



CSS3动画



今日主打

CSS3 动画

CSS3 animation

CSS3 @keyframes

CSS3 will-change

CSS3 动画

动画 (animation)

anima——灵魂、animate——赋予生命

动画可以定义为使用绘画的手法，创造生命运动的艺术。

视觉暂留原理

人类具有“**视觉暂留**”的特性，人的眼睛看到一幅画或一个物体后，在0.34秒内不会消失。

动画原理

通过把人物的表情、动作、变化等**分解**后画成许多动作瞬间的画幅，利用视觉暂留的原理，在一幅画还没有消失前播放下一幅画，就会给人造成一种流畅的**视觉变化**效果。

CSS3 动画

兼容性

IE10+、FireFox16+、Chrome43+、Safari9+、Opera30+、Android (-webkit-)

CSS3 动画

使元素从一种样式逐渐变化为另一种样式的效果。

CSS3 animation

animation-name属性

检索或设置对象所应用的动画名称。

语法

animation-name: keyframename | none;

参数说明

keyframename：指定要绑定到选择器的关键帧的名称；

none：指定有没有动画（可用于覆盖从级联的动画）。

CSS3 animation

animation-duration属性

检索或设置对象动画的持续时间

语法

animation-duration: time;

参数说明

time指定动画播放完成花费的时间。默认值为 0，意味着没有动画效果

CSS3 animation

animation-timing-function属性

检索或设置对象动画的过渡类型

语法

animation-timing-function:

**ease | linear | ease-in | ease-out | ease-in-out | step-start | step-end |
steps(<integer>[, [start | end]]?) | cubic-bezier(<number>, <number>,
<number>, <number>);**

CSS3 animation

参数说明

linear : 线性过渡。等同于贝塞尔曲线(0.0, 0.0, 1.0, 1.0)

ease : 平滑过渡。等同于贝塞尔曲线(0.25, 0.1, 0.25, 1.0)

ease-in : 由慢到快。等同于贝塞尔曲线(0.42, 0, 1.0, 1.0)

ease-out : 由快到慢。等同于贝塞尔曲线(0, 0, 0.58, 1.0)

ease-in-out : 由慢到快再到慢。等同于贝塞尔曲线(0.42, 0, 0.58, 1.0)

step-start : 等同于 steps(1, start)

step-end : 等同于 steps(1, end)

steps(<integer>[, [start | end]]?) : 接受两个参数的步进函数。第一个参数必须为正整数, 指定函数的步数。第二个参数取值可以是start或end, 指定每一步的值发生变化的时间点。第二个参数是可选的, 默认值为end。

cubic-bezier(<number>, <number>, <number>, <number>) : 特定的贝塞尔曲线类型, 4个数值需在[0, 1]区间内

CSS3 animation

animation-delay属性

检索或设置对象动画的延迟时间。

语法

animation-delay: time;

参数说明

可选。定义动画开始前等待的时间，以秒或毫秒计。默认值为0。

CSS3 animation

animation-iteration-count属性

检索或设置对象动画的循环次数。

语法

animation-iteration-count: infinite | <number>;

参数说明

<number>为数字，其默认值为“1”；infinite为无限次数循环。

CSS3 animation

animation-direction属性

检索或设置对象动画在循环中是否反向运动。

语法

`animation-direction: normal | reverse | alternate | alternate-reverse | initial | inherit;`

参数说明

`normal` : 正常方向 ;

`reverse` : 反方向运行 ;

`alternate` : 动画先正常运行再反方向运行 , 并持续交替运行 ;

`alternate-reverse` : 动画先反运行再正方向运行 , 并持续交替运行。

CSS3 animation

animation-fill-mode属性

规定当动画不播放时（当动画完成或当动画有延迟未开始播放时），要应用到元素的样式。

语法

`animation-fill-mode: none | forwards | backwards | both | initial | inherit;`

参数说明

`none`：默认值。不设置对象动画之外的状态；

`forwards`：设置对象状态为动画结束时的状态；

`backwards`：设置对象状态为动画开始时的状态；

`both`：设置对象状态为动画结束或开始的状态。

CSS3 animation

animation-play-state属性

指定动画是否正在运行或已暂停。

语法

`animation-play-state: paused | running;`

参数说明

`paused` : 指定暂停动画 ;

`running` : 默认值 , 指定正在运行的动画。

CSS3 animation

animation属性

复合属性。检索或设置对象所应用的动画特效。

语法

**animation: name duration timing-function delay iteration-count
direction fill-mode play-state;**

举个栗子

CSS3 @keyframes

Keyframes定义

关键帧，可以指定任何顺序排列来决定Animation动画变化的关键位置。

使用说明

使用@keyframes规则创建动画，通过逐步改变从一个CSS样式设定到另一个。

在动画过程中可以通过@keyframes规则多次更改CSS样式的设定。

CSS3 @keyframes

语法

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

参数说明

animationname : 必写项 , 定义animation的名称。

keyframes-selector : 必写项 , 动画持续时间的百分比 , 0-100%、from (0%)、to (100%)

css-styles : 必写项 , 一个或多个合法的CSS样式属性。

CSS3 will-change

目标

增强页面渲染性能。

CPU和GPU

CPU即中央处理器，解释计算机指令以及处理计算机软件中的数据。

GPU即图形处理器，专门处理和**绘制图形**相关的硬件。GPU是专为执行复杂的数学和几何计算而设计的，有了它，CPU就从图形处理的任务中解放出来，可以执行其他更多的系统任务。

硬件加速

在计算机中把计算量非常大的工作分配给专门的硬件来处理，减轻CPU的工作量。

CSS3 will-change

现状

CSS的动画、变形、渐变并不会自动的触发GPU加速，而是使用浏览器稍慢的软件渲染引擎。

在transition，transform和animation的世界里，应该卸载进程到GPU以加快速度。只有3D变形会有自己的layer，2D变形不会。

translateZ() (or translate3d()) Hack

为元素添加没有变化的3D变形，骗取浏览器触发硬件加速。

代价

这种情况通过向它自己的层叠加元素，占用RAM和GPU存储空间。

CSS3 will-change

will-change

提前通知浏览器元素将要做什么动画，让浏览器提前准备合适的优化设置。

语法

`will-change: auto|scroll-position|contents|<custom-ident>|<animateable-feature>;`

兼容性

IE13+、FireFox47+、Chrome49+、Safari9.1+、Opera39+、IOS9.3+、Android52+

CSS3 will-change

关键词说明

auto：此关键字表示没有特定的意图，适用于它通常所做的任何启发式和优化。

scroll-position：表示将要改变元素的滚动位置。

contents：表示将要改变元素的内容。

<custom-ident>：明确指定将要改变的属性与给定的名称。

<animateable-feature>：可动画的一些特征值，比如left、top、margin等。

举个栗子

使用注意（勿滥用、提前申明、remove）