

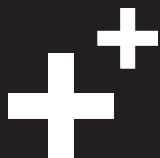
Python开发

NSD DEVWEB

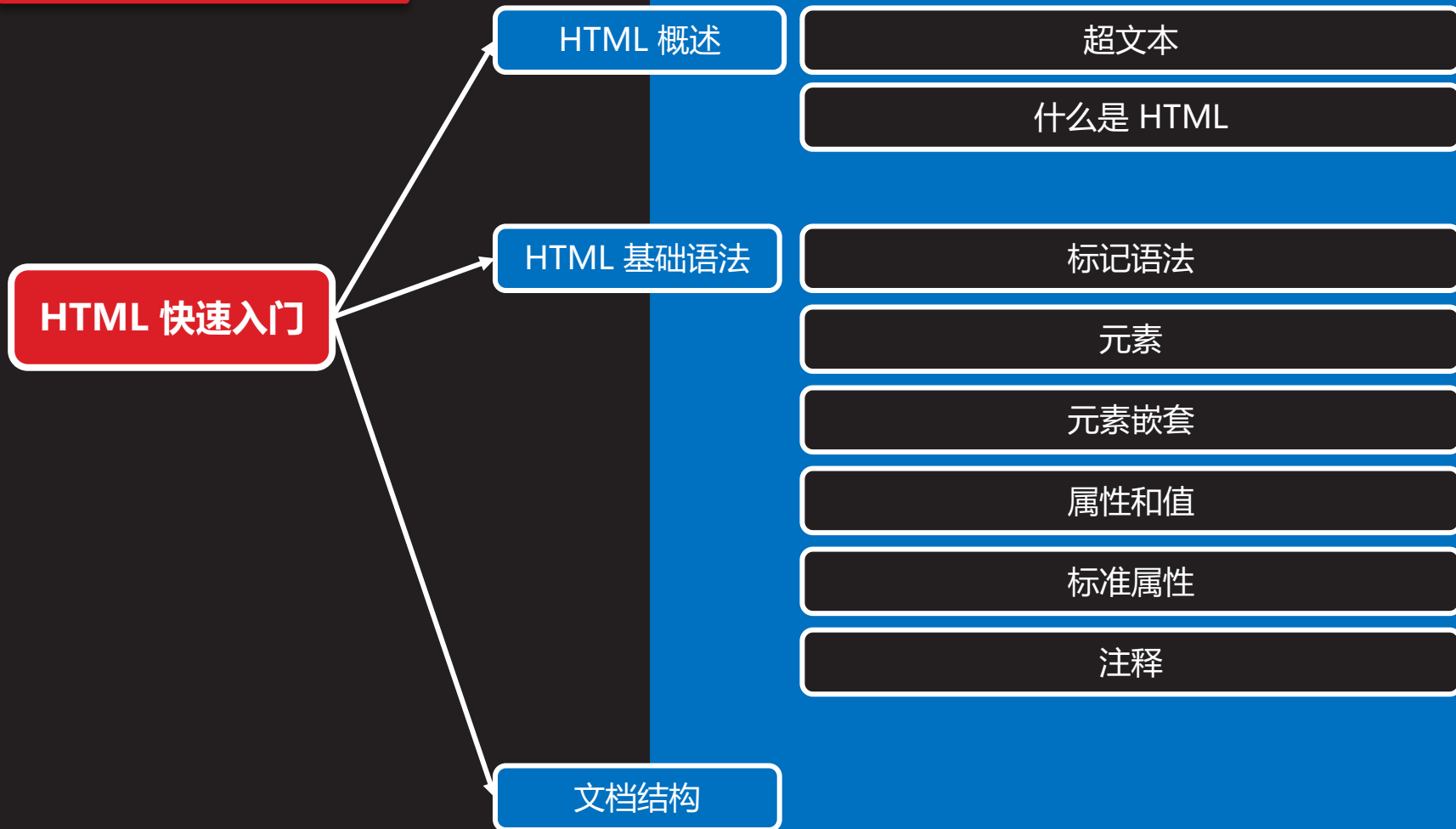
DAY01

内容

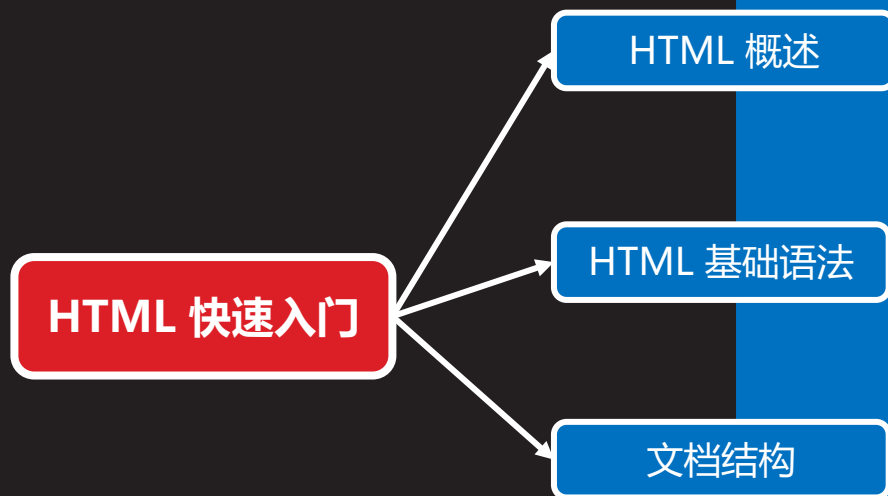
上午	09:00 ~ 09:30	HTML快速入门
	09:30 ~ 10:20	基本标签用法
	10:30 ~ 11:20	
	11:30 ~ 12:00	
下午	14:00 ~ 14:50	Form标点及控件
	15:00 ~ 15:50	
	16:10 ~ 17:00	CSS 概述
	17:10 ~ 18:00	总结和答疑



HTML 快速入门



HTML 快速入门



HTML 文档的结构

文档类型声明

<html> 元素

<head> 元素

文档头部内容 - <title>

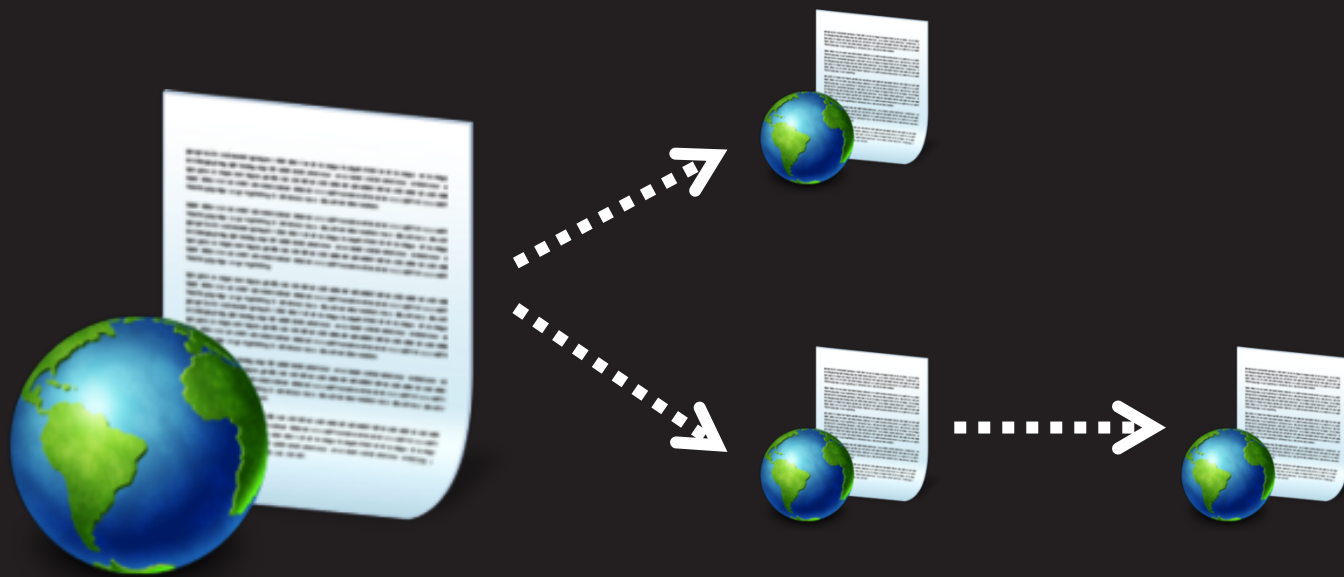
文档头部内容 - <meta>

<body> 元素

HTML 概述

超文本

- Web 是一个超文本文件的集合
- 超文本文件是 Web 的基本组成单元，也称为网页或 HTML 文档、Web 页等，通常是以.html或.htm为后缀的文件
- Web 页之间通过超文本中的超级链接组织在一起



什么是 HTML

- HTML (HyperText Markup Language) : 超文本标记语言, 一种纯文本类型的语言
 - 使用带有尖括号的“标记”将网页中的内容逐一标识出来
- 用来设计网页的标记语言
- 用该语言编写的文件, 以 .html 或者 .htm 为后缀
- 由浏览器解释执行
- HTML 页面上, 可以嵌套用脚本语言编写的程序段, 如: VBScript, JavaScript

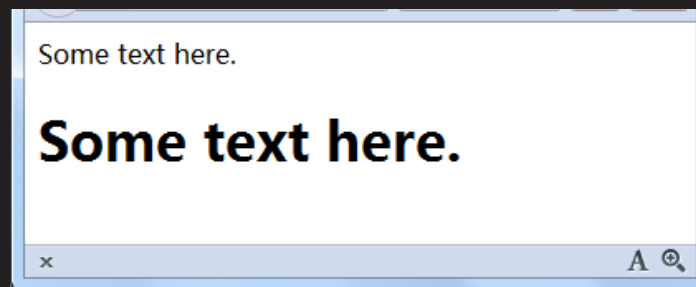


HTML 基础语法

标记语法

- HTML 用于描述功能的符号称为“标记”，比如 `<p>`、`<h1>` 等
 - 标记在使用时必须使用尖括号括起来
 - 有封闭类型标记，也有非封闭类型的标记

Some text here.
`<h1>Some text here.</h1>`



标记会导致不同的显示效果



标记语法（续1）

- 封闭类型标记（也叫双标记）：必须成对出现
 - <标记>内容</标记>
- 封闭类型的标记必须成对出现
 - 如果一个应该封闭的标记没有被封闭，则会产生**意料不到的错误**

<p>段落一</p>

<h1>另一个标题，忘了结束

<p>还是普通段落吗？</p>

段落一


另一个标题，忘了结束
还是普通段落吗？



标记语法（续2）

- 非封闭类型标记，也叫做空标记，或者单标记
 - <标记 />
 - 或者
 - <标记>
- 不能包含内容

普通文本一
普通文本二
普通文本三
普通文本四



普通文本一
普通文本二 普通文本三
普通文本四



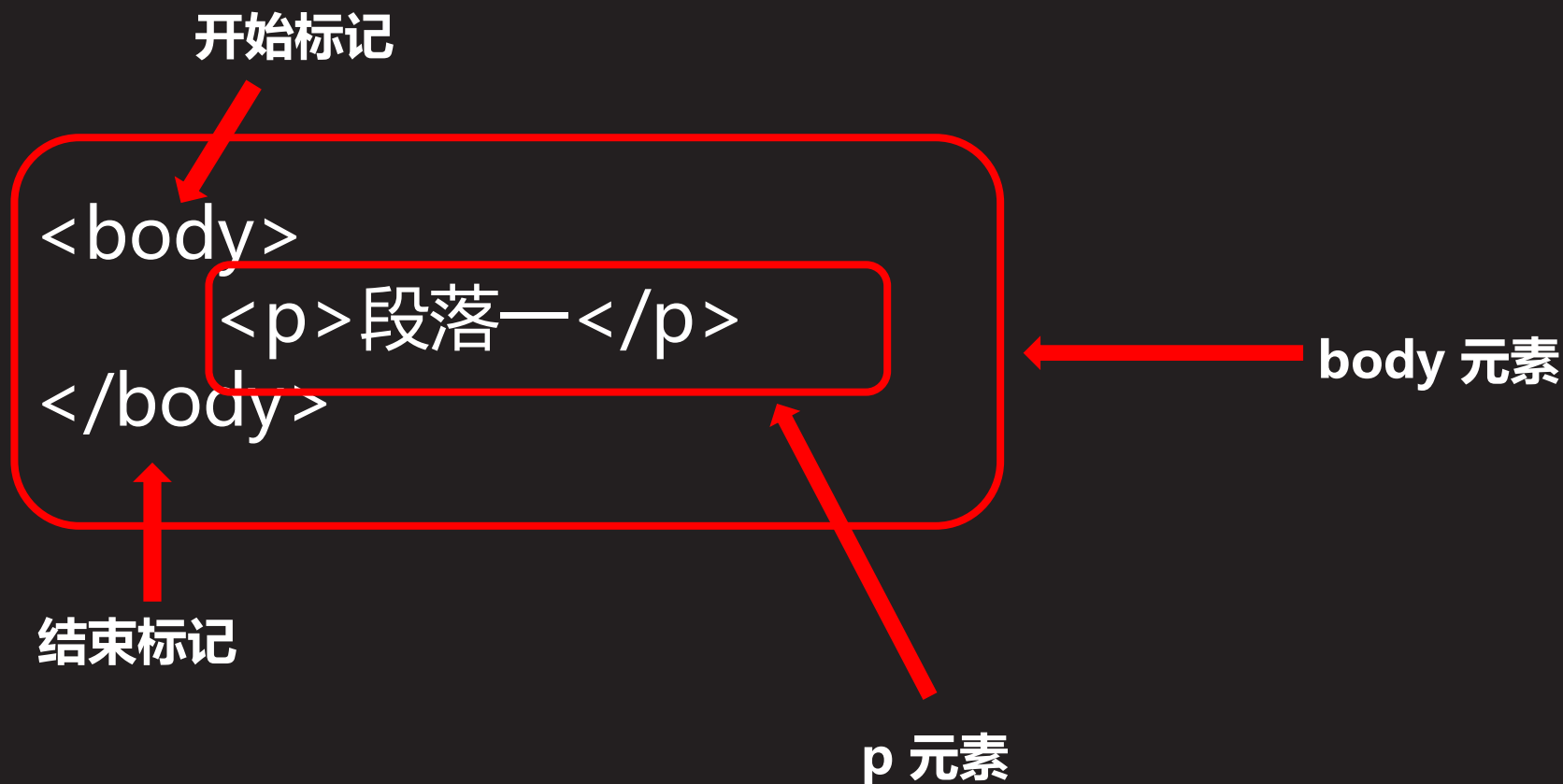
元素

- 元素，即标记
- 每一对尖括号包围的部分
 - 如 `<body></body>` 包围的部分就叫做 body 元素
 - 元素就像是小标签，用于标识网页文档的不同部分
- 元素可以包含文本内容和其他元素，也可以是空的，比如前面所述的空标记



元素嵌套

- 元素之间可以相互嵌套，形成更为复杂的语法
- 在嵌套元素时需要注意标记的嵌套顺序



属性和值

- 属性用来修饰元素
 - 属性的声明必须位于开始标记里
 - 一个元素的属性可能不止一个，多个属性之间用空格隔开
 - 多个属性之间不区分先后顺序
- 每个属性都有值
 - 属性和属性的值之间用等号连接
 - 属性的值包含在引号中

<p align = "center" >段落一</p>

属性名称

属性值



标准属性

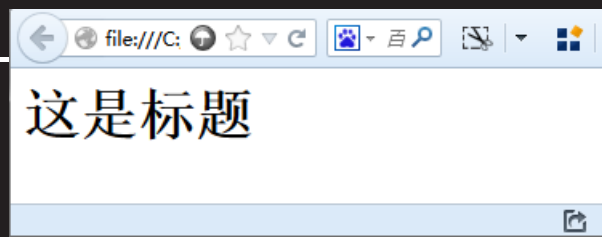
- 每个元素都有自己所特有的属性
- 有些属性是绝大多数元素都支持的属性，称为标准属性（或通用属性）
 - id
 - title
 - class
 - style



注释

- 为代码添加适当的注释是一种良好的编码习惯
- 注释只在编辑文档情况下可见，在浏览器展示页面时并不会显示
- 添加注释的语法如下：

`<!-- 注释的文本内容 -->`
`<!-- 注释部分 -->`
`<h1>这是标题</h1>`



- 1、 “<!--” 和 “-->” 之间的任何内容都不会显示在浏览器中
- 2、 注释不可以嵌套在其他注释中
- 3、 注释不可以位于嵌在 <> 中



HTML 文档片段

- 【参见COOKBOOK】



文档结构

HTML 文档的结构

- 文档类型声明
- html 页面
 - 文件头：<head> </head>
 - 文档主体部分：<body> </body>

文档类型声明

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
  </head>
  <body>
  </body>
</html>
```

文件头

文档主体

html 页面



文档类型声明

- 在文档的起始用DOCTYPE声明指定版本和风格
 - 让浏览器清楚文档的版本、类型和风格

- Strict DTD

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

- Transitional DTD

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

- Frameset DTD

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

- HTML 5

```
<!DOCTYPE HTML>
```



<html> 元素

- 整个文档的包含元素
 - 在DOCTYPE的文档类型声明之后
 - 按照严格版本（Strict XHTML 1.0），起始的 <html> 标记中必须包含命名空间标识符
- 有两个子元素
 - <head>：页面的头部内容
 - <body>：页面的主体内容



```
a.html - 记事本
文件(F)  编辑(E)  格式(O)  查看(V)  帮助(H)

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

<html xmlns="http://www.w3.org/1999/xhtml" >
  <head>

  </head>
  <body>

  </body>
</html>
```



<head> 元素

- <head> 元素用于为页面定义全局信息
 - 所有其他头元素的容器
 - 紧跟在起始标签 <html> 之后
- 可包含
 - title、meta、script、style、link等

<head>

<title>HTML 文档</title>

<meta name="keywords" content="html,css" />

<script type="javascript"> </script>

<style type="text/css"> </style>

</head>



文档头部内容 - - <title>

- 标题元素 <title></title> 用于为文档定义标题
 - 标题元素的内容出现在浏览器顶部
 - 没有属性
 - 必须出现在 <head> 元素中
 - 一个文档只能有一个标题元素

<!DOCTYPE HTML>

<html>

<head>

<title>我的第一个HTML页面</title>

</head>

<body>

</body>

</html>



文档头部内容 --<meta>

- 元数据元素 <meta>用于定义网页的基本信息
- 为空标记
- 常用属性有：content、http-equiv

<head>

<title>HTML 文档</title>

<meta http-equiv="content-type"

content="text/html;charset=utf-8" />

<meta name="keywords" content="html,css" />

</head>



<body> 元素

- <body> 元素出现在 <head> 元素之后，包含网页要显示给读者的内容，称为主体内容
- 可以包含除了html、head外所有元素

```
<!DOCTYPE HTML>
```

```
<html>
```

```
    <head>
```

```
        <title>我的第一个HTML页面</title>
```

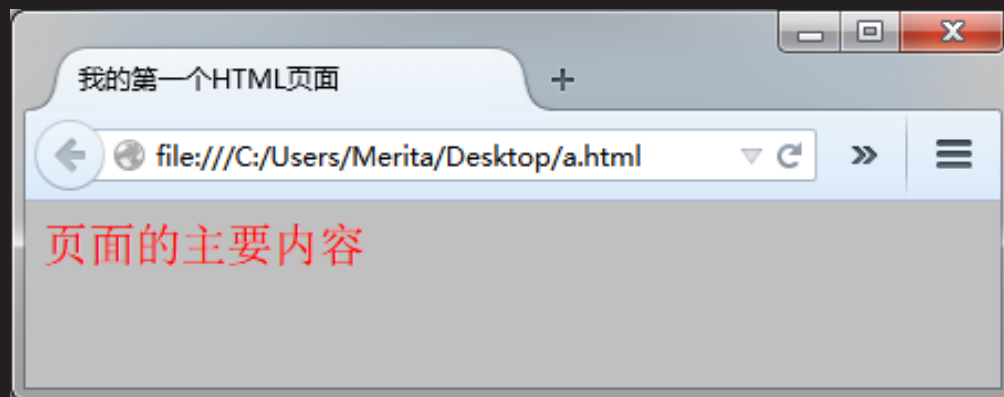
```
    </head>
```

```
    <body text="red" bgcolor="silver">
```

```
        页面的主要内容
```

```
    </body>
```

```
</html>
```



创建 HTML 文档并设置头元素

- 【参见COOKBOOK】
 - 1、创建标准结构的 HTML 文档
 - 2、为文档添加头元素 <head>
 - 3、为页面设置标题、编码格式



基本标签用法

基本标签用法

使用文本标记

文本与特殊字符

文本样式

标题元素 <h>

段落元素 <p>

换行元素

分区元素 和 <div>

行内元素与块级元素

分隔线元素 <hr>

图像和链接

URL

图像元素

链接元素<a>

链接的表现形式

锚点

基本标签用法

基本标签用法

使用文本标记

图像和链接

表格

创建表格

表格的常用属性

不规则表格

列表

列表的作用

有序列表

无序列表

列表嵌套

使用文本标记

文本与特殊字符

- 文本是网页上的重要组成部分
 - 直接书写的文本会用浏览器默认的样式显示
 - 包含在标记中的文本则会被显示为标记所拥有的样式
- 空格折叠
 - 多个空格或制表符压缩成单个空格，即只显示一个空格
- 特殊字符（如空格），需要进行转义（使用字符实体）

The <p> element. © 2011 by tarena.



The `<p>` element. © 2011 by tarena.

文本样式

- 文本样式的作用是对文本进行修饰，如加粗、倾斜等

- `...`：加粗
- `<i>...</i>`：倾斜
- `<u>...</u>`：下划线
- `<s>...</s>`：删除线
- `^{...}`：上标
- `_{...}`：下标

`粗体文本`

`<i>斜体</i>文本`

`<u>下划线</u>文本`

`<s>删除线</s>文本`

`_{下标}文本`

`^{上标}文本`



标题元素 <h#>

- 标题元素让文字以醒目的方式显示，往往用于文章的标题
- 基本语法：<h#> ... </h#>
 - # = 1, 2, 3, 4, 5, 6
 - 从<h1>到<h6>，即 标题1-标题6

<h1>h1 text</h1>

<h2>h2 text</h2>



h1 text

h2 text

段落元素 <p>

- <p> 元素提供了结构化文本的一种方式
- <p> 元素中的文本会用单独的段落显示
 - 与前后的文本都换行分开
 - 添加一段额外的垂直空白距离，作为段落间距
 - 常用属性： align

<p>The first paragraph.</p>

<p align="right">The second paragraph.</p>

The first paragraph.

The second paragraph.

换行元素

- 使用
 元素在任何地方创建手工换行
 - 空标记

<p>This is a
 paragraph.</p>



This is a
paragraph.

分区元素 和<div>

- 分区元素用于为元素分组，常用于页面布局
- 块分区元素：<div></div>
- 行内分区元素：
 - 设置同一行文字内的不同格式

<div style="color:blue;">

<p>first</p>

<p>second</p>

</div>

<p>

一周畅销榜

</p>

first

second

一周畅销榜

行内元素与块级元素

- 块级元素
 - 默认情况下，块级元素会独占一行，即元素前后都会自动换行
 - 如：`<p>`、`<div>`、`<h1>`
- 行内元素
 - 不会换行，可以和其他行内元素位于同一行
 - 如：``、``、`<i>`、`<u>`



使用文本标记为 HTML 页面添加内容

- 【参见COOKBOOK】
 - 1、创建 html 文档
 - 2、使用文本标记为页面添加内容



图像和链接



URL

- URL (Uniform Resource Locator) : 统一资源定位器 , 用来标识网络中的任何资源
 - 文本、图片、音视频文件 , 段落 , 或其他超文本
- 即路径 , 指从当前位置到目标位置所经过的路线
- Web 页面常用 URL 形式 :
 - 绝对路径
 - 相对路径



图像元素

- 使用 元素将图像添加到页面
 - 空标记
- 必须属性：src
- 常用属性：width、height

```

```



链接元素 <a>

- 使用 <a> 元素创建一个超级链接，语法如下：
- 文本
 - href 属性：链接 URL
 - target 属性：目标，可取值为 _blank、_self 等
 - name 属性：锚点名称

```
<a href="http://www.google.com.hk">To Google</a>
<br />
<a href="teacher/teacher.asp" target="_blank">
    To other page
</a>
```



[To Google](http://www.google.com.hk)
[To other page](teacher/teacher.asp)

链接的表现形式

- 目标文档为下载资源

`下载`

- 电子邮件链接

`联系我们`

- 返回页面顶部的空链接

`...`

- 链接到JavaScript

`JS 功能`



锚点

- 锚点是文档中某行的一个记号
 - 用于链接到文档中的某个位置

- 使用方式

- 1、定义锚点

`锚点一`

- 2、链接到锚点：在锚点名前加上#

`...`

如果文本/图像与锚点存在同一页面

`...`

如果文本/图像与锚点存在不同页面



锚点（续1）

``

定义锚点

`

`

`

`

`

`

`Return to top`

使用链接，导航到锚点



为HTML页面添加图像和链接

- 【参见COOKBOOK】
 - 1、创建 html 文档
 - 2、使用图像和链接元素为页面添加内容



表格

创建表格

- 定义表格：使用成对的 `<table></table>` 标记
- 创建表行：使用成对的 `<tr></tr>` 标记
- 创建单元格：使用成对的 `<td></td>` 标记

```
<table border="1">
```

```
<tr>
```

```
<td>第1行，第1列</td>
```

```
<td>第1行，第2列</td>
```

```
</tr>
```

```
<tr>
```

```
<td>第2行，第1列</td>
```

```
<td>第2行，第2列</td>
```

```
</tr>
```

```
</table>
```



第1行，第1列	第1行，第2列
第2行，第1列	第2行，第2列

表格的常用属性

- <table>元素
 - width , 设置表格宽度
 - height , 设置表格高度
 - align , 设置表格对齐方式(left|center|right)
 - border , 设置表格边框宽度
 - cellpadding , 设置内边距(单元格边框与内容之间的距离)
 - cellspacing , 设置外边距(单元格之间的距离)
 - bgcolor , 设置表格背景颜色



表格的常用属性（续1）

- <tr>元素
 - align , 设置水平对齐方式(left|center|right)
 - valign , 设置垂直对齐方式(top|middle|bottom)
- <td>元素
 - align , 设置水平对齐方式(left|center|right)
 - valign , 设置垂直对齐方式(top|middle|bottom)
 - width , 设置宽度
 - height , 设置高度
 - colspan , 设置单元格跨列
 - rowspan , 设置单元格跨行



不规则表格

- 设置单元格 `<td>` 的跨行或者跨列属性
- 跨列：`colspan`
 - 水平方向延伸单元格，值为一正整数，代表此单元格水平延伸的单元格数
- 跨行：`rowspan`
 - 垂直方向延伸单元格，值为一正整数，代表此单元格垂直延伸的单元格数

A单元格跨行

A	B	
	C	D

B单元格跨列

为 HTML文档添加表格

- 【参见COOKBOOK】



列表

列表的作用

- 列表是指将具有相似特征或者具有先后顺序的几行文字进行对齐排列

- one

- two

- three

1. one

2. two

3. three

- 所有的列表都由列表类型和列表项组成
 - 列表类型：有序列表和无序列表
 - 列表项：，用于指示具体的列表内容



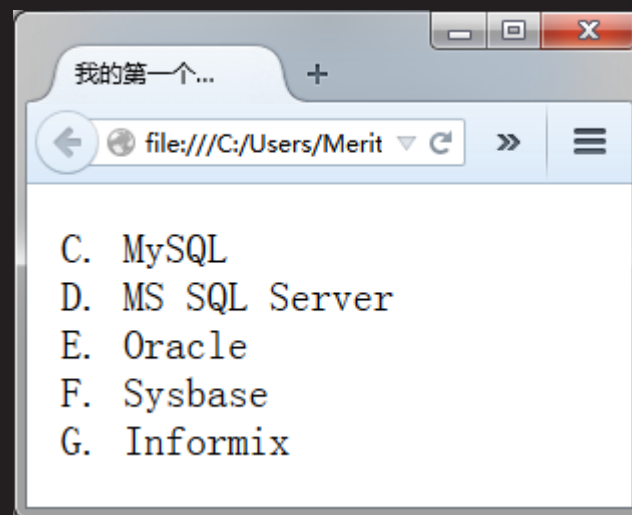
有序列表

- 元素编写有序列表，用于列出页面上有特定次序的一些项目
- 元素中只能包含列表项元素


```
<ol type="列表类型" start="起始编号">
    <li>...</li>
    <li>...</li>
    ...
</ol>
```
- type 属性值
 - 1，数字(默认)
 - a，小写字母
 - A，大写字母
 - i，小写罗马数字
 - I，大写罗马数字

有序列表 (续1)

```
<ol type="A" start="3">
  <li>MySQL</li>
  <li>MS SQL Server</li>
  <li>Oracle</li>
  <li>Sysbase</li>
  <li>Informix</li>
</ol>
```



无序列表

- 元素表示无序列表，用于列出页面上没有特定次序的一些项目
- 元素中只能包含具体的列表项元素

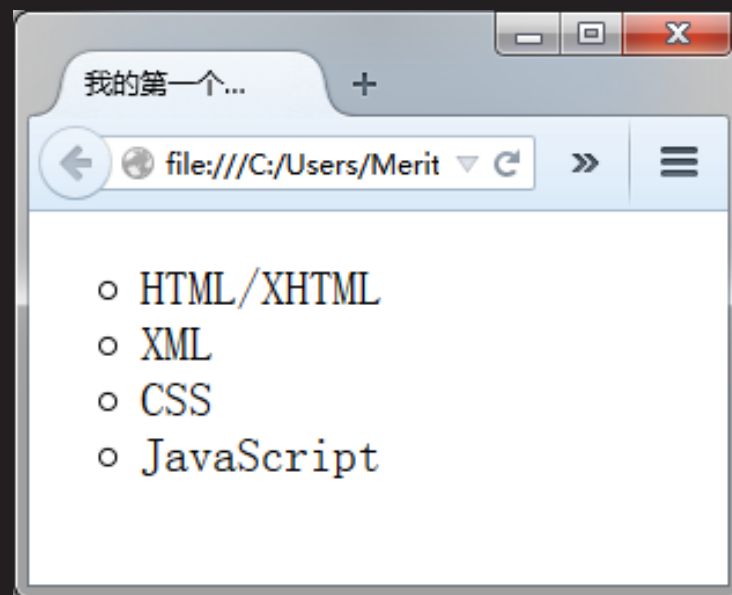
```
<ul type="列表类型" >
    <li>...</li>
    <li>...</li>
    ...
</ul>
```

- type 属性值
 - disc , 实心圆(默认)
 - circle , 空心圆
 - square , 实心矩形



无序列表 (续1)

```
<ul type="circle">
  <li>HTML/XHTML</li>
  <li>XML</li>
  <li>CSS</li>
  <li>JavaScript</li>
</ul>
```



列表嵌套

- 将列表元素嵌套使用，可以创建多层列表
 - 常用于创建文档大纲、导航菜单等

```
<ul>
  <li>
    Web基础知识
    <ul>
      <li>Web的工作原理</li>
    </ul>
  </li>
  <li>
    HTML快速入门
    <ul>
      <li>基础语法</li>
    </ul>
  </li>
</ul>
```

- Web基础知识
 - Web的工作原理
- HTML快速入门
 - 基础语法

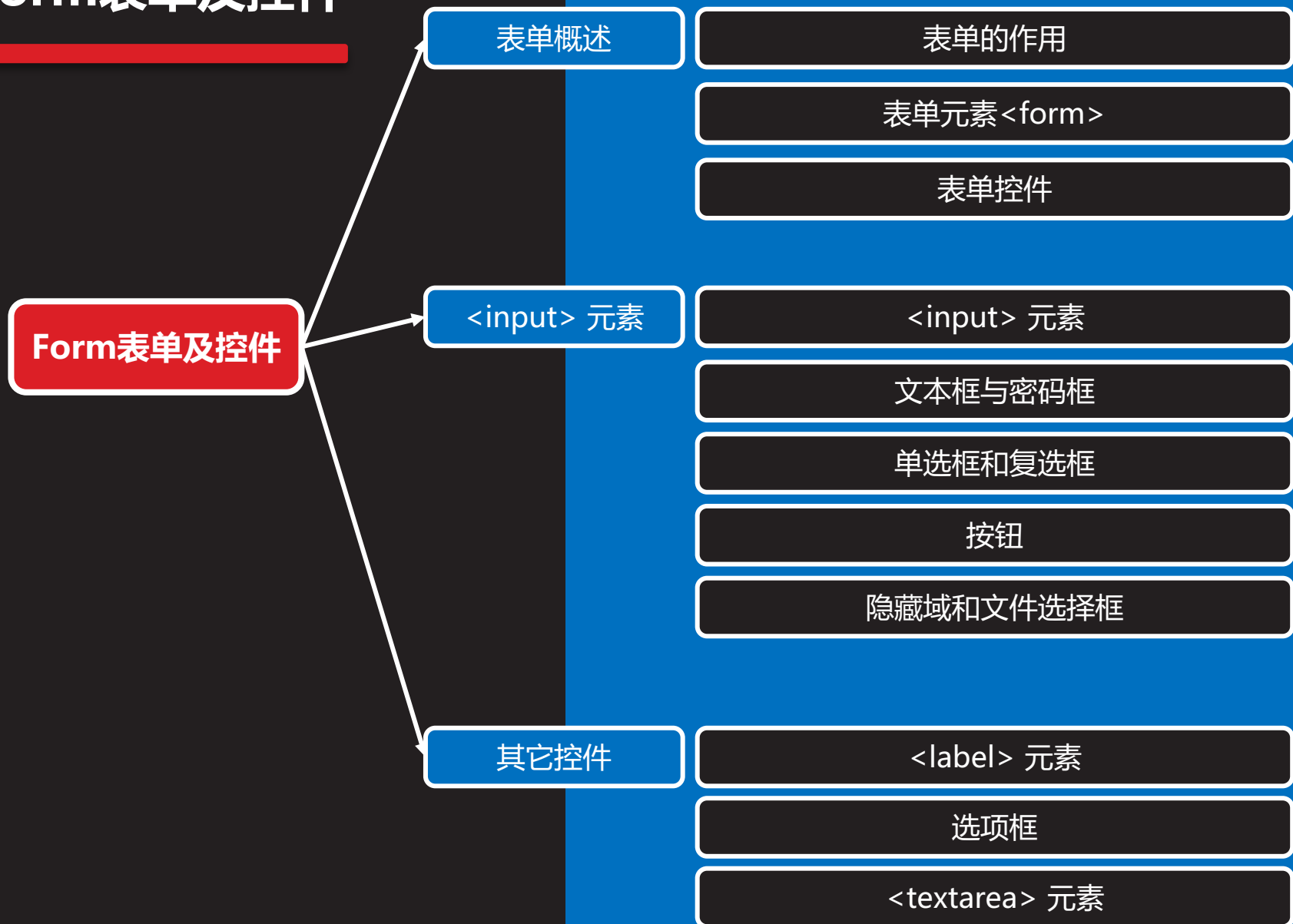


为 HTML 页面添加导航目录

- 【参见COOKBOOK】
 - 使用列表元素为 html 文档添加导航目录



Form表单及控件



表单概述

表单的作用



tarena

* 用户名 *****

* 密 码 0718

* 验证码  看不清, [换一张](#)

已有账号, 我要[登录](#)

☐ 同意服务使用协议 用户注册协议



立即注册



表单的作用（续1）

- 表单用于显示、收集信息，并提交信息到服务器
- 表单有两个基本部分
 - 实现数据交互的可见的界面元素，比如文本框或按钮
 - 提交后的表单处理
- 界面元素
 - 使用 `<form>` 元素创建表单
 - 在 `<form>` 元素中添加其他表单可以包含的控件元素



表单元素 <form>

- 定义表单：使用成对的 <form></form> 标记
- 主要属性
 - action：定义表单被提交时发生的动作，通常包含服务方脚本的URL（比如JSP、PHP）
 - method：指出表单数据提交的方式，取值为 get 或者 post
 - enctype：表单数据进行编码的方式
 - name：表单名称



表单控件

- 表单可以包含很多不同类型的表单控件
- 表单控件元素是包含在表单元素中具有可视化外观的HTML元素，用于访问者输入信息
- 表单控件元素有
 - input元素：文本输入控件、按钮、单选和复选按钮、选项框、文件选择框和隐藏控件等
 - textarea元素
 - select和option元素
 - 其他元素



<input> 元素

<input> 元素

- <input> 元素用于收集用户信息
- 该元素是一个单标记，语法为：<input />
- 主要属性
 - type：根据不同的 type 属性值，可以创建各种类型的输入字段，比如文本框、复选框等
 - value：控件的数据
 - name：控件的名称
 - disabled：禁用控件



文本框与密码框

- 文本框：`<input type = "text" />`
- 密码框：`<input type = "password" />`
- 主要属性
 - name：名称
 - value：由访问者自由输入的任何文本
 - maxlength：限制输入的字符数
 - readonly：设置文本控件只读

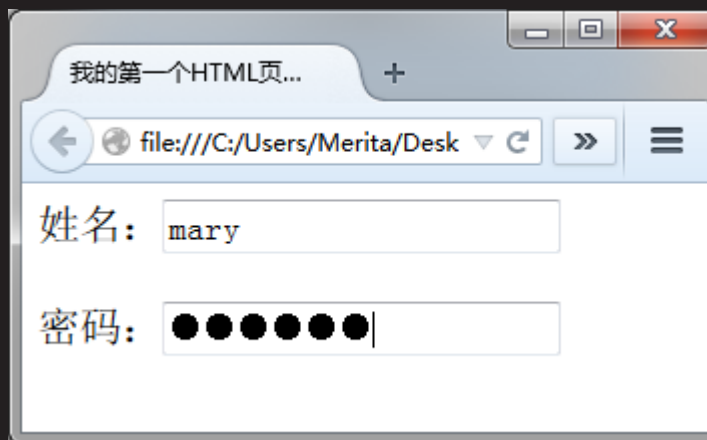


文本框与密码框（续1）

姓名：<input type="text" name="username"
value="mary" maxlength="6" />

密码：<input type="password" name="pwd" />

知识讲解



单选框和复选框

- 单选框： `<input type= "radio" />`
- 复选框： `<input type= "checkbox" />`
- 主要属性
 - name：设置名称，并用于分组，一组单选框或者复选框的名称必须相同
 - value：文本，当提交 form 时，如果选中了此单选按钮，那么 value 就被发送到服务器
 - checked：设置默认被选中



单选框和复选框（续1）

性别:

```
<input type="radio" name="sex" value="0"/>男
```

```
<input type="radio" name="sex" value="1"/>女
```

```
<input type="radio" name="sex" value="2" checked="checked"/>保密
```

```
<br /> <br />
```

喜欢的城市:

```
<input type="checkbox" name="cities" value="1"/>北京
```

```
<input type="checkbox" name="cities" value="2"/>厦门
```

```
<input type="checkbox" name="cities" value="3"/>敦煌
```

```
<input type="checkbox" name="cities" value="4" checked="checked"/>
```

杭州



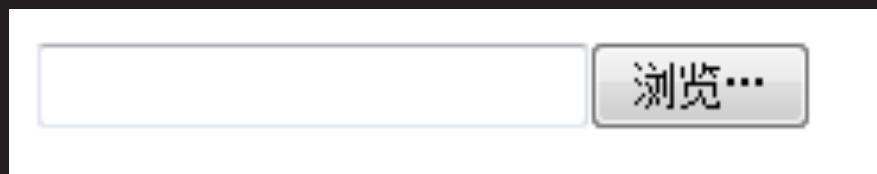
按钮

- 提交按钮：<input type= "submit" />
 - 传送表单数据给服务器端或其它程序处理
- 重置按钮：<input type= "reset" />
 - 清空表单的内容并把所有表单控件设置为最初的默认值
- 普通按钮：<input type= "button" />
 - 用于执行客户端脚本
- 主要属性
 - name：名称
 - value：按钮的标题文本



隐藏域和文件选择框

- 隐藏域： `<input type= "hidden" />`
 - 在表单中包含不希望用户看见的信息
 - name 属性：名称
 - value 属性：值
- 文件选择框： `<input type= "file" />`
 - name 属性：名称



其他控件



<label> 元素

- 语法： <label>文本</label>
- 主要属性：
 - for：表示与该元素相联系的控件的 ID 值
- 作用：
 - 将文本与控件联系在一起后，单击文本，效果就同单击控件一样

```
<input type="checkbox" name="chkHid" id="chkHid" />
<label for="chkHid">不要公开我的信息</label>
```



选项框

- 两种：下拉选项框和滚动列表
- `<select>`：创建选项框
 - name：选项框命名
 - size：大于 1，则为滚动列表
 - multiple：设置多选
- `<option>`：选项
 - value：选项的值
 - selected：预选中



选项框（续1）

选择课程：

<select>

<option value="1">Java</option>

<option value="2">C++</option>

<option value="3">PHP</option>

</select>

选择课程：

选择课程：

<select size="4">

<option value="1">Java</option>

<option value="2">C++</option>

<option value="3">PHP</option>

</select>

选择课程：

Java	▲
C++	
PHP	
	▼

<textarea> 元素

- 多行文本输入框
 - <textarea>文本</textarea>
- 主要属性：
 - name：名称
 - cols：指定文本区域的列数
 - rows：指定文本区域的行数
 - readonly：只读

多行文本框：

line 1

line 2

多行文本框：

<textarea name="txtInfo" rows="4" cols="20"></textarea>

创建表单

- 【参见 COOKBOOK】



CSS概述

CSS 概述

CSS 概述

CSS 的作用

什么是 CSS

CSS 与 HTML 之间的关系

HTML 属性与 CSS 样式的使用原则

使用 CSS 样式表

使用 CSS 样式表的方式

内联方式使用 CSS

内部样式表

外部样式表

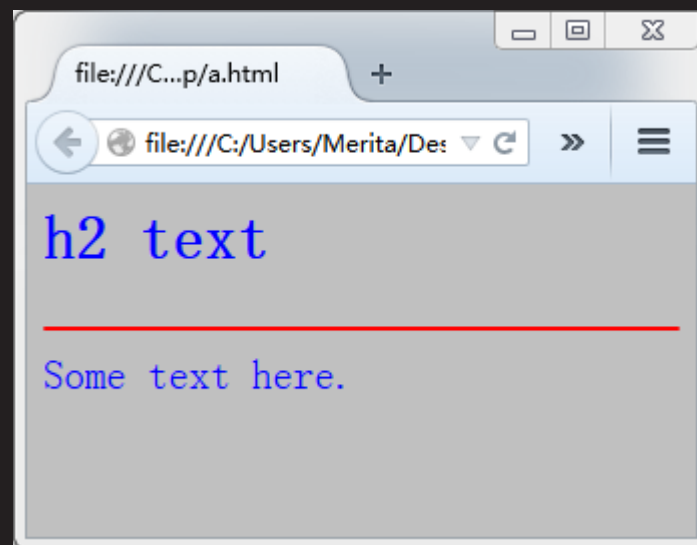
CSS 概述



CSS 的作用

- 查看如下代码

```
<body bgcolor="silver" text="blue">
  <h2>h2 text</h2>
  <hr color="red" />
  Some text here.
</body>
```



如果使用 HTML 标记中的属性来定义样式，各元素对于样式的属性定义各不相同！



CSS 的作用（续1）

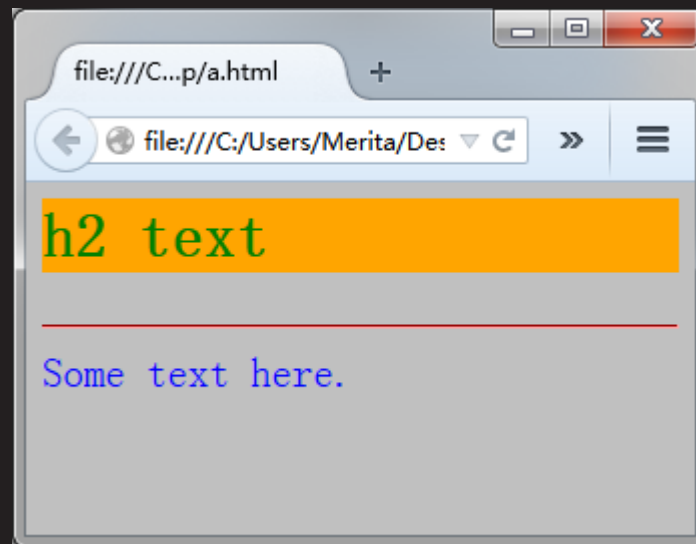
```
<html>
  <head>
    <style type="text/css">
      body
      {
        background-color:silver;
        color:blue;
      }
      h2
      {
        background-color:orange;
        color:green;
      }
      hr { color:red; }
    </style>
  </head>
```

使用 CSS 定义样式



CSS 的作用（续2）

```
<body>
  <h2>h2 text</h2>
  <hr />
  Some text here.
</body>
</html>
```



使用 CSS：

各元素使用统一的样式声明！

且提高了样式的可重用性和可维护性！



什么是 CSS

- CSS (Cascading Style Sheets) : 层叠样式表, 又叫级联样式表, 简称样式表
- 用于 HTML 文档中元素的样式定义
 - 实现了将内容与表现分离
 - 提高代码的可重用性和可维护性



CSS 与 HTML 之间的关系

- HTML 用于构建网页的结构
- CSS 用于构建 HTML 元素的样式
- HTML 是页面的内容组成，CSS 是页面的表现

```
<style type="text/css">
```

```
p{
```

```
  color:red;
```

```
  font-size:14px;
```

```
  text-align:center;
```

```
}
```

```
</style>
```

```
<p>段落1</p>
```

```
<p>段落2</p>
```

```
<p>段落3</p>
```

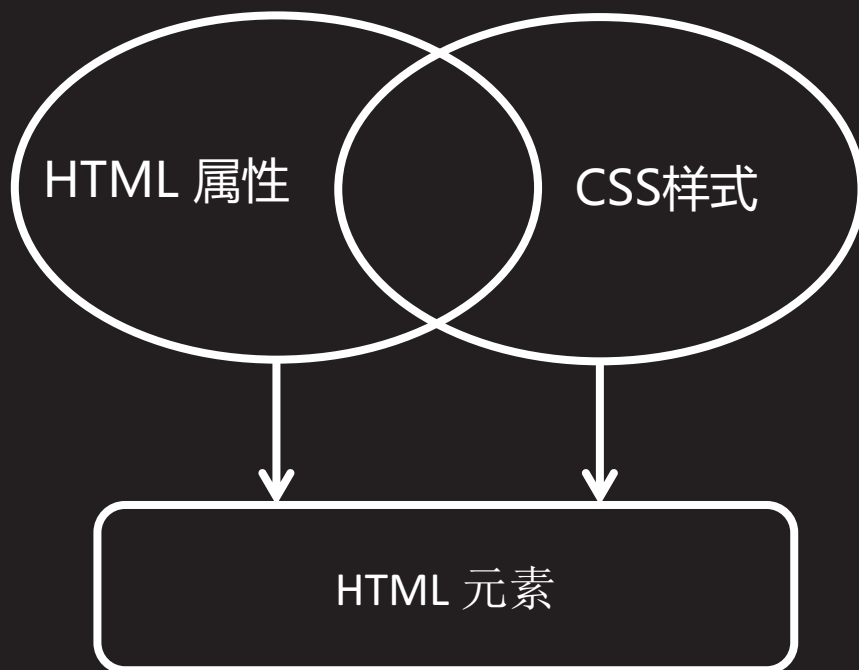
表现

内容



HTML 属性与 CSS 样式的使用原则

- W3C 建议尽量使用 CSS 样式取代 HTML 属性
 - 实现内容和表现的分离
 - 如果为 HTML 所特有的属性，则使用 HTML 属性



使用 CSS 样式表

使用 CSS 样式表的方式

- 内联方式
 - 样式定义在单个的 HTML 元素中
- 内部样式表
 - 样式定义在 HTML 页的头元素中
- 外部样式表
 - 将样式定义在一个外部的 CSS 文件中 (.css 文件)
 - 由 HTML 页面引用样式表文件



内联方式使用 CSS

- 样式定义在 HTML 元素的标准属性 style 里
- CSS 语法
 - 只需要将分号隔开的一个或者多个属性/值对作为元素的 style 属性的值
 - 属性和属性值之间用：连接
 - 多对属性之间用；隔开

```
<h1 style = "background-color : silver ; color : blue ;" >  
    文本  
</h1>
```

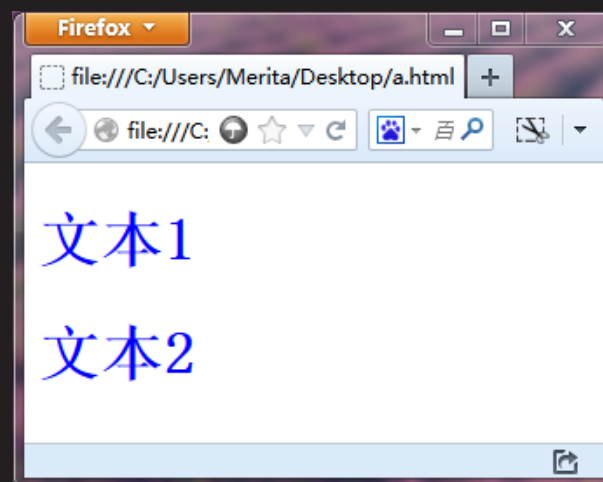


内部样式表

- 样式表规则位于文档头元素中的 `<style>` 元素内
 - 在文档的 `<head>` 元素内添加 `<style>` 元素
 - 在 `<style>` 元素中添加样式规则

```
<html>
  <head>
    <style type="text/css">
      h1 { color : blue ; }
    </style>
  </head>
  <body>
    <h1>文本1</h1>
    <h1>文本2</h1>
  </body>
</html>
```

可以在当前页面范围内重用



内部样式表（续1）

- 在 `<style>` 元素中添加样式规则
 - 可以定义多个样式规则
 - 每个样式规则有两个部分：选择器和样式声明
- 选择器：决定哪些元素使用这些规则
- 样式声明：一对大括号，包含一个或者多个属性/值对

`<head>`

`<style type="text/css">`

选择器

`h1`

`{`

`color : blue ;`

样式声明

`}`

`</style>`

`</head>`

外部样式表

- 第一步：创建一个单独的样式表文件用来保存样式规则
 - 一个纯文本文件，文件后缀为 .css
 - 该文件中只能包含样式规则
 - 样式规则由选择器和样式声明组成



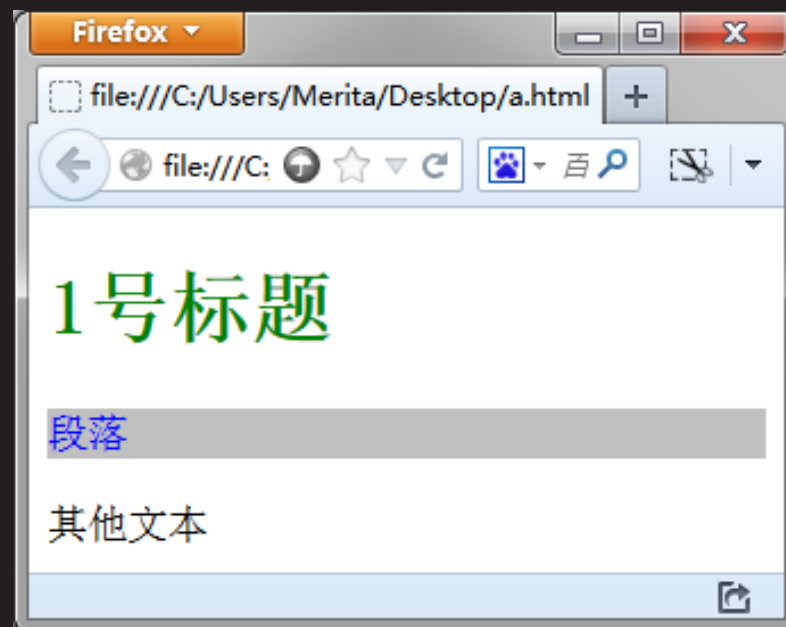
CSS 文件：
myStyle.css

外部样式表（续1）

- 第二步：在需要使用该样式表文件的页面上，使用 `<link>` 元素链接需要的外部样式表文件
 - 在文档的 `<head>` 元素内添加 `<link>` 元素

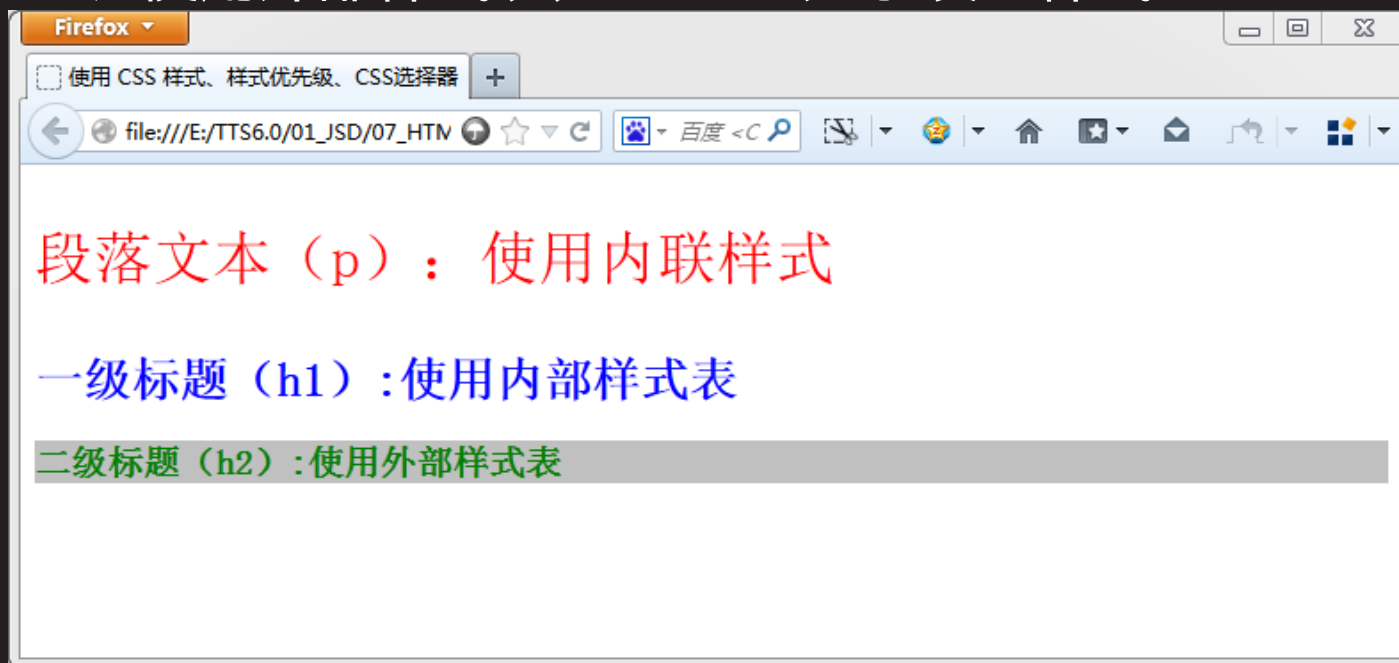
```
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="myStyle.css" />
  </head>
  <body>
    <h1>1号标题</h1>
    <p>段落</p>
    <span>其他文本</span>
  </body>
</html>
```

可以实现内容与表现分离
可以被站点中的所有页面重用



使用 CSS 样式

- 【参见 COOKBOOK】
 - 1、使用内联样式为 <p> 元素设置样式
 - 2、使用内部样式表为 <h1> 元素设置样式
 - 3、使用外部样式表为 <h2> 元素设置样式



总结和答疑
