

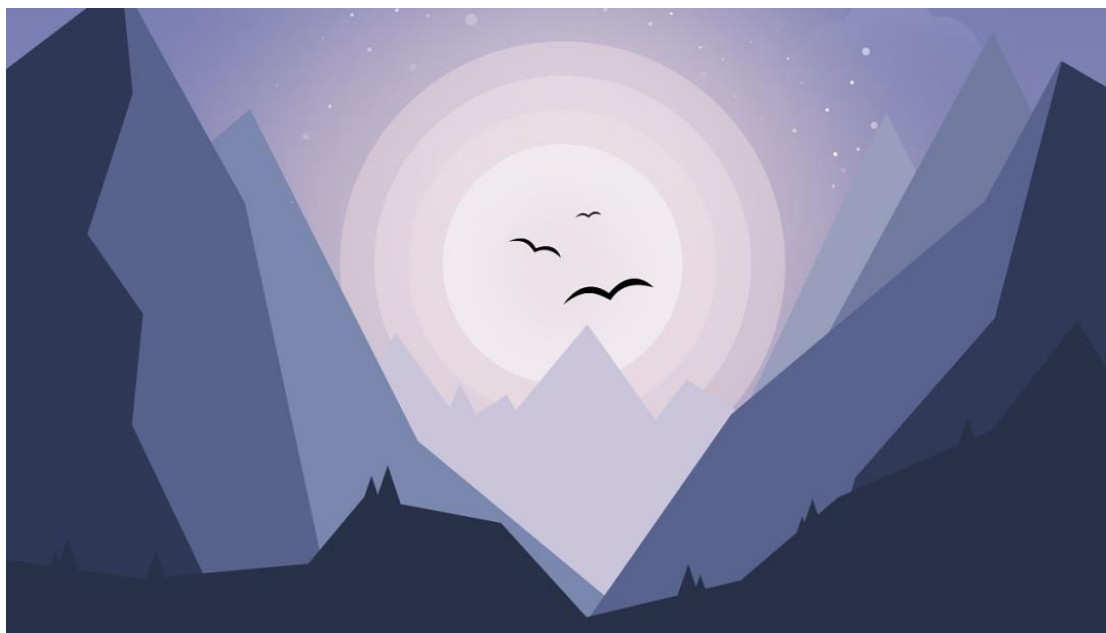
「学习笔记」HTML 基础

原创 饭老板 前端 fan

收录于话题

#前端入门

4 个



「学习笔记」HTML 基础

前言

勤做笔记不仅可以让自己学的扎实，更重要的是可以让自己少走弯路。有人说：“再次翻开笔记是什么感觉”，我的回答是：“初恋般的感觉”。或许笔记不一定十全十美，但肯定会让你有种初恋般的怦然心动。□□

本章着重复习 Html 的基础内容，学习 Html 究竟要学些什么呢？主要是学习各种标签，来搭建网页的“结构”。✧✧✧

本篇文章主要由五个章节构成，从 WEB 标准到初识 HTML，接着学习 HTML 常用标签，最后学习表格列表和表单。□□开始充电之旅啦~~~

一、认识 WEB

「网页」主要是由文字、图像和超链接等元素构成，当然除了这些元素，网页中还可以包括音频、视频以及 Flash 等。

「浏览器」是网页显示、运行的平台。

「浏览器内核」(排版引擎、解释引擎、渲染引擎)

负责读取网页内容，整理讯息，计算网页的显示方式并显示页面。

浏览器内核备注 IETridentIE、猎豹安全、360 极速浏览器、百度浏览器 firefoxGecko 可惜这几年已经没落了，打开速度慢、升级频繁、猪一样的队友 flash、神一样的对手 chrome。Safariwebkit 现在很多人错误地把 webkit 叫做 chrome 内核(即使 chrome 内核已经是 blink 了)。苹果感觉像被别人抢了媳妇，都哭晕在厕所里面了。chromeChromium/Blink 在 Chromium 项目中研发 Blink 渲染引擎(即浏览器核心)，内置于 Chrome 浏览器之中。Blink 其实是 WebKit 的分支。大部分国产浏览器最新版都采用 Blink 内核。二次开发 Operablink 现在跟随 chrome 用 blink 内核。

Web 标准

「构成」□ 结构标准，表现标准和行为标准

结构标准 用于对网页元素进行整理和分类(HTML)

表现标准 用于设置网页元素的版式、颜色、大小等外观属性(CSS)

行为标准 用于对网页模型的定义及交互的编写(JavaScript)

「Web 标准的优点」□

易于维护：只需更改 CSS 文件，就可以改变整站的样式

页面响应快：HTML 文档体积变小，响应时间短

可访问性：语义化的 HTML(结构和表现相分离的 HTML) 编写的网页文件，更容易被屏幕阅读器识别

设备兼容性：不同的样式表可以让网页在不同的设备上呈现不同的样式

搜索引擎：语义化的 HTML 能更容易被搜索引擎解析，提升排名

二、HTML 初识

HTML 初识

「HTML」 (Hyper Text Markup Language):超文本标记语言

「所谓超文本，有 2 层含义：」

因为它可以加入图片、声音、动画、多媒体等内容（**超越文本限制**）

不仅如此，它还可以从一个文件跳转到另一个文件，与世界各地主机的文件连接（**超级链接文本**）。

「HTML 骨架格式」

```
<!-- 页面中最大的标签 根标签 -->
<html>
  <!-- 头部标签 -->
  <head>
    <!-- 标题标签 -->
    <title></title>
  </head>
  <!-- 文档的主体 -->
  <body>
  </body>
</html>
```

「团队约定大小写」

HTML 标签名、类名、标签属性和大部分属性值统一用小写

「HTML 元素标签分类」

常规元素(双标签)

空元素(单标签)

常规元素(双标签)

`<标签名>` 内容 `</标签名>` 比如 `<body>我是文字</body>`

空元素(单标签)

`<标签名 />` 比如 `
` 或 `
`

「HTML 标签关系」

嵌套关系 父子级包含关系

并列关系 兄弟级并列关系

如果两个标签之间的关系是**嵌套关系**，子元素最好缩进一个 tab 键的身位（一个 tab 是 4 个空格）。如果是**并列关系**，最好上下对齐。

文档类型 `<!DOCTYPE >`

「**文档类型**」用来说明你用的 XHTML 或者 HTML 是什么版本。

`<!DOCTYPE html>`告诉浏览器按照 HTML5 标准解析页面。

页面语言 lang

lang 指定该 html 标签内容所用的语言

```
<html lang="en">
```

en 定义语言为英语 zh-CN 定义语言为中文

「lang 的作用」

根据根据 lang 属性来设定不同语言的 css 样式，或者字体

告诉搜索引擎做精确的识别

让语法检查程序做语言识别

帮助翻译工具做识别

帮助网页阅读程序做识别

字符集

「**字符集**」(Character set)是多个字符的集合,计算机要准确的处理

各种字符集文字，需要进行字符编码，以便计算机能够识别和存储各种文字。

UTF-8 是目前最常用的字符集编码方式

让 html 文件是以 UTF-8 编码保存的，浏览器根据编码去解码对应的 html 内容。

```
<meta charset="UTF-8" />
```

「meta viewport 的用法」

通常 viewport 是指视窗、视口。浏览器上(也可能是一个 app 中的 webview)用来显示网页的那部分区域。在移动端和 pc 端视口是不同的，pc 端的视口是浏览器窗口区域，而在移动端有三个不同的视口概念：布局视口、视觉视口、理想视口

meta 有两个属性 name 和 http-equiv

name 属性的取值

keywords(关键字) 告诉搜索引擎，该网页的关键字

description(网站内容描述) 用于告诉搜索引擎，你网站的主要内容。

viewport(移动端的窗口)

robots(定义搜索引擎爬虫的索引方式) robots 用来告诉爬虫哪些页面需要索引，哪些页面不需要索引

author(作者)

generator(网页制作软件)

copyright(版权)

http-equiv 有以下参数

http-equiv 相当于 http 的文件头作用，它可以向浏览器传回一些有用的信息，以帮助正确和精确地显示网页内容

content-Type 设定网页字符集(Html4 用法，不推荐)

Expires(期限),可以用于设定网页的到期时间。一旦网页过期，必须到服务器上重新传输。

Pragma(cache 模式),是用于设定禁止浏览器从本地机的缓存中调阅页面内容，设定后一旦离开网页就无法从 Cache 中再调出

Refresh(刷新),自动刷新并指向新页面。

cache-control (请求和响应遵循的缓存机制)

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

HTML 标签的语义化

方便代码的阅读和维护，样式丢失的时候能让页面呈现清晰的结构。

有利于 SEO，搜索引擎根据标签来确定上下文和各个关键字的权重。

方便其他设备解析，如盲人阅读器根据语义渲染网页

「拓展」 标签：规定页面上所有链接的默认 URL 和设置整体链接的打开状态

```
<head>
  <base href="http://www.baidu.com" target="_blank">
  <base target="_self">
</head>
<body>
  <a href="">测试 </a> 跳转到 百度
</body>
```

HTML 常用标签

常用标签

「1. 排版标签」 主要和 css 搭配使用，显示网页结构的标签，是网页布局最常用的标签。

标题标签 h(h1~h6)

段落标签 p,可以把 HTML 文档分割为若干段落

水平线标签 hr

换行标签 br

div 和 span 标签:是没有语义的,是我们网页布局最主要的 2 个盒子。

「2. 排版标签」

b 和 strong 文字以粗体显示

i 和 em 文字以斜体显示

s 和 del 文字以加删除线显示

u 和 ins 文字以加下划线显示

「3. 标签属性(行内式)」

使用 HTML 制作网页时,如果能让 HTML 标签提供更多的信息,可以使用 HTML 标签的属性加以设置。

```
<标签名 属性 1="属性值 1" 属性 2="属性值 2" ...> 内容 </标签名 >  
<手机 颜色="红色" 大小="5 寸"> </手机 >
```

「4. 图像标签 img」

标记属性

属性	属性值	描述
src	URL	图像的路径
alt	文本	图像不能显示时的替换文本
title	文本	鼠标悬停时显示的内容
width	像素 (XHTML不支持%页面百分比)	设置图像的宽度
height	像素 (XHTML不支持%页面百分比)	设置图像的高度
border	数字	设置图像边框的宽度

注意：

标签可以拥有多个属性,必须写在开始标签中,位于标签名后面。属性之间不分先后顺序,标签名与属性、属性与属性之间均以空格分开。

采取 键值对 的格式 key="value" 的格式

```

```

「5. 链接标签(重点)」

`文本或图像 `

`target="_self"` 默认窗口弹出方式

`target="_blank"` 新窗口弹出

属性作用 `href` 用于指定链接目标的 url 地址,(必须属性) 当为标签应用 `href` 属性时, 它就具有了超链接的功能 `target` 用于指定链接页面的打开方式, 其取值有 `_self` 和 `_blank` 两种, 其中 `_self` 为默认值, `_blank` 为在新窗口中打开方式。

src 和 href 的区别

一句话概括:`src` 是引入资源的 `href` 是跳转 url 的

`src` 用于替换当前元素, `href` 用于在当前文档和引用资源之间确立联系。

`src` 是 `source` 的缩写, 指向外部资源的位置, 指向的内容将会嵌入到文档中当前标签所在位置; 在请求 `src` 资源时会将其指向的资源下载并应用到文档内, 例如 `js` 脚本, `img` 图片和 `frame` 等元素。当浏览器解析到该元素时, 会暂停其他资源的下载和处理, 直到将该资源加载、编译、执行完毕, 图片和框架等元素也如此, 类似于将所指向资源嵌入当前标签内。这也是为什么将 `js` 脚本放在底部而不是头部。

`href` 是 `Hypertext Reference` 的缩写, 指向网络资源所在位置, 建立和当前元素(锚点) 或当前文档(链接) 之间的链接。如果我们在文档中添加那么浏览器会识别该文档为 `css` 文件, 就会并行下载资源并且不会停止对当前文档的处理。这也是为什么建议使用 `link` 方式来加载 `css`, 而不是使用 `@import` 方式。

注意 :

外部链接 需要添加 `http:// www.baidu.com`

内部链接 直接链接内部页面名称即可 比如 `< a href="index.html"> 首页`

如果当时没有确定链接目标时, 通常将链接标签的 `href` 属性值定义为 `"#"` (即 `href="#"`), 表示该链接暂时为一个空链接。

不仅可以创建文本超链接, 在网页中各种网页元素, 如图像、表格、音频、视频等都可以添加超链接。

锚点定位：通过创建锚点链接，用户能够快速定位到目标内容。

1. 使用相应的 id 名标注跳转目标的位置。（找目标）

```
<h3 id="two">第 2 集</h3>
```

2. 使用 `链接文本` 创建链接文本（被点击的）

```
<a href="#two">
```

「6. 注释标签」


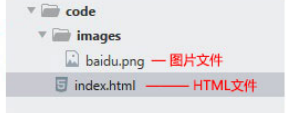
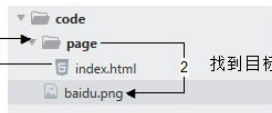
`<!-- 注释语句 -->`

快捷键是： ctrl + /

或者 ctrl + shift + /

团队约定：注释内容前后各一个空格字符，注释位于要注释代码的上面，单独占一行

「7. 路径」

路径分类	符号	说明
同一级路径		只需输入图像文件的名称即可，如 <code></code> 。 
下一级路径	"/"	图像文件位于HTML文件同级文件夹下（例如文件夹名称为：images） 如 <code></code> 。 
上一级路径	"../"	在文件名之前加入"../"，如果是上两级，则需要使用"../.."，以此类推， 如 <code></code> 。 1. 通过“../”返回到html文件的所在的上一级文件（也就是返回到和图片文件所在的同级目录） 2. 找到目标图片 

「8. 其他知识」

预格式化文本 pre 标签元素中的文本通常会保留空格和换行符。而文本也会呈现为等宽字体。格式化文本就是，按照我们预先写好的文字格式来显示页面，保留空格和换行等。

特殊字符

HTML为这些特殊字符准备了专门的替代代码

特殊字符	描述	字符的代码
	空格符	
<	小于号	<
>	大于号	>
&	和号	&
¥	人民币	¥
©	版权	©
®	注册商标	®
°	摄氏度	°
±	正负号	±
×	乘号	×
÷	除号	÷
²	平方2 (上标2)	²
³	立方3 (上标3)	³

什么是 XHTML

XHTML 指「可扩展超文本标签语言」(EXtensible HyperText Markup Language)。

XHTML 的目标是取代 HTML。

XHTML 与 HTML 4.01 几乎是相同的。

XHTML 是更严格更纯净的 HTML 版本。

XHTML 是作为一种 XML 应用被重新定义的 HTML,是严格版本的 HTML。例如它要求标签必须小写, 标签必须被正确关闭, 标签顺序必须正确排列, 对于属性都必须使用双引号等。

XHTML 是一个 W3C 标准。

写 HTML 代码时应注意什么 ?

尽可能少的使用无语义的标签 div 和 span ;

在语义不明显时, 既可以使用 div 或者 p 时, 尽量用 p, 因为 p 在默认情况下有上下间距, 对兼容特殊终端有利 ;

不要使用纯样式标签, 如 : b、font、u 等, 改用 css 设置。

需要强调的文本, 可以包含在 strong 或者 em 标签中 (浏览器预设样式, 能用 CSS 指定就不用他们), strong 默认样式是加粗 (不要用 b), em 是斜体 (不用 i);

使用表格时, 标题要用 caption, 表头用 thead, 主体部分用 tbody 包围, 尾部用 tfoot 包围。表头和一般单元格要区分开, 表头用 th, 单元格用 td ;

表单域要用 `fieldset` 标签包起来，并用 `legend` 标签说明表单的用途；

每个 `input` 标签对应的说明文本都需要使用 `label` 标签，并且通过为 `input` 设置 `id` 属性，在 `label` 标签中设置 `for` 来让说明文本和相对应的 `input` 关联起来。

表格

「1. 表格」

现在还是较为常用的一种标签，但不是用来布局，常见显示、展示表格式数据。因为它可以让数据显示的非常的规整，可读性非常好。特别是后台展示数据的时候表格运用是否熟练就显得很重要，一个清爽简约的表格能够把繁杂的数据表现得很有条理。

「2. 创建表格」

```
<table>
  <tr>
    <td>单元格内的文字</td>
    ...
  </tr>
  ...
</table>
```

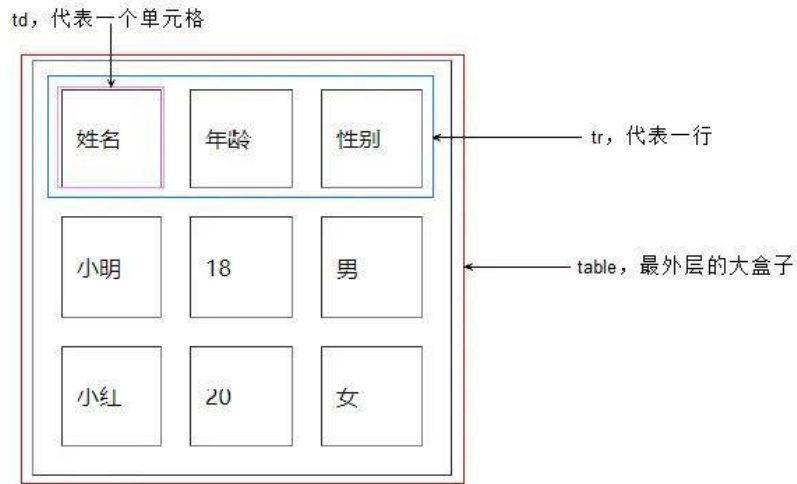
`table`、`tr`、`td`，他们是创建表格的基本标签，缺一不可

`table` 用于定义一个表格标签。

`tr` 标签 用于定义表格中的行，必须嵌套在 `table` 标签中。

`td` 用于定义表格中的单元格，必须嵌套在 `<tr></tr>` 标签中。

字母 `td` 指表格数据 (`table data`)，即数据单元格的内容，现在我们明白，表格最合适的地方就是用来存储数据的。`td` 像一个容器，可以容纳所有的元素。



表头单元格标签 th:一般表头单元格位于表格的第一行或第一列，并且文本加粗居中,只需用表头标签 `<th></th>` 替代相应的单元格标签 `<td></td>` 即可。

表头 →

姓名	性别	电话	
小王	女	11122233	海淀区
小李	男	55566677	
小张	男	88899900	

表头 →

表格标题 caption 通常这个标题会被居中且显示于表格之上。
caption 标签必须紧随 table 标签之后。这个标签只存在 表格里
面才有意义。你是风儿我是沙

```
<table>
  <caption>我是表格标题 </caption>
</table>
```

「3. 表格属性」

属性名	含义	常用属性值
border	设置表格的边框（默认border="0"无边框）	像素值
cellspacing	设置单元格与单元格边框之间的空白间距	像素值（默认为2像素）
cellpadding	设置单元格内容与单元格边框之间的空白间距	像素值（默认为1像素）
width	设置表格的宽度	像素值
height	设置表格的高度	像素值
align	设置表格在网页中的水平对齐方式	left、center、right

三参数为 0，平时开发的我们这三个参数 `border` `cellpadding` `cellspacing` 为 0

「4. 合并单元格」,合并的顺序我们按照 **先上 后下 先左 后右** 的顺序 ,合并完之后需要删除多余的单元格。

跨行合并：rowspan="合并单元格的个数"

跨列合并：colspan="合并单元格的个数"

「5. 总结表格」

标签名定义说明 `<table>` `</table>` 表格标签就是一个四方的盒子
`<tr>` `</tr>` 表格行标签行标签要再 table 标签内部才有意义
`<td>` `</td>` 单元格标签单元格标签是个容器级元素，可以放任何东西
`<th>` `</th>` 表头单元格标签它还是一个单元格，但是里面的文字会居中且加粗
`<caption>` `</caption>` 表格标题标签表格的标题，跟着表格一起走，和表格居中对齐
`colspan` 和 `rowspan` 合并属性用来合并单元格的

「6. 表格划分结构」

对于比较复杂的表格，表格的结构也就相对的复杂了，所以又将表格分割成三个部分：题头、正文和脚注。而这三部分分别用 `:thead`,`tbody`,`tfoot` 来标注，这样更好的分清表格结构。

注意：

1. `<thead>` `</thead>`：用于定义表格的头部。用来放标题之类的东

西。<thead> 内部必须拥有<tr> 标签！

2. <tbody></tbody>：用于定义表格的主体。放数据本体。

3. <tfoot></tfoot>放表格的脚注之类。

4. 以上标签都是放到 table 标签中。

列表

「列表 ul」容器里面装载着结构，样式一致的文字或图表的一种形式，叫列表。

列表最大的特点就是整齐、整洁、有序，跟表格类似，但是它可组合自由度会更高。

「1. 无序列表 ul」

中只能嵌套，直接在标签中输入其他标签或者文字的做法是不被允许的。

与之间相当于一个容器，可以容纳所有元素。

```
<ul>
  <li>列表项 1</li>
  <li>列表项 2</li>
  <li>列表项 3</li>
  .....
</ul>
```

「2. 有序列表 ol」

标签中的 type 属性值为排序的序列号，不添加 type 属性时，有序列表默认从数字 1 开始排序。

常用的 type 属性值分别为是 1, a, A, i, I

<ol reversed="reversed">中的 reversed 属性能够让有序列表中的序列倒序排列。

<ol start="3">中的 start 属性值为 3，有序列表中的第一个序列号将从 3 开始排列。

```
<ol type="A">
  <li>列表项 1</li>
```

```
<li>列表二 </li>
<li>列表三 </li>
</ol>
```

「2. 自定义列表 dl」

定义列表常用于对术语或名词进行解释和描述，定义列表的列表项前没有任何项目符号。

```
<dl>
  <dt>名词 1</dt>
  <dd>名词 1 解释 1</dd>
  <dd>名词 1 解释 2</dd>
  ...
  <dt>名词 2</dt>
  <dd>名词 2 解释 1</dd>
  <dd>名词 2 解释 2</dd>
  ...
</dl>
```



表单

在 HTML 中，一个完整的表单通常由**表单控件**（也称为表单元素）、**提示信息**和**表单域** 3 个部分构成。表单目的是为了收集用户信息。

欢迎注册无限容量的网易邮箱！邮件地址可以登录使用其他网易旗下产品。

注册字母邮箱 注册手机号码邮箱 注册VIP邮箱

* 手机号码 @163.com
请填写手机号码

* 图片验证码 
请填写图片中的字符，不区分大小写 看不清楚？换张图片
免费获取验证码

* 短信验证码 表单
请查收手机短信，并填写短信中的验证码

* 密码
6~16个字符，区分大小写

* 确认密码
请再次填写密码

同意“服务条款”和“用户须知”、“隐私权相关政策”

立即注册

表单控件：

包含了具体的表单功能项，如单行文本输入框、密码输入框、复选框、提交按钮、重置按钮等。

提示信息：

一个表单中通常还需要包含一些说明性的文字，提示用户进行填写和操作。

表单域：

它相当于一个容器，用来容纳所有的表单控件和提示信息，可以通过他定义处理表单数据所用程序的 url 地址，以及数据提交到服务器的方法。如果不定义表单域，表单中的数据就无法传送到后台服务器。

[1. input 控件]

```
<input type="属性值" value="你好">
```

input 输入的意思

<input /> 标签为单标签

type 属性设置不同的属性值用来指定不同的控件类型

除了 type 属性还有别的属性

常用属性：

属性	属性值	描述
type	text	单行文本输入框
	password	密码输入框
	radio	单选按钮
	checkbox	复选框
	button	普通按钮
	submit	提交按钮
	reset	重置按钮
	image	图像形式的提交按钮
	file	文件域
	name	由用户自定义
value	由用户自定义	input 控件中的默认文本值
size	正整数	input 控件在页面中的显示宽度
checked	checked	定义选择控件默认被选中的项
maxlength	正整数	控件允许输入的最多字符数

用户名：`<input type="text" />`

密码：`<input type="password" />`

value 属性

value 默认的文本值。有些表单想刚打开页面就默认显示几个文字，就可以通过这个 value 来设置。

用户名：`<input type="text" name="username" value="请输入用户名">`

name 属性

name 表单的名字，这样，后台可以通过这个 name 属性找到这个表单。页面中的表单很多，name 主要作用就是用于区别不同的表单。

name 属性后面的值，是我们自己定义的。

radio 如果是一组，我们必须给他们命名相同的名字 name 这样就可以多个选其中的一个啦

name 属性，我们现在用的较少，但是，当我们学 ajax 和后台的时候，是必须的。

```
<input type="radio" name="sex" />男  
<input type="radio" name="sex" />女
```

checked 属性

表示默认选中状态。较常见于 单选按钮和复选按钮。

性 别:

```
<input type="radio" name="sex" value="男" checked="checked" />男  
<input type="radio" name="sex" value="女" />女
```

input 属性小结

属性说明作用 type 表单类型用来指定不同的控件类型 value 表单值
表单里面默认显示的文本 name 表单名字页面中的表单很多，name
主要作用就是用于区别不同的表单。checked 默认选中表示那个单
选或者复选按钮一开始就被选中了

[2. label 标签]

label 标签为 input 元素定义标注 (标签)。

label 标签主要目的是为了提高用户体验。为用户提高最优秀的服
务。

作用：用于绑定一个表单元素，当点击 label 标签的时候，被绑定的
的表单元素就会获得输入焦点。

如何绑定元素呢

第一种用法就是用 label 标签直接包含 input 表单，适合单个表单
选择

第二种用法 for 属性规定 label 与哪个表单元素绑定(通过 id)。

第一种

```
<label> 用户名 :  
  <input type="radio" name="username" value="请输入用户名" >  
</label>
```

第二种

```
<label for="sex">男 </label>  
<input type="radio" name="sex" id="sex" >
```

[3. textarea 控件(文本域)]

通过 `textarea` 控件可以轻松地创建多行文本输入框。

`cols`="每行中的字符数" `rows`="显示的行数" 我们实际开发不用

网友评论 文明上网理性发言，请遵守新闻评论服务协议 腾讯牛评 135条评论

文本域 `textarea`

登录

```
<textarea >  
  文本内容  
</textarea >
```

文本框和文本域区别

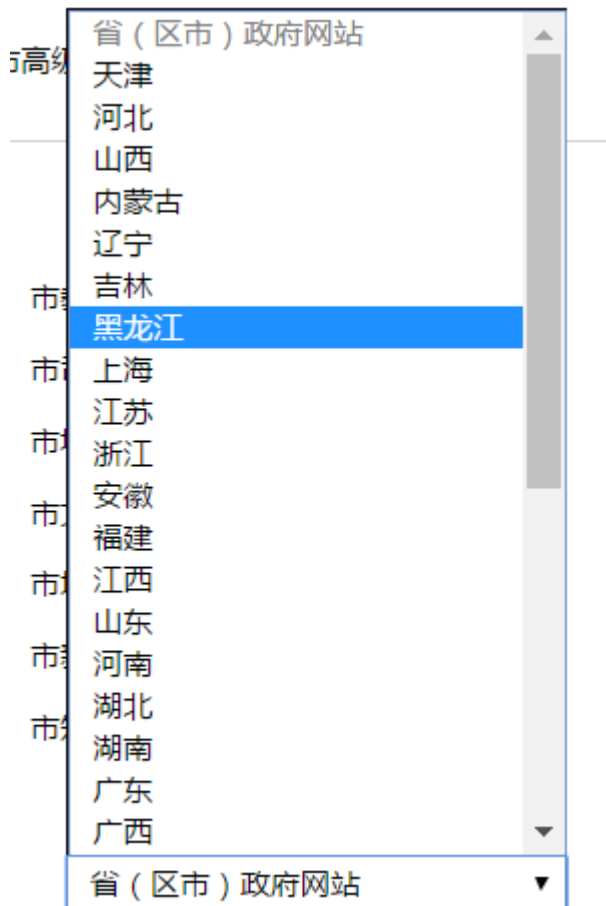
表单名称区别默认值显示用于场景 `input type="text"` 文本框只能显示一行文本单标签，通过 `value` 显示默认值用户名、昵称、密码等 `textarea` 文本域可以显示多行文本双标签，默认值写到标签中间留言板

「4. `select` 下拉列表」

如果有多个选项让用户选择，为了节约空间，我们可以使用 `select` 控件定义下拉列表。

在 `option` 中定义 `selected = "selected"` 时，当前项即为默认选中项。

我们实际开发会用的比较少



```
<select>
```

```
<option>选项 1</option>
```

```
<option>选项 2</option>
```

```
<option>选项 3</option>
```

```
...
```

```
</select>
```

form 表单域

收集的用户信息怎么传递给服务器？

通过 **form 表单域**

目的：

在 HTML 中，form 标签被用于定义表单域，以实现用户信息的收集和传递，form 中的所有内容都会被提交给服务器。

```
<form action="url 地址" method="提交方式" name="表单名称">
```

```
  各种表单控件
```

```
</form>
```

常用属性：

每个表单都应该有自己表单域。后面学 `ajax` 后台交互的时候，必须需要 `form` 表单域。

属性属性值作用 `actionurl` 地址用于指定接收并处理表单数据的服务器程序的 `url` 地址。`methodget/post` 用于设置表单数据的提交方式，其取值为 `get` 或 `post`。`name` 名称用于指定表单的名称，以区分同一个页面中的多个表单。

GET 和 POST 的区别

GET 在浏览器回退时是无害的，而 POST 会再次提交请求。

GET 请求会被浏览器主动 `cache`，而 POST 不会，除非手动设置。

GET 请求只能进行 `url` 编码，而 POST 支持多种编码方式。

GET 请求参数会被完整保留在浏览器历史记录里，而 POST 中的参数不会被保留。

GET 请求大小一般是(1024 字节)，`http` 协议并没有限制，而与服务器，操作系统有关，POST 理论上来说没有大小限制，`http` 协议规范也没有进行大小限制，但实际上 `post` 所能传递的数据量根据取决于服务器的设置和内存大小。

对参数的数据类型，GET 只接受 `ASCII` 字符，而 POST 没有限制。

GET 比 POST 更不安全，因为参数直接暴露在 `URL` 上，所以不能用来传递敏感信息。

团队约定：

元素属性值使用双引号语法

元素属性值可以写上的都写上

推荐

```
<input type="text" />
```

```
<input type="radio" name="name" checked="checked" />
```

从输入 url 到页面展示发生了什么(面试)

作者：Twinkle_

链接：<https://juejin.im/post/6869279683230629896>

来源：掘金

浏览器的多进程架构

从浏览器输入 URL 到页面渲染的整个过程都是由 浏览器架构中的各个进程之间的配合完成。

浏览器主进程：管理子进程、提供服务功能

渲染进程：将 HTML、CSS、JS 渲染成界面，js 引擎 v8 和排版引擎 Blink 就在上面，他会为每一个 tab 页面创建一个渲染进程

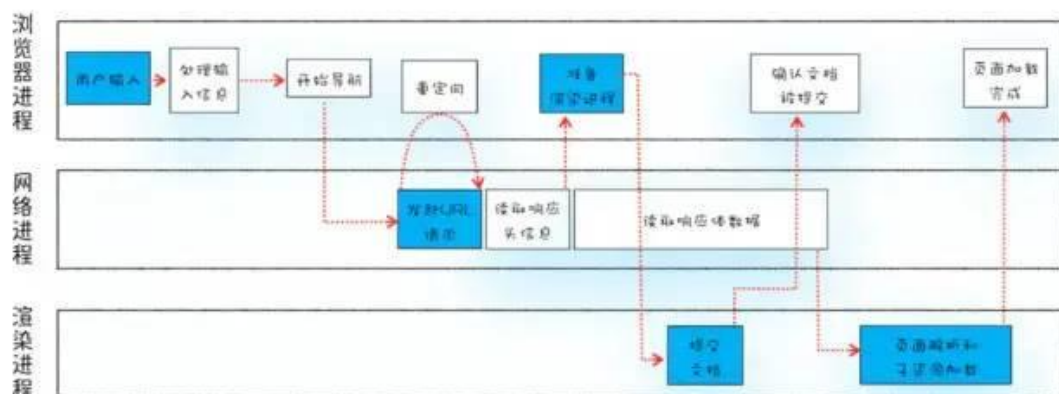
GPU 进程：本来是负责处理 3Dcss 的，后来慢慢的 UI 界面也交给 GPU 来绘制

网络进程：就是负责网络请求，网络资源加载的进程

插件进程：负责插件的运行的，因为插件很容易崩溃，把它放到独立的进程里不要让它影响别人

浏览器的多进程架构

从用户输入信息到页面展示的不同阶段，是不同的进程在发挥作用，示意图如下：



从图中可以看出，整个过程是需要各个进程之间相互配合完成的，过程大致可以描述为：

用户输入 url,处理输入信息，主进程开始导航，交给网络进程干活
网络进程发起网络请求，其中有可能发生重定向
服务器响应 URL 之后，主进程就要通知渲染进程，你要开始干活了
渲染进程准备好了，要想渲染进程提交数据，这个时间叫做提交文档
渲染进程接受到数据，完成页面渲染。

具体过程

输入 url

用户输入 url，处理输入信息：

如果为非 url 结构的字符串，交给浏览器 **默认引擎** 去搜索改字符串；
若为 url 结构的字符串，浏览器主进程会交给 **网络进程**，开始干活。

2.1 查找浏览器缓存

网络进程会先看看是否存在本地缓存，如果有就直接返回资源给浏览器进程，无则下一步 DNS-> IP -> TCP

2.2 DNS 解析

网络进程拿到 url 后，先会进行 DNS 域名解析得到 **IP 地址**。如果请求协议是 HTTPS，那么还需要建立 TLS 连接。

2.2 建立 TCP 连接，三次握手

接下来就是利用 IP 地址和服务器建立 TCP 连接。连接建立之后，向服务器发送请求。

服务器响应

服务器收到请求信息后，会根据请求信息生成响应行、响应头、响应体，并发给网络进程。网络进程接受了响应信息之后，就开始解析响应头的内容。

网络进程解析响应行和响应头信息的过程：

3.1 重定向

如果响应行状态码为 301 (永久重定向) 和 302 (临时)，那么说明需要重定向到其他 url。这时候网络进程会从响应头中的 Location

字段里读取重定向的地址，并重新发起网络请求。

3.2 响应数据处理

导航会通过请求头的 `Content-type` 字段判断响应体数据的类型。浏览器通过这个来决定如何显示响应体的内容。比如：若为 `application/octet-stream`，则会按照下载类型来处理这个请求，导航结束。若为 `text/html`，这就告诉浏览器服务器返回的是 html 格式，浏览器会通知渲染进程，你要干活了。

准备渲染进程

默认情况，每个页面一个渲染进程。但若处于 **同一站点**（同根域名 + 协议），那么渲染进程就会复用。

提交文档

渲染进程准备好后，浏览器进程发出 **“提交文档的消息”**，渲染进程接受了消息之后，会跟网络进程简历传输数据的管道。

等数据传输完成了，渲染进程会告诉浏览器进程，**确认文档提交**，这时候浏览器会更新页面，安全状态，url，前进后退的历史。

到这里导航结束，进入渲染阶段。

注：当浏览器刚开始加载一个地址之后，标签页上的图标便进入了加载状态。但此时图中页面显示的依然是之前打开的页面内容，并没立即替换为百度首页的页面。因为需要等待提交文档阶段，页面内容才会被替换。

前端 HTML 基础面试题

iframe 有哪些缺点？

iframe 是一种框架，也是一种很常见的网页嵌入方式。

【iframe 的优点】

iframe 能够原封不动的把嵌入的网页展现出来。

如果有多个网页引用 iframe，那么你只需要修改 iframe 的内容，就可以实现调用的每一个页面内容的更改，方便快捷。

网页如果为了统一风格，头部和版本都是一样的，就可以写成一个

页面，用 iframe 来嵌套，可以增加代码的可重用。

如果遇到加载缓慢的第三方内容如图标和广告，这些问题可以由 iframe 来解决。

「iframe 的缺点」

会产生很多页面，不容易管理。

iframe 框架结构有时会让人感到迷惑，如果框架个数多的话，可能会出现上下、左右滚动条，会分散访问者的注意力，用户体验度差。

代码复杂，无法被一些搜索引擎索引到，这一点很关键，现在的搜索引擎爬虫还不能很好的处理 iframe 中的内容，所以使用 iframe 会不利于搜索引擎优化。

很多的移动设备（PDA 手机）无法完全显示框架，设备兼容性差。iframe 框架页面会增加服务器的 http 请求，对于大型网站是不可取的。

现在基本上都是用 Ajax 来代替 iframe，所以 iframe 已经渐渐的退出了前端开发。

label 的作用是什么？是怎么用的？

例子 1: 点击 " 用户名:" 就可以定位光标到输入框

```
<form><label for="myid "> 用户名:</label>
<input type="text" id="myid" /></form>
```

例子 2: 点击 " 用户名:" 或按键 alt+1, 都可以定位光标到输入框

```
<form>
  <label for="myid" accesskey="1"> 用户名:</label>
  <input type="text" id="myid" tabindex="1" />
</form>
```

for 属性功能：表示 Label 标签要绑定的 HTML 元素，你点击这个标签的时候，所绑定的元素将获取焦点。

accesskey 属性

功能：表示访问 Label 标签所绑定的元素的热键，当您按下热键，所绑定的元素将获取焦点。

局限性：accessKey 属性所设置的快捷键不能与浏览器的快捷键冲突，否则将优先激活浏览器的快捷键。

HTML5 的 form 如何关闭自动完成功能？

HTML 的输入框可以拥有自动完成的功能，当你往输入框输入内容的时候，浏览器会从你以前的同名输入框的历史记录中查找出类似的内容并列在输入框下面，这样就不用全部输入进去了，直接选择列表中的项目就可以了。

但有时候我们希望关闭输入框的自动完成功能，例如当用户输入内容的时候，我们希望使用 AJAX 技术从数据库搜索并列举而不是在用户的历史记录中搜索。

关闭输入框的自动完成功能有 3 种方法：

在 IE 的 Internet 选项菜单里的内容--自动完成里面设置

设置 form 的 autocomplete 为 "on" 或者 "off" 来开启或者关闭自动完成功能

设置输入框的 autocomplete 为 "on" 或者 "off" 来开启或者关闭该输入框的自动完成功能

将 HTML5 看作成开放的网络平台

「什么是 HTML5 的基本构件 (building block) ?」

语义 - 提供更准确地描述内容。

连接 - 提供新的方式与服务器通信。

离线和存储 - 允许网页在本地存储数据并有效地离线运行。

多媒体 - 在 Open Web 中，视频和音频被视为一等公民 (first-class citizens)。

2D/3D 图形和特效 - 提供更多种演示选项。

性能和集成 - 提供更快的访问速度和性能更好的计算机硬件。

设备访问 - 允许使用各种输入、输出设备。

外观 - 可以开发丰富的主题。

浏览器是怎么对 HTML5 的离线储存资源进行管理和加载的呢？

在浏览器的 html 头部加上 manifest 属性，如果是第一次访问浏览器会根据 manifest 的内容进行下载存储离线内容，如果已经访问过则从离线存储中进行加载，然后在比对服务器如果有新内容在更新离线存储

离线的环境下，浏览器就直接使用离线存储的资源。

浏览器的渲染过程？

- 1、将获取的 html 解析成 dom 树
- 2、处理 css，构成层叠样式表模型 CSSOM
- 3、将 dom 树和 CSSOM 合并为渲染树
- 4、根据 CSSOM 将渲染树的节点布局计算
- 5、将渲染树节点样式绘制到页面上

// 注意

在渲染的过程中是自上而下渲染，
js 会阻塞页面的渲染，优先等 js 执行完成
如果在渲染的过程中改变了样式，会造成回流需要重新渲染

link 和 @import 的区别？

- 1、从属关系区别：
link 属于 html 标签，而 @import 是 css 提供的。
- 2、加载顺序区别：
页面被加载时，link 会同时被加载，而 @import 引用的 css 会等到页面被加载完再加载。
- 3、兼容性区别：
import 只在 IE5 以上才能识别，而 link 是 html 标签，无兼容问题。
- 4、dom 可操作性区别：
可以通过 JS 操作 DOM，插入 link 标签来改变样式；由于 DOM 方法是基于文档的，无法使用 @import 的方式插入样式
- 5、权重区别：
如果已经存在相同样式，@import 引入的这个样式将被该 CSS 文件本身的样式层叠掉，表现出 link 方式的样式权重高于 @import 的权重这样的直观效果。
(简而言之，link 和 @import，谁写在后面，谁的样式就被应用，后面的样式覆盖前面的样式。)

src 与 href 的区别？

- 1、href 是指向网络资源所在位置，建立和当前元素（锚点）或当前文档（链接）之间的链接，用于超链接。
- 2、src 是指向外部资源的位置，指向的内容将会嵌入到文档中当前标签所在位置；在请求 src 资源时会将其指向的资源下载并应用到文档内，例如 js 脚本，img 图片和 frame 等元素。当浏览器解析到该元素时，会暂停其他资源的下载和处理，直到将该资源加载、编译、执行完毕，图片和框架等元素也如此，类似于将所指向资源嵌入当前标签内。这也是为什么将 js 脚本放在底部而不是头部。

□ 感谢大家

如果你觉得这篇内容对你挺有帮助的话：

点赞支持下吧，让更多的人也能看到这篇内容（收藏不点赞，都是耍流氓 -_-）

关注公众号「**前端 fan**」，定期为你推送好文。

下一篇将着重记录 CSS 的学习笔记

HTML5 与 CSS3 新增特性单独写一章