**开发博客日记：**

\*有用该记住的css用法：

1. -webkit-box-shadow: 3px 3px 18px #888484; //阴影

2. -moz-border-radius: 100px; //圆角或圆形实现

3. -o-linear-gradient(left, #CDD0EF, #EBEAF5, #CDD0EF); //渐变颜色

4. text-decoration: none; //不要下划线

5. letter-spacing: 9px; //字间距设置

6. transition:background-color 1s ease-in-out; //动画平缓变化

7. font: lighter 2em/85px "Helvetica Neue", "Arial", "Helvetica", "sans-serif";//可调节字体距顶部的高度

8. .button2 a:first-child .button2 a:nth-of-type(2) //子元素选择

9. .button2:hover //做动画

10. animation:mymove 5s infinite; //infinite是不断重复执行

@keyframes mymove

{

from {left:0px;}

to {left:200px;}

// 0%{-moz-transform:rotate(0deg)} //transform还可以做scale放大

// 100%{-moz-transform:rotate(360deg)}

}

10. x=$("p").offset(); $("#span1").text(x.left/top); //距离顶部高度

11. glyphicon图标，属于font，可以调节字体大小或颜色

12. transform-origin: 1px 1px; //变化原点，不大懂

animation: hrhand 8s linear infinite (alternate); //动画，infinite无限重复

@-webkit-keyframes hrhand{

0%{-webkit-transform:rotate(0deg)}

100%{-webkit-transform:rotate(360deg)}

} //有时不到100%，有特别的收回效果，不懂

//alternate：交替起点终点效果

**13. cursor属性**，定义指针经过时光标形状。

14.在调节top\_bar三个按钮位置时，曾尝试多种方式，都无法做到具有响应式的居中。后来在三个按钮的span标签外，外套一层div，设置div的margin：0 auto ，同时设定div宽度固定为3\*300+200px （按钮宽度100px，间距100px）。然后再利用span的，position:relative 和left属性调节相对边距，从而达到响应式居中。

**15. white-space属性：**

有nowrap, pre, pre-wrap, pre-line这几个可选属性值。其中在文章正文内容中，空格符需要保留，换行符需要保留，自动运行允许，这时我们就需要使用pre-wrap属性值。

\*有用该记住的jquery用法：

1. var selector = $(this).children("a");

2. tID = setTimeout( function() {xxx} ，100)；clearTimeout(tId); //mouseenter时延迟100ms执行；mouseleave时清除延时，且调用stop方法，清空动画队列

3. $(this).children("a").stop().animate({"top":"-60px"} , "easeInOutCubic"); //动画函数animate

4. **$('html, body').animate({"scrollTop" : “200px”}, "linear");**

5. 手动制作向右淡入与淡出的效果：

$('#top\_panel .downward').fadeOut(100 , function(){

$('#top\_panel .panel\_content')

.css({"display":"none" , "margin-left":"-10%"})

//fadein顺序相反会出bug！！！！！！！！！！！！！！！！！！

.fadeIn(700)

.queue("fade\_in", function(next){

$(this).animate({"margin-left" : "0%"} , {"duration": 400 , "queue": false});

next();

})

.dequeue("fade\_in");

6.选择第n个子元素：

$(“#div1”).eq(n).addClass(“blue”); //注意，eq(n)是从第0个开始的

$(“div:eq(1)”).addClass(“blue”); //若eq(index)的index为负数，则从后开始

$(“body”).find(“div”).eq(n).addClass(“blue”); // find找出所有后代元素

$('#div1').children('span').length; //children找出所有子元素，注意区别

//length返回选择器选择的个数

/\*先把它定位到屏幕左侧，然后fadeIn 的效果同时，额外加入一个自定义队列，来做向右滑动的animate效果。注意，由于两个效果需要并行同时执行，所以在自定义队列的animate中，**"queue": false** 让其不加入默认的fx动画队列。\*/

7.用jquery动态生成的html，不能直接用click事件绑定，需要用on事件绑定

$('#top\_panel .panel\_content')

.queue("go\_right", function(next){

$(this).animate({"margin-left" : "50%"} , {"duration": 800 , "queue": false});

next();

})

.dequeue("go\_right")

.fadeOut(600 , function(){

$('#top\_panel .downward').fadeIn(150);

});

/\*向右淡出效果，思路同上。注意淡出时先animate后fadeOut，淡入时先fadeIn后animate。反过来可能会出bug！！\*/

8.若想刷新后返回页面顶部，可用

window.onbeforeunload = function(){

$(window).scrollTop('0');

}

1. 在同一行中想排多个元素时，有几种方法：
2. 用float:left ，多个排列
3. 用span标签，设置padding和margin值
4. 用Bootstrap的布局系统
5. 实现顶部图片（固定效果）：
6. 在top\_pic的子元素中，加入position：fixed。如果想要居中，同时设置left:0和right:0
7. Z-index要设置较小值
8. 字体大小: font: lighter 92px "KaiTi" , "FangSong" , "黑体" , "Microsoft YaHei" , "宋体" ,sans-serif;
9. 用background\_image无法实现图片百分百平铺+响应式，选择在div中加入img标签
10. 实现导航栏中三个动态按钮：
11. 设置一个固定长高的div，并设置边角圆形
12. 加入两个a标签，设置height与父div高度相同
13. 加入display：block , 会加入换行符
14. Position:relative ,然后在CSS的hover事件，或者jQuery的mouseenter/mousehover + mouserleave事件中，用animate改变top值()
15. 实现导航栏滚动到顶部时，固定
16. getheight函数求出top\_pic的height值global\_pic\_height，然后jquery赋值给top\_bar的height（用到substr去除字符串的“px”）
17. 赋值给图片上文字top\_pic h1/h2的height属性
18. $(document).ready( get\_height() )；加载页面时触发。
19. $(window).resize(function() { get\_height(); })；改变窗口大小或缩放时触发，动态调节top\_bar的高度。
20. 用scroll事件，求scrollTop值，与top\_pic的height比较
21. 改变top\_bar 的position为fixed。由于top\_bar变为fixed后不占据高度，为了让下边div元素的位置不移动，在top\_bar外加上一层父div（top\_bar\_father），并让父div的height固定（这里是60px），从而不影响下方元素
22. **实现窗口右侧目录**
23. jquery选取$(‘titles’).length，获取正文文章数，并用eq(n)与text()方法取得题目文本，存在全局数组。
24. 取得每个题目的offset().top()，存在全局数组，用于后边的锚点定位。
25. 判定每个标题长度。首先**用charCodeAt(j)区分中文与英文字符**，若值为[0 , 128]，为英文。中文字符计算2长度，英文计1。若总长度超过一定值，截去后部分字符串，并加上“…”
26. append方法动态插入目录，每一个加上id，并在最后加上“返回顶部”
27. $(document).on(“click” , “#catalogue h3,h4” , function(){ })，绑定目录的“click”（因为目录为动态生成，不能直接用click事件）。根据current\_id = $(this).attr(“id”)，取得对应的标题高度title\_height[current\_id]。
28. $('html , body').animate({"scrollTop" : title\_height[current\_id] + 'px'} , {"duration":600 ,"time-function":"linear" , "queue": false}) 滑动到达指定高度。
29. 与top\_bar绑定相同的scroll事件，切换position的fixed与relative属性值。在resize事件中，为了让目录在fixed状态下，也能相对窗口大小调整坐标，**考虑在他们外加上一层父div，并利用父div计算相对距离。**

var new\_top = parseInt( $('#catalogue').parent().offset().top)

- parseInt( $('#top\_bar\_father').offset().top );

var new\_left = parseInt( $('#catalogue').parent().offset().left) ;

$('#catalogue').css({"position":"fixed","top":new\_top+'px', "left":new\_left+'px'});

1. 使用：hover：after事件，在content中添加“<”用做指示
2. Scroll事件中，用当前的top\_dis = $(this).scrollTop()，与各个title\_height做对比，实现滑动到对应目录条内容时,在前面添加星号标记。提前在标题上-150px的高度，就添加星号。当top\_dis在两个title\_height之间时，$('.active\_catalogue').remove();

$('#catalogue' + (i-1) ) .prepend('<span class="active\_catalogue glyphicon glyphicon-star"></span>');

record[i] = 1; record[i-1] = 0; record[i+1] = 0;