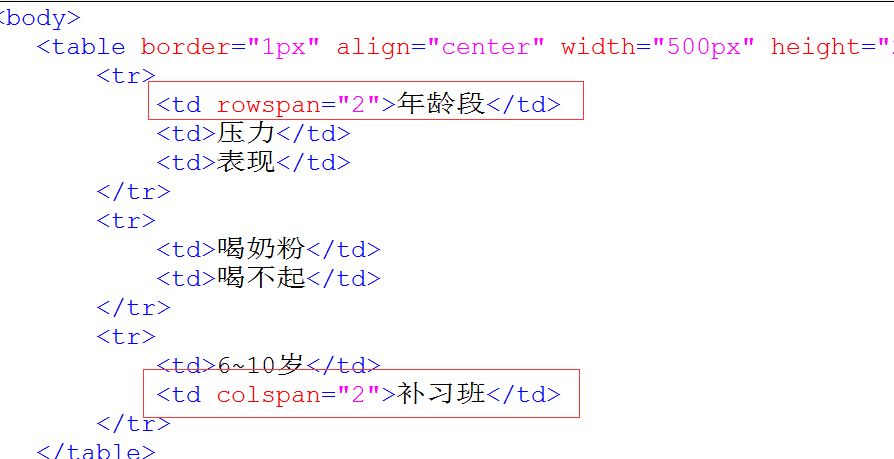
设置valign时，属性值：top middle bottom

Td/th的属性：

rowspan: 合并行

colspan: 合并列

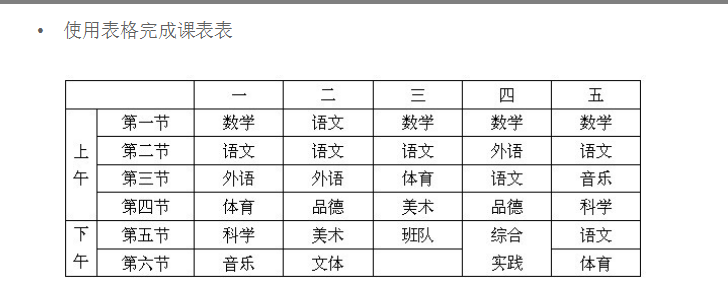
用法：



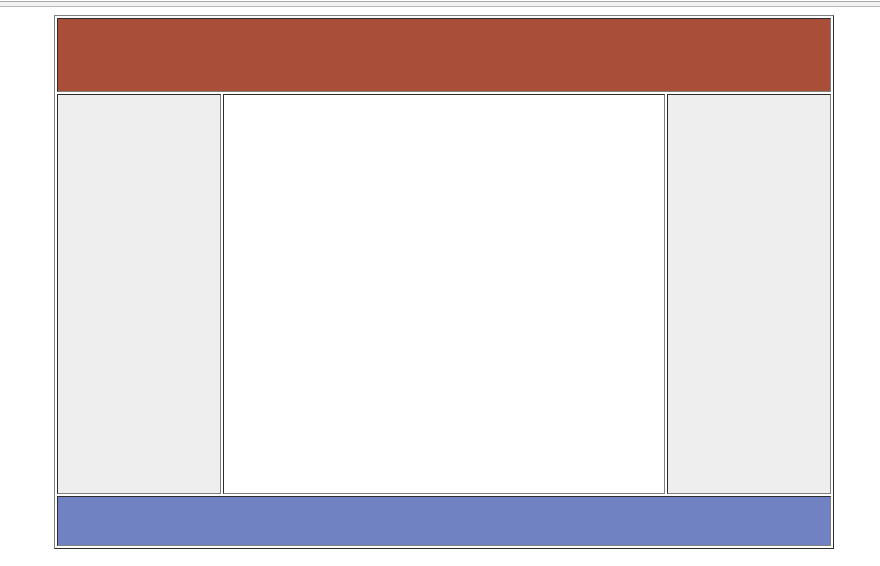
效果：



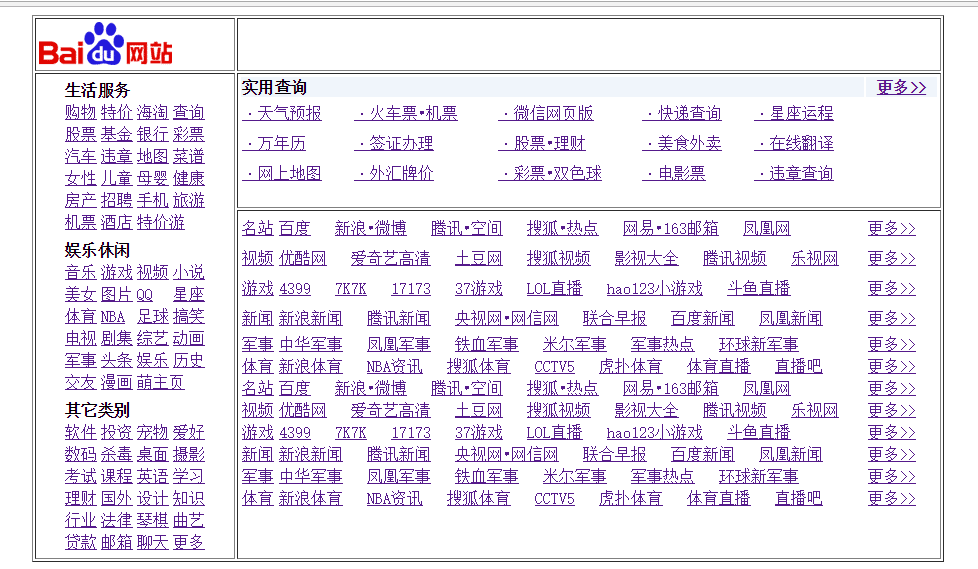
## 4，使用表格完成课程表



## 5，使用表格完成圣杯布局



## 6，使用表格完成布局 （老版的百度）



# 表单

## 1，表单赏析



## 2，了解表单

功能：用于搜集不同类型的用户输入的内容。

## 3，常用的表单元素

* <form>
* <input>
* <select>和<option>
* <textarea></textarea>

1. **form元素**

功能：用于创建一个表单。

格式：<form></form>

重要属性：

action: 表示当前表单需要提交给哪个页面进行处理。

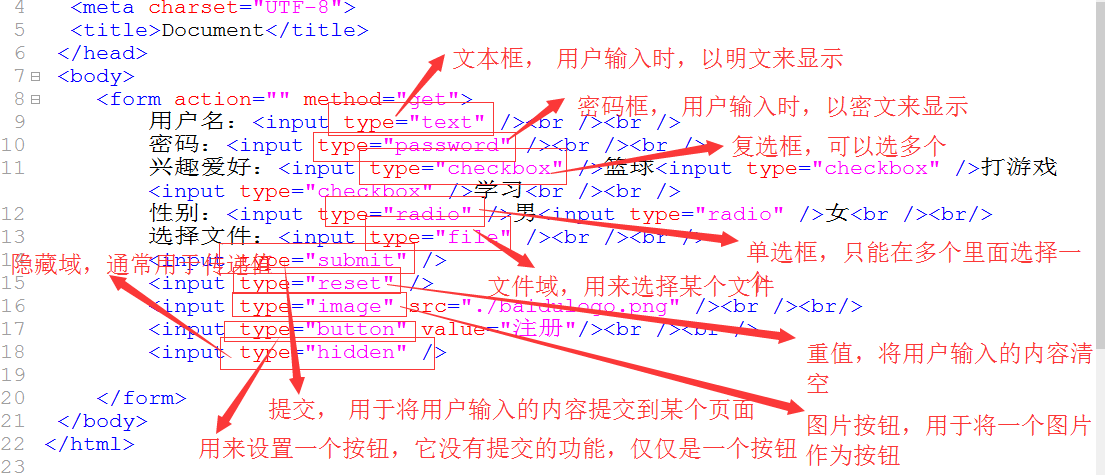
method: 表示当前表单提交的方式，常见的方式有get和post, 默认是get。

1. **Input**

表示输入框，有多种形式，可以根据type属性来确定显示哪种输入框。



基本用法：



效果 ：



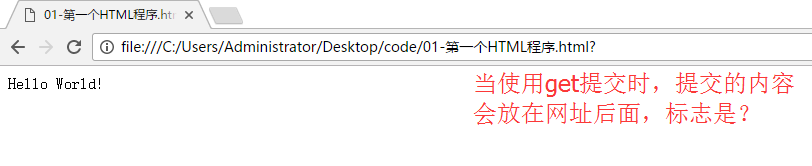
**一些细节：**

对于input来说，它的默认值就是type=”text”, 表示单行文本

对于input,最好给它添加一个name属性，作用是用于将用户填写的内容添加给要处理的页面。

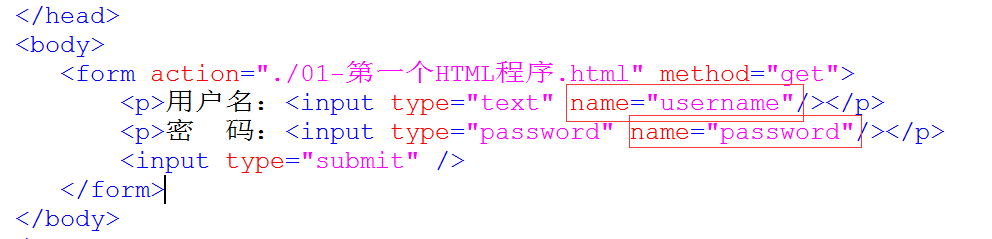
**GET和POST区别：**

**GET:**



提交的内容没有放到？后面，原因是input里面没有name属性。

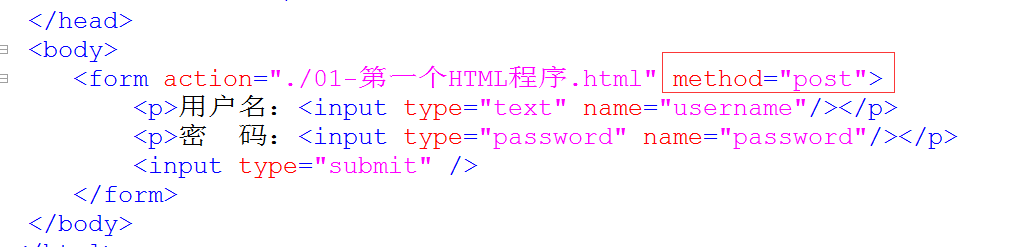
当加上Name属性时：



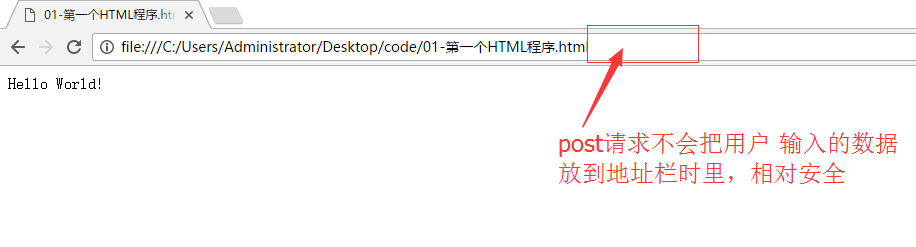
效果 ：



POST:



效果 ：



GET和POST请求小结：

1，GET会将用户输入的内容放到地址栏时面， 使用GET请求不安全。

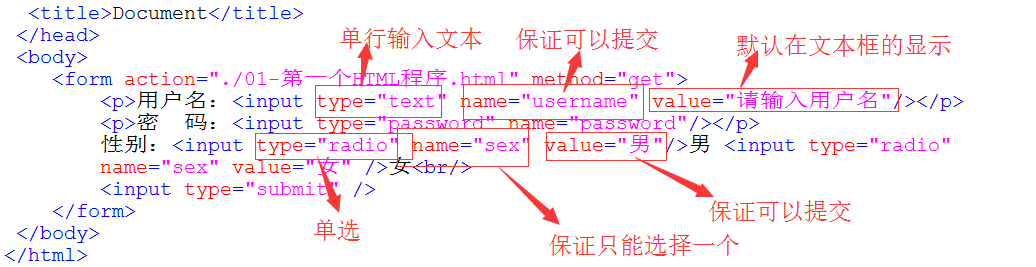
2，PoST不会将用户输入的内容放到地址栏里面，相对安全。

细节：

1，每一个Input标签都有一个value属性，不同的类型，value的作用不同

对于type=”text”来说，value可有可无，因为用户输入的内容都可以提交过去。

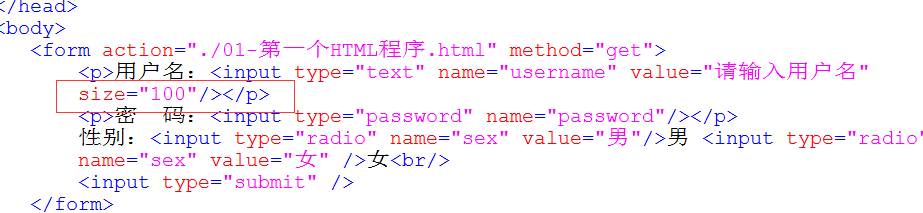
对于type=”radio” 或type=”checkbox”来说，没有value，是提交不过去的。



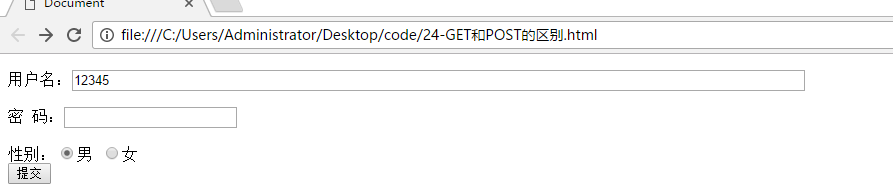
2，对于type=”radio”来说，如果你想让只能选择一个，你需要给它们添加一个name属性，如果不添加name属性，对于单选框来说，就没有意义。



3，关于单行文本的宽度，可以使用size属性来设置，表示字符长度，实际上不用，我们可以通过CSS来控制。



效果：



1. **select和option**

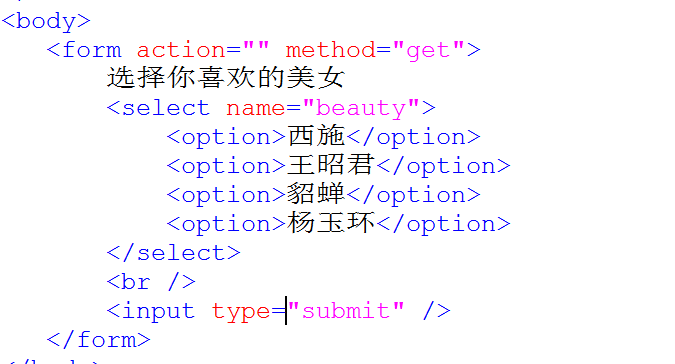
作用:用来实现下拉列表

格式：<select>

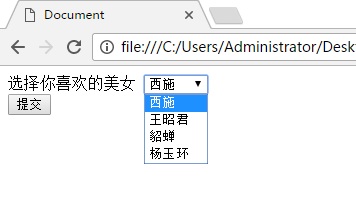
<option>.....</option>

</select>

基本使用：

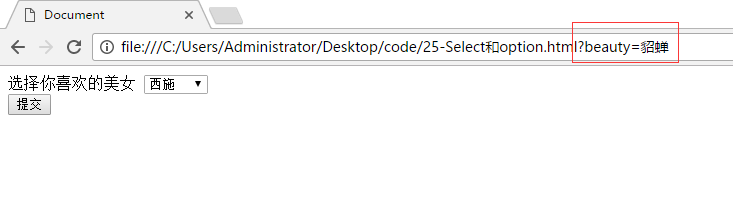


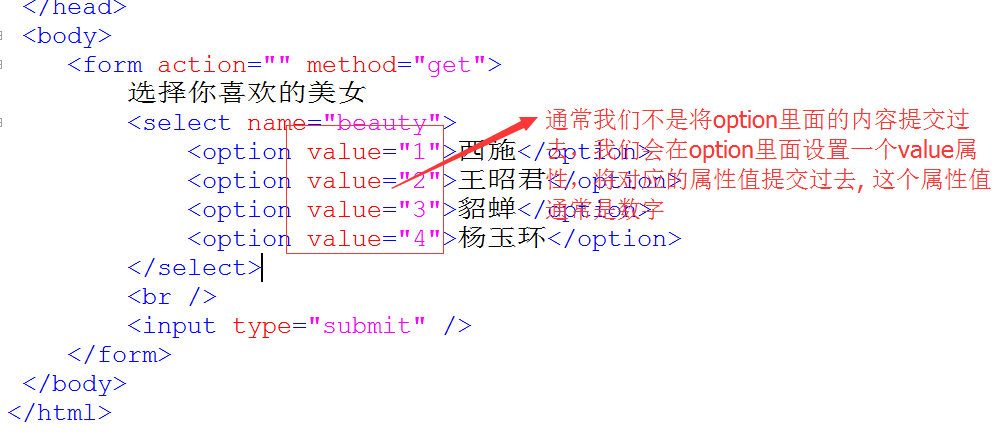
效果 ：



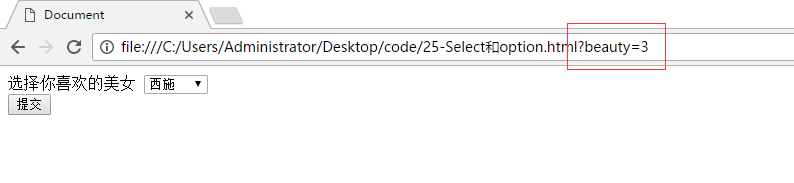
注意点：

1，默认情况下，它将option里面的内容作用提交的内容，但通常不这样做，我们需要给它设置value属性。

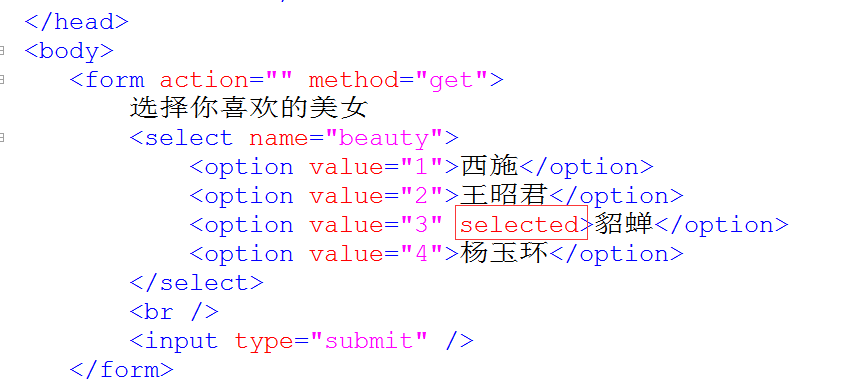




效果：



我们可以使用selected属性让某个option处于默认的选中项。



效果 ：

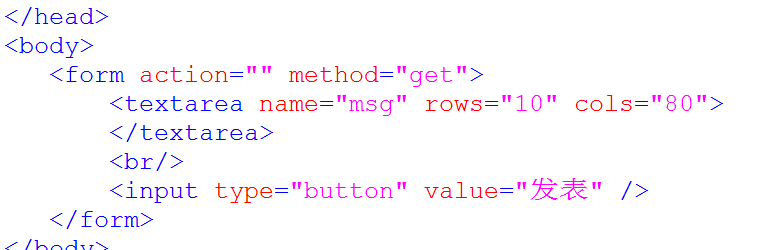


1. **textarea**

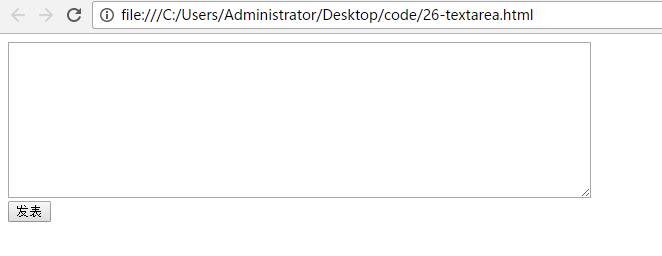
作用：提供输入大量文本的输入框，比如，论坛里发表贴子，QQ空间里发表说说

格式：<textarea></textarea>

基本使用：



效果 ：

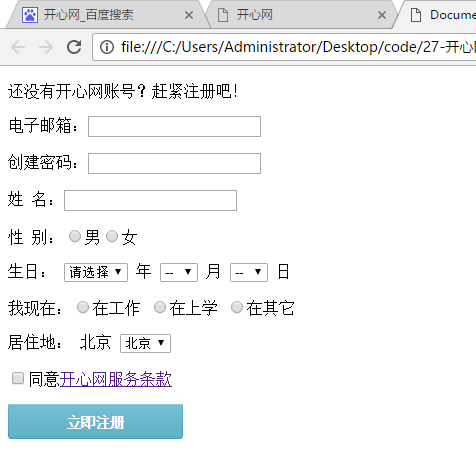


## 4，完成表单页面制作（开心网）



代码见27-表单案例

效果：



到此，表单内容没完。

# 框架

## 1，iframe标签及其属性

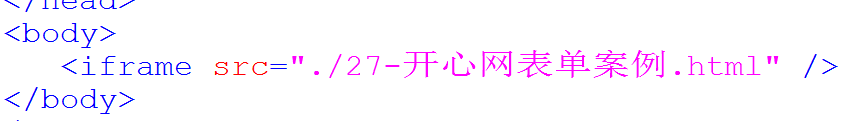
<iframe> 浮动框架， 或内嵌框架

使用比较简单

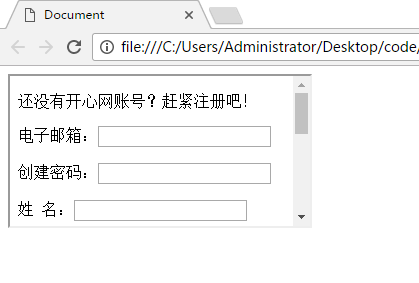
作用：嵌入第三方的内容

格式：<iframe src=”” />

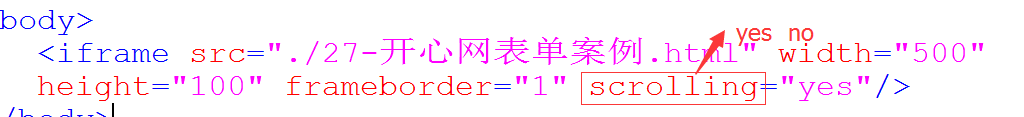
基本使用：



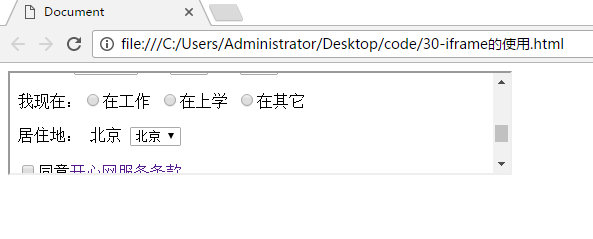
效果：



常见属性：



效果：

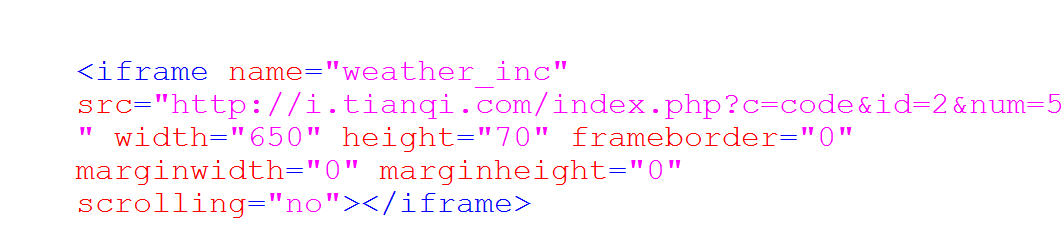


## 4，iframe应用--仿hao123中嵌入天气预报及优酷视频

天气预报（http://www.tianqi.com/dingzhi/）

视频播放（http://player.youku.com/embed/XNTcyMzIwMTE2）

1. **天气预报**



效果：



1. **优酷视频**

只需要在页面中，使用iframe，然后将src设置为视频的地址就可以了

使用：



效果：



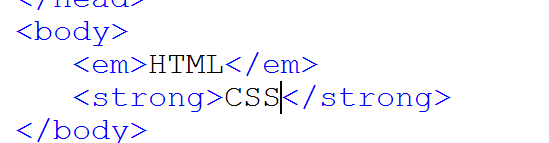
# 其它标签

## 1，em和strong

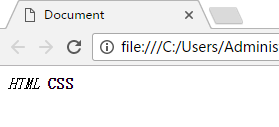
作用：用来强调某一个内容

格式：<em>内容</em> <strong>内容</strong>

基本用法：



效果：



说明：

尽管em在呈现上是斜体，但实际上是用来强调的。有利于语义化。

尽管strong在呈现上是粗体，但实际上也是用来强调的。也是有利于语义化。

它们的区别是：strong的语气更重一点。

# HTML5进阶

HTML 5 是一个新的网络标准，目标在于取代现有的 HTML 4.01, XHTML 1.0 and DOM Level 2 HTML 标准。它希望能够减少浏览器对于需要插件的丰富性网络应用服务（plug-in-based rich internet application，RIA)，如Adobe Flash, Microsoft Silverlight, 与 Sun JavaFX 的需求。

HTML 5 提供了一些新的元素和属性，反映典型的现代用法网站。其中有些是技术上类似 <div>和 <span> 标签，但有一定含义，例如 <nav>（网站导航块）和 <footer>。这种标签将有利于搜索引擎的索引整理、小屏幕装置和视障人士使用。同时为其他浏览要素提供了新的功能，通过一个标准接口，如 <audio> 和 <video> 标记。

一些过时的 HTML 4 标记将取消，其中包括纯粹用作显示效果的标记，如 <font> 和 <center>，因为它们已经被 CSS 取代。还有一些透过 DOM 的网络行为（via）。

## 下面我们来看一下HTML 5提供的一些新的标签用法以及和HTML 4的区别。

**Footer（底部）**

<footer> 标签定义 section 或 document 的页脚。典型地，它会包含创作者的姓名、文档的创作日期以及/或者联系信息。  
HTML5: <footer></footer>  
HTML4: <div></div>

**Header（头部）**

<header> 标签定义 section 或 document 的页眉。  
HTML5: <header></header>  
HTML4: <div></div>

**Nav（导航）**

<nav> 标签定义导航链接的部分。  
HTML5: <nav></nav>  
HTML4:<ul></ul>

**Section**

<section> 标签定义文档中的节（section、区段）。比如章节、页眉、页脚或文档中的其他部分。  
HTML5: <section></section>  
HTML4: <div></div>

**Time（日期和时间）**

<time> 标签定义日期或时间，或者两者。  
HTML5: <time></time>  
HTML4: <span></span>

**Video（视频）**

<video> 标签定义视频，比如电影片段或其他视频流。  
HTML5: <video src="movie.ogg" controls="controls">您的浏览器不支持 video 标签。</video>  
HTML4:<object type="video/ogg" data="movie.ogv"><param name="src" value="movie.ogv"></object>

**Audio（声音）**

<audio> 标签定义声音，比如音乐或其他音频流。  
HTML5:<audio src="someaudio.wav">您的浏览器不支持 audio 标签。</audio>  
HTML4:<object type="application/ogg" data="someaudio.wav"><param name="src" value="someaudio.wav"></object>