

网站的建站流程



页面图例





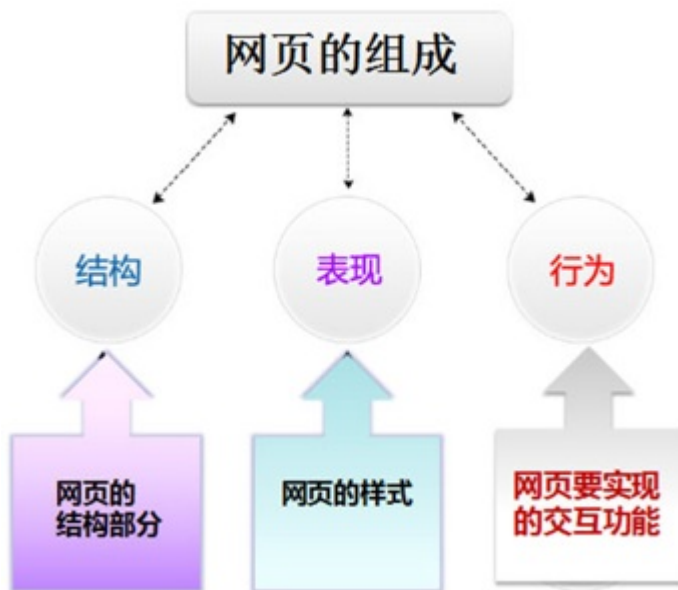
- 为什么《北京遇上西雅图》大爆，制作升级的《吹哨人》票房却扑街？



- 2019年内地电影票房突破600亿元 国产片表现抢眼
- 黄子韬表白周杰伦:好帅哥哥 温馨!朱丹高强度工作后获女儿按摩
 - 琼瑶御用男主秦汉近照曝光风度不减 周震南把粉丝名也改了...

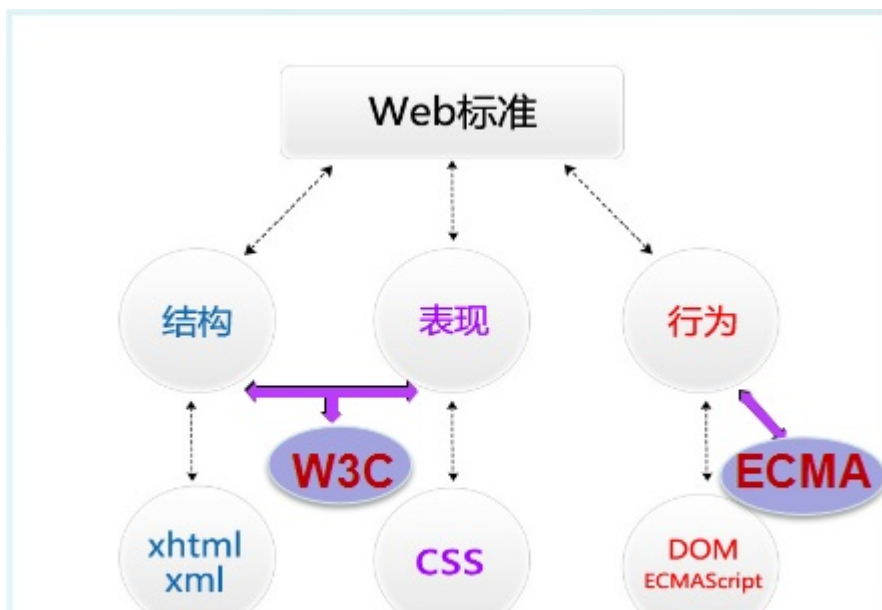


网页的结构



WEB标准

1. WEB标准是网页制作的标准，它不是一个标准，它是根据网页的不同组成部分生成的一系列标准。这些标准大部分由W3C起草发布，也有部分标准由ECMA起草发布



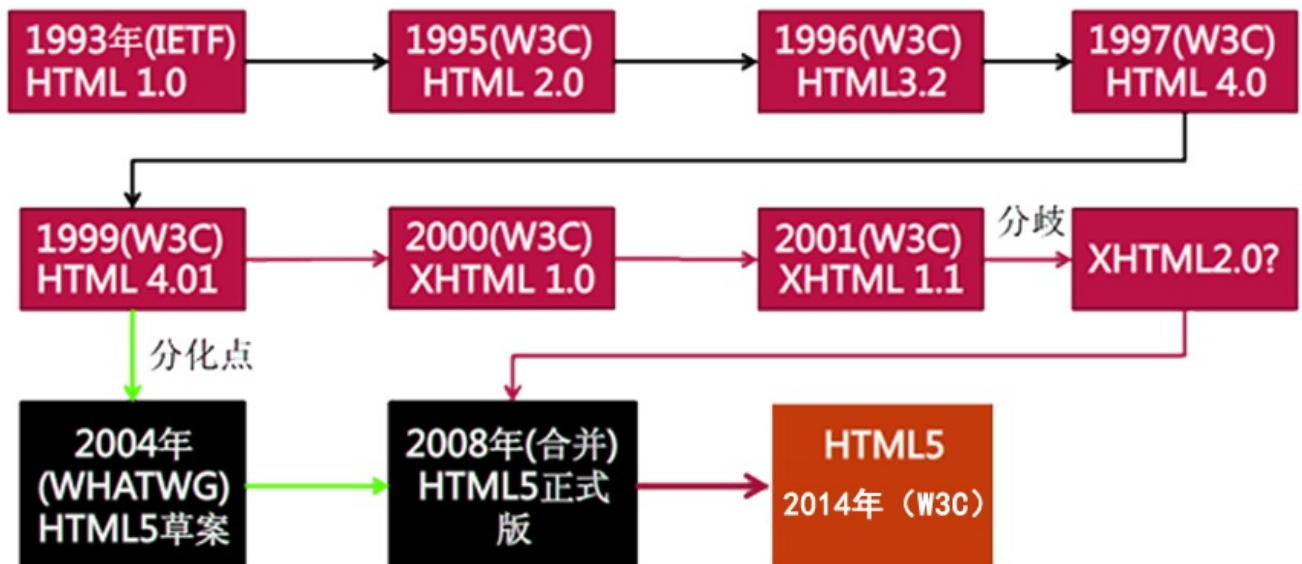
1. (1) W3C(World Wide Web Consortium)万维网联盟，创建于1994年是Web技术领域最具权威和影响力的国际中立性技术标准机构；是专门负责网络标准制定的非赢利组织。制定了结构标准和样式标准；
2. (2) ECMA：欧洲电脑网商联合会（厂商协会），制定了行为标准；

计算机语言

HTML

1. HTML 指的是超文本标记语言（Hyper Text Markup Language）www万维网的描述性语言。
- 2.
3. XHTML指可扩展超文本标记语言（标识语言）（EXtensible HyperText Markup Language）是一种置标语言，表现方式与超文本标记语言（HTML）类似，不过语法上更加严格。
- 4.
5. HTML5指的是HTML的第五次重大修改（第5个版本）

HTML发展



编辑器



建立站点

1. 规划网站的所有内容和代码
2. 整合资源

文件的命名规范

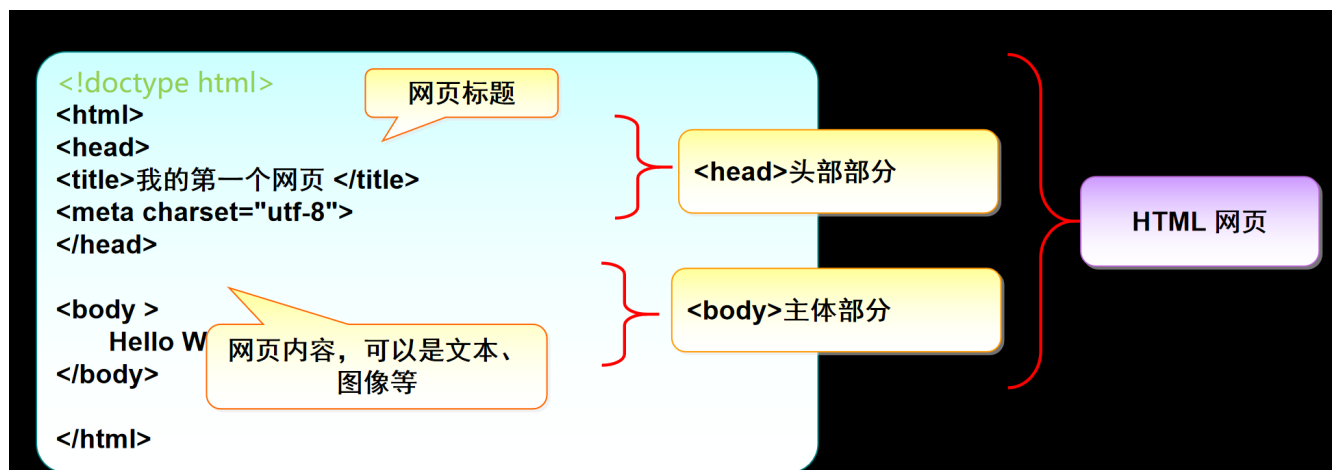
- 小写英文字母、数字、下划线的组合，
- 其中不得包含汉字、空格和特殊字符；
- 必须以英文字母开头。

HTML开始

1: HTML架构

1. `<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"`
2. `"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">`
- 3.
4. 有三种：Strict（严格型）、
5. Transitional（过渡型）、
6. Frameset（框架型）

2: HTML5基本结构



3:HTML语言

- HTML语言组成

- 标签

- 1. 写在尖角号<>里的第一个单词，叫做标记，也叫做标签，也称作元素；

- 属性

- 1. 标记和属性用空格隔开，属性和属性值用等号连接，属性值必须放在双引号内
 - 2. 一个标记可以有多个属性，属性和属性之间用空格隔开，属性不分先后顺序

- HTML语法

- 常规标记(双标记):

- <标记名称 属性1名=" 属性1值" 属性2名=" 属性2值" ></标记名>

- 空标记(单标记): <标记名 属性1名=" 属性1值" />

常用标签

1： 文本标题标签

1. 文本标题共有6个（h1-h6）
2. <h1>一级标题</h1>(唯一性, 放网站LOGO)
3. <h2>二级标题</h2>
4. ...
5. <h6>六级标题</h6>

2:字体倾斜&加粗标记

1. 文本倾斜:
2. <i></i>
3.
- 4.
5. 文本加粗:
6.
7.

3. 下划线

3.1 下划线

1. `<u></u>`

4：换行&水平线

1. `
`
2. `<hr>`

5：上标&下标

1. ``
2. ``

6：段落标记

1. `<p></p>`

7：字符（小段文本）

1. ``

8：常用转义字符

1. ` `；不换行空格
2. `>`；>右尖括号
3. `<`；<左尖括号
4. `©`；备案中图标版权 ©

8：列表

（1）无序列表

1. ``
2. `列表项内容`
3. `列表项内容`
4. `列表项内容`
5.
6. ``

（2）有序列表

1. ``

2. `列表项内容`
 3. `列表项内容`
 4. `列表项内容`
 5.
 6. ``
 1. type:规定列表中的列表项目的项目符号的类型
 2. 语法: `<ol type= " a">`
 3. l 数字顺序的有序列表(默认值)(1, 2, 3, 4)。
 4. a 字母顺序的有序列表, 小写(a, b, c, d)。
 5. A 字母顺序的有序列表, 大写(A, B, C, D)
 6. i 罗马数字, 小写(i, ii, iii, iv)。
 7. I 罗马数字, 大写(i, ii, iii, iv)。
 - 8.
 9. start 属性规定有序列表的开始点。(start的属性值必须是数字)
 10. 语法: `<ol start="5">`
- (3) 自定义列表

1. `<dl>`
2. `<dt></dt>`
3. `<dd></dd>`
4. `</dl>`

9 : 超链接

1. `<a>`
2. 属性:
3. `href = 'url'`
4. `target = "_black / _self";`
5. `title = '文本提示'`
- 6.
7. 拓展:
8. `rel = 'nofollow';`

10 : 图片

1. ``
2. 属性:
3. `src = 'url';`
4. `alt = ' 标签 实例 带有指定替代文本的图像'`
5. `title = '文本提示'`
6. `width = ''`

```
7. height = ''
8. border = ''
```

图片 title 和 alt区别:

1. alt:
2. 1、alt属性是考虑到不支持图像显示或者图像显示被关闭的浏览器的用户，以及视觉障碍的用户和使用屏幕阅读器的用户。当图片不显示的时候，图片的替换文字。
3. 2、alt属性值得长度必须少于100个英文字符
4. 3、alt属性是img标签的必须属性，如果没有特别意义的图片，可以写alt=""
5. 4、alt属性是搜索引擎判断图片与文字是否相关的重要依据，alt属性添加到img主要的目的是为了SEO
- 6.
7. title:
8. 1、title属性并不是必须的。
9. 2、title属性规定元素的额外信息，有视觉效果，当鼠标放到文字或是图片上时有文字显示。
10. 3、title属性并不作为搜索引擎抓取图片的参考，更多倾向于用户体验的考虑。

11 : 相对路径

1. (同级)
2. 1)当当前文件与目标文件在同一目录下，直接书写目标文件的文件名+扩展名;
3. (上级找下级)
4. 2)当当前文件与目标文件所处的文件夹在同一目录下，写法如下:
5. 文件夹名/目标文件全称+扩展名;
6. (下级找上级)
7. 3)当当前文件所处的文件夹和目标文件在同一目录下，写法如下:
8. ../目标文件文件名+扩展名;

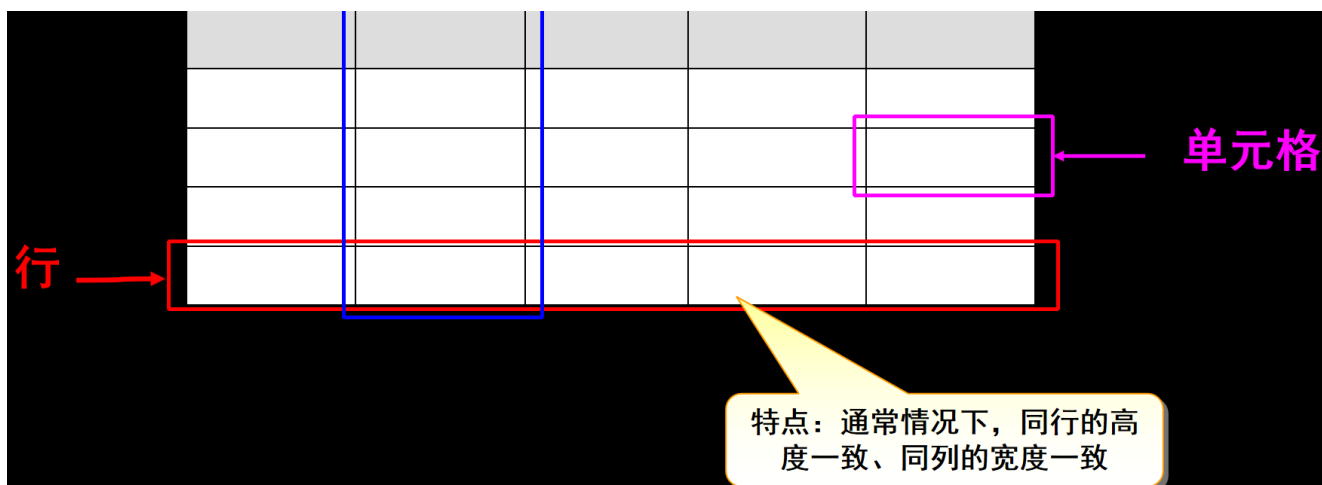
12 : DIV

13 : HTML注释

1. <!-- 注释 -->

表格





1：表格基本结构

1. `<table>`
2. `<tr>`
3. `<td></td>`
4. `<td></td>`
5. `</tr>`
6. `</table>`
- 7.
8. `<!--`
9. `table` 为表格
10. `tr` 行
11. `td` 列（每一个单元格）
12. `-->`

2：数据行分组

1. `<thead></thead>`
2. `<tbody></tbody>`
3. `<tfoot></tfoot>`

3：数据列分组

1. `<colgroup span="value"></colgroup>`
2. `<!--span属性为把几列分为一组-->`

3：列标题

1. `<th></th>`

.....

4：表格标题

1. `<caption></caption>`

5：表格的html属性

1. 1) width="表格的宽度"
2. 2) height="表格的高度"
3. 3) border="表格的边框"
4. 4) bordercolor="边框色"
5. 5) cellpadding="单元格与单元格之间的间距"
6. 6) cellspacing="单元格与内容之间的距离"
7. 7) align="表格水平对齐方式"
8. 取值：left、right、center、
9. valign="垂直对齐" top\bottom\middle
10. 8) 合并单元格属性：(td)
11. 合并列： colspan="所要合并的单元格的列数"
12. 合并行： rowspan="所要合并单元格的行数"

表单

1：表单标签

1. `<form></form>`
2. 属性：
3. action = '接口地址'
4. method = 'get / post'
5. name = '表单名称'

2：表单控件

1. `<input>`
2. 属性：
3. type = '控件类型'
4. name: 属性标识表单域的名称;
5. Value: 属性定义表单域的默认值，其他属性根据type的不同而有所变化。
6. maxlength: 控制最多输入的字符数，
7. Size: 控制框的宽度（以字符为单位）

1) 文本框

2. `<input type="text" value="默认值"/>`

2) 密码框

4. `<input type="password" value="默认值"/>`

```
4. <input type= password />
5. 3)提交按钮
6. <input type="submit" value="按钮内容" />
7. 4)重置按钮
8. <input type="reset" value="按钮内容" />
9. 5)空按钮
10. <input type="button" value="按钮内容" />
11. 6) 单选按钮组
12. <input type= "radio" name= "ral" />男
13. <input type= "radio" name= "ral"
14. checked= "checked" />(默认选中)女
15. 7) 复选框组
16. <input type="checkbox" name="" />
17. <input type="checkbox" name="" disabled="disabled" />
18. * disabled="disabled" (禁用)
19. * checked="checked" (默认选中)
20. 8) 下拉列表 (菜单) :
21. <select >
22. <option>下拉选项1</option>
23. <option>下拉选项2</option>
24. .....
25. </select>
26. 表示下拉列表, name属性不是必须的
27. 默认选择项用selected属性;
28. 9) 表单域多行文本定义:
29. 语法: <textarea name="" cols="" rows="" ></textarea>
30. 多行文本。rows属性和cols属性用来设置文本输入窗口的高度和宽度, 单位是字符。
31. 阻止浏览器对窗口的拖动设置:{resize:none;} (css属性)
32. 10) 上传文件:
33. 语法: <input type="file">
```

3 : 表单标签

```
1. 1) 表单字段集
2. 语法: <fieldset></fieldset>
3.
4. 说明: 相当于一个方框, 在字段集中可以包含文本和其他元素。该元素用于对表单中的元素进行分组
并在文档中区别标出文本。fieldset元素可以嵌套, 在其内部可以在设置多个fieldset对
象。disabled定义空间禁制可用;
5.
6. 2) 字段级标题:
7. 语法: <legend align="left/center/right/justify"></legend>
```

7. 语法: `<legend align="left/center/right/justify"></legend>`
8. 说明: legend元素可以在fieldset对象绘制的方框内插入一个标题。legend元素必须是fieldset内的唯一元素。
9. 3) 提示信息标签:
10. 语法: `<label for="绑定控件id名"></label>`
- 11.
12. 说明: label元素用来定义标签, 为页面上的其他元素指定提示信息。要将label元素绑定到其他的控件上, 可以将label元素的for属性设置为与该控件的id属性值相同。

拓展: post/get

- (1). 从功能上讲, GET一般用来从服务器上获取资源, POST一般用来更新服务器上的资源;
- (2). 从REST服务角度上说, GET是幂等的, 即读取同一个资源, 总是得到相同的数据, 而POST不是幂等的, 因为每次请求对资源的改变并不是相同的; 进一步地, GET不会改变服务器上的资源, 而POST会对服务器资源进行改变;
- (3). 从请求参数形式上看, GET请求的数据会附在URL之后, 即将请求数据放置在HTTP报文的 请求头 中, 以?分割URL和传输数据, 参数之间以&相连。特别地, 如果数据是英文字母/数字, 原样发送; 否则, 会将其编码为 application/x-www-form-urlencoded MIME 字符串 (如果是空格, 转换为+, 如果是中文/其他字符, 则直接把字符串用BASE64加密, 得出如: %E4%BD%A0%E5%A5%BD, 其中%XX中的XX为该符号以16进制表示的ASCII); 而POST请求会把提交的数据则放置在是HTTP请求报文的 请求体 中。
- (4). 就安全性而言, POST的安全性要比GET的安全性高, 因为GET请求提交的数据将明文出现在URL上, 而且POST请求参数则被包装到请求体中, 相对更安全。
- (5). 从请求的大小看, GET请求的长度受限于浏览器或服务器对URL长度的限制, 允许发送的数据量比较小, 而POST请求则是没有大小限制的。