css3动画效果小结

Css3的动画功能

1. Transition(过度属性)

2、Animation(动画属性)

3、Transform(2D/3D转换属性)

1、Transition:<过渡属性名称> <过渡时间> <过渡模式>

如-webkit-transition:color 1s;

等同于:

-webkit-transition-property:color;

-webkit-transition-duration:1s;

多个属性的过渡效果：

方法1：-webkit-transition:<属性1> <时间1> ,<属性2> <时间2> ,。。。

方法2：

-webkit-transition:<属性1> <时间1>;

-webkit-transition:<属性2> <时间2>;

Transition-timing-function:

Ease:缓慢开始，缓慢结束

Liner:匀速

Ease-in:缓慢开始

Ease-out:缓慢结束

Ease-in-out:缓慢开始，缓慢结束（和ease稍有区别）

实例：

1. 动画属性animation

animation: *name* *duration* *timing-function* *delay* *iteration-count* *direction*;

|  |  |
| --- | --- |
| **值** | **描述** |
| *[animation-name](http://www.w3school.com.cn/cssref/pr_animation-name.asp" \o "CSS3 animation-name 属性)* | 规定需要绑定到选择器的 keyframe 名称。。 |
| *[animation-duration](http://www.w3school.com.cn/cssref/pr_animation-duration.asp" \o "CSS3 animation-duration 属性)* | 规定完成动画所花费的时间，以秒或毫秒计。 |
| *[animation-timing-function](http://www.w3school.com.cn/cssref/pr_animation-timing-function.asp" \o "CSS3 animation-timing-function 属性)* | 规定动画的速度曲线。 |
| *[animation-delay](http://www.w3school.com.cn/cssref/pr_animation-delay.asp" \o "CSS3 animation-delay 属性)* | 规定在动画开始之前的延迟。 |
| *[animation-iteration-count](http://www.w3school.com.cn/cssref/pr_animation-iteration-count.asp" \o "CSS3 animation-iteration-count 属性)* | 规定动画应该播放的次数。 |
| *[animation-direction](http://www.w3school.com.cn/cssref/pr_animation-direction.asp" \o "CSS3 animation-direction 属性)* | 规定是否应该轮流反向播放动画。 |

注释：Internet Explorer 9 以及更早的版本不支持 animation 属性。

@keyframes *animationname* {*keyframes-selector* {*css-styles*;}}

|  |  |
| --- | --- |
| **值** | **描述** |
| *animationname* | 必需。定义动画的名称。 |
| *keyframes-selector* | 必需。动画时长的百分比。  合法的值：   * 0-100% * from（与 0% 相同） * to（与 100% 相同） |
| *css-styles* | 必需。一个或多个合法的 CSS 样式属性。 |

以百分比来规定改变发生的时间，或者通过关键词 "from" 和 "to"，等价于 0% 和 100%。

0% 是动画的开始时间，100% 动画的结束时间。

Animation:mymove 5s infinite;

@keyframes mymove{

from{ top:0px; }

to{ top:200px; }

}

还可以这么写：

@keyframes mymove{

0%{ top:0px; }

25%{ top:200px; }

50%{ top:100px; }

75%{ top:200px; }

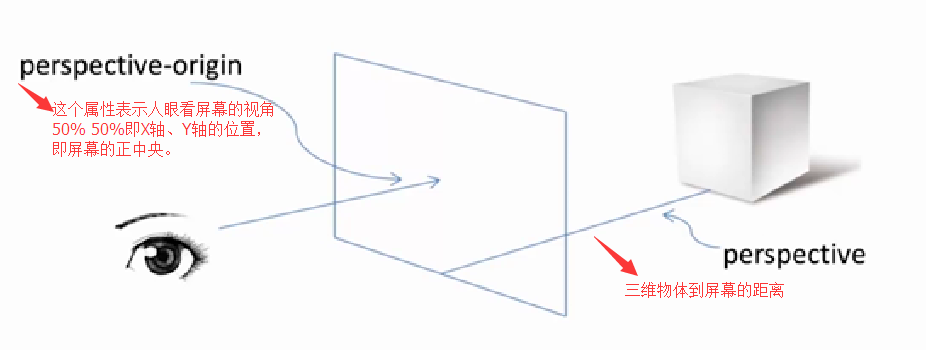
100%{ top:0px; }

}

3、设置3D场景

-webkit-perspective:800;(单位为像素)--即三维物体距离屏幕的距离。

-webkit-perspective-origin:50% 50%;(这个属性代表了人眼观察的视野。50% 50%为X轴、Y轴相应的位置，即屏幕的正中央。)



使用transform属性调整元素：-webkit-transform-style:-webkit-perserve-3d;(这个属性是告诉浏览器我们是在一个三维空间中对元素进行操作)

1. Translate(移动距离)

translateX（x px）

translateY（y px）

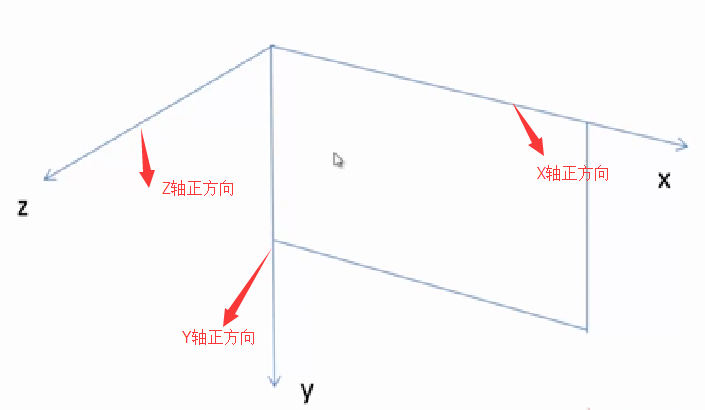
translateZ（z px）

1. Rotate(旋转角度)

rotateX(x deg)

rotateY(y deg)

rotateZ(z deg)



Transform：rotate（45deg）

rotateX:向屏幕上边沿向内旋转为正方向。

rotateY:向屏幕竖直向下为正方向。

rotateZ:向屏幕外为正方向。

一个div块，右边沿向屏幕内旋转45deg,即应设置为：Transform：rotateY（45deg）。

实例：

使用transform-origin属性调整旋转中心。

默认旋转中心点为div盒子的正中心。

这个旋转中心是可以改变的：

X轴：left、center、right.

Y轴：top、center、bottom.

Z轴：length px(一个长度值)。