



HTML和CSS简介

讲师：李立超

问HTML为何物，其实就是标记语言

HTML

HTML

- HTML (Hypertext Markup Language)
超文本标记语言。
- 它负责网页的三个要素之中的结构。
- HTML使用标签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。

- 一个最基本的HTML页面：

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>网页标题</title>
</head>
<body>
    <h1>网页正文</h1>
</body>
</html>
```


标签

- HTML中的标记指的就是**标签**。
- HTML使用标记标签来描述网页。
- 结构：

<**标签名**>标签内容</**标签名**>

<**标签名** />

元素

- 我们还将一个完整的标签称为元素。
- 这里我们可以将元素和标签认为是一个同义词。

`<h1>一级标题</h1>`

上边的h1我们就称为元素

`<p>我是一个段落</p>`

p也是一个元素，em是p的子元素，p是em的父元素。

```
<body>  
    <p> <em>内容</em> </p>  
</body>
```

- body也是一个元素。
- body是p和em的祖先元素。
- p和em是body的后代元素。

属性

- 可以为HTML标签设置属性。
- 通过属性为HTML元素提供附加信息。
- 属性需要设置在开始标签或自结束标签中。
- 属性总是以名称/值对的形式出现。
- 比如：name= "value"
- 有些属性可以是任意值，有些则必须是指定值。

```
<h1 title="我是一个标题">标题</h1>
```

```
<img src="" alt="" />
```


常见属性

- id
 - id属性作为标签的唯一标识，在同一个网页中不能出现相同的id属性值。
- class
 - class属性用来为标签分组，拥有相同class属性的标签我们认为就是一组，可以出现相同的class属性，可以为一个元素指定多个class。
- title
 - title属性用来指定标签的标题，指定title以后，鼠标移入到元素上方时，会出现提示文字。

注释

- HTML注释中的内容不会在网页中显示。
- 格式:
`<!-- 注释内容 -->`
- 合理的使用注释可以帮助开发人员理解网页的代码。
- 注释不能嵌套！

HTML从哪来，又会到哪去

HTML的发展

HTML的发展

- 1993年6月：HTML第一个版本发布。
- 1995年11月：HTML2.0
- 1997年1月：HTML3.2（W3C推荐）
- 1999年12月：HTML4.01（W3C推荐）
- 2000年底：XHTML1.0（W3C推荐）
- 2014年10月：HTML5（W3C推荐）

doctype

- HTML总共有那么多的版本，而且这其中至少有三个版本在广泛使用，那么浏览器怎么知道我们在使用哪个版本呢？
- 为了让浏览器知道我们使用的HTML版本我们还需要在网页的最上边添加一个doctype声明，来告诉浏览器网页的版本。

html4

- 过渡版

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">
```

- 严格版

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  
"http://www.w3.org/TR/html4/strict.dtd">
```

- 框架集

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN"  
"http://www.w3.org/TR/html4/frameset.dtd">
```

xhtml 1.0

- 过渡版

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "  
http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

- 严格版

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

- 框架集

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
```

html5

- 我们会发现html4.01和xhtml的文档声明十分的麻烦。不过不用担心，以上的内容都不是我们使用的，我们使用的是html5的文档声明，而且非常简单：

```
<!DOCTYPE html>
```


怪异模式

- 为了兼容一些旧的页面，浏览器中设置了解析模式：
 - 标准模式 (Standards Mode)
 - 怪异模式 (Quirks Mode)
- 怪异模式解析网页时会产生一些不可预期的行为，所以我们应该避免怪异模式的出现。
- 避免的最好方式就是在页面中编写正确的doctype。

唉唉唉~ 我写的中文怎么都变成鸟语了！

编码问题

编码问题

- 在计算机的内部，文件都是以二进制编码保存的。
- 所谓的二进制编码就是指1和0，也就是我们的所有内容都需要转换为1和0。
- **中国**两个字在计算机的底层保存的可能要转换为**10100101**这种二进制码，这一过程称为**编码**。
- 计算机在读取文件时需要将10100101在转换为**中国**给我们显示这一过程称为**解码**。

字符集

- 这就带来一个问题，中国到底是10100101还是01011010到底由谁说了算。
- 所以我们还需要一个东西称为**字符集**，字符集规定了如何将文本转换为二进制编码。
- 常见的字符集：ASCII、ISO8859-1、GBK、GB2312、UTF-8。

乱码

- 如果我们保存文件时使用的是utf-8进行编码，而浏览器读取页面时使用gb2312，这样就会导致页面中的内容不能正常显示，也就是我们所说的乱码。
- 所以我们只需要统一两者使用的字符集就可以解决乱码问题。
- 这里为了页面有更好的使用性，我们一般使用utf-8。

解决

- 保存文件的编码我们直接通过编辑器即可指定，接下来就是要告诉浏览器使用什么字符集去解析文件。
- 在html5中只需要使用meta标签即可完成这个任务：

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

<meta>

- 作用：

- <meta> 标签可提供有关页面的元信息，比如针对搜索引擎和更新频度的描述和关键词。
- <meta> 标签位于文档的头部，不包含任何内容。<meta> 标签的属性定义了与文档相关联的名称/值对。

meta的用法

- 设置页面的字符集
 - `<meta charset="utf-8">`
- 设置网页的描述
 - `<meta name="description" content="">`
- 设置网页的关键字
 - `<meta name="keywords" content="">`
- 请求的重定向
 - `<meta http-equiv="refresh" content="5;url=地址"/>`

这么多标签都是干啥的啊？

常用标签

<html>

- 作用：
 - <html> 标签用于告诉浏览器这个文档中包含的信息是用HTML编写的。
- 用法：
 - 所有的网页的内容都需要编写到html标签中，一个页面中html标签只能有一个。
 - html标签中有两个子标签head和body。

<head>

- 作用：
 - <head> 标签用来表示网页的元数据，head 中包含了浏览器和搜索引擎使用的其他不可见信息。
- 用法：
 - head 标签作为html标签的子元素的出现，一个网页中只能有一个head。

<title>

- 作用：
 - <title> 标签表示网页的标题，一般会在网页的标题栏上显示。
 - title 标签中的文字，是页面优化的最重要因素。在搜索引擎的搜索时最先看到的、最醒目的内容。
- 用法：
 - 建议将title 标签紧贴着head 标签编写，这样搜索引擎可以快速检索到标题标签。
 - 网站中的多个页面的title 也不应该重复，这样不利于搜索隐藏检索。

<body>

- 作用：
 - <body> 标签用来设置网页的主体，所有在页面中能看到的內容都应该编写到body标签中。
- 用法：
 - body标签作为html的子标签使用。

<h1> ~ <h6>

- 作用：

- **h1~h6**都是网页中的标题标签，用来表示网页中的一个标题，不同的是，从h1~h6重要性越来越低。
- 标题标签相当于正文的标题，通常认为重要性仅次于页面的title。
- 一般标题标签我们只会使用到h3，h3以后的标题标签对于搜索引擎就没有什么意义了。
- 一个页面中只会使用一个h1标签。

<p>

- 作用：

- <p> 标签表示网页中的一个段落。
- 一般浏览器会在段落的前和后各加上一个换行，也就是段落会在页面中自成一行。

- 作用

-
 标签表示一个换行标签，使用br标签可以使br标签后的内容另起一行。

<hr />

- 作用：

- <hr /> 标签是水平线标签，使用hr标签可以在页面中打印一条水平线，水平线可以将页面分成上下两个部分。

- 作用：
 - 标签是图片标签，可以用来向页面中引入一张外部的图片。
- 属性：
 - src
 - 指向一个外部的图片的路径。
 - alt
 - 图片的描述

<a>

- 作用：
 - <a> 标签是超链接标签，通过a标签，可以快速跳转到其他页面。
- 属性：
 - href
 - 指向一个链接地址
 - target
 - 设置打开目标页面的位置，可选值：_blank新窗口、_self当前窗口。

HTML这么厉害，但是多写几个空格就不行了呢！

实体（转义字符）

实体

- 在HTML中预留了一些字符。
- 这些预留字符是不能在网页中直接使用的。
- 比如<和>,我们不能直接在页面中使用<和>号,因为浏览器会将它解析为html标签。
- 为了可以使用这些预留字符,我们必须在html中使用字符实体。
- 语法: **&实体名;**

字符实体

- 小于号<
 - <
- 大于号>
 - >
- 空格
 -
- 和符号&
 - &
- 版权©
 - ©
- 引号"
 - "
- 注册商标®
 - ®
- 商标™
 - ™

工欲善其事必先利其器！

开发工具

文本编辑器

- 在windows中我们只需要使用最简单的**记事本**就可以完成所有的网页的开发。
- 但是一般我们会使用一些具有提示功能的纯文本编辑器：
 - Notepad++(**免费**)
 - Sublime(**收费**)
- 当然还有很多其他的工具。

IDE

- IDE（集成开发工具）
- IDE拥有比纯文本编辑器更加强大的提示功能，也是我们开发中用的比较多的工具。
 - DreamWeaver（**收费**）
 - WebStorm（**收费**）
 - Hbuilder（**免费**）
- 当然也有其他的IDE。

工具的选择

- 上边说了那么多工具我们要使用哪个呢？
- 其实使用哪个工具都不重要，我们也不用费劲心机去讨论工具的好坏，找一个自己喜欢用的即可。
- 而且我们也要做到不依赖于某一个工具，我们要做到，即使只使用最简单的记事本，我们也可以照常开发。

HTML页面实在是太丑了，怎么破？

CSS

CSS

- **层叠样式表** (Cascading Style Sheets)
- css可以用来为网页创建样式表，通过样式表可以对网页进行装饰。
- 所谓层叠，可以将整个网页想象成是一层一层的结构，层次高的将会覆盖层次低的。
- 而css就可以分别为网页的各个层次设置样式。

基本语法

- CSS的样式表由一个一个的样式构成，一个样式又由**选择器**和**声明块**构成。
- 语法：
 - **选择器** {**样式名:样式值** ; **样式名:样式值** ; }
 - **p** {**color:red** ; **font-size:12px;**}

行内样式

- 可以直接将样式写到标签内部的style属性中，这种样式不用填写选择器，直接编写声明即可。

```
<p style="color: red;font-size: 30px"></p>
```

- 这种方式编写简单，定位准确。但是由于直接将css代码写到了html标签的内部，导致结构与表现耦合，同时导致样式不能够复用，所以这种方式我们不使用。

内部样式表

- 可以直接将样式写到 **<style>** 标签中。

<style>

```
p{color:red; font-size: 30px;}
```

</style>

- 这样使css独立于html代码，而且可以同时为多个元素设置样式，这是我们使用的比较多的一种方式。
- 但是这种方式，样式只能在一个页面中使用，不能在多个页面中重复使用。

外部样式表

- 可以将所有的样式保存到一个外部的css文件中，然后通过<link>标签将样式表引入到文件中。

```
<link rel="stylesheet" type="text/css"
href="style.css">
```

- 这种方式将样式表放入到了页面的外部，可以在多个页面中引入，同时浏览器加载文件时可以使用缓存，这是我们开发中使用的最多的方式。

