

web上渐进使用jQuery Mobile中animate相关CSS

这篇文章发布于 2012年11月2日, 星期五, 12:30, 归类于 Web综合。 阅读 82013 次, 今日 13 次 21 条评论

by zhangxinxu from <http://www.zhangxinxu.com>

本文地址: <http://www.zhangxinxu.com/wordpress/?p=2748>

一、关于animate.css

在介绍主人公之前, 先说说他的亲戚。

有个叫“蛋一灯(Dan Eden)”的人弄了个名叫 `animate.css` 的开源项目, 实际上就是使用CSS3 animation 实现了各种蛋疼或菊紧的动画效果。

官方主页地址: <http://daneden.me/animate>

github地址: <https://github.com/daneden/animate.css>

该项目不错之处在于, 你可以自定义你自己需要的CSS效果, 自定义页面访问[点击这里](#)。

更新于2014年3月18日

该项目已经迁移至Github有些时日了, 上面的自定义页面已经不存在了, 可以[访问这里](#)查看一些效果。

做移动开发时候很方便。使用Phonegap开发, 性能问题也不用担心了, 很流畅, 很有用。



在目标浏览器下, 鼠标移到文字上即可预览到效果, 勾选与去勾选, 点击最下面的”Build it”按钮, 可以生成你自己需要的效果的CSS文件。如下示意:



然而, 从实际的角度来讲, 这个项目基本上属于华丽包装的中国式礼品, 除了让我们学习如何写效果相对应的CSS3代码。换句话说, 有技术学习的价值, 缺少实际应用的价值! 因此, 本文的主人公不是他, 而是她!

二、jQuery Mobile中animate.css

jQuery Mobile中也有animate相关的CSS代码(没有animate.css之类名称, 标题这么写是为了与前一个标题呼应🙄)。

虽然同是animation相关动画的CSS代码，但是，jQuery Mobile中的这个显然更简单更实用，更值得一说！

我抽了点拉便便的功夫，把jQuery Mobile项目中的这部分CSS代码提出来了，放在了一个独立的CSS文件中（有改动，兼容、命名、属性等），您可以狠狠地点击[这里](#)查看或下载：[animate.css](#)

其中，相关animation动画有：spin, fade, pop, slide, flip, turn, flow. 至于具体分别指代什么，下面会讲，稍安勿躁！

有了animate.css, 实现一些动画效果那是非常的简单——几个class类名的切换而已，就算是不懂JavaScript的人，松松的几十分钟，也可以弄出效果来。

对于大多数同行，虽未实践过animation动画，但肯定也有所耳闻，这些CSS3属性IE6~9甚至Opera浏览器都不支持，顶多手机项目或者iPad项目上用用，至于传统Web上，啧啧，估计吃翔的可能性居多！

所以，可能“米娜桑^①”现在对于animation等东西更多的是观望，了解或等待之类！No, no, no, 诸位，今天我就要告诉大家，就算是需要支持IE6浏览器，面向各类普通用户的传统web项目，animate也是可以渐进使用的，而且使用成本相当相当的低的哦😎！哟，还不信，咱们骑驴看唱本——走着瞧！

注解：① 大家的意思。

三、热热身 – slide相关CSS与幻灯片切换效果

面对新事物，鲜活的实例远比生僻头大的代码、陈述之类更加吸引人。

您可以狠狠地点击[这里](#)：[slide动画与幻灯片切换浏览效果demo](#)

在Chrome浏览器或者FireFox浏览器下（或360，遨游，搜狗浏览器的极速模式），点击图片，您会看到图片们向左不停地、以流畅动画形式，显示啊显示啊显示.....



在不知CSS3为何物的浏览器下，例如IE7这厮，图片也会一个一个切换，只是木有动画效果而已——对于实际应用而言，足够了！！——我们平时的效果基本上就是这样的，FireFox等浏览器更加better而已！

实现

要说如何实现的，咳咳，说穿了真是简单地让人吐血。

1. 显然的，调用我提出并编辑过的[animate.css](#)文件，如下代码：

```
<link rel="stylesheet" href="http://www.zhangxinxu.com/study/css/animate.css" type="text/css" />
```

2. 给要动画的元素添加几个关键的类名，例如这里是slide效果，因此加一个名叫 `slide` 的类名，如下截图：

```
<a href="/" class="list slide out">  
鑫旭-鑫空间-鑫生活

好了，可以去吃晚饭了 – 🍲

## 四、animate.css回锅再炒

上面的例子作用有2个：1. 提起兴趣；2. 大致认识。

于是，现在，到了可以好好讲讲 animate.css 相关内容了：

- animate.css 驱动下的各种动画效果都是通过切换类名实现的；
- 类名分为三类：公用类名、动画关键字类名以及可有可无的两个3D视角类名。  
公共类名有3个： in , out 和 reverse . 分别指无到有、有到无、反向。  
关键字类名9个： fade , pop , slide , slidefade , slidedown , slideup , flip , turn , flow . 各个效果后面有展示；  
3D视角类名2个： viewport-flip , viewport-turn . 从名字就可以看出，是 flip 效果和 turn 效果需要的。
- 虽说IE10也会支持animation动画，但是，这里只有moz, webkit前缀驱动，因此，IE10下无效果（您自然可以添加更多CSS使IE10以及后续的Opera浏览器支持）。

```
.flip {
 -webkit-backface-visibility: hidden;
 -webkit-transform: rotateY(180deg);
 -moz-backface-visibility: hidden;
 -moz-transform: rotateY(180deg);
}
```

仅2个私有前缀驱动

从实际应用的角度讲，为了可以准确判断向下不支持的浏览器，这样的命名是比较推荐的。但是，5年之后，必然，这里的命名等需要大动。

- flip 效果和 turn 效果属于3D变幻的范畴，因此父级元素上有必要设置：

```
perspective: 1000px
```

您可以使用 animate.css 中的 viewport-flip , viewport-turn 或者使用自己定义类名。因此，我说对于 animate.css 而言， viewport-flip , viewport-turn 不是必须的。

- animate.css 中的每一行的CSS代码都是比较高级的CSS3属性，因此包括IE9在内的浏览器都是根本不认识的。这种完全不认识性，使得我们的兼容性处理就变得相当简单了。

in, out, reverse类名的理解

各种动画效果的实现的本质就是“使用JavaScript对in, out, reverse三个类名颠来倒去切换”。

一般而言，`in` 表示元素从看不见出现到出现的动画效果。例如 `fade + in` 的动画效果就是淡入（图片透明度从0到1）。而 `out` 指代元素隐藏，逝去，例如 `fade + out` 的动画效果就是淡出（图片透明度从1到0）。

类似的 `slide+in` 效果就是移入，`slide+out` 效果就是移出；`pop+in` 效果就是弹出；`pop+out` 效果就是收进去；等等！

`reverse` 的作用是反向。举个例子，最简单的slide效果：进来是 `slide+in`，即从右往左。如果移出是 `slide+out` 则还是从右往左移出，如果移出是 `slide+reverse+out`，则是从左往右移出，也就是原路返回！

因此，`reverse` 一般用在独立元素的交互效果上，例如弹框出现和弹框关闭的效果应该是完全相反的，这时候就需要用到类名 `reverse`。

在以前的jQuery版本中，in, out动画的时间都是一样的，如下代码：

```
.in, .out {
 -webkit-animation-timing-function: ease-in-out;
 -webkit-animation-duration: 350ms;
}
```

不过现在做了不同处理，默认动画进入 350 毫秒，动画移出 225 毫秒。至于为什么做这番修改，我也不得而知，总之对相关并木有什么糟糕的影响，我们仍可从容使用之。

## 五、animate.css的向下兼容

前面说过，包括IE9, Opera浏览器在内的浏览器都是不支持 `animate.css` 的动画CSS的，如果保证这些浏览器的显示也是正常的。

其实很简单，只要让这些浏览器有下面这一行CSS代码就可以了：

```
.out { display: none!important; }
```

`animate.css` 中 `out` 类名的本质就是以动画形式让元素隐藏（不可见）；其本质与直接的元素隐藏(`display:none`)是一样的。

然后，什么 `in`，`out` 之类的切换就完全不会影响在低版本浏览器上的显示了。

下面问题来了，如何让非目标浏览器上渲染 `display:none` 呢？

考虑到CSS hack太难搞，@supports目前仅Firefox17支持，我是借助JavaScript实现的，完整代码如下：

```
var BROWSER = function() {
 var ua = navigator.userAgent.toLowerCase();

 var match = /(webkit)[\/>]([\w.]+)/.exec(ua) ||
 /(opera)(?:.*version)?[\/>]([\w.]+)/.exec(ua) ||
 /(msie) ([\w.]+)/.exec(ua) ||
 !/compatible/.test(ua) && /(mozilla)(?:.*? rv:([\w.]+))?/.exec(ua) ||
 [];

 return { browser: match[1] || "", version: match[2] || "0" };
}();
```

```

if ((BROWSER.animate = (BROWSER.browser !== "mozilla" && BROWSER.browser !== "webkit"))) {
 // 不是目标浏览器，创建CSS向下兼容
 var oStyle = document.createElement("style"), cssText = ".out{display:none!important;}";
 oStyle.type = "text/css";
 if (BROWSER.browser === "msie") {
 oStyle.styleSheet.cssText = cssText;
 } else {
 oStyle.innerHTML = cssText;
 }
 document.getElementsByTagName("head")[0].appendChild(oStyle);
}

```

建议代码放在页面的头部，或者在头部放入如下的JS文件链接代码：

```
<script src="http://www.zhangxinxu.com/study/201210/animate-fix.js"></script>
```

如果您的头部已经链接了例如jQuery框架，更简单，直接（未测试）：

```

if (!$.browser.webkit && !$_.browser.mozilla) $("head").append('<style>.out{display:none!important;}</style>');

```

## 六、fade以及slidefade动画体验

### fade动画效果

**fade** 效果可以说是最好理解，最易识别的效果了。淡入淡出效果为jQuery内置动画效果，如果两者画个等号的话，类似这样：

```

$.fadeIn() = $.addClass("fade in");
$.fadeOut() = $.addClass("fade out");

```

您可以狠狠地点击这里：[fade动画下的图片轮播效果demo](#)

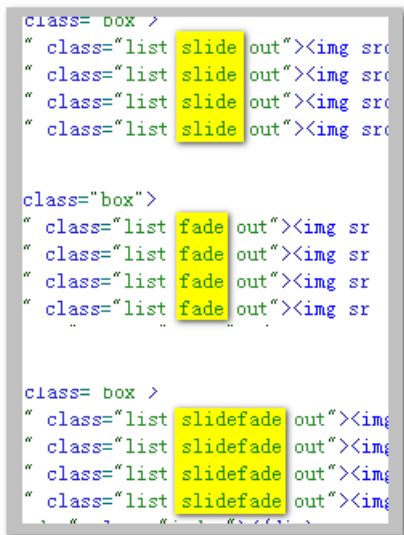
// zxx: 下面的N多demo中也夹杂着 **fade** 效果，同时 **fade** 动画是唯一没有 **reverse** 参与的动画类型。

### slidefade动画效果

**slidefade** 是 **slide** 移动和 **fade** 淡入淡出效果的结合，您可以狠狠地点击这里：[slidefade动画下的图片轮播效果demo](#)



到目前为止展示的三个demo，唯一不同的就是HTML代码中的类名：



可见，CSS3 `animate.css` 下的动画效果完全由类名驱动的。

对于多元素且有规律的动画效果，一般关键类名 `reverse` 是不参与进来的。

但是，对于单独的元素动画，`reverse` 就不可或缺了。

## 七、slideup动画效果展示

这里展示 `slideup` 动画，同时更重要的是介绍以下几个知识点：

1. CSS控制下的动画元素隐藏
2. 定时器控制下的动画元素隐藏
3. `reverse` 使用的一般规律

您可以狠狠地点击这里：[含提示的图片列表删除demo](#)

本demo含有两个slideup效果，一个是鼠标经过图片时候出现的含有“删除”文字的黑色半透明提示条，如下截图：



另外一个就是点击“删除”出现的“是否删除”的提示框，如下截图：

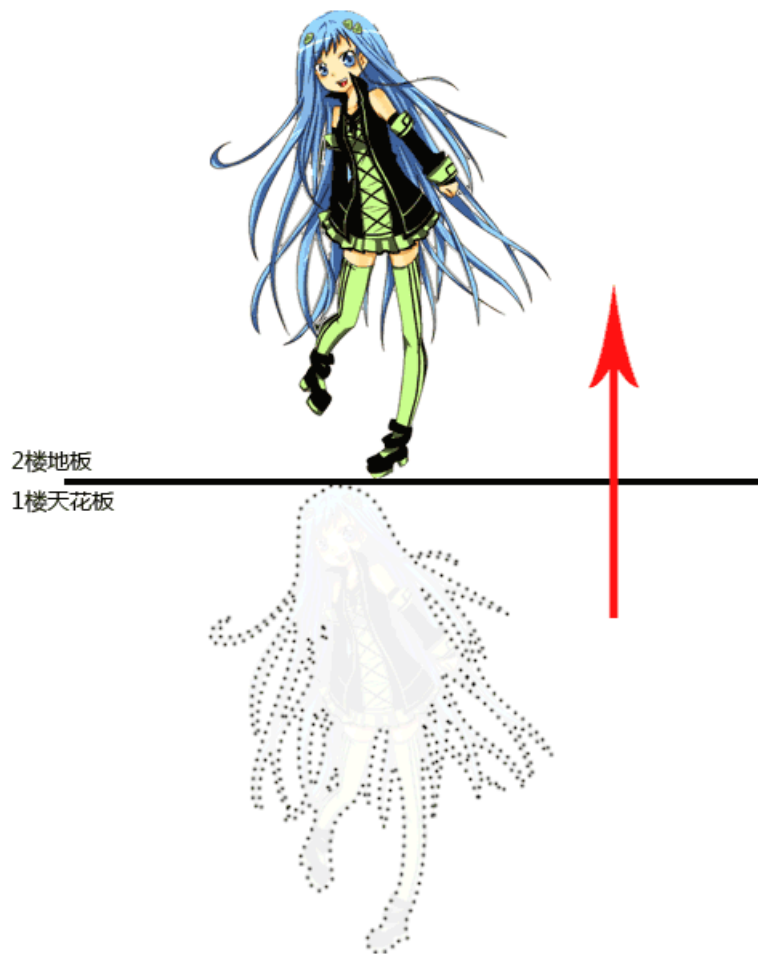


其中，前面slideup元素的隐藏是通过CSS限制实现的；后者是JavaScript定时器实现的。

在介绍两种元素隐藏方法之前有必要先要脱下 `slideup` 动画的衣服，好好窥视其真实的肌体。

`slideup+in` 效果是向上移动到当前位置，距离为自身高度。举个例子，一个身高170cm的妹子站在在

二楼吹头发；则该妹子应用 `slideup+in` 动画的效果就是：妹子瞬间脑袋顶着1楼天花板，倏地向上移动到正好站在2楼的位置。如果我们在2楼的话，看到的就是妹子的脑袋开始从楼板上冒出来——一直到整个身体出现 🤖 ！



### 1. CSS限制下的slideup效果

这类效果，需要容器（类似上面的楼层）限制(`overflow:hidden`), HTML结构如下：

```
外部限制容器(overflow:hidden)
 slideup效果元素(slideup + in/out)
```

于是，slideup动画执行时候我们就会看到元素慢慢“冒出来”的效果了。

如果没有外部容器的限制，slideup效果就是完整元素（妹子不会被楼层截掉）的上移或下降。这显然不是我们需要的，当 `out` 触发的时候，我们希望元素不可见（下降只是位置改变，元素依然可见）。这种情况，就需要借助JS脚本。

### 2. JavaScript定时器的限制

动画的执行的时候是固定的（CSS限制的），因此，我们可以使用JS让动画效果结束的时候，让元素不可见，如 `display:none` .demo中相关代码如下：

```
// 点击取消按钮
$("#取消按钮").bind("click", function() {
 // 提示框下移动画触发
 $("#提示框").addClass("reverse out").removeClass("in");
 setTimeout(function() {
 // 200毫秒后提示框隐藏
 $("#提示框").hide();
 }, 200);
});
```



```
});
```

我们无需担心IE6~9之类浏览器的兼容性问题，因为，当元素添加类名 `out` 的时候，元素就已经隐藏了，所以延时什么的无需担心影响交互效果。

### 3. reverse的一般使用规律

如果您希望元素的out动画与in动画是“原路返回”的关系，则需要用到类名 `reverse`。例如demo页面提示框的显示与隐藏完全相反效果，则需要用到 `reverse`。

其使用是一个路子（其他各种animate效果也是如此），我是这样操作的：  
元素进入动画：

```
$("#元素").addClass("in").removeClass("reverse out");
```

元素移出动画：

```
$("#元素").addClass("reverse out").removeClass("in");
```

`addClass`、`removeClass` 顺序不分先后。

于是，完整流程的效果即可实现。

初始化的时候，我都是把 `out`，`reverse` 预先放在元素上了，例如这里的：`slideup reverse out`。

//zxx: 下面为广告~~注意不要勿点~~嘻嘻~~



## 八、pop效果和flow效果综合实例

pop效果是元素从正面弹出弹入；flow效果是元素先变小然后再向两侧偏移。

您可以狠狠的点击这里：[pop/flow效果下的图片移入回收站demo](#)



弹框提示为 `pop` 效果，图片移入移出回收站为 `flow` 效果。均使用到了 `reverse` 使动画效果镜像，`flow` 效果使用了 `setTimeout` 定时器控制元素的隐藏。都是上面 `slideup` 提到的东西，不再赘述。

不过，回收站的摇动效果可能大家会比较感兴趣，该效果并不出自jQuery Mobile中的animate动画效果之列，而是来自文章一开始提到的那个“蛋一灯”的animate.css中的 `tada` 效果。

相关CSS代码demo页面有展示，该动画触发模式与jQuery Mobile更重用的 `in/out` 模式不同，其直接添



加动画关键字类名就可以了，例如这里，直接：

```
$().addClass("tada");
```

就可以了。

## 九、3D效果之flip翻转

flip效果为中轴翻转，具有代表性的效果就是翻纸牌。

您可以狠狠地点击这里：[flip动画与翻转纸牌动画效果](#)



因为是3D效果，如果希望呈现一定的3D视角，需要在父级元素上添加类名 `viewport-flip` 或者直接添加如下CSS：

```
-webkit-perspective: 1000px;
-moz-perspective: 1000px;
```

`perspective` 属性具体含义可参见我之前的“[CSS3 3D transform详解](#)”一文。

### 原理简述

1. 当前在前显示的元素翻转90度隐藏, 动画时间225毫秒
2. 225毫秒结束后，之前显示在后面的元素逆向90度翻转显示在前
3. 完成翻面效果

也就是纸牌的前后面在两个不同的时间点进行flip效果，构成完整的纸牌翻面效果。

注：Chrome浏览器下需要让元素屏幕垂直居中，以保证元素均在视角内，避免部分区域不显示的情况发生。

## 十、3D效果之trun翻转

trun效果为沿着侧边翻转，类似翻书，开关门效果。

您可以狠狠地点击这里：[trun动画与门的开关模拟效果demo](#)



与上面flip效果类似，父标签需要添加视角样式，或类名 `viewport-turn` 或自己写两行 `perspective` 相关CSS.

如果我们把page页面整个应用turn效果，web页面的浏览就像翻书那样，很酷的！

## 十一、其他相关的总结

### 绝对定位元素

所有这些animation动画效果，元素本身所占据的空间至始至终都是不变的。因此，类似幻灯片之类多元素切换的效果，势必需要将元素设为绝对定位元素，以占据同一垂直空间。

再考虑到动画会造成强烈的重绘与渲染，从性能角度讲，我们必须将动画元素脱离文档流，也就是设置成绝对定位元素（避免强烈的回流）。

因此，这里，我认为：如果您想让一个元素应用animation驱动动画效果，请将其设置为绝对定位元素。

### 与transition动画对比

`transition` 也是有动画效果的，其特定是简单灵活，代码精简。不足之处在于：

1. 不能通过CSS控制动画的起点；
2. 不能设置动画的断点；
3. 动画的驱动与值类型甚至单位有关；
4. 动画只能是一次性的；
5. 动画不能延迟；

等。

各有裨益，这里不展开。

《CSS世界》签名版独家发售，包邮，可指定寄语，点击显示购买码

（本篇完） // 想要打赏？点击[这里](#)。有话要说？点击[这里](#)。



## 猜你喜欢

- HTML5+JS手机web开发之jQuery Mobile初涉
- 手机网页UI框架jQuery Mobile介绍之按钮篇
- cssSandpaper-兼容IE的CSS3 JavaScript库
- 好吧, CSS3 3D transform变换, 不过如此!
- JavaScript实现图片幻灯片滚动播放动画效果
- jQuery - 鼠标经过(hover)事件的延时处理
- 团购类网站倒计时的js实现
- 基于canvas画布的两个炫酷效果展示
- CSS CSS3实现3D开门动画效果
- 小tip: 微博新版查看大图前后浏览的另外一种实现
- 不同CSS布局实现与文字鼠标选择的可用性

分享到:  0

标签: 3D效果, animate, animation, jQuery Mobile, setTimeout, 回流, 绝对定位

## 发表评论 (目前21条评论)

名称 (必须)

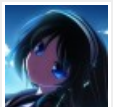
邮件地址(不会被公开) (必须)

网站

提交评论

### 1. mir说道:

2018年07月21日 10:55



flip翻牌效果没有必要用到setTimeout这么繁琐, 增加了脚本复杂度。既然用了animation, 可以使用它的animation-delay和animation-fill-mode来实现推迟播放, 这样脚本只要一句话就能实现切换。在线例子: <https://codepen.io/mirari/pen/oMBwxQ>

[回复](#)

### 2. 欣欣一单说道:

2017年06月28日 14:32



上述的效果都是支持ie8浏览器的么? 可是animate不是只支持ie10+么?

[回复](#)

### 3. Asiaidc.net说道:

2015年10月15日 18:39



完整而又详尽的难得好教程,学习了!

[回复](#)

4. nan说道:

2015年07月23日 09:47



之前在邮件中间的flip下, 翻转不彻底的问题, 找到了原因。一般而言, 扑克牌翻转后, 我们理解为隐藏, 但是自己照例子做的效果却是有变形却不隐藏。后来, 找到了原因, body中有text-align:center(本文中并没加, 是我自己做的时候加的), 扑克牌容器box中没有定义text-align, list属性又是position:absolute, 在box容器中加入了text-align:left; 效果就与预期一致了。

css代码如下, 其他代码, 与flip下贴出来的一致。

原:

```
body{text-align:center;}//大概怕box的margin左右auto不居中, 顺手写上的
box {width: 355px;
height: 500px;
padding-top: 30px;
padding-bottom: 30px;
margin-left: auto;
margin-right: auto;
position: relative;}
```

修改之后:

```
body{text-align:center;}
box {width: 355px;
height: 500px;
padding-top: 30px;
padding-bottom: 30px;
margin-left: auto;
margin-right: auto;
position: relative;
text-align:left; // 修改后, 增加了text-align:left; flip效果与预期一致, 也与本文一致
}
```

问题来了, 这个text-align对齐方式的属性, 为何有这么大的影响呢?

[回复](#)

5. 游戏币萧何说道:

2015年04月25日 11:48



九、3D效果之flip翻转——这个例子当快速点击扑克牌多次之后会出现某一面不翻转的问题

[回复](#)

臭豆腐的说道:

2015年07月3日 18:04

我也看到了这个问题。

[回复](#)



6. 猫猫说道:

2013年02月26日 11:29



焦点图的例子比较常见, 我抽取了jquery mobile中的滑动事件给焦点图例子增加了带滑动事件可定时也可以左滑右滑触发。

SORRY, 刚才评论例子URL写错了, 这个是正确的。麻烦管理员删除之前的评论。

<http://runjs.cn/detail/awbl45bs>

[回复](#)

7. awen说道:

2012年11月21日 13:57

不错



[回复](#)

8. 星仔说道:

2012年11月10日 21:32

这个可以有啊！正琢磨jQuery Mobile呢。

[回复](#)



9. cnPolan说道:

2012年11月9日 23:11

这里的JQ是什么版本的。好像是豆瓣的，怎么解，里面的JQ跟原来的不同，是插件吗？

[回复](#)



10. filod说道:

2012年11月8日 17:07

“蛋一灯”这样的译法对项目作者非常地不尊重

[回复](#)



11. 蓝面小生说道:

2012年11月7日 10:33

隐藏如果使用setTimeout的话会导致延时的时间依赖于动画的时间，所以JS提供了一个事件来支持动画完成，我们可以监听这个事件完成来让元素隐藏

```
$(el).one('webkitAnimationEnd animationend',function(){
$(this).hide();
});
```

[回复](#)



张鑫旭说道:

2012年11月9日 15:46

@蓝面小生 这个赞，学习了！！

[回复](#)



12. 浅笑说道:

2012年11月5日 14:06

好文章 必须顶，学习了。

[回复](#)



13. www.greebell.com说道:

2012年11月3日 13:31

看不太懂，很多年没弄这玩意儿了

[回复](#)



14. steve说道:

2012年11月3日 12:27

经常来博主这里学习，许多文章都视为经典，但是现在文章越来越多，可不可建一个只有文章目录的索引页面，这样找文章的经验就会好很多。

[回复](#)



15.

yetazhan说道：

2012年11月3日 00:36

什么时候能前缀都不用带，那就好了

回复


16.

\_t说道：

2012年11月2日 20:28

你可以把 -moz- 删了。当前的火狐支持不带有前缀的版本而且 IE 也支持。。。

回复


17.

hgh 569457128说道：

2012年11月2日 17:33

不知道帮主对淘宝触屏版的图片滚动有研究没 期待此专题博文 主要是手动触屏

回复


18.

tc dona说道：

2012年11月2日 13:52

顶！完整的方案呀~~

回复

tomxzzhou说道：

2017年08月23日 15:41

CSS3 animate flip下的纸牌翻转效果实例页面 这个例子不能在IOS上有兼容问题

回复




- ## 最新文章
- » [常见的CSS图形绘制合集](#)
  - » [粉丝群第1期CSS小测点评与答疑](#)
  - » [分享三个纯CSS实现26个英文字母的案例](#)
  - » [小tips: 纯CSS实现打字动画效果](#)
  - » [CSS/CSS3 box-decoration-break属性简介](#)
  - » [CSS :placeholder-shown伪类实现Material Design占位符交互效果](#)
  - » [从天猫某活动视频不必要的3次请求说起](#)
  - » [CSS vector-effect与SVG stroke描边缩放](#)
  - » [CSS ::backdrop伪元素是干嘛用的？](#)
  - » [周知：CSS -webkit-伪元素选择器不再导致整行无效](#)
- ## 今日热门
- » [常见的CSS图形绘制合集](#) <sup>(193)</sup>
  - » [未来必热：SVG Sprite技术介绍](#) <sup>(120)</sup>
  - » [粉丝群第1期CSS小测点评与答疑](#) <sup>(115)</sup>
  - » [HTML5终极备忘大全（图片版+文字版）](#) <sup>(93)</sup>
  - » [让所有浏览器支持HTML5 video视频标签](#) <sup>(86)</sup>
  - » [Selectivizr-让IE6~8支持CSS3伪类和属性选择器](#) <sup>(82)</sup>
  - » [CSS3下的147个颜色名称及对应颜色值](#) <sup>(80)</sup>
  - » [视区相关单位vw, vh..简介以及可实际应用场景](#) <sup>(76)</sup>
  - » [写给自己看的display: flex布局教程](#) <sup>(76)</sup>
  - » [小tips: 纯CSS实现打字动画效果](#) <sup>(76)</sup>
- ## 今年热议
- » [《CSS世界》女主角诚寻靠谱一起奋斗之人](#) <sup>(76)</sup>
  - » [不借助Echarts等图形框架原生JS快速实现折线图效果](#) <sup>(64)</sup>

- » [看，for..in和for..of在那里吵架！](#) <sup>(60)</sup>
- » [是时候好好安利下LuLu UI框架了！](#) <sup>(47)</sup>
- » [原来浏览器原生支持JS Base64编码解码](#) <sup>(35)</sup>
- » [妙法攻略：渐变虚框及边框滚动动画的纯CSS实现](#) <sup>(33)</sup>
- » [炫酷H5中序列图片视频化播放的高性能实现](#) <sup>(31)</sup>
- » [CSS scroll-behavior和JS scrollIntoView让页面滚动平滑](#) <sup>(30)</sup>
- » [windows系统下批量删除OS X系统.DS\\_Store文件](#) <sup>(26)</sup>
- » [写给自己看的display: flex布局教程](#) <sup>(26)</sup>

## 猜你喜欢

- [HTML5+JS手机web开发之jQuery Mobile初涉](#)
- [手机网页UI框架jQuery Mobile介绍之按钮篇](#)
- [cssSandpaper-兼容IE的CSS3 JavaScript库](#)
- [好吧，CSS3 3D transform变换，不过如此！](#)
- [JavaScript实现图片幻灯片滚动播放动画效果](#)
- [jQuery - 鼠标经过\(hover\)事件的延时处理](#)
- [团购类网站倒计时的js实现](#)
- [基于canvas画布的两个炫酷效果展示](#)
- [CSS CSS3实现3D开门动画效果](#)
- [小tip: 微博新版查看大图前后浏览的另外一种实现](#)
- [不同CSS布局实现与文字鼠标选择的可用性](#)