昨日知识点回顾：

01、HTML5-CSS3简介

02、CSS3选择器

03、颜色模式设置

04、阴影的设置

05、盒模型

06、边框圆角

今日知识点：

01、线性渐变 径向渐变

02、background新增属性

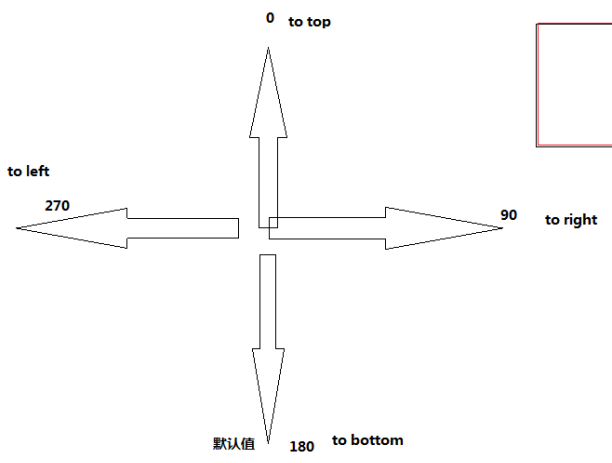
03、图片边框的基本用法

04、过渡效果transition属性

05、2d&3d变换效果transform属性

# 1 线性渐变 & 径向渐变

## 1.1 01-HTML5-线性渐变



**1、线性渐变属性的理解**

**添加渐变：从一边到另一边，渐变不是一个单一色，它产生的是图像，所以需要使用background  
 linear-gradient(方向，开始颜色 位置，颜色2 位置，颜色3 位置...);  
方向：  
to top:0deg  
to right:90deg  
to bottom:180deg --默认值 从上到下  
to left:270deg**

**2、线性渐变linear-gradiet()的使用**

background: linear-gradient(red yellow); **从上到下显示是：红到黄**

background: linear-gradient(to right,red,blue); **从左到右显示是：红到蓝**

background:linear-gradient(to right,red 0%,red 50%,blue 50%,blue 100% );

**0%-50%显示红色，50%-100%显示蓝色**

## 1.2 02-HTML5-径向渐变

1. **径向渐变属性的理解**

**添加径向渐变：产生也是图像 从中间向四周  
background: radial-gradient(red,blue);  
语法：radial-gradient(形状 大小 坐标,颜色1，颜色2...)：  
形状shape:circle:产生正方形的渐变色**

**ellipse:适配当前的形状,如果是正方形的容器，两者效果一样.如果宽高不一样，默认效果切换到ellipse**

**circle: 也是正方形形状但有些区域会溢出  
at position:坐标，默认在正中心。可以赋值坐标（参照元素的左上角），也可以赋值关键字(left center right top bottom)**

**background: radial-gradient(at left top,red,blue);**

**大小size: closest-side：最近边； farthest-side：最远边； closest-corner：最近角； farthest-corner：最远角。默认是最远的角farthest-corner**

**background:radial-gradient(circle farthest-side at 50px 50px,red,blue);**

## 1.3 03-HTML5-重复渐变

1. **重复渐变的语法**
   1. **径向渐变**

background: repeating-radial-gradient(circle closest-side at center center,  
#fff 0%,#fff 10%,  
#000 10%,#000 20%);

b) 线性渐变

background: repeating-linear-gradient(45deg, **线性的方向是可以设置角度**  
 #fff 0%,#fff 10%,  
 #000 10%,#000 20%);

# 2 background新增属性

## 2.1 04-HTML5-background(上)

**1、背景平铺如何设置？ (background-repeat)**

**background-repeat**

**round:会将图片进行缩放之后再平铺  
space:图片不会缩放平铺，只是会在图片之间产生相同的间距值**

1. **如何设置在滚动容器的背景行为？ (background-attachment)**
   1. 固定滚动和跟随滚动 (background-attachment：fixed/scroll)
      1. fixed**滚动窗口的滚动条** 图片不会随着滚动条滚动但是图片被盖住
      2. scroll**滚动窗口的滚动条** 图片会随着滚动条滚动
   2. local和scroll的区别
      1. local：**滚动内容的滚动条** 背景图片会跟随内容一起滚动
      2. scroll：**滚动内容的滚动条** 背景图片不会跟随内容一起滚动

## 2.2 05-HTML5-background(中)

1. **background-size图片尺寸大小如何设置**

**语法： cover会填充满盒子，contain盒子可能会有空隙**

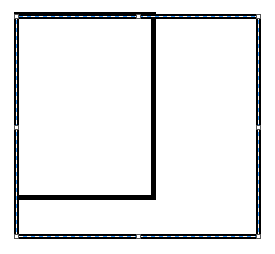
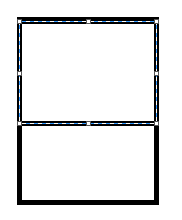
background-size: auto(原始图片大小) || number(数值) || percentage(百分比) || cover(放大铺满) || contain(缩小铺满)

**1、background-size: 300px 500px;  
 2、background-size: 300px;  
 3、background-size: 50% 50%; 是参照父容器可放置内容区域的百分比  
 4、background-size: contain;**

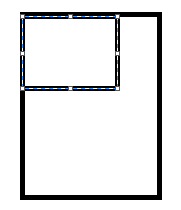
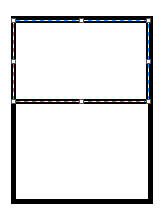
**设置contain:按比例调整图片大小，使用图片宽高自适应整个元素的背景区域，使图片全部包含在容器内  
 1.图片大于容器：有可能造成容器的空白区域,将图片缩小  
 2.图片小于容器：有可能造成容器的空白区域，将图片放大**

**总结：只要有一边靠近边框就停止缩小或放大，其余的以空白显示**

**图片大于容器： 开始如左图，设置contain后：如右图**

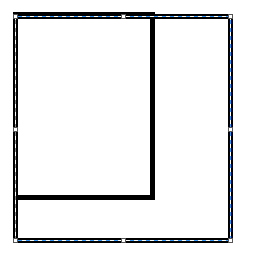
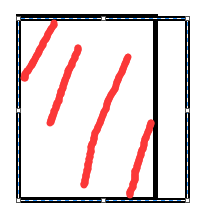
 

**图片小于容器： 开始如左图，设置contain后：如右图**

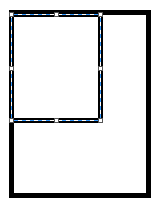
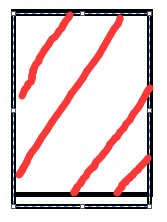
 

1. **background-size: cover; 容器会全部显示，有部分会不可见  
    cover:与contain刚好相反，背景图片会按比例缩放自适应整个背景区域，如果背景区域不足以包含所有背景图片，图片内容会溢出  
    1.图片大于容器：等比例缩小，会填满整个背景区域，有可能造成图片的某些区域不可见**

**总结：先缩小图片，满足图片长和宽都在容器都填充满，超出的就不可见，只显示红色区**

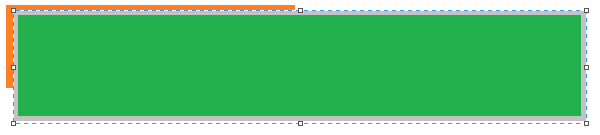
  **2.图片小于容器：等比例放大，填满整个背景区域，图片有可能造成某个方向上内容的溢出**

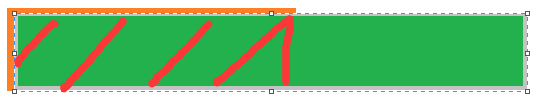
**总结：先放大图片，满足图片长和宽在容器都填充满，超出的就不可见，只显示红色区**

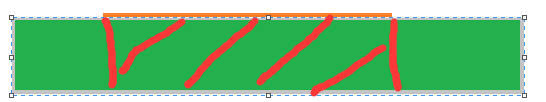
 

## 2.3 06-HTML5-background(backgroundSize案例)

1. **如何让图片居中且正常覆盖在盒子中?**
   1. 给图片设置大小：background-size:cover;
   2. 给图片设置定位：background-position:center







**实现原理：**

1.图片正常显示第一张效果，

1. 设置cover属性显示第二张效果，

3.设置定位center 显示第三张效果 》》》就实现了居中全屏显示

## 2.4 07-HTML5-background(下)

1. **以定位显示：background-origin:**

a) 作用：background-origin 属性规定 background-position 属性相对于什么位置来定位。默认值是left top左上角

b) 语法：background-origin: padding-box|border-box|content-box;

c) 属性值说明：

|  |  |
| --- | --- |
| padding-box | 背景图像相对于内边距框来定位。 |
| border-box | 背景图像相对于边框盒来定位。 |
| content-box | 背景图像相对于内容框来定位。 |

1. **以内容显示：background-clip：**

a) background-clip 属性规定背景的绘制区：虽然是设置裁切，但是控制的是显示。说白了，就是设置最终显示那些区域

b) 语法：background-clip: border-box|padding-box|content-box;

c) 属性值说明：

|  |  |
| --- | --- |
| **值** | **描述** |
| border-box | 背景被裁剪到边框盒。 |
| padding-box | 背景被裁剪到内边距框。 |
| content-box | 背景被裁剪到内容框。 |

**基本常用:background-origin:content-box/background-clip:content-box**

# 3 图片边框的基本用法

## 3.1 08-HTML5-边框图片(上)

1. **边框图片如何添加**

**border-image-source:可以指定边框图片的路径,默认只是填充到容器的四个角  
border-image-source: url("../images/border1.png");  
让它成为九宫格：border-image-slice:设置四个方向上的裁切距离.fill:做内容的内部填充  
border-image-slice: 27 fill;**

## 3.2 09-HTML5-边框图片(中)

**1、边框图片宽度设置 (background-image-width)**

**border-image-width:27px**

**边框图片的宽度:如果没有设置这个属性，那么宽度默认就是元素的原始的边框宽度。 (即border的宽度)  
细节：1.边框图片的本质是背景，并不会影响元素内容的放置 2.内容只会被容器的border和padding影响  
建议：一般将值设置为原始的边框的宽度**

1. **边框扩展 (background-image-outset)** 
   1. 对边框进行扩展，但会影响周围元素，基本不使用，了解即可

**3、边框图片平铺 (background-image-repeat)**

* 1. Background-image-repeat:repeat/round
  2. repeat会有重复一般都是用round

**repeat:直接重复平铺  
round:将内容缩放进行完整的重复平铺  
border-image-repeat: round;**

**4、边框图片缩写**

**缩写：  
 border-image: source slice / width/outset repeat;  
 border-image: url("../images/border1.png") 27 / 27px /0px round;**

## 3.3 10-HTML5-边框图片(案例)

div{  
 width: 500px;  
 height: auto;  
 border: 10px solid red;  
 margin:100px auto;  
 /\*添加边框图片\*/  
 border-image-source: url("../images/btn\_bg.png");  
 /\*设置受保护的区域大小\*/  
 border-image-slice: 10 fill;   
 /\*设置边框图片的宽度  
 1.明确圆角的大小  
 2.明确受保护的区域的大小\*/  
 border-image-width: 10px;  
 /\*设置背景平铺效果 默认是stretch：拉伸\*/  
 border-image-repeat: stretch;  **默认值：strech，其他组会有纹路**}

# 4 过渡效果transition属性

## 4.1 11-HTML5-过渡

**1、过渡的语法**

**transition: property duration timing-function delay;**

**属性名称 过渡时间 时间函数-速度 延迟**

1. **transition-property:添加过渡效果的样式属性名称  
    transition-property: left;**

**2.transition-duration:过渡效果的耗时 以秒做为单位  
 transition-duration: 2s;  
3.transition-timing-function:设置时间函数--控制运动的速度  
 transition-timing-function: linear;  
4.transition-delay:过渡效果的延迟**

**transition-delay: 2s;**

|  |  |
| --- | --- |
| **值** | **描述** |
| [transition-property](http://www.w3school.com.cn/cssref/pr_transition-property.asp" \o "CSS3 transition-property 属性) | 规定设置过渡效果的 CSS 属性的名称。 |
| [transition-duration](http://www.w3school.com.cn/cssref/pr_transition-duration.asp" \o "CSS3 transition-duration 属性) | 规定完成过渡效果需要多少秒或毫秒。 |
| [transition-timing-function](http://www.w3school.com.cn/cssref/pr_transition-timing-function.asp" \o "CSS3 transition-timing-function 属性) | 规定速度效果的速度曲线。 |
| [transition-delay](http://www.w3school.com.cn/cssref/pr_transition-delay.asp" \o "CSS3 transition-delay 属性) | 定义过渡效果何时开始。 |

|  |  |
| --- | --- |
| **值** | **描述** |
| linear | 规定以相同速度开始至结束的过渡效果（等于 cubic-bezier(0,0,1,1)）。 |
| ease | 规定慢速开始，然后变快，然后慢速结束的过渡效果（cubic-bezier(0.25,0.1,0.25,1)）。 |
| ease-in | 规定以慢速开始的过渡效果（等于 cubic-bezier(0.42,0,1,1)）。 |
| ease-out | 规定以慢速结束的过渡效果（等于 cubic-bezier(0,0,0.58,1)）。 |
| ease-in-out | 规定以慢速开始和结束的过渡效果（等于 cubic-bezier(0.42,0,0.58,1)）。 |
| cubic-bezier(*n*,*n*,*n*,*n*) | 在 cubic-bezier 函数中定义自己的值。可能的值是 0 至 1 之间的数值。 |

**2、transition的简写**

**简写：transition:属性名称 过渡时间 时间函数 延迟  
transition: left 2s linear 0s;**

**3、为多个添加过渡效果**  (以逗号 ，来分割)  
**transition: left 2s linear 0s,background-color 5s linear 0s;**

**4、为所有样式属性添加过渡**

**为所有样式添加过渡效果 all:所有样式**

**transition:all 2s   
1.所有样式的过渡效果一样  
2.效率低下，它会去查询所有添加的样式  
3.建议：以后不要这么写**

## 4.2 12-HTML5-过渡案例-手风琴菜单

**1、过渡分步设置**

transition:all 2s steps(4);

1. **过渡如何做兼容性处理？**

transition:all 2s steps(4);  
-moz-transition: all 2s steps(4);  
-webkit-transition: all 2s steps(4);  
-o-transition: all 2s steps(4);

1. **添加过渡效果有哪些要求？**
2. **一定要设置为哪些css样式添加过渡效果 [transition-property](http://www.w3school.com.cn/cssref/pr_transition-property.asp" \o "CSS3 transition-property 属性)**

**2、一定要设置过渡效果的耗时 [transition-duration](http://www.w3school.com.cn/cssref/pr_transition-duration.asp" \o "CSS3 transition-duration 属性)**

**添加过渡效果:过渡效果只能产生从某个值到另外一个具体的值的过渡**

**类似于display:block就不能设置过渡**

1. **.item:hover >.itemBox{}的理解**
   1. 对类名为item添加hover，靠近就让.itemBox执行设置的样式

# 5 2d&3d变换效果transform属性

## 5.1 13-HTML5-transform2D转换(translate)

1. **Transform:translate()移动语法**
2. **移动是参照元素的左上角**

**b.执行完毕之后会恢复到原始状态**

**语法：transform:translate(X,Y)/translateX()/translateY()  
1.如果只有一个参数就代表x方向 transform:translate(100px)   
2.如果有两个参数就代表x/y方向 transform:translate(100px,100px)**

## 5.2 14-HTML5-transform2D转换(scale)

**1、transform:scale()的语法**

**实现缩放 1指不缩放，>1.01放大 <0.99缩小 参照元素的几何中心  
1.如果只有一个参数，就代表x和y方向都进行相等比例的缩放  
2.如果有两个参数，就代表x/y方向**

**transform: scale(2); 代表X和Y方向都进行以中心放大2倍  
transform: scale(2,1); 代表X方向以中心放大2倍Y方向不变  
transform:scaleX(0.5); 代表X方向以中心缩写0.5倍**

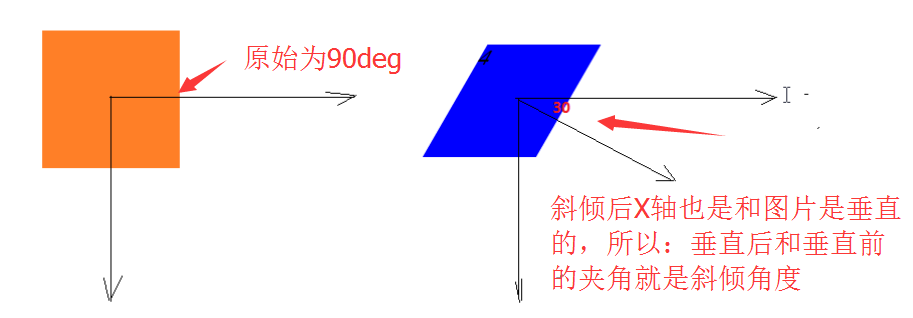
**transform:scaleY(0.5); 代表Y方向以中心缩写0.5倍**

## 5.3 15-HTML5-transform2D转换(rotate)

1. **transform:rotate()的语法**
   1. 始终是沿着默认的Z轴(垂直屏幕的方向)旋转
   2. 顺时针为正，逆时针为负 单位：deg

## 5.4 16-HTML5-transform2D转换(skew)

**1、transform:skew()的语法**



**如果角度为正，则往当前轴的负方向斜切，如果角度为负，则往当前轴的正方向斜切  
transform:skew(-30deg);  
transform:skew(30deg,-30deg);  
设置某个方向的斜切值  
transform:skewX(30deg);  
transform:skewY(30deg);**

## 5.5 17-HTML5-transform-origin

1. **旋转轴心的设置**

**设置旋转轴心  
1.x y (第一个值为X轴，第二个为Y轴)  
2.关键字：left top right bottom center**  
transform-origin: left top;  **(旋转轴心为左上角)**

1. **扑克牌旋转案例：**
   1. 所有图片通过定位重合在一起
   2. 分别在给每张图片设置一个不同的旋转角度，添加过渡效果

## 5.6 18-HTML5-同时添加多个transform属性值

1. **如何同时添加多个transform属性？**
   1. **同时添加移动和旋转**

transform: translateX(700px) rotate(-90deg); 先移动后旋转

* 1. **同时添加旋转在添加移动**

transform: rotate(-90deg) translateX(700px);

**先旋转后移动，坐标系也会跟着旋转，最后移动就是朝着旋转后坐标系移动**

1. **盾牌案例**
   1. **给每个图片设置translate和rotate打乱**
   2. **给每张图片添加过渡效果**
   3. **鼠标靠近就设置trnsform：none**

## 5.7 19-HTML5-实现任意元素居中

1. **实现原理**
   1. 通过定位position：absolute;left:50%;top:50%;
   2. 在通过transform：translate(50%，50%) 50%是参照元素本身的宽高，也可以写具体宽度的值

## 5.8 20-HTML5-3d移动

**1、3d移动translate3d的语法**

**translate3d(X方向的偏移，Y方向的偏移，Z方向的偏移)**

**transform: translate3d(0px,0px,400px);**

**始终垂直屏幕的那个方向就是Z轴的正方形**

## 5.9 21-HTML5-3d缩放

**1、3d放大scale3d的语法**

**scale3d(X方向的缩放，Y方向的缩放，Z方向的缩放)**

**transform: scale3d(2,0.1,1.7);**

**始终垂直屏幕的那个方向就是Z轴的正方形**

## 5.10 22-HTML5-3d旋转

**1、3d放大scale3d的语法**

rotate3d(x,y,z,angle):  
x:代表x轴方向上的一个向量值  
y:代表y轴方向上的一个向量值  
z:代表z轴方向上的一个向量值  
transform: rotate3d(1,1,1,330deg);

**X,y,z都有值旋转原理是：根据他的向量值来旋转的**

