# 软件的架构

- 软件分成两种架构：C/S 和 B/S

## - C/S

- 客户端/服务器

- 我们在桌面上使用的软件都属于C/S架构

- 用户通过客户端来使用软件

- 在远程服务器中来处理软件的业务逻辑

- 特点：

1.使用C/S架构的软件必须要安装

2.软件更新时服务器和客户端需要同时更新

3.C/S架构的软件不能跨平台使用

4.C/S架构的软件服务器和客户端之间的通信采用自有的协议，比较安全

## - B/S

- 浏览器/服务器

- B/S本质上也是一种C/S

- 不同的是B/S架构的软件使用浏览器作为客户端

- B/S架构的软件都是通过访问网页的形式使用的

- 比如：京东、淘宝、12306

- 特点：

1.使用B/S架构的软件不需要安装，系统中只要有浏览器即可

2.软件更新时客户端不需要更新

3.B/S架构的软件可以跨平台使用

4.B/S架构的软件服务器和浏览器之间的通信采用的是通用的HTTP协议，安全性较差

# 网页的组成

根据W3C（Word Wid Web Consortium）标准，一个网页由三部分组成：结构、表现、行为

结构：HTML用于描述网页的结构

表现：CSS用于控制页面中元素的样式

行为：JavaScript用于响应用户操作

# HTML结构

HTML（Hypertext Markup Language）超文本标记语言

* HTML使用标签的的形式来标识网页中的不同组成部分。
* 所谓超文本指的是超链接，使用超链接可以让我们从一个页面跳转到另一个页面。
* **HTML中不区分大小写**,但是尽量用小写

## 文档声明

HTML5的文档声明：<!doctype html>

文档声明，标识当前网页的版本的

编写网页时，一定要编写的文档声明

在某些浏览器中，如果不写文档声明，会导致浏览器进入到怪异模式

## 注释<!-- -->

HTML注释中的内容，不会在页面中显示，但是可以在源代码中查看

通过注释可以对代码进行解释说明

一定要养成良好的编写注释的习惯

编写注释时，要求简单明了

通过注释可以将不希望在页面中显示代码隐藏

HTML注释不能嵌套（所有的多行注释都不能嵌套）

## 根标签<html> </html>

html网页中根标签，一个页面中有且只有一个根标签，网页中的所有内容都应该写在根标签的内部

## 网页的头部<head></head>

网页的头部，head标签中的内容，不会在页面中直接显示，浏览器根据head中的内容来解析网页，搜索引擎也可以根据他们来检索网页

## 网页的标题<title></title>

网页的标题，一般会在页面的标签头部显示,搜索引擎在检索一个网页时，会主要检索title中的内容,并依据该内容，来判断网页的主要内容， title中的内容会影响到网站在搜索引擎中的排名

## 字符集

使用meta来设置页面的字符集

<meta charset=”utf-8” /> 表示网页的编码是utf-8

## 网页的主体<body></body>

网页的主体，网页中所有的可见内容，都应该写在body中

## 属性

可以在标签中来添加属性，从而设置浏览器如何显示内容

属性需要写在开始标签中或自结束标签中

属性实际上就是一个名值对结构 属性名="属性值" 属性名='属性值'

属性名和标签名之间要使用空格分开，一个标签中可以设置多个属性，多个属性之间使用空格分开

通过font标签可以来设置字体相关的效果

color 用来设置字体颜色

size 用来设置字体大小（1-7）

在HTML中所有和样式相关的标签都是不推荐使用的，因为样式需要通过CSS来设置

<h1>这是一个非常<font color="red" size="7">漂亮</font>的网页</h1>

# 进制

- 我们日常生活中使用十进制比较多，所谓的十进制指的是满10进1

- 十进制

0 1 2 3 4 5 6 7 8 9

10 11 12 ...

- 在计算机中，我们往往还会遇到二进制 八进制 十六进制

- 二进制，满2进1

- 在计算机底层保存数据时都是采用的二进制

0 1

10 11

100

- 八进制，满8进1

0 1 2 3 4 5 6 7

10 11 12 13 ... 17

20

- 十六进制，满16进1

- 由于16进制是满16进1的，但是单个的数字，最大就到9

所以在十六进制中使用了 a b c d e f 六个字母分别表示10 11 12 13 14 15

0 1 2 3 4 5 6 7 8 9 a b c d e f

10 11 12 ... 1a 1b 1c 1d 1f

20

# 乱码问题

- 乱码产生的根本原因

- 计算机底层保存一切内容时，都需要将其转换为二进制编码进行保存

- 在计算机中显示一个内容时，需要将二进制编码转换为对应的字符

- 将字符转换为二进制码的过程，我们称为编码

- 将二进制码转换为字符串的过程，我们称为解码

- 编码和解码所遵循的规则我们称为字符集

- 常见的字符集

- ASCII

- ISO-8859-1

- GB2312

- GBK

- UTF-8

- 万国码，几乎支持地球上所有的字符

- 乱码产生的原因就是编码和解码所采用的字符集不同

我们在编写一个网页时，编码是由文本编辑器进行操作，而解码是由浏览器完成

- 使用NotePad++编写一个页面时，默认使用utf-8进行编码

- 浏览器默认使用gb2312进行解码

# 常用标签

## 标题标签

在HTML一共有h1 ~ h6 六级标题 <h1> </h1>

6级标题中 h1最大 h6最小

从h1 到 h6 重要性依次降低，h1最重要，h2其次，依次递减

搜索引擎检索页面时，h1仅次于title，也会影响到页面在搜索引擎中的排名

一个页面一般只有一个h1标签

一般页面中只会使用h1 ~ h3

## 段落标签

使用p标签来表示一个段落 <p> </p>

段落标签会独占一行，并和其他内容会有一个距离

## 换行标签

在HTML中，默认将多个空格和换行认为是一个空格，使用br标签来表示一个换行 <br />

## 水平线标签

可以在页面的指定位置输出一条水平线 <hr />

## 图片标签

使用图片标签可以向当前页面中引入一个外部图片

使用img标签来引入图片，img是一个自结束标签

属性：

src：要引入图片的路径，目前我们使用相对路径

alt：表示对图片的描述，该描述会在图片无法显示时出现

搜索引擎在检索图片时，会根据alt来判断图片的内容

一般创建一个img时都需要指定alt，如果不指定，则搜索引擎不会对图片进行收录

width: 可以设置图片的宽度

height：可以设置图片的高度

- 宽度和高度，如果只修改一个，则剩下会等比例缩放

# 相对路径

- 相对路径指外部文件相对于当前资源所在目录的位置

- 使用 ../ 返回一级目录,返回几级使用几个../

- 使用 文件夹名/ 进入文件夹

# 实体

在HTML中可以使用实体（转义字符），来表示一些特殊的字符

语法：

&实体的名字;

空格 &nbsp;

< &lt;

> &gt;

版权符号 &copy;

使用转义字符，来表示一个Unicode编码

&#x编码; 编码需要的是4位的十六进制编码

# 列表

列表类似于购物清单

HTML中一共有三种列表

## 1.无序列表

- 使用ul标签来创建一个无序列表

- 使用li向无序列表中添加列表项

- 无序列表默认使用 圆点 来作为项目符号

<ul>

<li>西红柿</li>



<li>大茄子</li>

<li>小辣椒</li>

</ul>

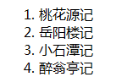
## 2.有序列表

- 有序列表和无序列表类似，不同的是它使用ol来创建，其他的都和无序列表一样

- 有序列表使用有序的序号 作为项目符号

<ol>

<li>桃花源记</li>



<li>岳阳楼记</li>

<li>小石潭记</li>

<li>醉翁亭记</li>

</ol>

## 3.定义列表

- 定义列表用来对一些内容做解释说明的

- 使用 dl 来创建一个定义列表

- 在dl中使用dt，来创建一个定义项

- 使用dd来创建一个对定义项的描述

<dl>

<dt>武松</dt>

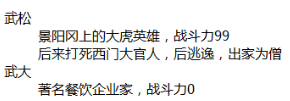
<dd>景阳冈上的大虎英雄，战斗力99</dd>

<dd>后来打死西门大官人，后逃逸，出家为僧</dd>

<dt>武大</dt>

<dd>著名餐饮企业家，战斗力0</dd>

</dl>



列表之间可以互相嵌套，可以在有序列表中放无序列表，也可以在无序列表放有序列表，定义列表也可以

<p>菜谱</p>

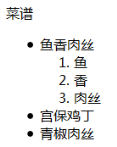
<ul>

<li>

鱼香肉丝

<ol>

<li>鱼</li>



<li>香</li>

<li>肉丝</li>

</ol>

</li>

<li>宫保鸡丁</li>

<li>青椒肉丝</li>

</ul>