

# Web前端开发技术与实践

## 第11章：背景与边框

阮晓龙

13938213680 / rxl@hactcm.edu.cn  
<http://web.51xueweb.cn>

河南中医学院管理信息工程学科  
河南中医学院网络信息中心

2015.9

# 本章主要内容

- 背景属性
- 边框属性
- 案例：图片轮转的实现
- 案例：网页课程表的实现



# 1.背景属性

## 1.1基本属性

- **背景的相关属性**是用于控制元素背景色、背景图片等，同时可以设置背景图片的排列方式。
- **背景的基本属性**主要用于定义背景的色彩、图片、重复方式等基本内容。

表 11-01 背景的基本属性

属性	描述
background-color	设置元素的背景颜色
background-image	把图像设置为背景
background-repeat	设置对象背景的背景图片是否平铺
background-attachment	背景图像是否固定或者随着页面的其余部分滚动
background-position	设置背景图像的起始位置
background	简写属性，作用是将背景属性设置在一个声明中

# 1.背景属性

## 1.1基本属性

### □ background-color

- background-color属性用于设置元素的背景颜色。它的默认值取值为none，允许下面的任何内容显示出来。
- background-color属性和color属性一样支持多种颜色格式。

```
p{ background-color:yellow;}
body{ background-color:#ccc;}
.fun{ background-color:#000;}
#test{ background-color:rgb(0,0,0);}
```

# 1.背景属性

## 1.1基本属性

### □ background-image

- background-image属性可以为元素指定背景图像。
- 如果图像包括透明区域，下面的内容将会显示出来。为防止这一点，网页设计者通常将background-image属性和background-color属性一起使用。背景颜色显示在背景图像下面，提供了不透明的背景。
- background-image属性需要一个URL来选择作为背景的背景的相应图像。
- 用作背景图像的图片类型可以为浏览器支持的任何格式，通常为GIF、JPG和PNG格式。

```
b{background-image:url(img.gif);background-color:#fff;}  
body{ background-image:url(img.gif);}  
.brick{ background-image:url(img.gif);}  
#prison{ background-image:url(img.gif);}
```

# 1.背景属性

## 1.1基本属性

### □ background-repeat

- background-repeat属性决定当背景图像比元素的空间小时将如何排列。
- 该属性的默认值为repeat，这将让图像在水平和垂直两个方向上平铺。当该属性取值为repeat-x时，背景图像将仅在水平上平铺。当该属性取值为repeat-y时，背景图像将仅在垂直方向上平铺。当该属性取值为no-repeat时，背景图像将不会平铺。

```
p{background-image:url(img.gif);background-repeat:repeat-x;}  
.titleup{background-image:url(img.gif);background-repeat:repeat-y;}  
body{background-image:url(img.gif);background-repeat:no-repeat;}
```

# 1.背景属性

## 1.1基本属性

### □ background-attachment

- background-attachment属性决定背景图像在元素的内容进行相应滚动时是应该滚动还是应该停留在屏幕的一个固定位置。
- 这个属性的默认值取值为scroll。当取值为fixed时，可实现水印效果，这和<body>元素的专有属性bgproperties（由Microsoft引入）相似。

```
body{background-image:url(images/logo.png);background-attachment:fixed;}
```

# 1.背景属性

## 1.1基本属性

### □ background-position

- background-position属性指定背景图像，在元素的画布空间中的定位方式。
- 有三种方法指定位置：
  - 为图像的左上角指定一个绝对距离，通常以像素为单位。
  - 可以使用水平和垂直方向的百分比来指定位置。
  - 可以使用关键字来描述水平和垂直方向的位置。水平方向上的关键字为left、center和right。垂直方向上的关键字为top、center和bottom。在使用关键字时，未指明的方向上默认的取值为center。

# 1.背景属性

## 1.1基本属性

表 11-02 背景位置关键字定位方式

关键字对	水平位置	垂直位置
top left	0%	0%
top center	50%	0%
top right	100%	0%
center left	0%	50%
center center	50%	50%
center right	100%	50%
bottom left	0%	100%
bottom center	50%	100%
bottom right	100%	100%

# 1.背景属性

## 1.1基本属性

### □ background-position

- 具体语法如下：

```
p{background-image:url(img.gif);background-position:10px 10px;}  
p{ background-image:url(img.gif); background-position:20% 20%;}  
body{background-image:url(img.gif);background-position:center center;}
```

**注意：**如果仅仅设置了一个关键字，那么第二个关键字将取默认值center。因此，在上面的例子中，关键字center只需用一次即可。

# 1.背景属性

## 1.1基本属性

### □ background

- background属性用于全面设置背景样式。该属性是一个复合属性，可用于同时设置背景色、背景图片、背景重复模式等。为更好地控制背景，一般不建议通过该属性来控制背景。
- 具体语法如下：

```
p{ background:white url(img.gif) repeat-y center;}  
body{ background:url(img.gif) top center fixed;}  
.bricks{ background:repeat-y top url(img.gif);}
```

# 1.背景属性

## 1.2 CSS3新增背景属性

### □ CSS3新增的背景属性：

表 11-03 CSS 3 新增背景样式

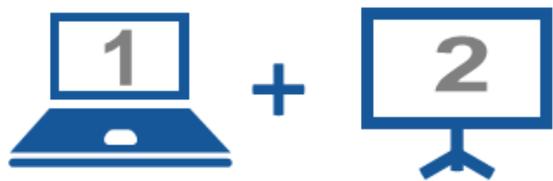
属性	描述
background-clip	规定背景的绘制区域
background-origin	规定背景图片的定位区域
background-size	规定背景图片的尺寸
background-break	规定内联元素的背景图像进行平铺时的循环方式

# 1.背景属性

## 1.2 CSS3新增背景属性

### □ background-clip

- 在HTML页面中，具有背景的元素通常有元素内容、内部补白（padding）、边框、外部补白（margin）四部分组成。
- 元素背景的显示范围在CSS2与CSS2.1、CSS3中并不相同。
  - 在CSS2中，背景的显示范围是指内部补白之内的范围，不包括边框。
  - 在CSS2.1和CSS3中，背景的显示范围是指包括边框在内的范围。
  - 在CSS3中，也可以使用background-clip来修饰背景的显示范围。
- 如果将background-clip的属性值设定为border，则背景范围包括边框区域，如果设定为padding，则不包括边框区域。



现场演示:

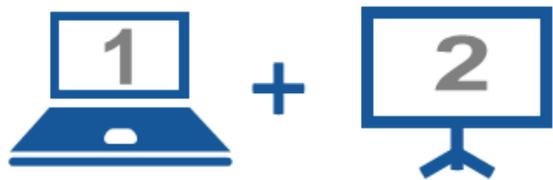
- 案例11-01: 两种background-clip属性值的对比

# 1.背景属性

## 1.2 CSS3新增背景属性

### □ background-origin

- 在绘制背景图像时，默认是从内部区域（padding）的左上角开始，但是可以利用background-origin属性来指定绘制时从边框的左上角开始，或者从内容的左上角开始。
- 在Firefox浏览器中指定绘制起点时，需要在样式代码中将background-origin属性书写成“-moz-background-origin”的形式；在Safari浏览器或者Chrome浏览器中指定绘制起点时，需要在样式代码中将background-origin属性写成“-webkit-background-origin”的形式。



现场演示:

- 案例11-02: background-origin属性

# 1.背景属性

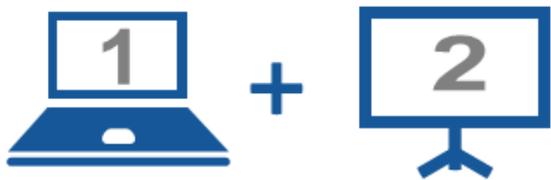
## 1.2 CSS3新增背景属性

### □ background-size

- 在CSS3中，可以使用background-size属性来指定背景图像的尺寸。使用background-size属性来指定背景图像尺寸的最简单的方法类似如下所示：

```
background-size: 40px 20px;  
-webkit-background-size: 40px 20px;
```

- 其中，40px为背景图像的宽度，20px为背景图像的高度，中间用半角空格进行分隔。
- 如果要维持图像比例的话，可以设定图像宽度与高度的同时，将另一个参数设定为“auto”。



现场演示:

- 案例11-03: background-size属性

# 1.背景属性

## 1.3多背景

### □ CSS3多背景语法及参数

- CSS3多背景语法和CSS中背景语法其实没有本质上的区别，只是在CSS3中可以给多个背景图像设置相同或者不相同的background-  
(position||repeat||clip||size||origin||attachment) 属性。
- 在CSS3多背景中，相邻背景之间必须使用逗号分隔开。

# 1.背景属性

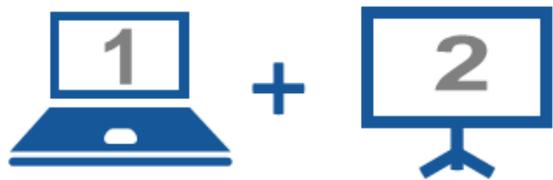
## □ CSS3多背景语法及参数

- 具体语法如下：

```
background: [background-image] | [background-position] [/background-size] | [background-repeat] | [background-attachment] | [background-clip] | [background-origin],*
```

- 可以把上面的缩写拆解成以下形式：

```
background-image: url1,url2,...,urlN;  
background-repeat:repeat1,repeat2,...,repeatN;  
background-position:position1,position2,...,positionN;  
background-size:size1,size2,...,sizeN;  
background-attachment: attachment1, attachment2,..., attachmentN;  
background-clip:clip1,clip2,...,clipN;  
background-origin:origin1,origin2,...,originN;  
background-color:color1,color2,...,colorN;
```



现场演示:

- 案例11-04: 多背景属性使用

# 1.背景属性

---

## 1.3多背景

### □ CSS3多背景的优势

- 使设计师部分摆脱了对Adobe Photoshop等绘图工具的依赖。
- 具有CSS 3多背景的层次之分。
- 可单独指定背景图像中某个图像的平铺方式、图像的大小以及位置。
- 可图片合成，代码易于维护。

# 1. 背景属性

## □ 线性渐变

- 渐变是通过背景设置的，所以必须使用background或background-image属性。具体语法如下：

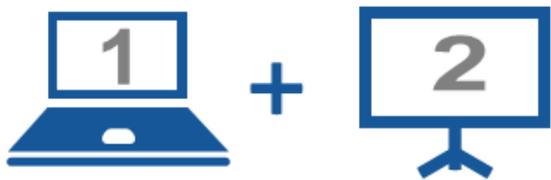
```
linear-gradient ( start position, from color, to color ) ;
```

- linear-gradient () 函数的属性规定了创建渐变效果的开始位置与颜色。第一个值可以是像素值、百分比或使用关键字top、bottom、left和right。
- 开始位置可以用角度替换，指定渐变效果的方向。基本语法如下：

```
background: linear-gradient(30deg,#fff,#000);
```

- 可以声明各种颜色的结束位置。基本语法如下：

```
background: linear-gradient(top,#fff 50%,#000 90%);
```



现场演示:

- 案例11-05: 线性渐变属性

# 1.背景属性

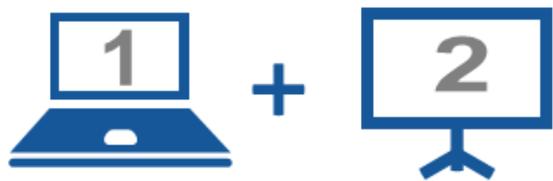
---

## 1.4渐变背景

### □ 放射渐变

- 放射渐变的使用方法与前一方法相似。

```
radial-gradient ( start position, from color, to color ) ;
```



现场演示:

- 案例11-06: 放射渐变属性

## 2.边框属性

### 2.1基本属性

表 11-04 边框基本属性

属性	描述
border	简写属性，用于把针对四个边的属性设置在一个声明
border-style	用于设置元素所有边框的样式，或者单独地为各边设置边框样式
border-width	简写属性，用于为元素的所有边框设置宽度，或者单独地为各边边框设置宽度
border-color	简写属性，设置元素的所有边框中可见部分的颜色，或为 4 条边分别设置颜色
border-bottom	简写属性，用于把下边框的所有属性设置到一个声明中
border-bottom-color	设置元素的下边框的颜色
border-bottom-style	设置元素的下边框的样式
border-bottom-width	设置元素的下边框的宽度
border-left	简写属性，用于把左边框的所有属性设置到一个声明中
border-left-color	设置元素的左边框的颜色

## 2.边框属性

### 2.1基本属性

表 11-04 边框基本属性

属性	描述
border-left-style	设置元素的左边框的样式
border-left-width	设置元素的左边框的宽度
border-right	简写属性，用于把右边框的所有属性设置到一个声明中
border-right-color	设置元素的右边框的颜色
border-right-style	设置元素的右边框的样式
border-right-width	设置元素的右边框的宽度
border-top	简写属性，用于把上边框的所有属性设置到一个声明中
border-top-color	设置元素的上边框的颜色
border-top-style	设置元素的上边框的样式
border-top-width	设置元素的上边框的宽度

## 2.边框属性

### 2.1基本属性

#### □ border-style

- border-style属性用来设置边框的显示外观。它默认的取值为none，即没有边框。

表 11-05 边框 style 样式

属性	描述
dotted	一个点边框
dashed	一个虚线边框
solid	一个实线边框
double	一个双线边框
groove	一个蚀刻边框
ridge	一个突出边框
inset	一个凹进的边框，使得对象看起来嵌入了页面
outset	一个凸起的边框，使得对象看起来凸起

- 具体语法：

```
h1{border-style:solid;}  
p.boxed{border-style:double;}  
.button{border-style:outset;}
```

## 2.边框属性

### 2.1基本属性

#### □ border-width

- 可通过border-top-width、border-right-width、border-left-width、border-bottom-width分别设置边框的宽度。也可通过border-width属性对全部边框进行设置。
- border-width属性可分别赋1-4个值，多个值按照顺时针方向应用到顶部、右边、底部、左边的边框。边框的宽度可以使用像素值和关键字来指定，关键字允许为thin，medium和thick。
- 具体语法如下：

```
p{border-width:10px;}  
p.double{border-width:thick;}  
.thickandthin{border-width:thick thin;}  
.fun{border-width:thick;}
```

## 2.边框属性

### 2.1基本属性

#### □ border-color

- 通过border-color属性可以为边框指定颜色。指定边框颜色时可以使用支持的颜色英文名称、数字的RGB颜色规范、十六进制色彩值。border-color属性可以分别赋1-4个值，多个值按照顺时针方向应用到顶部、右边、底部、左边的边框。也可以通过border-top-color、border-right-color、border-bottom-color、border-left-color为各边框单独设置颜色值。
- 具体语法如下：

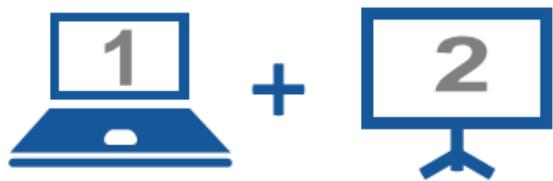
```
p.all{  
border-style:solid;  
border-top-color:green;  
border-right-color:#ff0000;  
border-bottom-color:yellow;  
border-left-color:rgb(12,255,255);  
}
```

## 2.边框属性

### 2.1基本属性

#### □ border属性快速设置

- border属性允许使用组合属性值对边框的宽度、颜色、风格等进行设置，也可通过border-top、border-right、border-bottom、border-left属性使用组合属性值对单一边框的宽度、颜色、风格等进行设置。



现场演示:

- 案例11-07: 边框基本属性

## 2.边框属性

### 2.2 CSS3新增边框属性

- CSS3新增的边框属性具有强大的生命力，灵活使用这些属性可以设计很多优美精巧的UI界面效果。
- CSS3增加了圆角边框（border-radius）、图片边框（border-image）、渐变边框、阴影（box-shadow）。

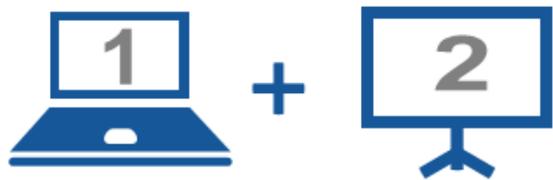
## 2.边框属性

### 2.3圆角边框

- 在border-radius属性中指定两个半径
  - 具体语法如下：

```
div{border-radius: 40px 20px;}
```

- 针对这种情况，各种浏览器的处理方式并不一致。在Chrome浏览器和Safari浏览器中，会绘制出一个椭圆形边框，第一个半径为椭圆的水平方向半径，第二个半径为椭圆的垂直方向半径。在Firefox浏览器与Opera浏览器中，将第一个半径作为边框左上角与右下角的圆半径来绘制，将第二个半径作为边框左下角与右上角的圆半径来绘制。



现场演示:

- 案例11-08: 在border-radius属性中指定两个半径

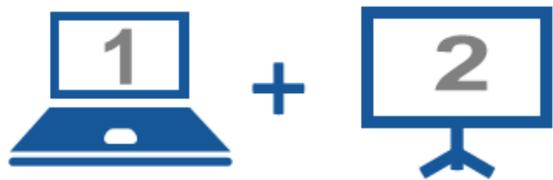
## 2.边框属性

---

### 2.3圆角边框

#### □ 不显示边框

- 如使用border-radius属性但把边框设定为不显示时，浏览器将把背景的四个角绘制为圆角。



现场演示：

- 案例11-09：不显示边框的使用

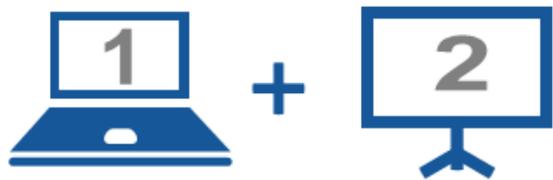
## 2.边框属性

---

### 2.3圆角边框

#### □ 修改边框种类

- 使用border-radius属性后，不管边框是什么种类，都会将边框沿着圆角曲线进行绘制。



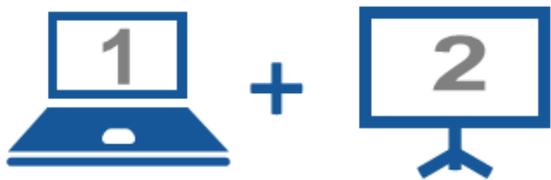
现场演示:

- 案例11-10: 修改边框种类

## 2.边框属性

### 2.3圆角边框

- 绘制四个不同半径的圆角边框
  - 可以通过分别定义border-top-left-radius、border-top-right-radius、border-bottom-right-radius、border-bottom-left-radius属性值，分别定义边框四个圆角半径。



现场演示:

- 案例11-11: 绘制四个不同半径的圆角边框

## 2. 边框属性

### □ border-image

#### ■ border-image 具体语法：

```
border-image: none | <image> [number | <percentage>] [stretch | repeat | round] {0,2}
```

#### ■ 参数的含义与使用方法：

- none：默认值，表示边框无背景图片。
- <image>：设置背景图片，这跟background-image一样，可以使用绝对或相对的URL地址，来指定边框的背景图片。
- <number>：number是一个数值，用来设置边框或者边框背景图片的大小，其单位是像素（px），可以使用1-4个值，表示4个方位的值。

## 2.边框属性

### 2.4图片边框

#### □ border-image

##### ■ 参数的含义与使用方法:

- `<percentage>`: `percentage`也是用来设置边框或者边框背景图片的大小, 和`number`不同之处是, `percentage`使用的是百分比。
- `stretch`、`repeat`、`round`: 这三个属性参数是用来设置边框背景图片的铺放方式, 类似于`background-position`。其中`stretch`会拉伸边框背景图片; `repeat`会重复边框背景图片; `round`会平铺边框背景图片, 其中`stretch`为默认值。

## 2.边框属性

### 2.4图片边框

#### □ border-image

- 为了便于理解，暂把border-image在语法的表达式形式进行属性的分解阐述（实际应用中是不能分解的），分解后变成以下四个部分：
  - 引入背景图片：border-image-source。
  - 切割引入背景图片：border-image-slice。
  - 边框图片的宽度：border-image-width。
  - 边框背景图片的排列方式：border-image-repeat。

## 2.边框属性

### 2.4图片边框

#### □ border-image-source

- border-image-source具体语法:

```
border-image: none | <image> [number | <percentage>] [stretch | repeat | round] {0,2}
```

- border-image-source和CSS 2中的background-image属性相似，也是通过url () 来调用背景图片，图片的路径可以是相对地址，也可以是绝对地址，其默认值为none。

## 2.边框属性

### 2.4图片边框

#### □ border-image-slice

- border-image-slice具体语法：

```
border-image-slice:[<number> | <percentage>] {1,4} && fill ?
```

- border-image-slice是用来分解引入的背景图片，这个参数相对来说比较复杂，主要有以下几个方面：
  - 取值支持<number> | <percentage>。其中number是没有单位的，因为其默认值的单位就是像素。除了直接用number来设置外，还可以使用百分比值来表示，即相对于边框背景图片而言的。

## 2.边框属性

### 2.4图片边框

#### □ border-image-slice

- border-image-slice是用来分解引入的背景图片，这个参数相对来说比较复杂，主要有以下几个方面：
  - 例如，边框图片的大小是300px\*240px，取百分比为25%，30%，15%，20%，实际相对应的效果就是剪切了图片的60px，90px，36px，60px的四边大小。
  - border-image-slice中的number或者percentage都可以取1-4个值，这个类似于CSS 2中的border-width的取值方式，遵循top、right、bottom、left的规则。
  - fill从字面上说就是填充的意思，如果使用这个关键字时，图片边界的中间部分将保留下来，默认情况下为空。

## 2.边框属性

### 2.4图片边框

#### □ border-image-slice

- border-image-slice是用来分解引入的背景图片，这个参数相对来说比较复杂，主要有以下几个方面：
  - 剪切的特性 (slice)。在border-image中slice是一个关键部分，border-image-slice虽然表面上说不是剪切，但实际应用中就是一种纯粹的剪切，把取到的边框图片切成九份，再像background-image一样重新布置（如图1所示），在该图中，border-image-slice属性在距边框背景图片的top、right、bottom、left四边的18px分别切了一刀，这样一来就把背景图片切成了九个部分，称为“九宫格”（如图2所示），分割后的图像在CSS 3中的名称如表11-06 所示。

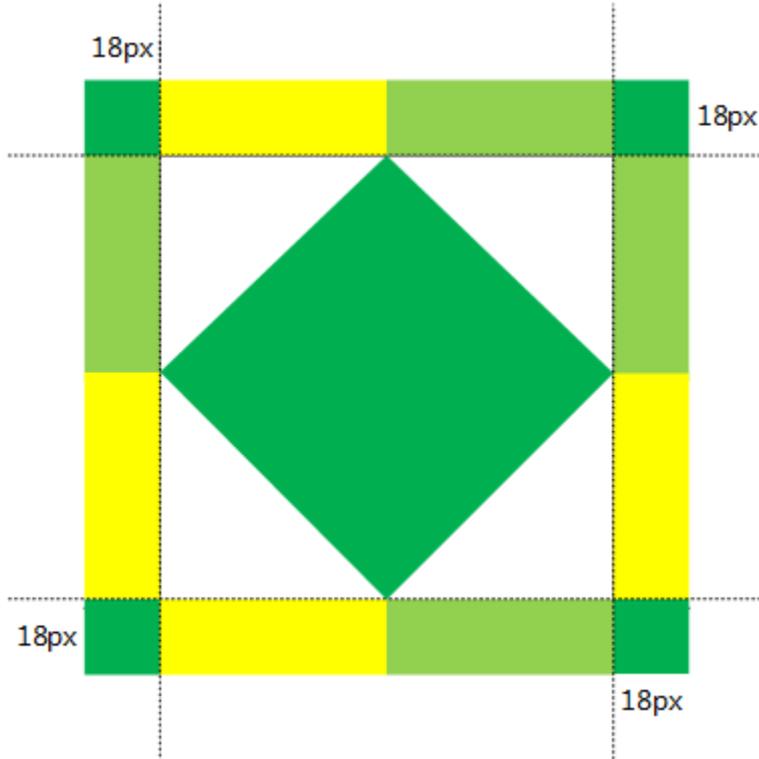


图1 浏览器对于图像文件的分割

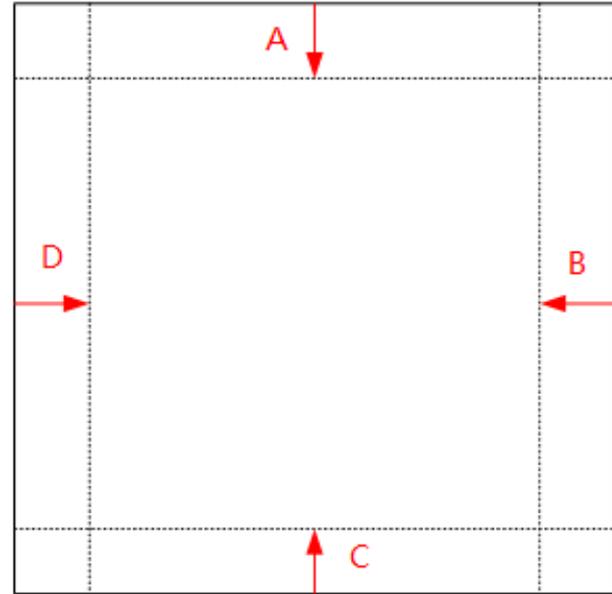


图2 A、B、C、D、四个参数

表 11-06 被分割为九个部分的图像名称

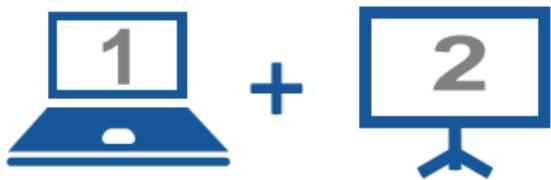
border-top-left-image	border-top-image	border-top-right-image
border-left-image		border-right-image
border-bottom-left-image	border-bottom-image	border-bottom-right-image

## 2.边框属性

### 2.4图片边框

#### □ border-image-slice

- border-top-right-image、border-bottom-right-image、border-bottom-left-image 以及border-top-left-image四个边角部分，在border-image中是没有任何展示效果的，把这四个部分称为盲区。
- border-top-image、border-right-image、border-bottom-image 以及border-left-image属于展示效果区域。其中上下区域border-top-image和border-bottom-image区域受到水平方向效果影响，border-right-image和border-left-image区域受到垂直方向效果影响。



## 现场演示:

- 案例11-12: 水平round效果
- 案例11-13: 水平repeat效果
- 案例11-14: 水平stretch效果
- 案例11-15: 垂直round效果
- 案例11-16: 垂直repeat效果

## 2.边框属性

### 2.4图片边框

□ 通过上面几个案例比较可知：

- repeat属性是边框中间向两端不断平铺，在平铺的过程中保持边框背景图片切片的大小，这样就造成了图示中的两端边缘处有被切的现象；
- round属性则会对边框背景图的切片进行压缩（拉伸）来适应边框宽度大小进行排列，使其正好显示在区域内；
- stretch属性只会把相应的切片进行拉伸，适应边框的大小。

## 2.边框属性

### 2.4图片边框

#### □ border-image-width

- border-image-width具体语法：

```
border-image-width:[<length> | <percentage> | <number> | auto]{1,4}
```

- 该属性用来设置边框背景图片的显示大小，其实也可以理解为border-width，虽然W3C定义了border-image-width属性，但各浏览器还是将其视为border-width来用，也就是border-image-width属性用法和border-width属性用法一样。

## 2. 边框属性

### □ border-image-repeat

- border-image-repeat 具体语法：

```
border-image-repeat [stretch | repeat | round]{1,2}
```

- 该属性用来指定边框背景图片的排列方式，默认值为stretch。这个属性设置参数和其他属性不同，border-image-repeat不遵循top、right、bottom、left的方位原则，只接受两个（或一个）参数值，第一个值表示水平方向上的排列方式，第二个值表示垂直方向上的排列方式。当取一个值时，表示水平和垂直方向上的排列方式相同。如果不设置任何值时，水平和垂直方向上都会以默认值stretch方式进行排列，如图3所示。

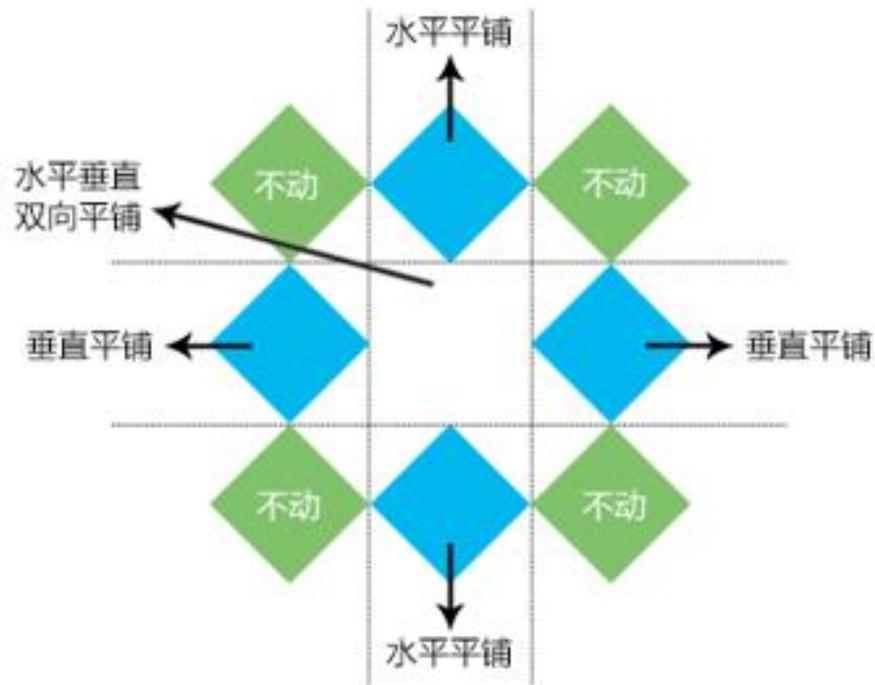


图3 背景图片平铺方式

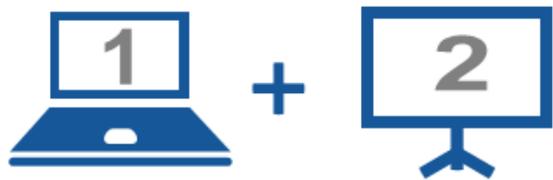
## 2. 边框属性

- CSS3提供了4个属性来支持渐变边框，但这4个属性目前只有Firefox浏览器能够支持：
  - `border-top-colors`：该属性用于设置目标组件上边框颜色。如果设置上边框的宽度是Npx，那么就可以为该属性设置N种颜色，每种颜色显示1px的宽度。但如果设置的颜色数量小于边框的宽度，那么最后一个颜色将会覆盖该边框剩下的宽度。
  - `border-right-colors`：该属性用于设置目标组件的右边框颜色。该属性指定多个颜色值的意义与`border-top-colors`属性里各颜色值的意义相同。

## 2.边框属性

### 2.5渐变边框

- CSS3提供了4个属性来支持渐变边框，但这4个属性目前只有Firefox浏览器能够支持：
  - border-bottom-colors: 该属性用于设置目标组件的下边框颜色。该属性指定多个颜色值的意义与border-top-colors属性里各个颜色值的意义相同。
  - border-left-colors: 该属性用于设置目标组件的左边框颜色。该属性指定多个颜色值的意义与border-top-colors属性里各个颜色值的意义相同。



现场演示:

- 案例11-17: 渐变边框

## 2.边框属性

### 2.6盒子阴影

#### □ 语法

- box-shadow具体语法：

```
box-shadow: none | [inset x-offset y-offset blur-radius spread-radius color], [inset x-offset y-offset blur-radius spread-radius color]
```

- box-shadow属性可以使用一个或多个投影，如果使用多个投影时必须使用逗号“,” 隔开。
- box-shadow参数的具体内容：
  - none：默认值，元素没有任何阴影效果。
  - inset：阴影类型，可选值。如果不设置，其默认的投影方式是外阴影；如果取其唯一值“inset”，就是给元素设置内阴影。

## 2.边框属性

### 2.6盒子阴影

#### □ 语法

##### ■ box-shadow参数的具体内容：

- x-offset：阴影水平偏移量，其值可以是正负值。其值取正值，则阴影在元素右边；取负值，则阴影在元素的左边。
- y-offset：阴影垂直偏移量，其值可以是正负值。其值取正值，则阴影在元素底部；取负值，阴影在元素的顶部。
- blur-radius：阴影模糊半径，可选参数。其值只能是正值，如果取值为“0”时，表示阴影不具有模糊效果，取值越大，阴影的边缘就越模糊。

## 2.边框属性

### 2.6盒子阴影

#### □ 语法

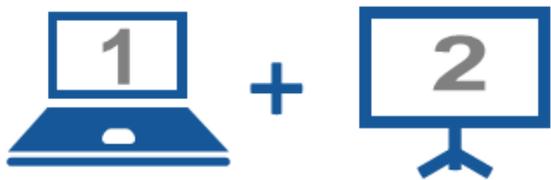
- box-shadow参数的具体内容：
  - spread-radius：阴影扩展半径，可选参数。其值可以是正负值，取值为正值，则整个阴影都延展扩大；取负值，则整个阴影都缩小。
  - color：阴影颜色，可选参数。如果不设定任何颜色时，浏览器会取默认色，但各浏览器默认色不一样，特别是在webkit内核下的浏览器将无色，也就是透明。

## 2.边框属性

### 2.6盒子阴影

#### □ 应用

- 和Adobe Photoshop软件制作图片相比，box-shadow修改元素阴影效果要方便得多，因为box-shadow可以修改六个参数，从而得到不同的效果。
- 单边阴影效果
  - 从下面案例中分析单边阴影效果。



现场演示:

- 案例11-18: 单边阴影效果

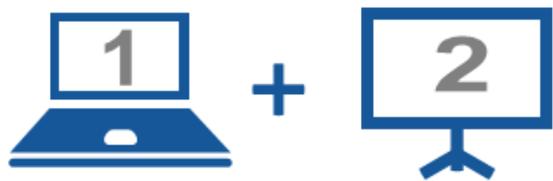
## 2.边框属性

### 2.6盒子阴影

#### □ 应用

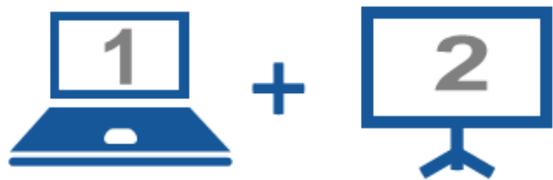
##### ■ 四边相同阴影效果

- box-shadow给元素设置相同的四边阴影效果，其实分为两种：
  - 只设置阴影模糊半径和阴影颜色
  - 只设置扩展半径和阴影颜色
- 从案例中进行分析只设置阴影模糊半径和阴影颜色与只设置扩展半径和阴影颜色的效果。



## 现场演示：

- 案例11-19：设置相同四边阴影效果（只设置阴影模糊半径和阴影颜色）
- 案例11-20：设置相同四边阴影效果（只设置扩展半径和阴影颜色）



现场演示:

- 案例11-21: 内阴影效果

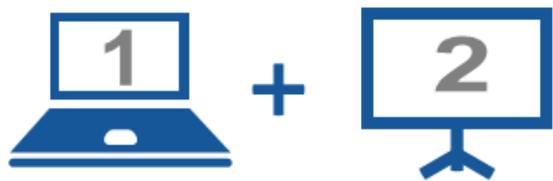
## 2.边框属性

### 2.6盒子阴影

#### □ 应用

##### ■ 图片内阴影效果

- box-shadow的inset内阴影直接运用在图片“img”元素上是没有任何效果的。只有在img外添加一个容器便签，例如“div”便签，不将img转换成div便签的背景，只是将box-shadow的内阴影使用在div标签上，img才会出现内阴影效果。



现场演示:

- 案例11-22: 图片内阴影效果

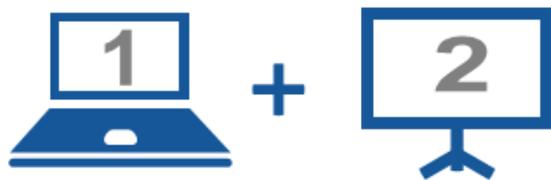
## 2.边框属性

### 2.6盒子阴影

#### □ 应用

##### ■ 多层阴影

- 前几种都是单阴影效果的使用，其实`box-shadow`可以和多层阴影同时使用，每层阴影之间用逗号“,” 隔开。
- 制作多层阴影效果时，不设置模糊半径，只设置扩展半径，并配合多个阴影颜色，还可以制作多色边框效果。
- 使用多层级`box-shadow`制作多色边框效果需要注意阴影的顺序，最先写的阴影将显示在最顶层。如果前面的阴影太大，顶层的阴影就会遮盖底部的阴影。



## 现场演示:

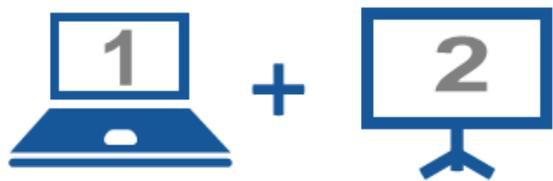
- 案例11-23: 多层阴影效果
- 案例11-24: 使用box-shadow制作多色边框效果

## 2.边框属性

### 2.6盒子阴影

#### □ 优势

- 从实现盒子阴影来说，`box-shadow`是最方便的，不论是使用背景图片，还是使用滤镜或JavaScript，都无法与`box-shadow`属性相比。
  - `box-shadow`具有多个属性参数可选，能制作出圆润平滑的阴影效果。
  - 代码维护方便，可以随时更改参数来实现效果和更新。



现场演示:

- 案例11-25: 制作3D搜索表单

## 2.边框属性

### 2.6盒子阴影

#### □ 浏览器兼容性

- 目前box-shadow属性得到各种浏览器的支持，IE 8及以前版本的浏览器不支持box-shadow属性，在现代浏览器的新版本中无需加各浏览器的前缀。考虑到向前兼容的需求，Firefox 3.5-3.6下需要添加“-moz-”，Chrome 4-9和Safari 3.1-5.0浏览器中需要添加“-webkit-”。
- 虽然IE浏览器低版本不支持这个属性，但目前box-shadow在实际项目运用越来越普遍。因为box-shadow实现阴影比使用背景图片的方法方便，同时能为Web前端开发减少很多时间，维护也方便。要兼容IE浏览器低版本，可以使用IE的滤镜来模拟实现。

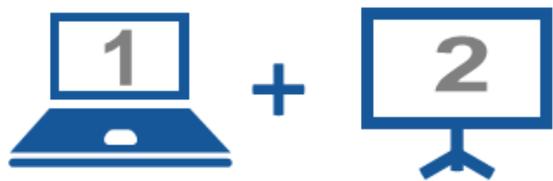
```
filter:progid:DXImageTransform.Microsoft.Shadow(color='颜色值',Direction=阴影角度(数值),Strength=阴影半径(数值));
```

### 3.案例：图片轮转的实现

- 效果图：



POP图片轮转



现场演示:

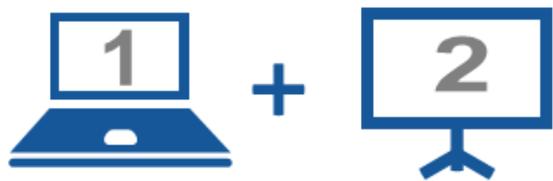
- 案例11-26: 图片轮转的实现

## 4.案例：网页课程表的实现

### ■ 效果图：

我的课程表						
日	一	二	三	四	五	六
1	高等数学 A301	经济学原理 A521	管理学概论 A521	C语言程序设计 A533	高等数学 A301	玉工艺品 鉴赏 A701
2						
3		马克思主义基本原理 概论 A632	高等数学 A301		经济学原理 A521	玉工艺品 鉴赏 A701
4						
5	大学英语 A709	大学英语 A709	经济学原理 A521	大学英语 A709	管理学概论 A521	
6						
7	C语言程序设计 A533		体育 西操场	马克思主义基本原理 概论 A632		
8						
9						
10						

我的课程表



现场演示:

- 案例11-27: 网页课程表的实现

Thanks.