判断元素是否隐藏

1. $("xxx:visible").length , $("xxx:hidden").length
2. $("xxx ").is(":visible") $("xxx ").is(":hidden")
3. $('#id').css('display')=="none" , "display"

判断是否获得焦点

1. **var** isFocus=$("#tRow").is(":focus");
2. **if**(**true**==isFocus){
3. alert("focus");
4. }**else**{
5. alert("blur");
6. }

//光标放在最后 $("#文本框ID").textFocus();光标放在第二个字符后面 $("#文本框ID").textFocus(2);  
(**function**($){  
 $.fn.textFocus=**function**(v){  
 **var** range,len,v=v===undefined?0:parseInt(v);  
 **this**.each(**function**(){  
 **if**($.browser.msie){  
 range=**this**.createTextRange();  
 v===0?range.collapse(**false**):range.move("character",v);  
 range.select();  
 }**else**{  
 len=**this**.value.length;  
 v===0?**this**.setSelectionRange(len,len):**this**.setSelectionRange(v,v);  
 }  
 **this**.focus();  
 });  
 **return this**;  
 }  
})(jQuery);

浏览器

$(function() {   
if($.browser.msie) {   
alert("this is msie");   
} else if($.browser.safari) {   
alert("this is safari!");   
} else if($.browser.mozilla) {   
alert("this is mozilla!");   
} else if($.browser.opera) {   
alert("this is opera");   
} else {   
alert("i don't konw!");   
}

浏览器版本

if($.browser.msie&&($.browser.version == "6.0")&&!$.support.style){   
alert("ie6");   
}

if($.browser.msie&&($.browser.version == "7.0")){   
alert("ie7");   
}

<a data-xxx="yyy"></a>

$().data("xxx") -> "yyy"

$("[data-xxx]") -> all jQuery objects with html attribute data-xxx

parent:just parent,1 layer outside

parents:including ancestors x layer outside

必须在$()插入到文档中后才能使用 parents() find() 等遍历方法，不然得不到对象

跨域问题：server

String referer = request.getHeader("Referer");  
URL u = new URL(referer);   
String host = u.getHost().toLowerCase();

// 如果域名包含.zhihuishu.com，则允许其跨域  
 if(StringUtils.*contains*(host, ".zhihuishu.com")) {  
// response.setHeader("Access-Control-Allow-Origin", referer);  
 response.setHeader("Access-Control-Allow-Origin", "\*");  
 response.setHeader("Access-Control-Allow-Methods", "POST, GET, OPTIONS, DELETE");  
 response.setHeader("Access-Control-Max-Age", "3600");  
 response.setHeader("Access-Control-Allow-Headers", "x-requested-with");  
 }

jquery插件编写

**if** (typeof params === 'string' && typeof plugin[params] === 'function') {  
 retval = plugin[params].apply(plugin, [].slice.call(args, 1));  
}**else** {  
 retval = plugin;  
}

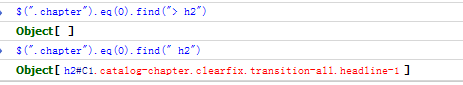
jquery 选择器

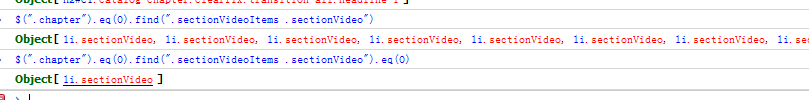
空格：$('parent childchild')表示获取parent下的所有的childchild节点（所有的子孙）。

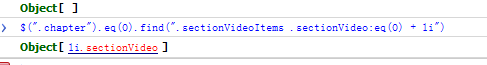
大于号：$('parent > child')表示获取parent下的所有child的儿子( 第一代)。

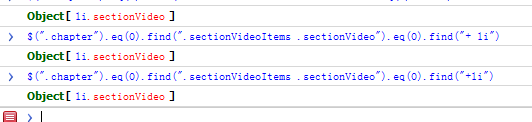
加号：$('pre + nextbrother')表示获得pre节点的下一个兄弟节点，相当于next()方法

波浪号：$('pre ~ brother')表示获取pre节点的后面的所有兄弟节点，相当于nextAll()方法。









都是找下一个li



找剩下的所有li

滚轮事件

$obj.delegate('#sideToolbar','mousewheel DOMMouseScroll', **function** (e) {  
 e.preventDefault();  
 e.stopPropagation();  
  
// $obj.bind('mousewheel', function (e) {  
// console.log('asdasdsad');  
 **var** delta = (e.originalEvent.wheelDelta && (e.originalEvent.wheelDelta > 0 ? 1 : -1)) || // chrome & ie  
 (e.originalEvent.detail && (e.originalEvent.detail > 0 ? -1 : 1)); // firefox

**if** (delta > 0) {  
 // 向上滚  
 $('#sideCatalog-up').trigger("doClick");  
} **else if** (delta < 0) {  
 // 向下滚  
 $('#sideCatalog-down').trigger("doClick");  
}  
step =oldStep;  
**return false**;

}

平滑滚动效果

**var** $body = (window.opera) ? (document.compatMode == "CSS1Compat" ? $('html') : $('body')) : $('html,body');

$body.animate({scrollTop: $target.offset().top}, 500,"swing");

鼠标持续点击效果

**var** downInterval;  
**var** upInterval;  
**var** intervalTime=80;  
$('#sideCatalog-down').bind('mousedown', **function** (e) {  
 downInterval=setInterval(**function** () {  
 $('#sideCatalog-down').trigger("doClick");  
 }, intervalTime);  
});  
$('#sideCatalog-up').bind('mousedown', **function** (e) {  
 upInterval=setInterval(**function** () {  
 $('#sideCatalog-up').trigger("doClick");  
 }, intervalTime);  
});  
  
$('body').bind('mouseup', **function** (e) {  
 // console.log('mouseup');  
 clearInterval(downInterval);  
 clearInterval(upInterval);  
});