

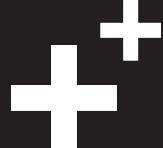
Python开发

NSD DEVWEB

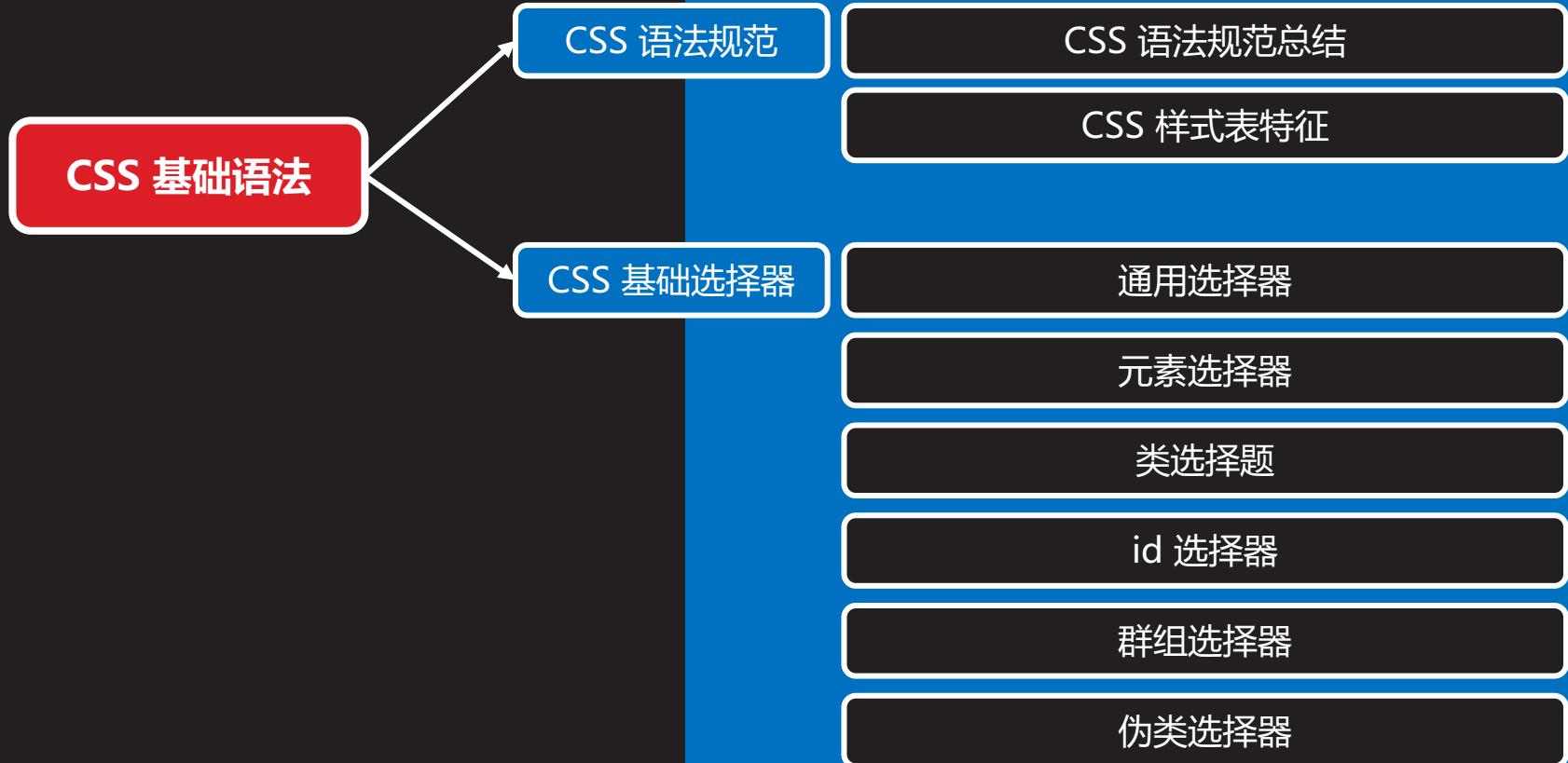
DAY02

内容

上午	09:00 ~ 09:30	作业讲解和回顾
	09:30 ~ 10:20	CSS 基础语法
	10:30 ~ 11:20	尺寸与框模型
	11:30 ~ 12:00	常用样式属性
下午	14:00 ~ 14:50	Bootstrap 起步
	15:00 ~ 15:50	全局 CSS 样式和组件
	16:10 ~ 17:00	
	17:10 ~ 18:00	总结和答疑



CSS 基础语法



CSS 语法规范

CSS 语法规规范总结

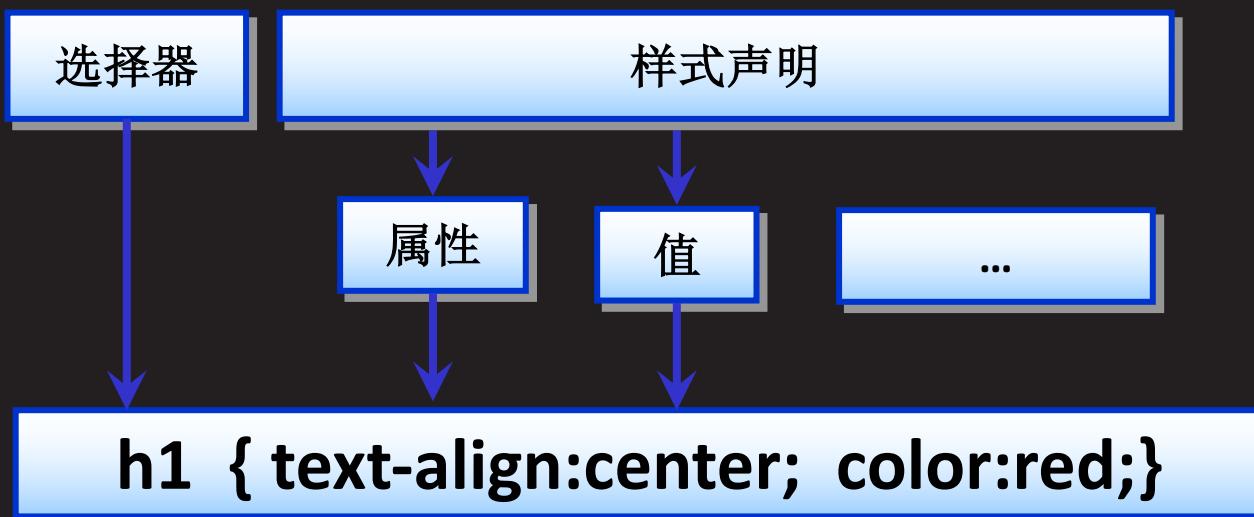
- 内联样式：由样式声明组成

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

- 样式表（内部样式表或者外部样式表）

- 由多个样式规则组成

- 每个样式规则有两个部分：选择器和样式声明



CSS 样式表特征

- 继承性
 - 大多数 CSS 的样式规则可以被继承
- 层叠性
 - 可以定义多个样式
 - 不冲突时，多个样式表中的样式可层叠为一个
- 优先级
 - 样式定义冲突时，按照不同样式规则的优先级来应用样式

CSS 基础选择器

通用选择器

- 通用选择器，显示为一个星号（ * ）
 - 可以与任何元素匹配
 - 常用于设置一些默认样式，比如设置整个文档的文本的默认字体和大小

```
*  
{  
    font-size : 9pt;  
    font-family : "Times New Roman";  
}
```

元素选择器

- html 文档的元素就是选择器
 - 比如 <p>、<h1> 等

```
html
{
    color : black ;
}
h1
{
    color : blue ;
}
h2
{
    color : silver ;
}
```

类选择器

- 语法为 : .className { color:red; }
 - 类名称不能以数字开头
- 所有能够附带class属性的元素都可以使用此样式声明
 - 将元素的 class属性的值设置为样式类名

样式表中 : 声明一个样式类

```
.myClass
{
    background-color : Pink;
    font-size : 35pt;
}
```

html 文档中 : 将元素的 class 属性的值设置为样式类的名称

```
<h2 class="myClass">h2文本</h2>
<p class="myClass">段落文本</p>
```



类选择器（续1）

- 可以将多个类选择器应用于同一个元素（多类选择器）
 - HTML 元素的 class 属性的值中可能包含一个词列表
 - 各个词之间用空格分隔，每个词都是一个类选择器

知识讲解

样式表中：声明两个样式类

```
.important {  
    font-weight : bold;  
}  
.warning {  
    color : red;  
}
```

html 文档中：将元素的 class 属性的值设置为多个样式类名的列表

```
<p class="warning">警告</p>  
<p class="important">重要</p>  
<p class="important warning">哈哈</p>
```



类选择器（续2）

- 可以将类选择器和元素选择器结合起来使用，以实现对某种元素中不同样式的细分控制（分类选择器）
- 语法为：元素选择器 .className { }
 - 先声明一个元素选择器
 - 然后使用一个点号（.）代表将使用类选择器
 - 然后声明一个类的名称
 - 最后使用一对大括号声明样式规则
- 将样式规则与附带 class 属性的某种元素匹配
 - 元素的 class 属性的值为分类选择器中指定的样式类名



类选择器（续3）

样式表中：定义一个分类选择器

```
p.important  
{  
    color : red ;  
    font-size : 20pt;  
    border:1px solid black;  
}
```

html 文档中：将元素的 class 属性的值设置为样式类的名称

```
<h2 class="important">h2文本</h2>  
<p class="important">段落文本</p>
```



只有 <p> 元素使用样式



id 选择器

- id 选择器以一种独立于文档元素的方式来指定样式
- 它仅作用于 id 属性的值
- 语法为：
 - 选择器前面需要有一个 # 号
 - 选择器本身则为文档中某个元素的 id 属性的值



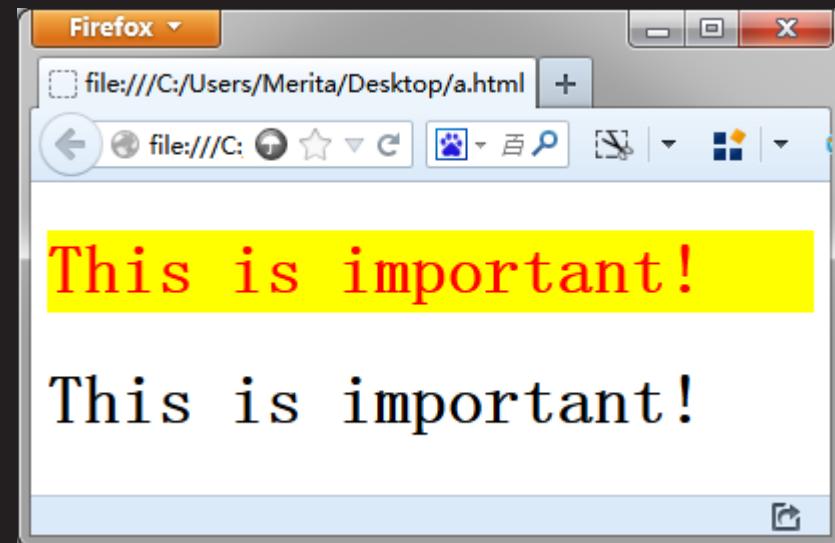
id 选择器 (续1)

样式表中 : 定义一个 id 选择器

```
#mostImportant
{
    color:red;
    background:yellow;
}
```

html 文档中 : 将元素的 id 属性的值设置为选择器的名称

```
<h1 id="mostImportant">This is important!</h1>
<h1>This is important!</h1>
```



群组选择器

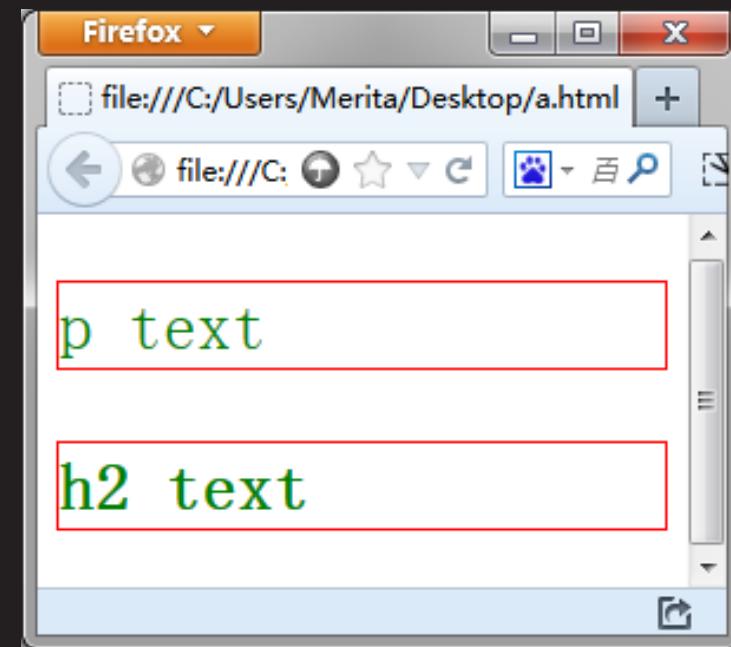
- 选择器声明为以逗号隔开的选择器列表
 - 将一些相同的规则作用于多个元素

样式表中：定义选择器分组

```
h2, p.important
{
    color:green;
    font-size:20pt;
    border:1px solid red;
}
```

html 文档中：

```
<p class="important">p text</p>
<h2>h2 text</h2>
```



伪类选择器

- 伪类用于向某些选择器添加特殊的效果
- 使用冒号(:)作为结合符，结合符左边是其他选择器，右边是伪类
 - 选择器：伪类选择器
- 链接伪类
 - :link，适用于尚未访问的链接
 - :visited，适用于访问过的链接
- 动态伪类，用于呈现用户操作
 - :hover，适用于鼠标悬停在 HTML 元素时
 - :active，适用于 HTML 元素被激活时
 - :focus，适用于 HTML 元素获取焦点时



伪类选择器（续1）

样式表中：定义伪类选择器

```
a:link {  
    color: black;  
    font-size:15pt;  
}  
a:visited {  
    color: pink;  
    font-size:15pt;  
}  
a:hover {  
    font-size : 20pt;  
}  
a:active {  
    color : red;  
}
```

知识讲解

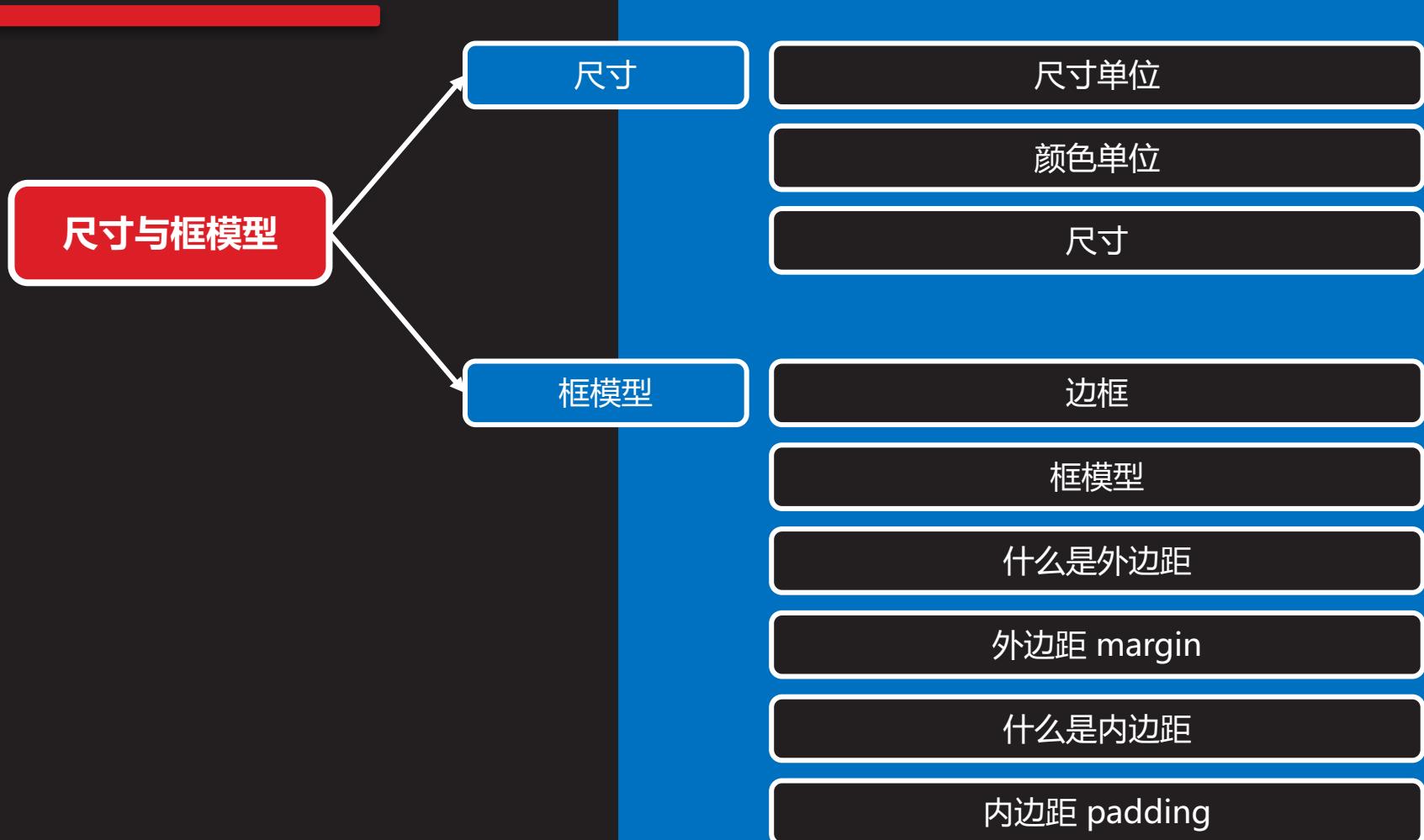


CSS 基础选择器

- 【参见 COOKBOOK】
 - 使用 CSS 基础选择器为页面元素设置样式



尺寸与框模型



尺寸



尺寸单位

- % : 百分比
- in : 英寸
- cm : 厘米
- mm : 毫米
- pt : 磅(1pt等于1/72英寸)
- px : 像素(计算机屏幕上的一一个点)
- em : 1em等于当前的字体尺寸，2em等于当前字体尺寸的两倍，继承父级元素的字体大小
- rem : 与em类似，但是相对于根元素设置字体尺寸的倍数



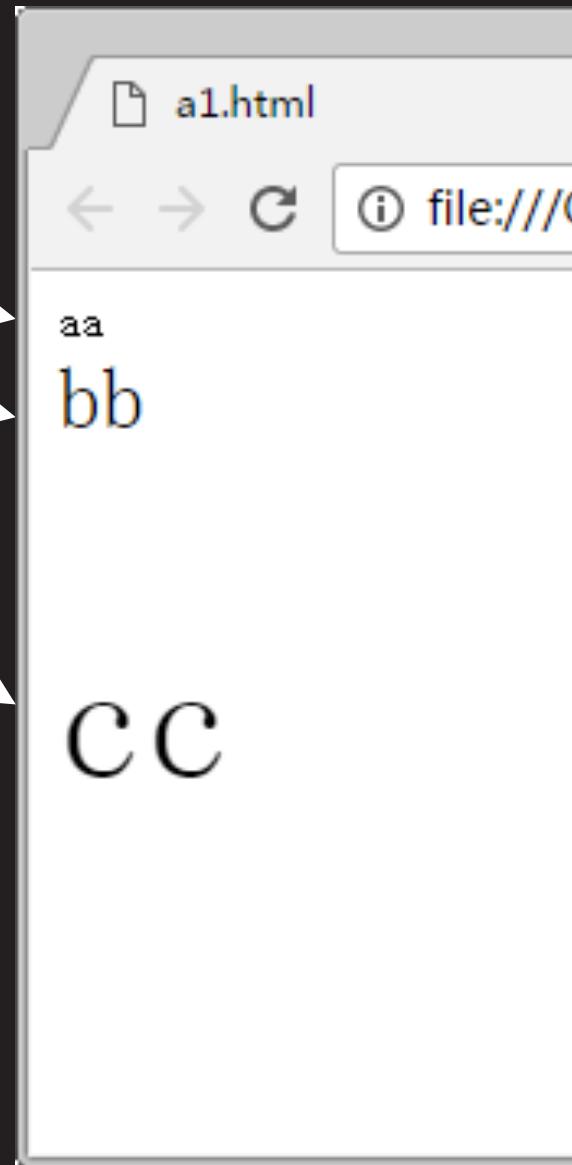
尺寸单位（续1）

样式表中：

```
Html {font-size:10pt;}  
div {font-size:2rem;}  
p {font-size:2em;}
```

HTML文档中：

```
<body>  
    aa  
    <div>  
        bb  
        <p>cc</p>  
    </div>  
</body>
```



颜色单位

- `rgb(x,x,x)` : RGB 值，如 `rgb(255,0,0)`
- `rgb(x%,x%,x%)` : RGB百分比值，如`rgb(100%,0%,0%)`
- `#rrggbb` : 十六进制数，如`#ff0000`
- `#rgb` : 简写的十六进制数，如`#f00`
- 表示颜色的英文单词，如 `red`



尺寸

- 尺寸属性是用于设置元素的宽度和高度
 - 单位一般为像素或百分比
- 宽度属性
 - width
 - max-width
 - min-width
- 高度属性
 - height
 - max-height
 - min-height



框模型



边框

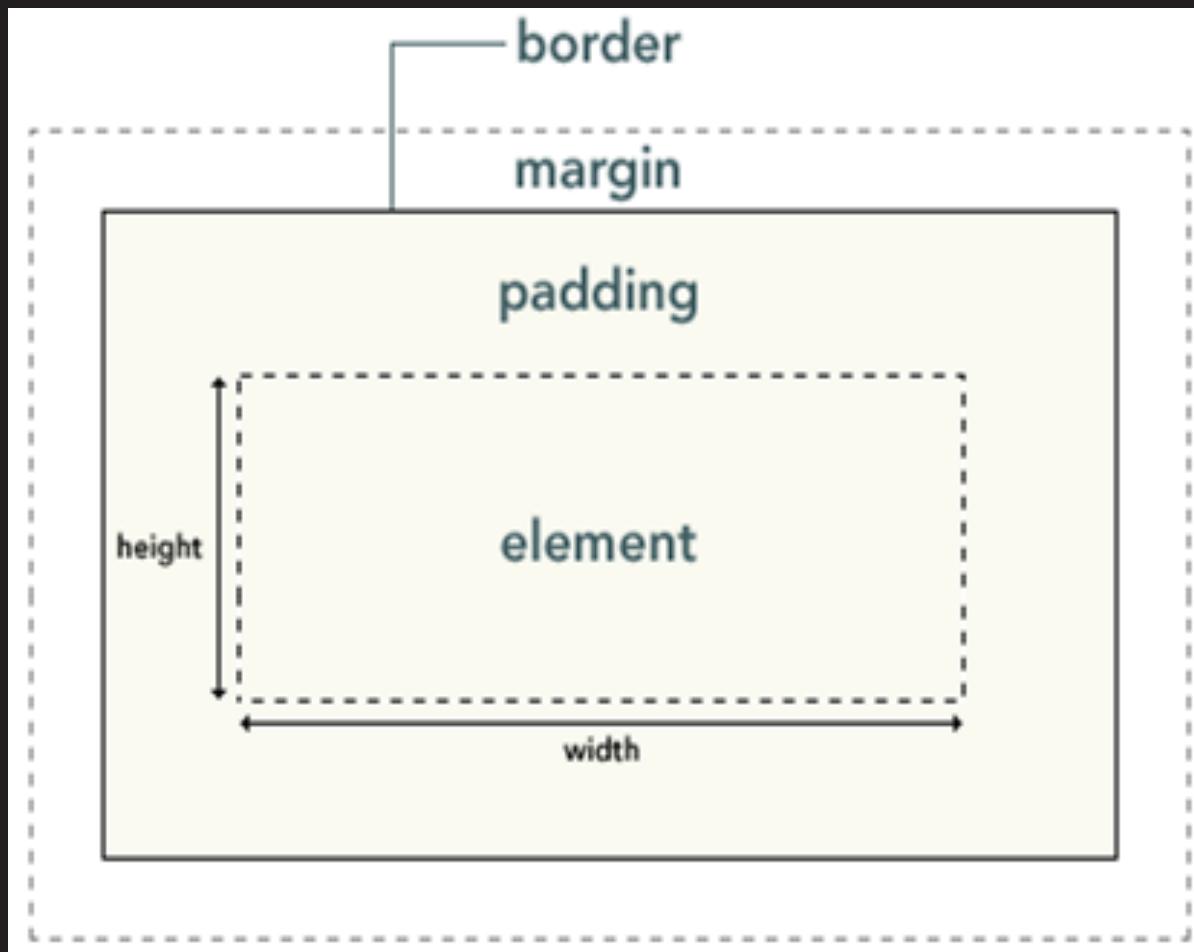
- 简写方式
 - border:width style color;
- 单边定义
 - border-left/right/top/bottom:width style color;
- border-width
 - border-left/right/top/bottom-width
- border-style
 - border-left/right/top/bottom-style
- border-color
 - border-left/right/top/bottom-color

边框颜色，可设置为值 transparent (用于创建有宽度的不可见边框)



框模型

- 框模型（Box Model）定义了元素框处理元素内容、内边距、边框和外边距的方式

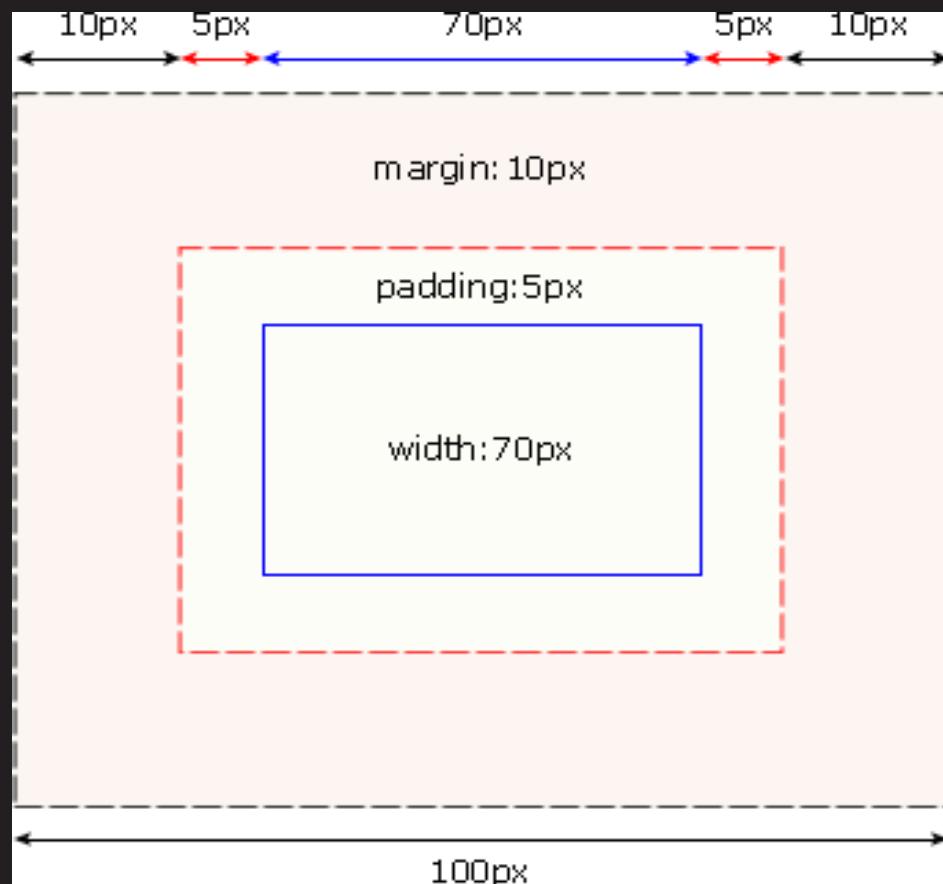


框模型（续1）

- width 和 height 指内容区域的宽度和高度
- 增加内边距、边框和外边距不会影响内容区域的尺寸，但是会增加元素框的总尺寸

```
#box
{
    width: 70px;
    margin: 10px;
    padding: 5px;
}
```

对象实际宽度 = 左侧外边距 +
左侧边框 + 左侧内边距 + 宽度 +
右侧内边距 + 右侧边框 +
右侧外边距



框模型（续2）

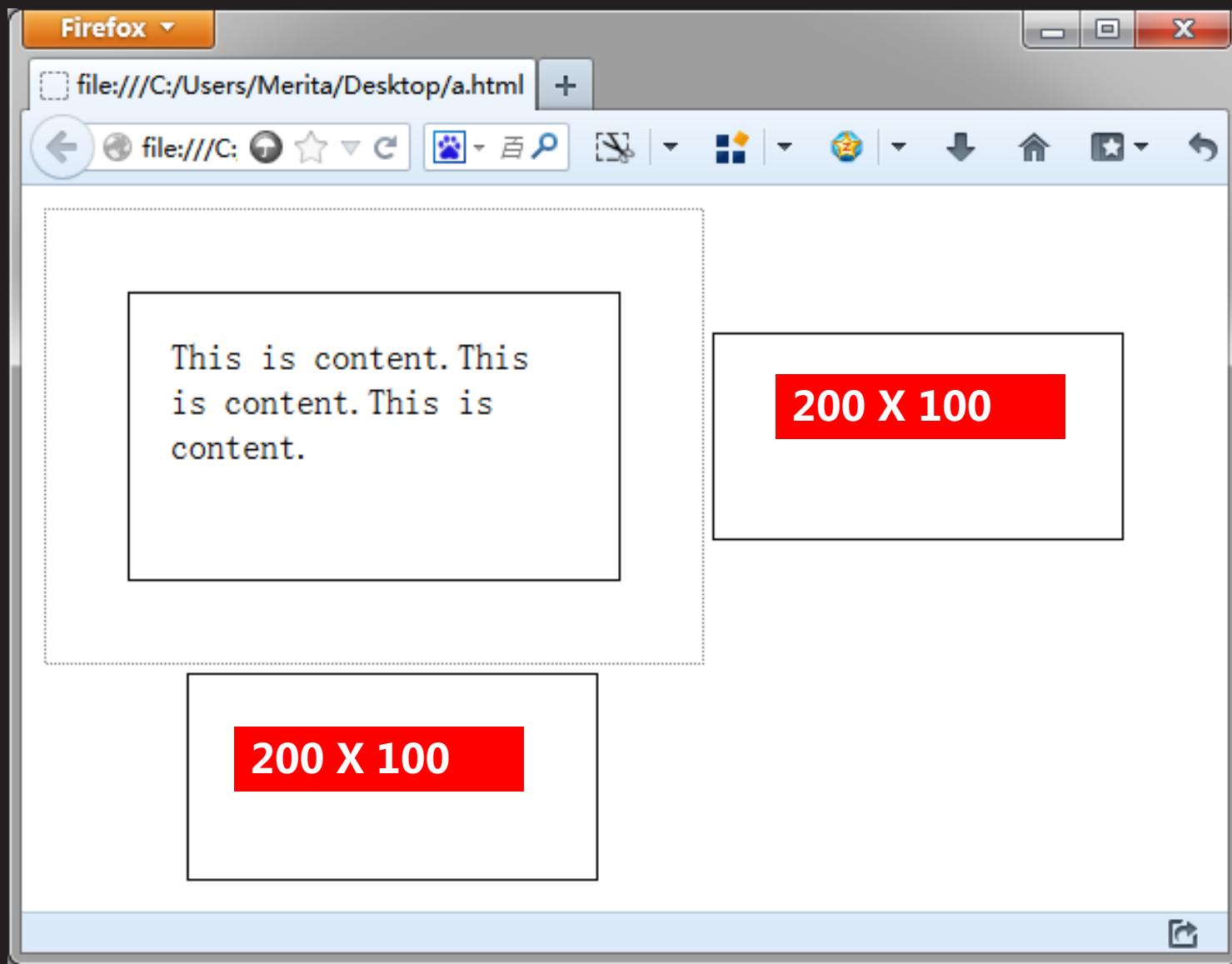
样式表中：

```
div.box
{
    width:200px;
    height:100px;
    padding:20px;
    margin:40px;
    border:1px solid black;
}
```

html 文档中：

```
<div class="box">
    This is content.
    This is content.This is content.
</div>
```

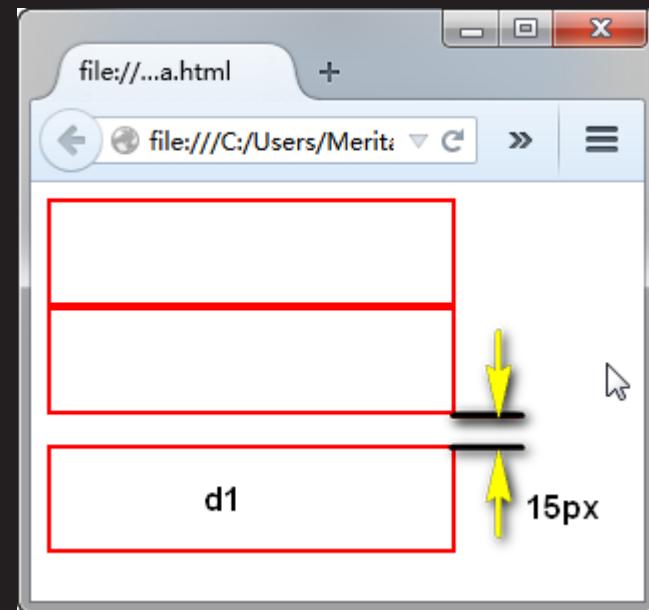
框模型 (续3)



什么是外边距

- 围绕在元素边框周围的空白区域是外边距
 - 会在元素外创建额外的空白
 - 外边距是透明的

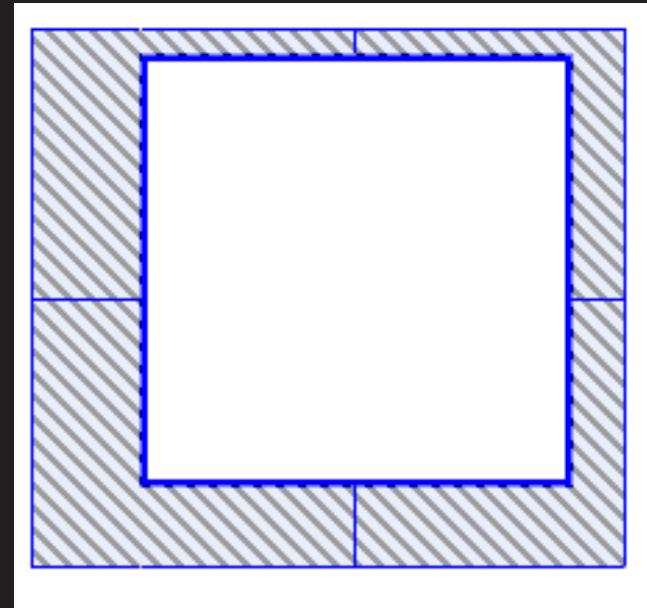
```
div
{
    border:2px solid red;
    width:200px;
    height:50px;
}
#d1 {margin-top:15px;}
```



外边距 margin

- 外边距：与下一个元素之间的空间量
 - margin : value;
- 单边设置
 - margin-top/right/bottom/left : value;
- 外边距的属性值可能为px(像素) , 百分比(%)或自动(auto) , 也可以为负值

```
div {  
    width:150px;  
    height:150px;  
    border:2px solid #0f0;  
    margin-top:10px;  
    margin-right:20px;  
    margin-bottom:30px;  
    margin-left:40px;  
}
```



外边距 margin (续1)

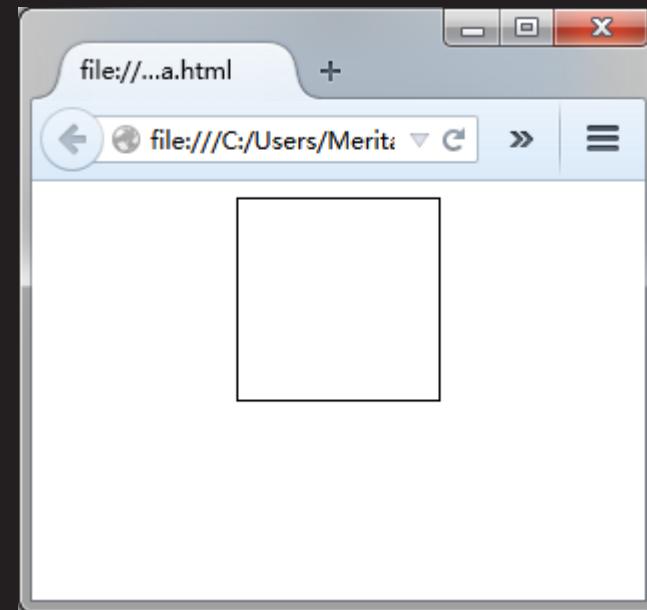
- 默认情况下，以下 HTML 块级元素都存在外边距
 - body
 - h1,h2,h3,h4,h5,h6
 - p...
- 声明 margin 属性，可以覆盖默认样式

```
body,p,div,h1,h2,h3,h4,h5,h6,pre {  
    margin:0;  
}
```

外边距 margin (续2)

- margin 可取值为 auto , 由浏览器计算外边距
 - 实现水平方向居中显示的效果

```
div
{
    width:100px;
    height:100px;
    border:1px solid black;
    margin:auto;
}
```



设置水平方向居中

什么是内边距

- 内边距：内容区域和边框之间的空间
 - 会扩大元素边框所占用的区域

```
div
{
    border:2px solid red;
    width:200px;
    height:50px;
}
#d1 {padding:20px;}
```



内边距 padding

- 内边距：内容与框线之间的距离
 - padding : value;
- 内边距属性值可以为像素、百分比，但不能为负数
- 单边设置
 - padding-top/right/bottom/left : value;

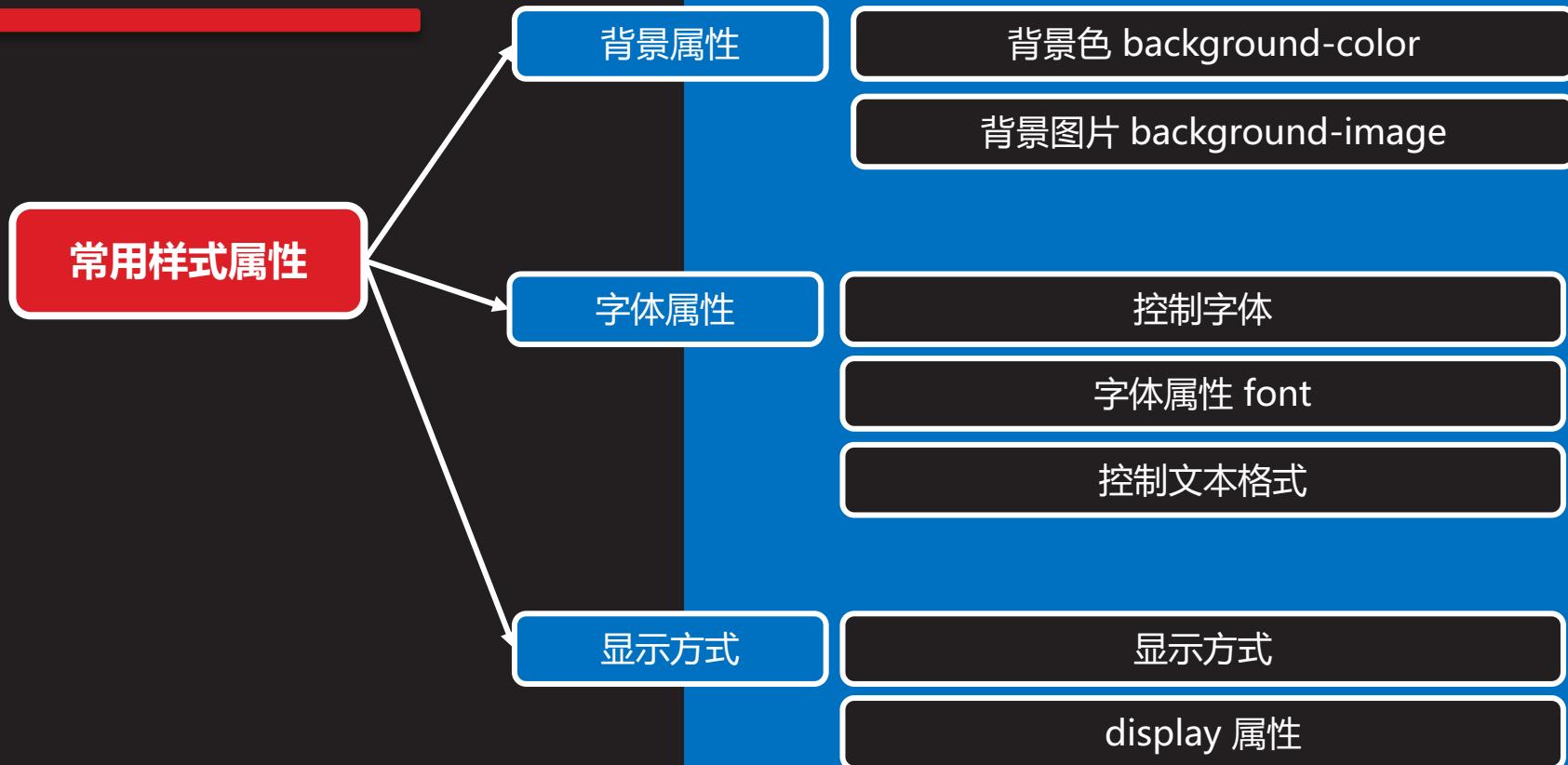


尺寸与框模型

- 使用尺寸、边框、边距属性为页面元素定义样式



常用样式属性



背景属性

背景色 background-color

- background-color 属性用于为元素设置背景色
 - 接受任何合法的颜色值
 - 可取值为 transparent
- 为元素背景设置一种纯色
 - 会填充元素的内容、内边距和边框区域
 - 如果边框有透明部分，会透过这些透明部分显示出背景色。

```
div
{
    border:2px dashed black;
    width:200px;
    height:50px;
    background-color:#ccc;
}
```



背景图片 background-image

- 默认值是 none，表示背景上没有放置任何图像
- 如果需要设置一个背景图像，需要用起始字母 url 附带一个图像的 URL 值
 - 可以是相对 URL 或者绝对 URL

```
body
{
    background-image: url("image/bg_01.gif");
}
```

字体属性

控制字体

- 指定字体
 - font-family : value1,value2;
- 字体大小
 - font-size : value;
- 字体加粗
 - font-weight : normal/bold/value;
- 字体样式
 - font-style : normal/italic;
- 小型大写字母显示
 - font-variant : normal/small-caps;



控制字体（续1）

样式表中：

```
p {  
    font-family : Times, 'New York', serif ;  
    font-size : 14pt ;  
    font-weight : bold ;  
    font-style : italic;  
    font-variant : small-caps;  
}
```

html 文档中：

```
<p>this is a paragraph.</p>
```



字体属性 font

- 字体属性 font 用于把所有针对字体的属性设置在一个声明中
- 为简写属性，包含6个值，可以按顺序设置
 - `font : font-style font-variant font-weight font-size font-family;`
 - 不设置的值，则使用默认设置



控制文本格式

- 文本颜色 color : value;
- 文本排列 text-align : left/right/center;
- 文字修饰 text-decoration : none/underline;
- 行高 line-height : value;



控制文本格式（续1）

```
div {  
    color : #FF0000 ;  
    text-align : left ;  
    text-decoration : underline ;  
    height:50px;  
    border : 1px solid black;  
}  
  
div {  
    color : #FF0000 ;  
    text-align : left ;  
    text-decoration : underline ;  
    height:50px;  
    border:1px solid black;  
    line-height:50px;  
}
```

div中的文本

<div>div中的文本</div>

div中的文本

Bootstrap起步

Bootstrap起步

基本概念

什么是Bootstrap

构成

Bootstrap目录说明

Bootstrap相关技术

基本概念



什么是Bootstrap

Bootstrap

简洁、直观、强悍的前端开发框架，让web开发更迅速、简单。

Bootstrap3中文文档(v3.3.4)



- 移动设备优先
- 所有的主流浏览器都支持 Bootstrap
- 是由动态CSS语言Less写成，Github热门开源项目

构成

起步

全局 CSS 样式

组件

JavaScript 插件

定制

下载

包含的内容

编译 CSS 和 JavaScript 文件

基本模板

实例精选

工具

社区

禁止响应式布局

从 2.x 版本升级到 3.0 版本

对浏览器和设备的支持情况

对第三方组件的支持

可访问性

许可证 FAQ

文档翻译

概览

栅格系统

排版

代码

表格

表单

按钮

图片

辅助类

响应式工具

使用 Less

使用 Sass

概览

过渡效果

模态框

下拉菜单

滚动监听

标签页

工具提示

弹出框

警告框

按钮

Collapse

Carousel

Affix

Glyphicons 字体图标

下拉菜单

按钮组

按钮式下拉菜单

输入框组

导航

导航条

路径导航

分页

标签

徽章

巨幕

页头

缩略图

警告框

进度条

媒体对象

列表组

面板



Bootstrap目录说明

- css目录 – 用于存放Bootstrap框架使用的样式文件
 - bootstrap.css文件 : Bootstrap框架的样式文件
 - bootstrap.min.css文件 : Bootstrap框架的样式压缩文件
 - bootstrap-theme.css文件 : Bootstrap框架的主题文件
- fonts目录 – 用于存放Bootstrap框架使用的字体文件
- js目录 – 用于存放Bootstrap框架使用的核心javascript文件
 - bootstrap.js文件 : Bootstrap框架的核心javascript文件
 - bootstrap.min.js文件 : Bootstrap框架的核心javascript压缩文件

Bootstrap相关技术

- 由于Bootstrap框架是基于jQuery使用的javascript代码，所以想要使用Bootstrap框架，必须要先引入jQuery文件
- 解决浏览器兼容性：
 - html5shiv.min.js文件：解决IE8浏览器支持HTML5的新元素
 - respond.min.js文件：解决IE8浏览器支持CSS Media Query



全局CSS样式

全局CSS样式

全局CSS样式

概述

CSS全局样式

布局容器

按钮

文本

列表

表单

常用组件

字体图标

下拉菜单

按钮组

警告框

全局CSS样式



概述

- HTML5文件类型
 - Bootstrap使用到的某些HTML元素和CSS属性需要将页面设置为HTML5

```
<!DOCTYPE html>
<html lang="zh-CN"> ... </html>
```

- 移动设备优先
 - 为了确保适当的绘制和触屏缩放，需要在<head>之中添加viewport元数据元素

```
<meta name="viewport"
content="width=device-width, initial-scale=1">
```



CSS全局样式

- 为 body 元素设置 background-color: #fff;
- 使用 @font-family-base、@font-size-base 和 @line-height-base 变量作为排版的基本参数
- 为所有链接设置了基本颜色 @link-color，并且当链接处于 :hover 状态时才添加下划线



布局容器

- `.container` 类用于固定宽度并支持响应式布局的容器

```
<div class="container">  
  ...  
</div>
```

- `.container-fluid` 类用于 100% 宽度，占据全部视口（viewport）的容器

```
<div class="container-fluid">  
  ...  
</div>
```



按钮

Bootstrap提供的五种颜色



Bootstrap提供的四种尺寸



文本

- 对齐
 - text-left、text-center、text-right、text-justify、text-wrap
- 大小写
 - text-lowercase、text-uppercase、text-capitalize
- 颜色
 - text-success、text-warning、text-error、text-info、text-primary

列表

- 无序列表和有序列表
 -
 -
- 列表样式
 - .list-unstyled
 - .list-inline
- 描述列表
 - <dl>
 - .dl-horizontal

表单

- 基本表单

```
<div class="form-group">
    <label for="exampleInputEmail1">Email address</label>
    <input type="email" class="form-control" id="exampleInputEmail1" placeholder="Enter email">
</div>
```

- 水平表单

```
<form class="form-horizontal">
    <div class="form-group">
        <label for="inputEmail3" class="col-sm-2 control-label">Email</label>
        <div class="col-sm-10">
            <input type="email" class="form-control" id="inputEmail3" placeholder="Email">
        </div>
    </div>
    <\n>
```

- 内联表单

```
<form class="form-inline">
    <div class="form-group">
        <label class="sr-only" for="exampleInputEmail3">Email address</label>
        <input type="email" class="form-control" id="exampleInputEmail3" placeholder="Enter email">
    </div>
    <\n>
```



表单 (续1)

- 单行输入框
 - form-control
- 多行文本域
 - form-control
- 多选和单选框
 - 内联多选和单选框
- 下拉列表
 - form-control

使用全局 CSS 样式

- 【参见 COOKBOOK】
 - 使用全局 CSS 样式定义页面

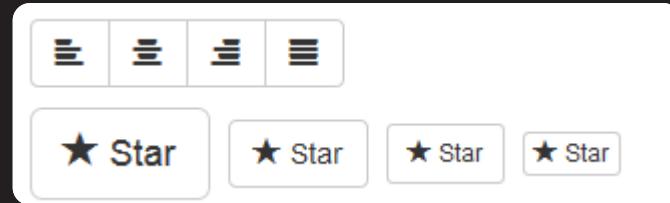


常用组件



字体图标

- Bootstrap提供了一套专用于Web开发/移动开发常用的
一套图标字体——Glyphicons Halflings
 - 基本样式类 glyphicon
 - 其他修饰类
- 注意事项
 - 引入 fonts 文件
 - 不要和其他组件混合使用
 - 只对内容为空的元素起作用



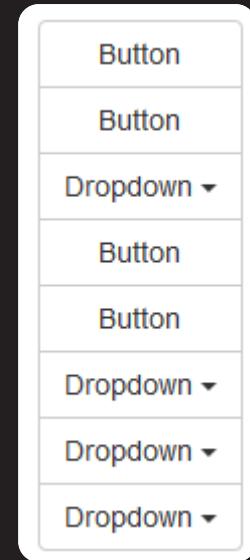
下拉菜单

- 基本格式
 - <div class="dropdown | dropup">
 <button data-toggle="dropdown">
 <ul class="dropdown-menu">
- 标题
 - <li class="dropdown-header">
- 分割线
 - <li class="divider">
- 禁用
 - <li class="disabled">



按钮组

- 基本按钮组
 - <div class="btn-group">
- 按钮工具栏
 - <div class="btn-toolbar">
- 尺寸
 - .btn-group-lg | sm | xs
- 自适应
- 垂直排列
 - <div class="btn-group-vertical">



警告框

```
<div class="alert alert-success" role="alert">
```

Well done! You successfully read this important alert message.

```
<div class="alert alert-info" role="alert">
```

Heads up! This alert needs your attention, but it's not super important.

```
<div class="alert alert-warning" role="alert">
```

Warning! Better check yourself, you're not looking too good.

```
<div class="alert alert-danger" role="alert">
```

Oh snap! Change a few things up and try submitting again.



使用常用组件

- 【参见 COOKBOOK】
 - 使用常用组件为页面添加元素



总结和答疑

