

网页编程基础 CSS3基础

WebBasic CSS3BASIC Unit02

内容

上午	09:00 ~ 09:30	作业讲解和回顾
	09:30 ~ 10:20	框模型
	10:30 ~ 11:20	
	11:30 ~ 12:00	背景
下午	14:00 ~ 14:50	
	15:00 ~ 15:50	渐变
	16:00 ~ 16:50	
	17:00 ~ 17:30	总结和答疑



框模型

框模型

框模型

框模型

外边距

什么是外边距

外边距 margin

外边距的简洁写法

外边距合并

内边距

什么是内边距

内边距 padding

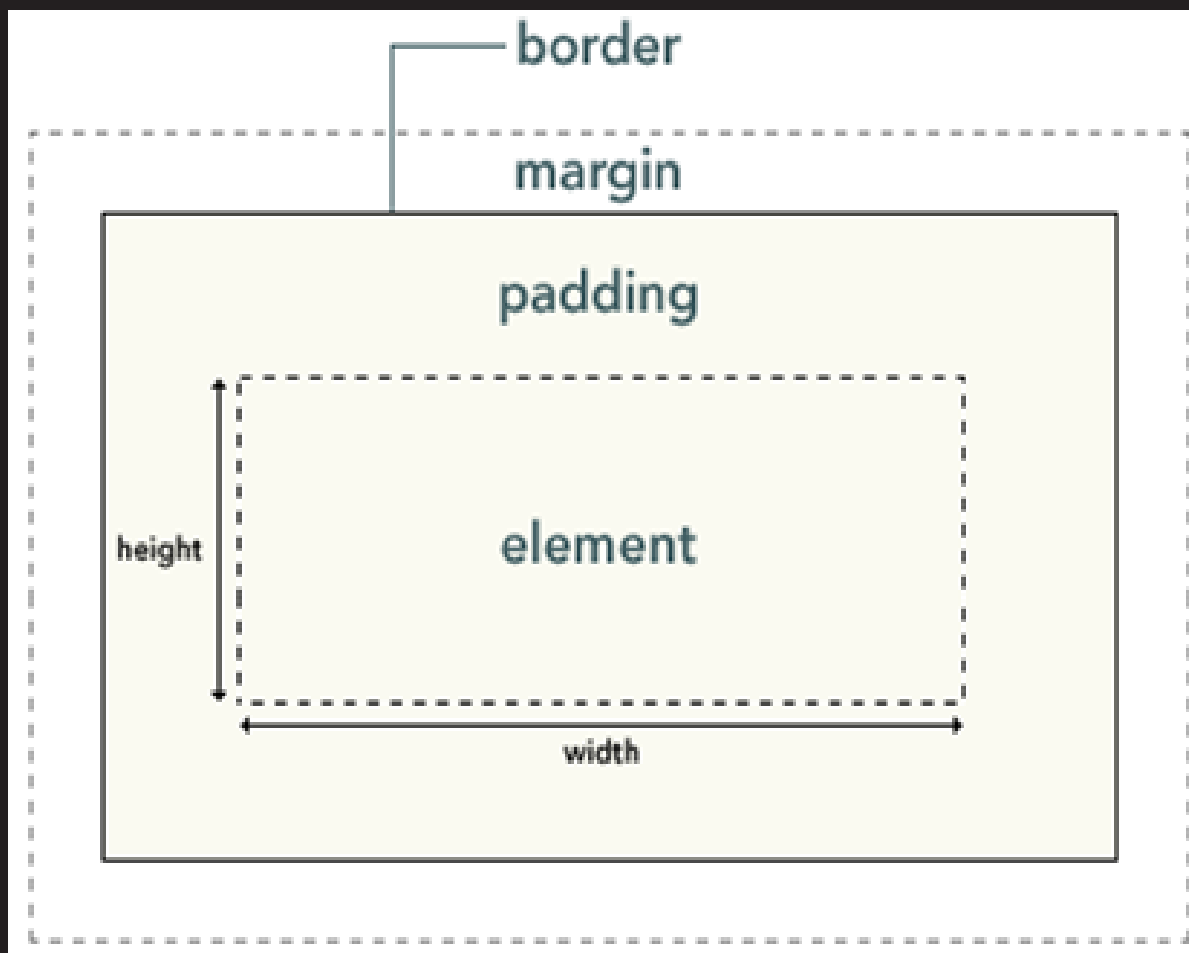
内边距的简洁写法

框模型



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

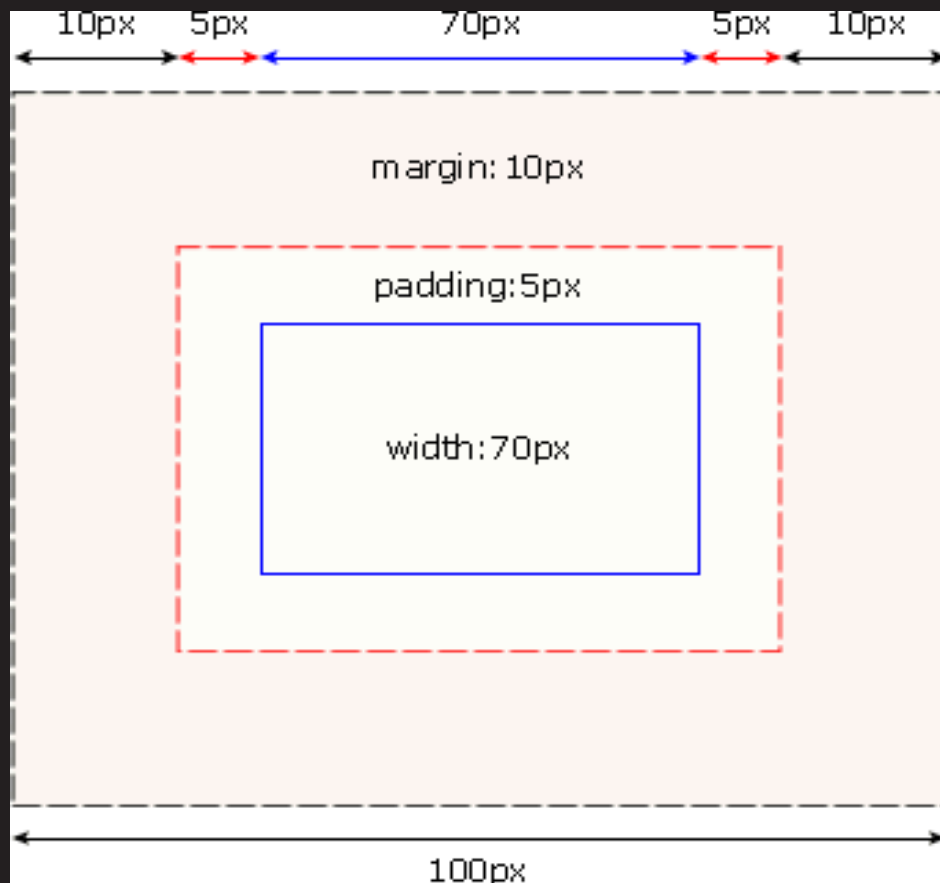
#box

{

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：

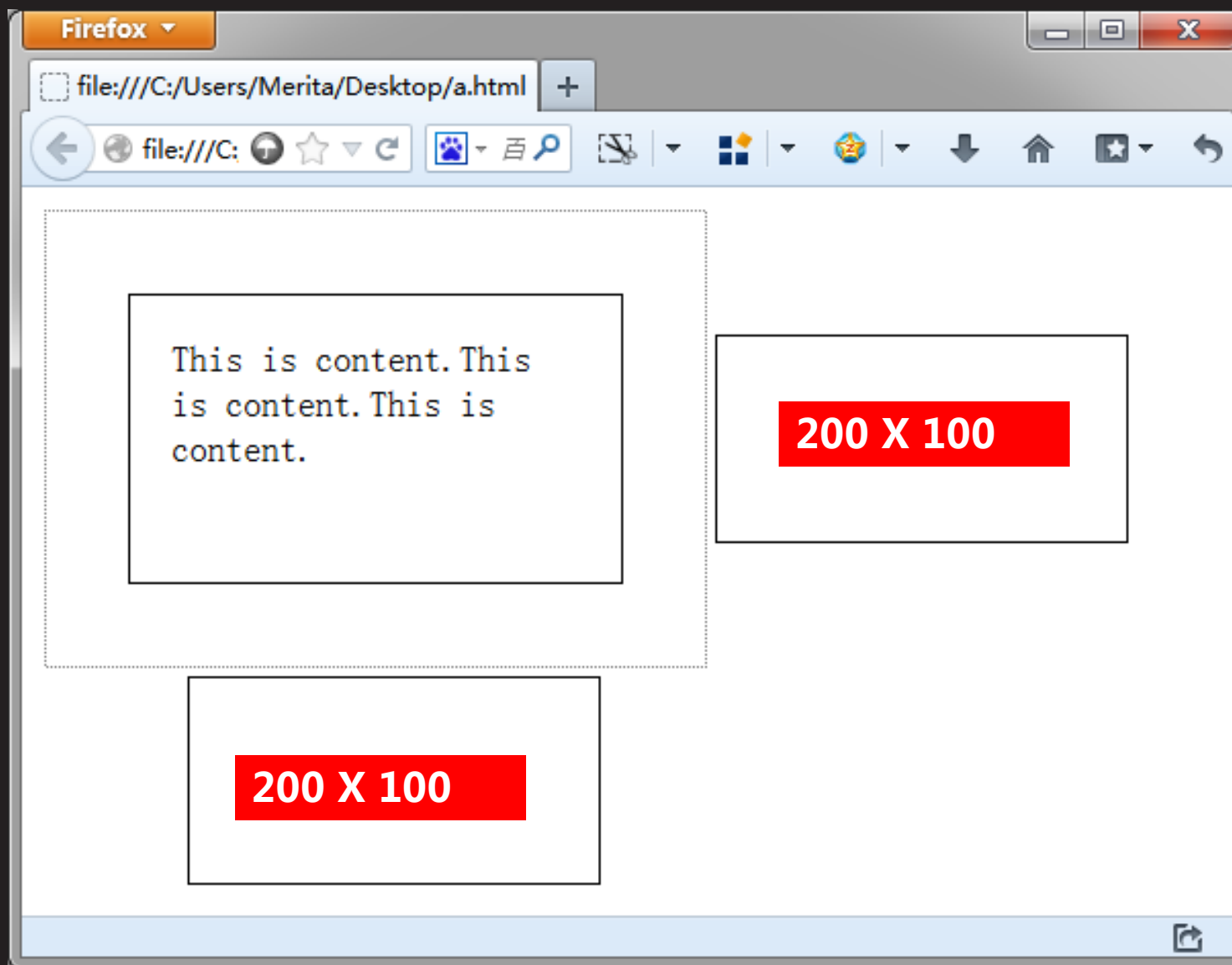
```
div.box
{
    width:200px;
    height:100px;
    padding:20px;
    margin:40px;
    border:1px solid black;
}
```

html 文档中：

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



框模型（续3）



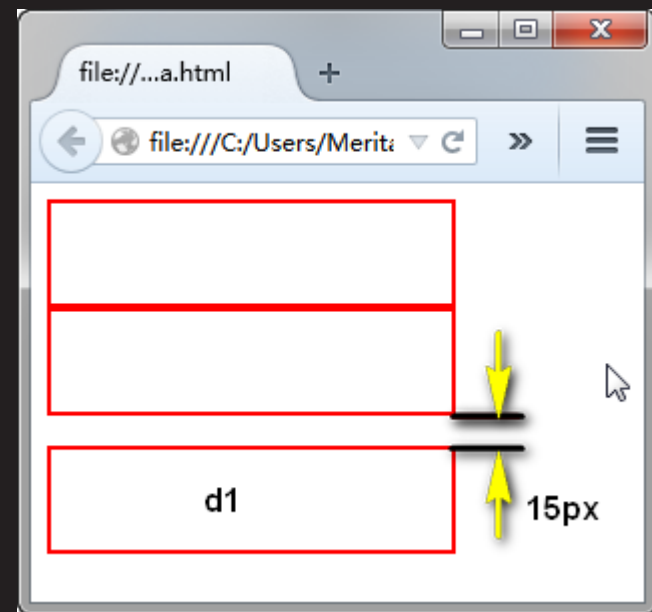
外边距



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

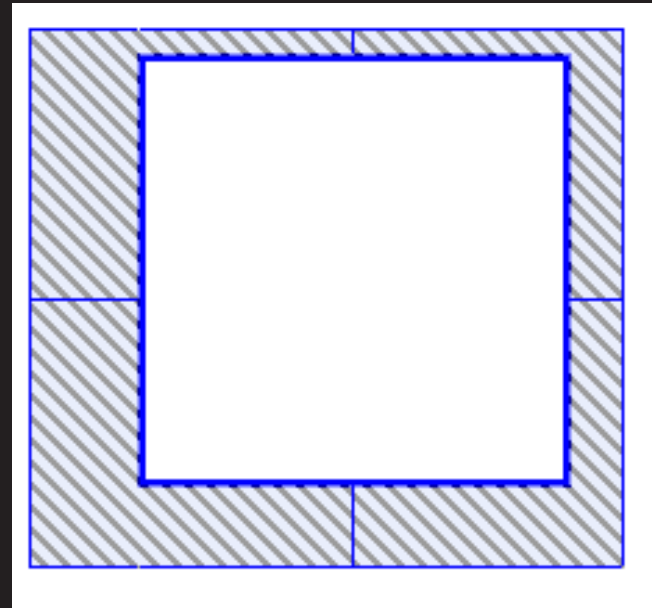
```
div
{
    border:2px solid red;
    width:200px;
    height:50px;
}
#d1 {margin-top:15px;}
```



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

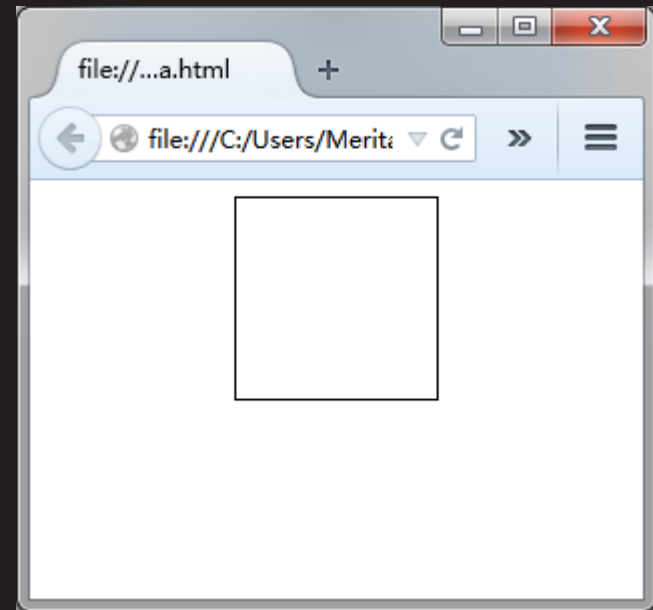
```
body,p,div,h1,h2,h3,h4,h5,h6,pre {
    margin:0;
}
```



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;
}
```

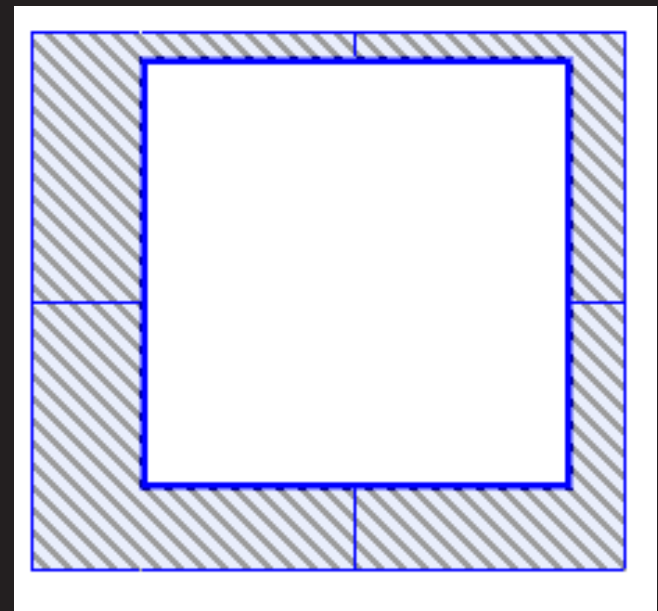


设置水平方向居中

外边距的简洁写法

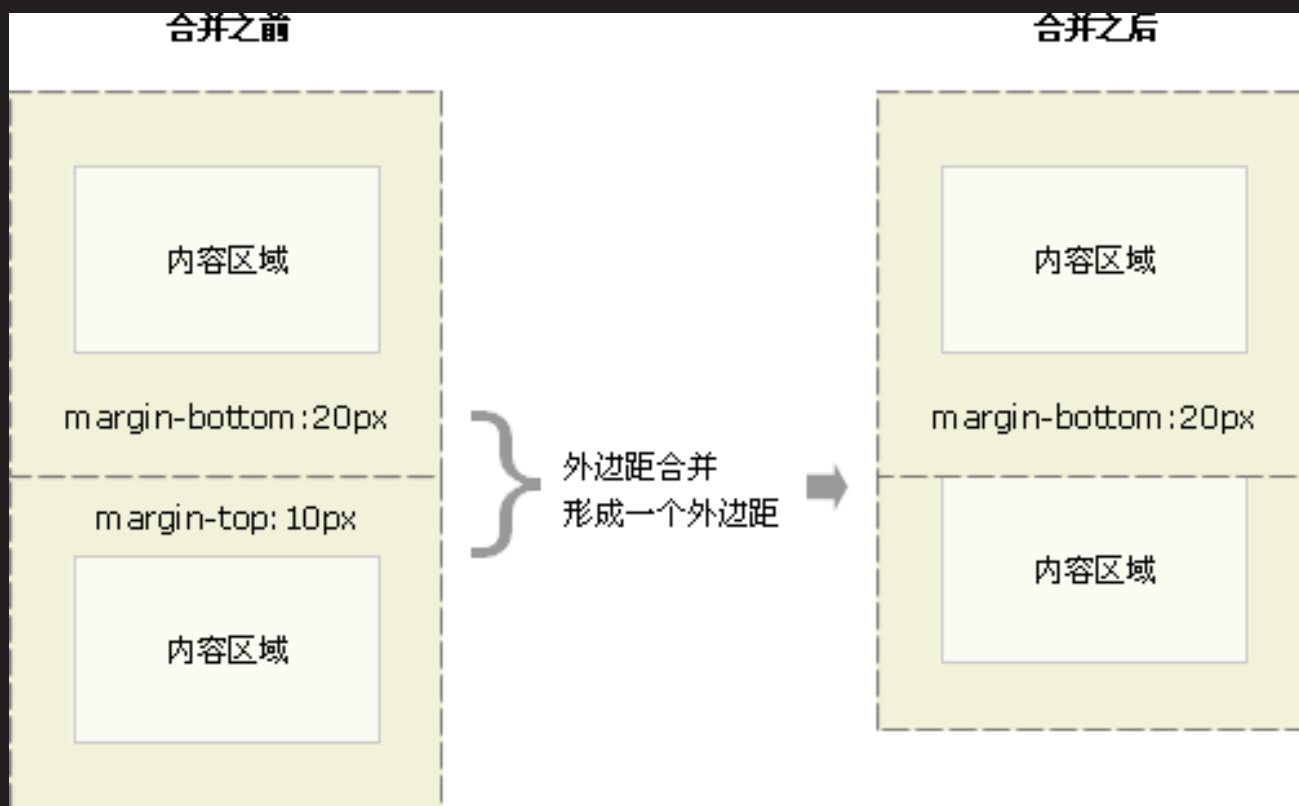
- 外边距的简捷写法有以下几种：
 - margin : value(四个方向相同) ;
 - margin : value(上下) value(左右) ;
 - margin : value(上) value(左右) value(下) ;
 - margin: value(上) value(右) value(下) value(左) ;

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



外边距合并

- 当两个垂直外边距相遇时，它们将形成一个外边距，称为外边距合并
- 合并后的外边距的高度等于两个发生合并的外边距的高度中的较大者



内边距



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;



内边距的简洁写法

- 内边距的简捷写法有以下几种：
 - padding : value(四个方向相同);
 - padding : value(上下) value(左右);
 - padding : value(上) value(左右) value(下);
 - padding : value(上) value(右) value(下) value(左) ;



边距属性的应用

- 使用边距属性为页面元素定义样式



背景

背景

背景概述

背景属性的作用

背景属性

背景色 background-color

背景图片 background-image

背景重复 background-repeat

背景图片尺寸 background-size

背景图片的固定 background-attachment

背景定位 background-position

背景属性 background

背景概述



背景属性的作用

- 背景样式可以控制 HTML 元素的背景颜色、背景图像等
- 背景色
 - 设置单一的颜色作为背景
- 背景图像
 - 以图片作为背景
 - 可以设置图像的位置、平铺、尺寸等



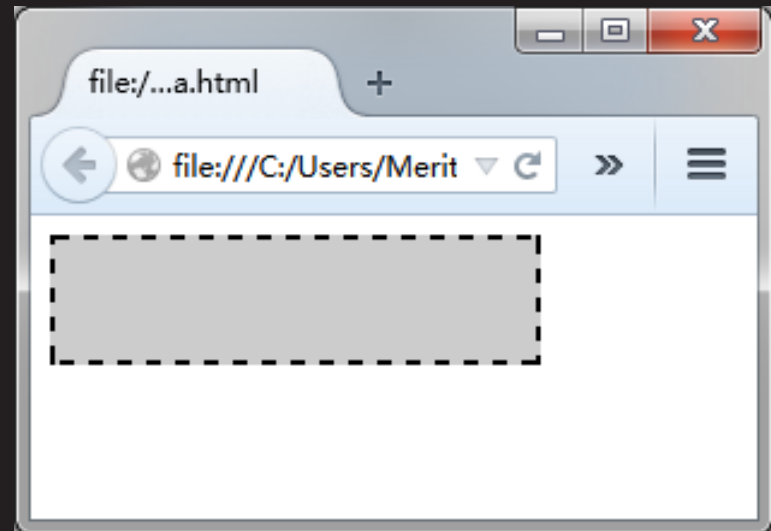
背景属性



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");
}
```



背景重复 background-repeat

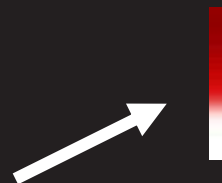
- 默认情况下，背景图像在水平和垂直方向上重复出现（平铺），创建一种称为“墙纸”的效果
- 可以使用 background-repeat 属性控制背景图像的平铺效果
- background-repeat 可取值为
 - repeat：在垂直方向和水平方向重复，为默认值
 - repeat-x：仅在水平方向重复
 - repeat-y：仅在垂直方向重复
 - no-repeat：仅显示一次



背景重复 background-repeat (续1)

样式表中：

```
#bar
{
    height : 76px;
    width : 200px;
    background-image : url(bg.jpg);
    background-repeat : repeat-x;
}
```



html 文档中：

```
<div id="bar"></div>
```



创建渐变色的背景效果

背景图片尺寸 background-size

- 规定背景图像的尺寸，可取值为
 - value1 value2 : 宽度，高度
 - value1% value2% : 百分比
 - cover : 把背景图像扩展至足够大，以使背景图像完全覆盖背景区域，图片的某些部分也许无法显示
 - contain : 把背景图像扩展至最大尺寸，以使其宽度和高度完全适应内容区域



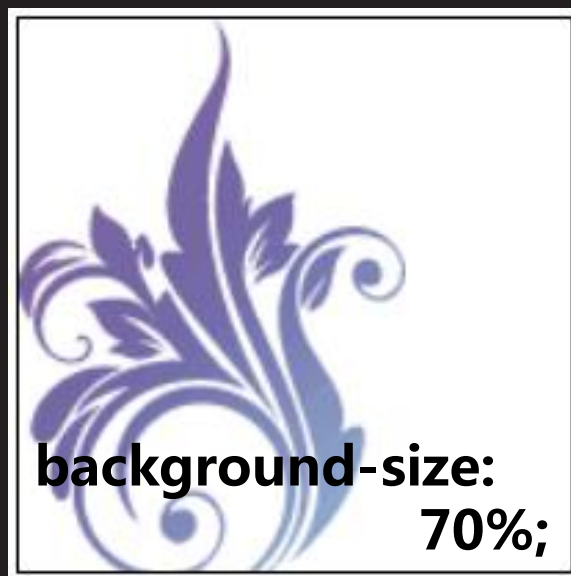
背景图片尺寸 background-size (续1)

样式表中：

```
div
{
    border:1px solid black;
    width:300px;
    height:200px;
    background-image:url(bg_flower.gif);
    background-repeat:no-repeat;
}
```



背景图片尺寸 background-size (续2)



背景图片的固定 background-attachment

- 默认情况下，背景图像会随着页面的滚动而移动
- 可以通过 background-attachment 属性来设置
 - 默认值是 scroll：默认情况下，背景会随文档滚动
 - 可取值为 fixed：背景图像固定，并不会随着页面的其余部分滚动，常用于实现称为水印的图像

body

```
{  
    background-image : url("image/Tulips.jpg");  
    background-repeat : no-repeat;  
    background-attachment : fixed;  
}
```



背景定位 background-position

- background-position 属性改变图像在背景中的位置
- 该属性的取值为：

值	说明
x% y%	第一个值是水平位置，第二个值是垂直位置 表示沿着x轴（水平）和y轴（垂直）的百分比 左上角是 0% 0%，右下角是 100% 100%
x y	第一个值是水平位置，第二个值是垂直位置 表示沿着x轴（水平）和y轴（垂直）的绝对长度 左上角是 0 0
left	在页面或者包含元素的左边显示
center	在页面或者包含元素的中间显示
right	在页面或者包含元素的右边显示
top	在页面或者包含元素的顶部显示
bottom	在页面或者包含元素的底部显示

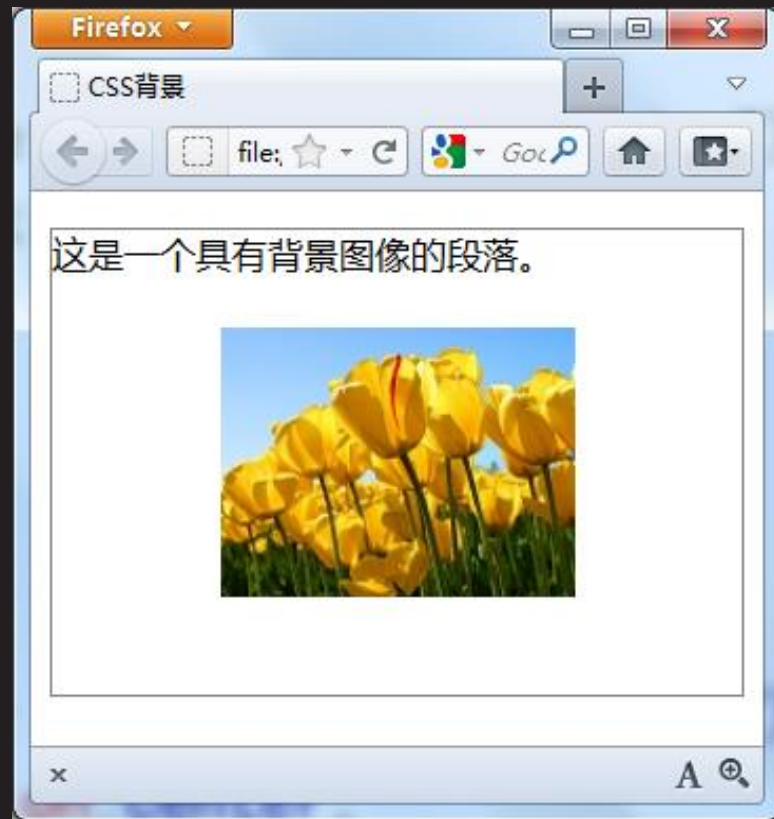
背景定位 background-position (续1)

样式表中：

```
p
{
    background-image : url("image/bg_02.jpg");
    background-position : center;
    background-repeat : no-repeat;
    height : 200px;
    border : 1px solid gray;
}
```

html 文档中：

```
<p>
这是一个具有背景图像的段落。
</p>
```



背景属性 background

- background 属性在一个声明中设置所有的背景属性
- 语法结构是：
 - background : color url(图像URL) repeat attachment position ;
- 如果不设置其中的某个值，将使用默认值

```
body
{
    background:#9dcdfe url("image/Tulips.jpg")
no-repeat fixed left top;
}
```



背景属性的应用

- 【参见 COOKBOOK】
 - 使用背景属性为页面元素定义样式



渐变

渐变

渐变属性

什么是渐变

渐变语法

线性渐变

径向渐变

重复渐变

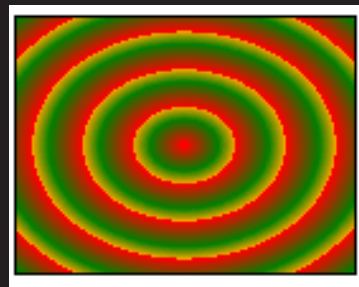
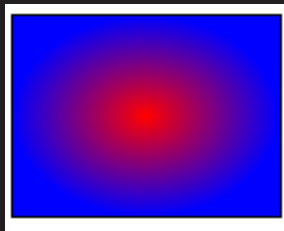
浏览器兼容性

渐变属性



什么是渐变

- 渐变指，两种或者多种颜色之间的平滑过渡
- 可以指定多个中间颜色值（色标）
 - 每个色标包含一种颜色和一个位置
 - 从每个色标的颜色淡出到下一个，以创建平滑的渐变
- 渐变可以用在任何使用背景图片的地方
 - 线性渐变
 - 径向渐变
 - 重复渐变



渐变语法

- 使用 background - image 属性进行设置
- 可以取值
 - linear-gradient : 线性渐变
 - radial-gradient : 径向渐变
 - repeating-linear-gradient : 重复线性渐变
 - repeating-radial-gradient : 重复径向渐变



线性渐变

- linear-gradient(angle, color-point1, color-point2,...)
 - angle : 为第一个参数, 指定渐变的方向, 可以是角度值, 也可以是关键词, 如 to top (对应 0deg), to right (对应 90deg), to bottom (对应 180deg), to left (对应 270deg)
 - color - point : 表示颜色的起始点、中间点或者结束点, 取值为颜色和位置的组合, 如 red 0%、green 50%



线性渐变（续1）

样式表中：

```
div
```

```
{
```

```
width:400px;
```

```
height:300px;
```

```
border:1px solid black;
```

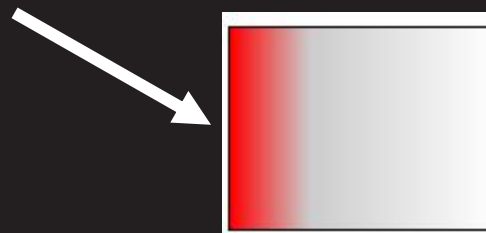
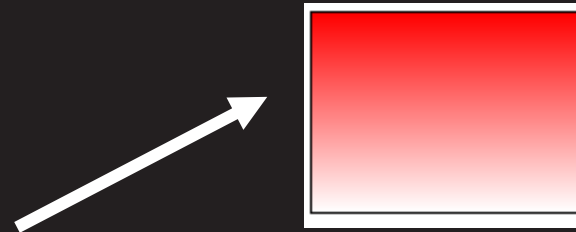
```
}
```

```
#d1 {
```

```
background-image : linear-gradient(to bottom,red,#fff);}
```

```
#d2 {
```

```
background-image : linear-gradient(90deg,red 0,#ccc  
30%,#fff 100%);}
```



径向渐变

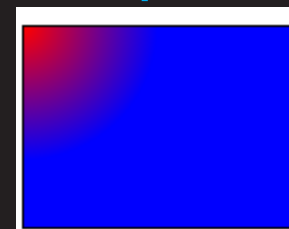
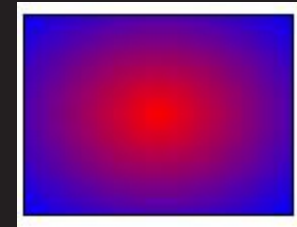
- radial-gradient([size at position], color-point1, color-point2,...)
 - position : 为第一个参数，指定渐变的圆心位置，默认值为 center；可以取值为 数值、百分比，或者关键字；此参数可以省略
 - color - point : 表示颜色的起始点、中间点或者结束点，取值为颜色和位置的组合，如 red 0%、green 50%



径向渐变（续1）

样式表中：

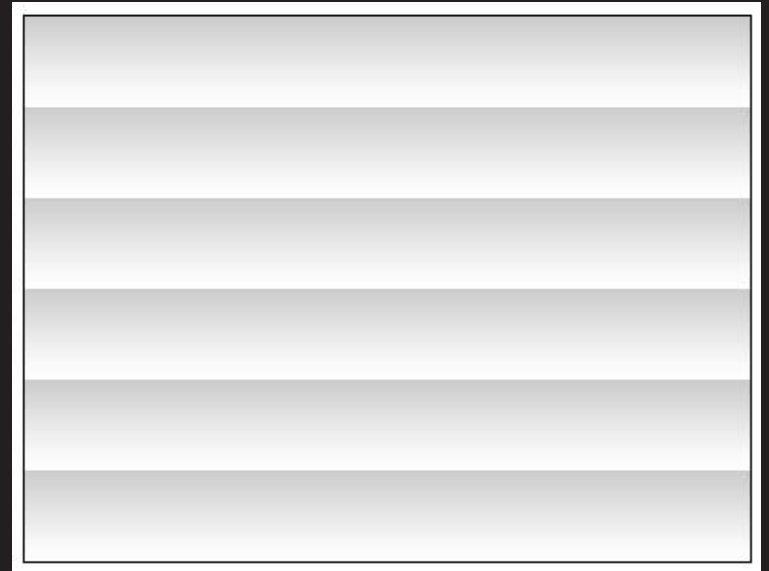
```
div
{
    width:400px;
    height:300px;
    border:1px solid black;
}
#d1{
    background-image:radial-gradient(red,blue);
}
#d2{
    background-image:radial-gradient(200px at left
top,red,blue);
}
```



重复渐变

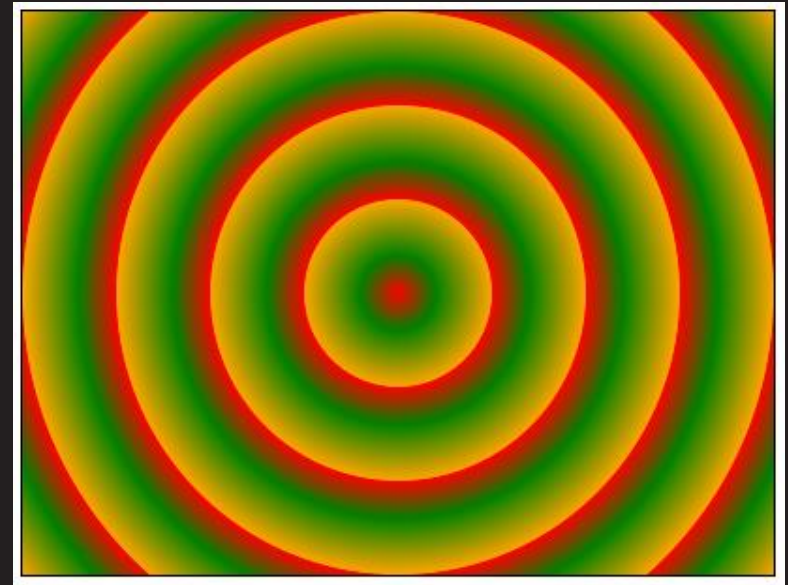
- 重复线性渐变
 - repeating-linear-gradient(angle, color-point1, color-point2,...)

```
div
{
    width:400px;
    height:300px;
    border:1px solid black;
    background-image:repeating-linear-gradient(to
    top,#fff,#f9f9f9 10px,#ccc 50px);
}
```



重复渐变（续1）

- 重复径向渐变
 - repeating-radial-gradient([size at position], color-point1, color-point2,...)



div
{
width:400px;
height:300px;
border:1px solid black;
background-image:repeating-radial-gradient(50px at center,red 0px,green 20px,orange 50px);
}

浏览器兼容性

- 目前，各浏览器的新版本均支持渐变属性
- 对于不支持的版本
 - Firefox 需要前缀 -moz-
 - Chrome 和 Safari 需要前缀 -webkit-
 - Opera 需要前缀 -o-

```
#d1 {
    background-image :
        linear-gradient(to bottom,red,#fff);
    background-image :
        -moz-linear-gradient(to bottom,red,#fff);
    background-image :
        -webkit-linear-gradient(to bottom,red,#fff);
    background-image :
        -o-linear-gradient(to bottom,red,#fff);
}
```



渐变属性的应用

- 【参见 COOKBOOK】
 - 使用渐变属性为页面元素定义样式



总结和答疑

