

Animation

一、兼容性

Chrome 13+、Firefox 5+、Safari 5.1+、Opera 12+、IE10+均支持 animation

常用前缀：

Firefox：-moz

Safari 和 Chrome：-webkit

Opera：-o

二、语法介绍

1 定义 keyframes 语法

@keyframes 规则用于创建动画。在 @keyframes 中规定某项 CSS 样式，就能创建由当前样式逐渐改为新样式的动画效果。

1.1 语法：

```
@keyframes animation-name{keyframes-selector {css-styles;}}
```

1.2 值和描述

animation-name：(必需)指定要绑定到选择器的关键帧的名称，即动画的名称。

keyframes-selector：(必需)动画时长的百分比。

建议：在 CSS3 中，我们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%，其中，0%是动画的开始时间。100%是动画的结束时间。

例如：

```
@keyframes colorchange {
```

```

0%    { background-color: #00F; /* from: blue */ }

25%   { background-color: #F00; /* red      */ }

50%   { background-color: #0F0; /* green    */ }

75%   { background-color: #F0F; /* purple   */ }

100%  { background-color: #00F; /* to: blue  */ }

}

```

```

@-webkit-keyframes colorchange {

    0%    { background-color: #00F; /* from: blue */ }

    25%   { background-color: #F00; /* red      */ }

    50%   { background-color: #0F0; /* green    */ }

    75%   { background-color: #F0F; /* purple   */ }

    100%  { background-color: #00F; /* to: blue  */ }

}

```

解释：这个例子中定义了 background-color 这个属性。其中 0%这可以用 from 关键字来代替，100%可以用 to 代替。即：

```

@keyframes colorchange {

    from { background-color: #00F; /* from: blue */ }

    25%  { background-color: #F00; /* red      */ }

    50%  { background-color: #0F0; /* green    */ }

    75%  { background-color: #F0F; /* purple   */ }

    to   { background-color: #00F; /* to: blue  */ }

}

```

2 调用动画语法

2.1 语法：

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

例：`animation:changeColor 6s ease-in-out .2s infinite alternate;`

2.2 属性值

`animation` 属性是一个简写属性，用于设置动画的属性。

`animation-name`：该属性主要是用来调用@keyframes 定义好的动画。

`animation-duration`：动画指定需要多久完成，默认是 0。

`animation-timing-function`：设置动画的速度曲线，默认是“ease”。

`animation-delay`：设置在动画将启动之前经过的延迟间隔，默认是 0。

`animation-iteration-count`：设置动画周期的播放次数，默认是 1。

`animation-direction`：设置对象动画在循环中是否反向运动，默认是“normal”。

3 属性值详解

3.1 animation-duration

指定动画需要多久完成。请始终规定 `animation-duration` 属性，否则时长为 0，就不会播放动画了。

3.2 animation-timing-function

`animation-timing-function` 值

`ease`：默认值，元素样式从初始状态过渡到终止状态时速度由快到慢，逐渐减慢。

linear：元素样式从初始状态过渡到终止状态速度是恒速。

ease-in：元素样式从初始状态过渡到终止状态时，速度越来越快，呈一种加速状态。常称这种状态为渐显效果。

ease-out：元素状态从初始状态过渡到终止状态时，速度越来越慢。呈一种减速状态。常称这种状态为渐隐效果。

ease-in-out：元素样式从初始状态到终止状态时，先加速再减速。常称这种效果为渐显渐隐效果。

3.3 animation-iteration-count

指定动画的播放次数，取值可以是：

n：即具体的数字，表示具体的播放次数；

infinite：表示无数次播放。

3.4 animation-direction

有四个值：

normal：默认，从 0%到 100%；

reverse：动画从 100%执行到 0%；

alternate：动画在 0%到 100%之间往复执行；

alternate-reverse 与 alternate 一致，不过是从 100%开始。

三、综合示例

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta charset="utf-8">
```

```
<title>animation</title>
```

```
<style>
```

```
@keyframes slide{
```

```
    0%    {background:red; left:0px; top:0px;}
```

```
    100%  {background:yellow; left:200px; top:0px;}
```

```
}
```

```
@-moz-keyframes slide{
```

```
    0%    {background:red; left:0px; top:0px;}
```

```
    100%  {background:yellow; left:200px; top:0px;}
```

```
}
```

```
@-webkit-keyframes slide{
```

```
    0%    {background:red; left:0px; top:0px;}
```

```
    100%  {background:yellow; left:200px; top:0px;}
```

```
}
```

```
@-o-keyframes slide{
```

```
    0%    {background:red; left:0px; top:0px;}
```

```
    100%  {background:yellow; left:200px; top:0px;}
```

```
}
```

```
.box{
```

```
    width:100px;
```

```
    height:100px;
```

```
background:red;

position:relative;

animation:slide 5s linear 2s infinite alternate;

/* Firefox: */

-moz-animation:slide 5s linear 2s infinite alternate;

/* Safari and Chrome: */

-webkit-animation:slide 5s linear 2s infinite alternate;

/* Opera: */

-o-animation:slide 5s linear 2s infinite alternate;

}

</style>

</head>

<body>

    <div class="box"> </div>

</body>

</html>
```

说明：这段代码首先用@keyframes 定义了一个名称为“slide”的动画，效果是使一个背景为红色的元素从左向右滑动 200px，同时背景色变为黄色；然后将这个动画效果和 box 类绑定，使这个动画：先延迟 2s、然后无限循环播放此动画、5s 完成一次动画、每次动画都以匀速演示、奇数次动画正向播放、偶数次动画逆向播放。