浏览器前缀

1. -ms- -ms-box-shadow IE浏览器专属的CSS属性需添加-ms-前缀
2. -moz- -moz-box-shadow 所有基于Gecko引擎的浏览器（如Firefox）专属的CSS属性需添加-moz-前缀
3. -o- -o-box-shadow Opera浏览器专属的CSS属性需添加-o-前缀
4. -webkit- -webkit-box-shadow 所有基于Webkit引擎的浏览器（如Chrome、Safari）专属的CSS需添加-webkit-前缀

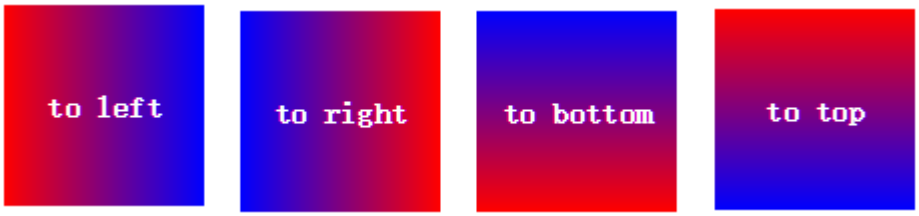
css3 渐变

1. CSS3 渐变（gradient）可以让你在两个或多个指定的颜色之间显示平稳的过渡。 以前，你必须使用图像来实现这些效果，现在通过使用 CSS3 的渐变（gradients）即可实现。此外，渐变效果的元素在放大时看起来效果更好，因为渐变（gradient）是由浏览器生成的。

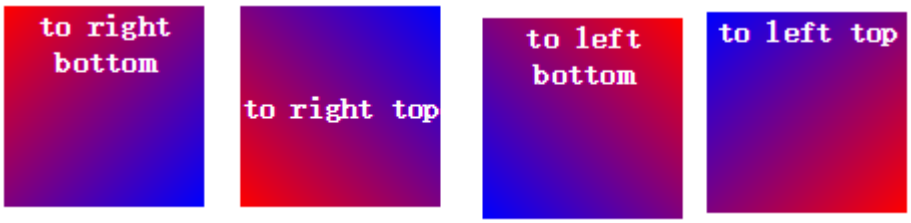
线性渐变

1. 语法:
2. background: linear-gradient(direction, color-stop1, color-stop2, ...);
3.
4. 说明:
5. direction: 默认为to bottom，即从上向下的渐变;
6. stop: 颜色的分布位置，默认均匀分布，例如有3个颜色，各个颜色的stop均为33.33%。

- 示例1：to left、top right、to bottom、to top



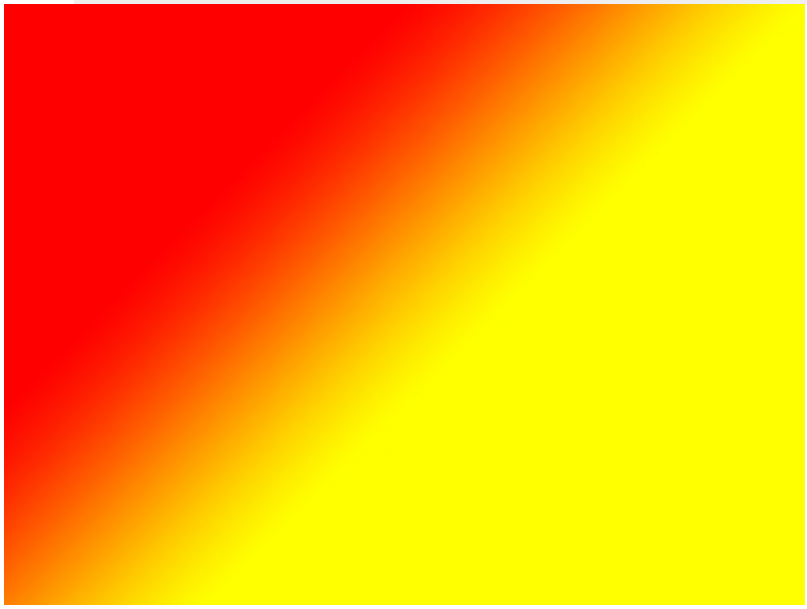
- 示例2：to right bottom、top right top、top left bottom、top left top



- 示例3：使用角度渐变linear-gradient(10deg, red, blue)

- 示例3：使用角度渐变linear-gradient(10deg, red, blue)

1. 角度是指水平线和渐变线之间的角度，逆时针方向计算。换句话说，0deg 将创建一个从下到上的渐变，90deg 将创建一个从左到右的渐变。
2. 但是，请注意很多浏览器(Chrome,Safari,fiefox等)的使用了旧的标准，即 0deg 将创建一个从左到右的渐变，90deg 将创建一个从下到上的渐变。换算公式 90 - x = y 其中 x 为标准角度，y为非标准角度。



径向渐变

1. 径向渐变不同于线性渐变，线性渐变是从“一个方向”向“另一个方向”的颜色渐变，而径向渐变是从“一个点”向四周的颜色渐变
1. 语法：
2. background: radial-gradient(center, shape, size, start-color, ..., last-color);
3. 说明：
4. center：渐变起点的位置，可以为百分比，默认是图形的正中心。
5. shape：渐变的形状，ellipse表示椭圆形，circle表示圆形。默认为ellipse，如果元素形状为正方形的元素，则ellipse和circle显示一样。
6. size：渐变的大小，即渐变到哪里停止，它有四个值。 closest-side：最近边； farthest-side：最远边； closest-corner：最近角； farthest-corner：最远角。

- 示例1：多颜色节点均匀分布

1. div { background: -webkit-radial-gradient(50% 50%, farthest-corner, red, green, blue); }

2. div { background: -webkit-radial-gradient(center, farthest-corner, red, green, blue); }

- 示例2：多颜色节点均匀分布

1. div { background: radial-gradient(circle, red, yellow, green); }
2. div { background: radial-gradient(ellipse, red, yellow, green); }

2. div { background: radial-gradient(ellipse, red, yellow, green); }

- 示例3：设置渐变形状

1. circle：渐变为最大的圆形； ellipse：根据元素形状渐变，元素为正方形是显示效果与circle无异

- 示例4：不同尺寸的渐变

```
1. div { background: radial-gradient(60% 40%, closest-side, blue, green, yellow, black); }
2. div { background: radial-gradient(60% 40%, farthest-side, blue, green, yellow, black); }
3. div { background: radial-gradient(60% 40%, closest-corner, blue, green, yellow, black); }
4. div { background: radial-gradient(60% 40%, farthest-corner, blue, green, yellow, black);}
```

重复性渐变

1.重复性线性渐变

```
1. div { background: repeating-linear-gradient(red, yellow 10%, green 20%); }
```

2.重复性径向渐变

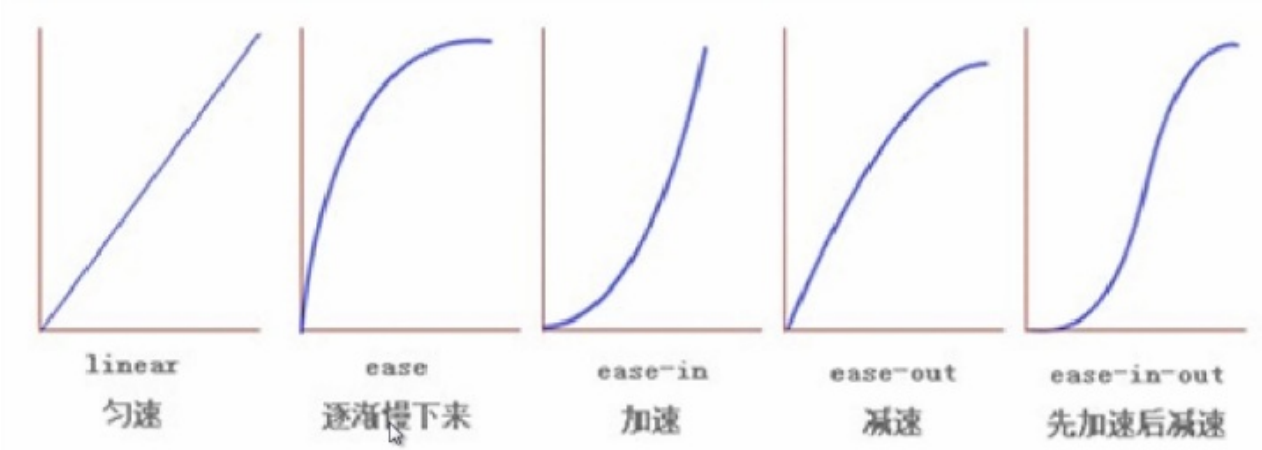
```
1. div { background: repeating-radial-gradient(red, yellow 10%, green 20%); }
```

过渡

1. css3的transition允许css的属性值在一定的时间区间内平滑地过渡。这种效果可以在鼠标单击、获

得焦点、被点击或对元素任何改变中触发，并圆滑地以动画效果改变CSS的属性值

1. transition-property：检索或设置对象中的参与过渡的属性
2. transition-duration：检索或设置对象过渡的持续时间
3. transition-delay：检索或设置对象延迟过渡的时间
4. transition-timing-function：检索或设置对象中过渡的动画类型
    1. 检索或设置对象中过渡的动画类型
    2. http://cubic-bezier.com/



简写：transition:all/具体属性值 运动时间s/ms 延迟时间s/ms 动画类型

案例

变形属性：transform

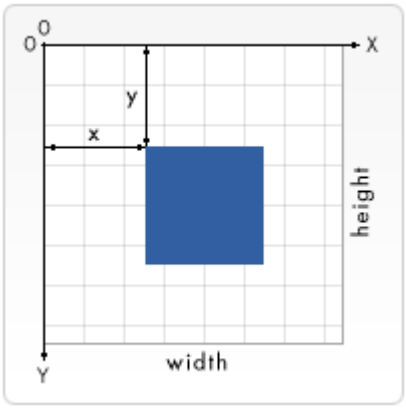
1. transform翻译成汉语具有"变换"或者"改变"的意思。
2. 通过此属性具有非常强大的功能，比如可以实现元素的位移、拉伸或者旋转等效果
3. 最能体现transform 属性强大实力的是实现元素的3D变换效果。

2D

2D

1. 2D变换，是在一个平面对元素进行的操作。
2. 可以对元素进行水平或者垂直位移、旋转或者拉伸.

- 明确一下坐标系



1. 对上面坐标系简单分析如下：
2. （1）.默认状态下，x轴是水平的，向右为正。
3. （2）.默认状态下，y轴是垂直的，向下为正，这与传统的数学坐标系不同。

2D功能函数

2D位移 translate()

- 将元素向指定的方向移动，类似于position中的relative。
- 水平移动：向右移动translate(tx,0)和向左移动translate(-tx,0)；
- 垂直移动：向上移动translate(0,-ty)和向下移动translate(0,ty)；
- 对角移动：右下角移动translate(tx,ty)、右上角移动translate(tx,-ty)、左上角移动translate(-tx,-ty)和左下角移动translate(-tx,ty)。

2D缩放scale()

- 让元素根据中心原点对对象进行缩放。默认的值1。因此0.01到0.99之间的任何值，使一个元素缩小；而任何大于或等于1.01的值，让元素显得更大。

- 缩放scale()函数和translate()函数的语法非常相似，他可以接受一个值，也可以同时接受两个值，如果只有一个值时，其第二个值默认与第一个值相等。例如，scale(1,1)元素不会有任何变化，而scale(2,2)让元素沿X轴和Y轴放大两倍。

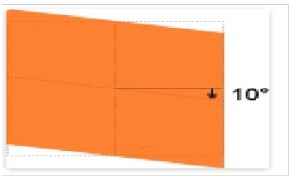
- scaleX()：相当于scale(sx,1)。表示元素只在X轴（水平方向）缩放元素，其默认值是1。
- scaleY()：相当于scale(1,sy)。表示元素只在Y轴（纵横方向）缩放元素，其默认值是 1 。

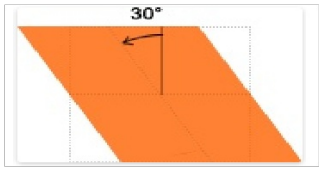
3、rotate()

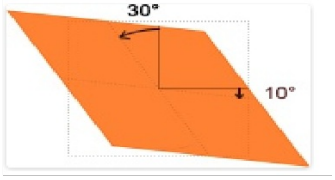
1. 旋转rotate()函数通过指定的角度参数对元素根据对象原点指定一个2D旋转。它主要在二维空间内进行操作，接受一个角度值，用来指定旋转的幅度。如果这个值为正值，元素相对原点中心顺时针旋转；如果这个值为负值，元素相对原点中心逆时针旋转。
2. rotateX() 方法，元素围绕其 X 轴以给定的度数进行旋转
3. rotateY() 方法，元素围绕其 Y 轴以给定的度数进行旋转

4、skew()

1. 倾斜skew()函数能够让元素倾斜显示。它可以将一个对象以其中心位置围绕着X轴和Y轴按照一定的角度倾斜。
2. 一个参数时：表示水平方向的倾斜角度；
3. 两个参数时：第一个参数表示水平方向的倾斜角度，第二个参数表示垂直方向的倾斜角度



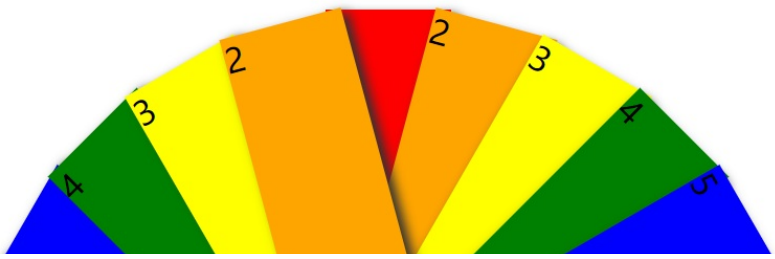


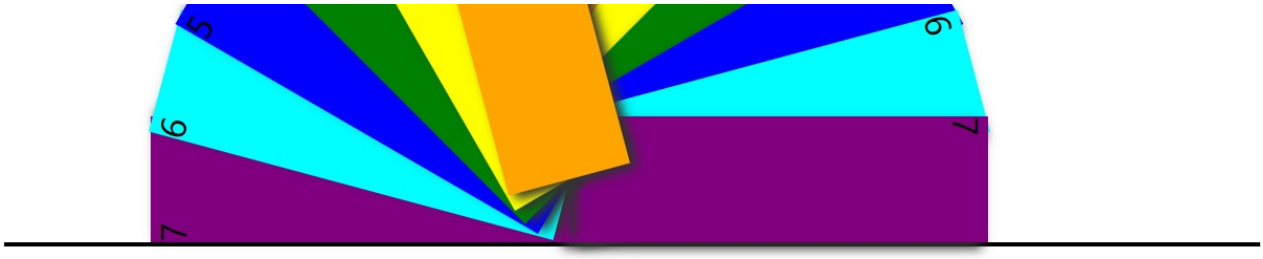


变形原点

1. transform-origin
2. transform-origin是变形原点，也就是该元素围绕着那个点变形或旋转，该属性只有在设置了transform属性的时候起作用；
3. 因为我们元素默认基点就是其中心位置，换句话说我们没有使用transform-origin改变元素基点位置的情况下，transform进行的rotate,translate,scale,skew等操作都是以元素自己中心位置进行变化的。

2d案例




5
6
7
6
7