1. **背景渐变**

1、背景色的渐变线性渐变background: gradient（有兼容性的问题-webkit- -moz-）【老式写法】



div{

width: 500px;

height:500px;

margin:10px;

float: left;

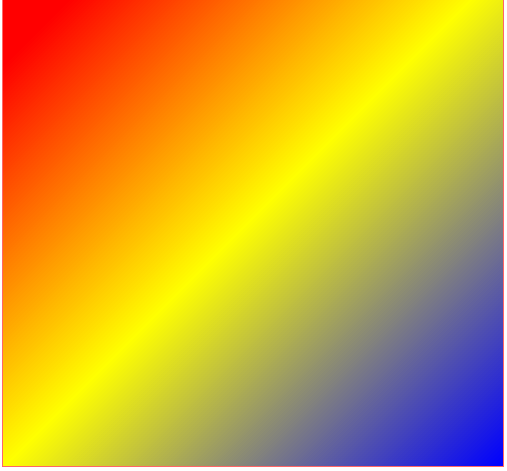
border: 1px solid #f66;

background: -webkit-gradient(linear,0 0,0 100%,from(#f00),to(#0000FF));

background: -moz-gradient(linear,0 0,0 100%,from(#f00),to(#0000FF));

background:-webkit-gradient(linear,left top,left bottom,from(#f00),to(#00f));

}



div{

width: 500px;

height:500px;

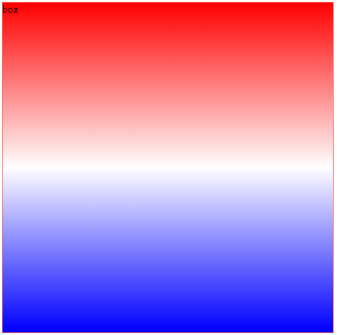
margin:10px;

border: 1px solid #f66;

background:-webkit-gradient(linear,0% 0%,100% 100%,color-stop(0.1,rgb(255,0,0)),color-stop(0.5,rgb(255,255,0)),color-stop(1,rgb(0,0,255)));

}

1、线性渐变



div{

width: 500px;

height:500px;

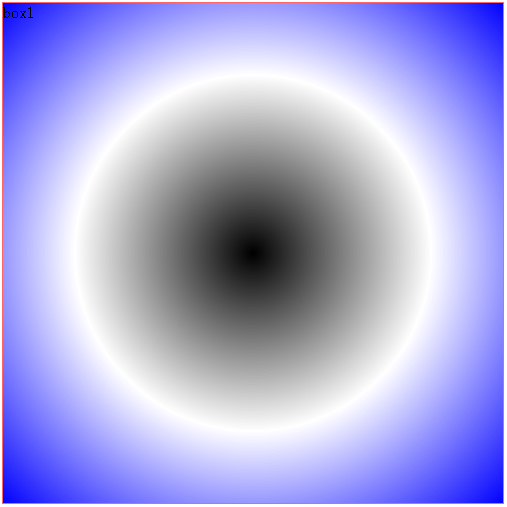
margin:10px;

background: -webkit-linear-gradient(top,red,white,blue);

background: -moz-linear-gradient(top,red,white,blue);

}

1. 径向渐变



div{

width: 500px;

height:500px;

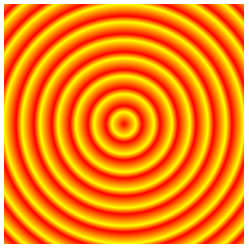
margin:10px;

background: -webkit-radial-gradient(50% 50%,black,white,blue);

background: -moz-radial-gradient(50% 50%,black,white,blue);

}

1. background:repeating-radial-gradient(*center,shape size, start-color, ..., last-color*)重复径向渐变



div{

width:200px;

height:200px;

padding: 20px;

background:-webkit-repeating-radial-gradient(circle contain,#f00,#ff0 10%,#f00 15%)

}

1. **过渡transition**

transition-property规定应用过渡的 CSS 属性的名称。

transition-duration定义过渡效果花费的时间。默认是 0。

transition-timing-function规定过渡效果的时间曲线。默认是 "ease"。

ease规定慢速开始，然后变快，然后慢速结束的过渡效果

linear规定以相同速度开始至结束的过渡效果

ease-in规定以慢速开始的过渡效果

ease-out规定以慢速结束的过渡效果

ease-in-out规定以慢速开始和结束的过渡效果

transition-delay规定过渡效果何时开始。默认是 0。

transition ：[ transition-property ] || [ transition-duration ] || [ transition-timing-function ] || [ transition-delay ]简写属性，用于在一个属性中设置四个过渡属性。

**注释：**CSS3 过渡是元素从一种样式逐渐改变为另一种的效果。要实现这一点，必须规定两项内容：

1.规定把效果添加到哪个 CSS 属性上

2.规定效果的时长

Internet Explorer 10、Firefox、Chrome 以及 Opera 支持 transition 属性。

Safari 需要前缀 -webkit-。

Internet Explorer 9 以及更早的版本，不支持 transition 属性。Chrome 25 以及更早的版本，需要前缀 -webkit-。

当鼠标光标移动到该元素时，它逐渐改变它原有样式

**宽度的过渡案例：（当鼠标划上时宽度由小逐渐变大）**



div{

width:100px;

height:100px;

background:red;

transition:width 2s;

-webkit-transition:width 2s;

}

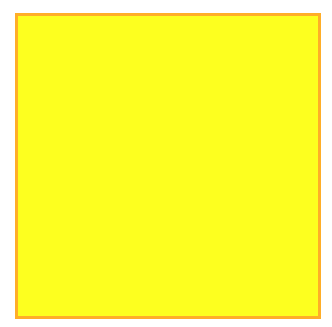
div:hover{

width:300px;

}

<div></div>

**多属性过渡**

div{

width: 300px;

height: 300px;

background: #fdff1f;

border: #ffae28 solid 3px;

margin: 100px auto;

transition: all 0.5s;

}

div:hover{

width: 500px;

height: 400px;

border-radius: 15px;

background: #ff67c2;

}

<div></div>

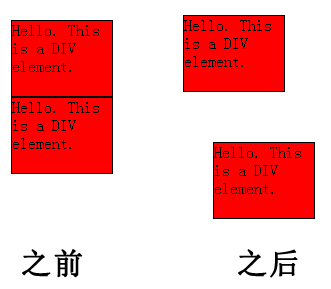
1. **2D转换transform**
2. **translate( ,)偏移方法**

translate()根据左(X轴)和顶部(Y轴)位置给定的参数，从当前元素位置移动。

Internet Explorer 10, Firefox, 和 Opera支持transform 属性.

Chrome 和 Safari 要求前缀 -webkit- 版本.

**注意：** Internet Explorer 9 要求前缀 -ms- 版本.



div {

width:100px;

height:75px;

background-color:red;

border:1px solid black;

}

#div2{

transform:translate(30px,50px);

-ms-transform:translate(30px,50px); IE 9

-webkit-transform:translate(30px,50px); \*/

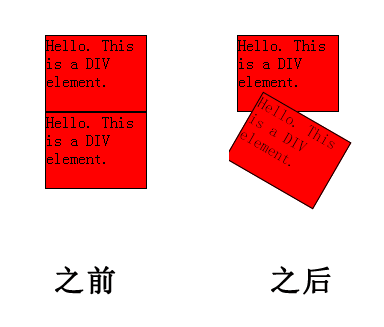
}

<div>Hello. This is a DIV element.</div>

<div id="div2">Hello. This is a DIV element.</div>

1. **rotate()旋转方法**

rotate()方法，在一个给定度数顺时针旋转的元素。负值是允许的，这样是元素逆时针旋转。



div {

width:100px;

height:75px;

background-color:red;

border:1px solid black;

}

#div2{

transform:rotate(30deg);

-ms-transform:rotate(30deg); -webkit-transform:rotate(30deg);

}

<div>Hello. This is a DIV element.</div>

<div id="div2">Hello. This is a DIV element.</div>

1. **scale( ,)缩放方法**

scale()方法，该元素增加或减少的大小，取决于宽度（X轴）和高度（Y轴）的参数：

div 元素的宽度是原始大小的两倍，高度是原始大小的三倍。

div {

margin: 150px;

width: 200px;

height: 100px;

background-color: yellow;

border: 1px solid black;

border: 1px solid black;

-ms-transform: scale(2,3); /\* IE 9 \*/

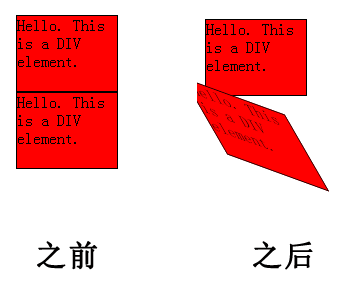
-webkit-transform: scale(2,3); /\* Safari \*/

transform: scale(2,3); /\* 标准语法 \*/

}

1. **skew( ,)扭曲方法**

skew()方法，该元素会根据横向（X轴）和垂直（Y轴）线参数给定角度：



div{

width:100px;

height:75px;

background-color:red;

border:1px solid black;

}

#div2{

transform:skew(30deg,20deg);

-ms-transform:skew(30deg,20deg);

-webkit-transform:skew(30deg,20deg);

}

<div>Hello. This is a DIV element.</div>

<div id="div2">Hello. This is a DIV element.</div>

1. **360度旋转案例**



img{

width: 320px;

height: 350px;

display: block;

margin: 40px auto;

border-radius: 50%;

border: #2e51ff solid 2px;

transition: all 5s; /\*all来表示所有的属性都变化\*/

}

img:hover{

transform: rotate(360deg);

}

<img src="img/img1.jpg">

1. **translateX(x) 定义转换，只是用 X 轴的值。**
2. **translateY(y) 定义转换，只是用 Y 轴的值。**
3. **transform-origin设置旋转元素的基点位置： transform-origin: 0% 0%;**
4. **transition-delay延迟**