

HTML5 过渡 动画

五、过渡与动画

过渡

动画

五、过渡与动画

过渡

1. `transition: property duration delay timing-function`

`transition-property` : 过渡属性的名称

`none` : 没有过渡属性

`all` : 所有的属性都过渡(默认值)

`property` : 具体的属性名称

`transition-duration` : 过渡属性花费的时间 s/ms

`transition-delay` : 过渡效果延时时间 s/ms

`transition-timing-function` : 过渡效果速度曲线

`linear / ease / ease-in / ease-out / ease-in-out`

1. 过渡完成事件:
2. Webkit内核: `obj.addEventListener('webkitTransitionEnd', function() {});`
3. 标准: `obj.addEventListener('transitionend', function() {});`

动画

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

animation-name: 动画名称

keyframename 自定义的名字

none 无动画效果

animation-duration: 动画执行时间

animation-delay: 动画效果延迟时间

animation-timing-function: 动画速度曲线

linear / ease / ease-in / ease-out / ease-in-out

animation-iteration-count: 动画执行次数

次数数值 / infinite 无限重复

animation-direction: 动画执行方向

normal 正常 / alternate 动画轮流反向播放

animation-play-state: 动画执行状态

paused 暂停动画 / running 运行动画

animation-fill-mode: 动画执行过程效果是否可见

none 不改变 (默认)

forwards 当动画完成后, 保持最后一个属性值 (在最后一个关键帧中定义)

backwards 在 animation-delay 所指定的一段时间内, 在动画显示之前, 应用开始属性值 (在第一个关键帧中定义)

both 向前和向后填充模式都被应用, forwards 与 backwards 的

1. `@keyframes`: 规定动画
2. `@keyframes animationname {keyframes-selector {css-styles;}}`

animationname 定义动画的名称。

keyframes-selector 动画时长的百分比。

0-100%

from (与 0% 相同)

to (与 100% 相同)

可以只有to

css-styles 一个或多个合法的 CSS 样式属性。

1. 动画事件:
- 2.
3. 动画开始触发: `obj.addEventListener("webkitAnimationStart", fn);`
4. `obj.addEventListener("animationstart", fn);`
5. 动画执行过程中触发: `obj.addEventListener("webkitAnimationIteration", fn);`
6. `obj.addEventListener("animationiteration", fn);`
7. 动画结束触发: `obj.addEventListener('webkitAnimationEnd', fn);`
8. `obj.addEventListener('animationend', fn);`

HTML5中的 requestAnimationFrame 动画优化

1. `window.requestAnimFrame = (function () {`
2. `return window.requestAnimationFrame ||`
3. `window.webkitRequestAnimationFrame ||`
4. `window.mozRequestAnimationFrame ||`
5. `function (callback) {`
6. `window.setTimeout(callback, 1000 / 60);`
7. `};`
8. `})();`