1. JQuery
2. 概述

JQuery是一个快速，简洁的JavaScript框架，它封装了JavaScript常用的功能代码，提供一种简便的JavaScript设计模式，优化HTML文档操作，事件处理，动画设计和Ajax交互。

1. JQuery的使用
2. 下载JQuery

JQuery一般有两个文件，一个是jquery-xxx.js，一个是jquery-xxx.min.js，两个js文件有什么区别呢？

jquery-xxx.js是开发版本，有良好的缩进和注释，利于阅读，文件体积大一些。

jquery-xxx.min.js是生产版本，程序中使用，没有缩进，体积小一些，程序能够更快地加载。

2. 导入JQuery的js文件

导入min.js文件，即导入一个js文件：

<script src=”js/jquery-3.3.1.min.js”></script>

1. JQuery对象和JS对象的区别和转换
2. JQuery对象和JS对象的方法是不通用的。
3. JQuery对象要转换为JS对象，则用以下方式：

JQuery对象[索引]或者JQuery对象.get(索引)

1. JS对象要转换为JQuery对象，则使用：

$(JS对象)

1. 事件绑定与样式控制
2. 事件绑定

将click函数绑定到id为b1的元素上：

$("#b1").click(function(){

alert("abc");

});

1. 入口函数

$(function () {

});

window.onload和$(function)的区别：

window.onload

只能定义一次,如果定义多次，后边的会将前边的覆盖掉

$(function)可以定义多次的。

1. 样式控制

通过css方法进行样式控制：  
$(“#div”).css(“color”, “red”);

1. 选择器
2. 基本选择器
3. 标签选择器

获取所有匹配标签名称的元素。

$(“html标签名”)

1. id选择器

获取与指定id属性值匹配的元素

$(“#id的属性值”)

1. class选择器

获取与指定的class属性值匹配的元素

$(“.class的属性值”)

1. 并集选择器

获取多个选择器选中的元素：

$(“选择器1，， 选择器2…”)

1. 层级选择器
2. 后代选择器

A和B之间用空格隔开，选择A元素的所有后代B元素，后代指的是子辈，孙辈…

$(“A B”)

1. 子选择器

选择A元素内部的所有B子元素，仅仅是儿子辈的。

$(“A> B”)

1. 属性选择器
2. 属性名称选择器

选择包含指定属性的元素，如查找所有包含了属性B的A元素。

$(“A[属性名B]”)

1. 属性选择器

选择包含指定属性等于指定值的元素，如查找所有属性名B=值C的元素A：

$(“A[属性名B=”值C”]”)

1. 复合属性选择器

包含多个属性条件的选择器：

$(“A[属性名=”值”][]…”)

1. 正则方式选择

选择以指定字符串开始的属性值，如选择title属性以”te”开始的div元素。

$(“div[title^=’es’]”)

选择属性值以指定字符串结尾的元素，如选择title属性以字符串”est”结尾的div元素：

$(“div[title$=”est”]”)

选择属性值包含指定字符串的元素，如选择title属性值包含”es”的div元素：

$(“div[title\*=”es”]”)

1. 过滤选择器
2. 首元素选择器

获得选择元素中的第一个元素：  
:first

1. 尾元素选择器

获得选择的元素中的最后一个元素：

:last

1. 偶数选择器

选择偶数，从0开始计数

:even

1. 奇数选择器

:odd

1. 等于索引选择器

:eq(index)

1. 大于索引选择器

:gt(index)

1. 小于索引选择器

:lt(index)

1. 标题选择器

:header

1. 表单过滤选择器
2. 可用元素选择器

获取可用的元素

:enabled

1. 不可用元素选择器

获取不可用元素

:disabled

1. 选中选择器

获得单选/复选框中选中的元素：

:checked

1. 选中选择器

获得下拉框中选中的元素：  
:selected

1. DOM操作
2. 内容操作
3. html()

获取或设置元素的标签体的内容，html()获取的是完整的标签体内容，而不是文本内容，如：

<font>内容</font>

1. text()

获取或设置元素的标签体纯文本内容

1. val()

获取或设置元素的value属性值

1. 属性操作
2. attr()

获取或设置元素的属性，多用于元素自定义的属性

获取属性值：

$(“#b”).attr(“name”)

设置属性值：

$(“#b”).attr(“name”, “exx”)

1. removeAttr()

删除属性

$(“#b”).removeAttr(“name”)

1. prop()

获取或者设置元素的属性，多用于元素的固有属性。

$(“#b”).prop(“checked”)

1. removeProp()

删除属性

1. 操作class属性
2. addClass()

添加class属性值

1. removeClass()

删除class属性值

1. toggleClass()

切换class属性，若存在该属性值，则删除，弱不存在，则添加。如：

toggleClass(“one”)，如果该元素对象上存在class=’one”，则删除one属性值，如果不存在class=”one”，则添加该属性值。

1. CRUD操作
2. append()

对象1.append(对象2)，将对象2添加到对象1元素的内部末尾、

1. prepend()

对象1.prepend(对象2)，将对象2添加到对象1的内部开头。

1. appendTo()

对象1.appendTo()，将对象1添加到对象2的内部末尾。

1. prependTo()

对象1.perpendTo()，将对象1添加到对象2内部的开头

1. after()

对象1.after（对象2），将对象2添加到对象1后面，两者为兄弟关系。

1. before()

对象1.before（对象2），将对象2添加到对象1前面，两者为兄弟关系。

1. insertAfter()

对象1.insertAfter(对象2)，将对象2添加到对象1后面，两者兄弟关系。

1. insertBefore()

对象1.insertBefore(对象2)，将对象2添加到对象1的前面，两者兄弟关系。

1. remove()

删除对象

1. empty()

清空对象的所有后代元素，但是保留当前对象以及其属性节点。

1. JQuery动画
2. 显示和隐藏元素
3. show([speed], [easing], [fn])

展示该元素。

参数介绍：

speed:动画的速度，预定义值为slow, normal ,fast，或者传入动画时长的毫秒数值。

easing:指定切换效果，默认是swing，可选linear。swing的效果是先慢，后快，再慢；linear的效果是匀速的。

fn:在动画完成时执行的函数，每个元素执行一次

$(“#a”).show(“fast”, “linear”);

1. hide([speed], [easing], [fn])

隐藏该元素。

$(“#a”).hide(“fast”, “linear”);

1. toggle([speed], [easing], [fn])

切换显示和隐藏元素。

1. 滑动显示和隐藏
2. slideDown(speed, easing,fn)

滑动下拉显示元素。

1. slidUp(speed, easing, fn)

滑动上升隐藏元素。

1. slideToggle(speed, easing, fn)

切换滑动下拉与上升。

1. 淡入淡出显示和隐藏
2. fadeIn(speed, easing, fn)

淡入效果

1. fadeout(speed, easing ,fn)

淡出效果

1. fadeToggle(speed, easing ,fn)

切换效果

1. JQuery的遍历
2. JS的遍历

JQuery中可以使用JS的遍历方式，，即通过for循环的方式。

1. each(callback)方法

JQuery对象的each方法需要传入一个callback函数。

1. 直接用this获取对象

this代表着遍历循环中的一个对象，该对象为一个JS对象，可以调用JS方法或者转换为JQuery对象调用JQuery方法。

citys.each(function(){

alert(this.innerHTML);

}

)

1. callback函数中传入参数

回调函数中可以传入参数index, element，代表着遍历循环中的索引值和该索引对应的对象。

citys.each(function(index, element){

alert(index + “:” + element.innerHTML);

}

)

1. 循环的结束

如果function函数返回的是false，则结束循环（break）；

如果function函数返回的是true，则结束本次循环，继续下次循环（continue）

（三）$.each(obj, callback)

obj为要遍历的数组，callback与之前一样是回调函数。

$.each(citys, function(){

alert($(this).html());

});

（四）xx of xx

与增强for循环的用法一样。

for(li of citys){

alert($(li).html());

}

1. 事件的绑定
2. JQuery标准的绑定方式

JQuery对象.事件方法（回调函数）;

$(“#name”).click(function(){

alert(“click”);

})

1. on绑定事件/off解绑

JQuery对象.on(“事件名称”, 回调函数);

JQuery对象.off(“事件名称”, 回调函数);

如果off方法不传递任何参数，则将组建上的所有事件全部解绑

$(“#tag”).on(“click”, function(){

alert(“click”);

})

1. toggle事件切换

JQuery对象.toggle(fn1, fn2…)

当党纪JQuery对象对应的组件后，会执行fn1，第二次点击会执行fn2

$(“#btn”).toggle(function(){

$(“#div”).css(“color”, “red”);

}, function(){

alert(“color”);

});

1. JQuery的插件

JQuery允许我们自定义插件来增强JQuery的功能，即增加JQuery的方法。

1. $.fn.extend(object)

增强JQuery获取的对象的功能，即为JQuery选中的元素添加方法。

如：为所有的JQuery对象添加check()方法：

$.fn.extend({

check:function(){

this.prop(“checked”, true);

}

})

1. $.extend(object)

增强JQuery对象自身的功能,可以不用通过获取JQuery对象来调用，可以直接通过$调用。即添加全局方法。

如，添加全局方法max，求最大值：

$.extend({

max:function(a, b){

return a >= b ? a : b;

}

})

调用max方法的话直接通过$.max（）调用即可。