jQuery框架

目录

[jQuery框架 1](#_Toc475345294)

[一、事件编程 2](#_Toc475345295)

[1、回顾JS中的三种绑定事件 2](#_Toc475345296)

[2、jQuery中的事件绑定 2](#_Toc475345297)

[3、jQuery中的事件冒泡 4](#_Toc475345298)

[4、jQuery中的默认行为 5](#_Toc475345299)

[二、jQuery中的动画效果 6](#_Toc475345300)

[1、基本效果（显示与隐藏） 6](#_Toc475345301)

[2、滑动效果（向上或向下滑动） 7](#_Toc475345302)

[3、淡入淡出效果 8](#_Toc475345303)

[4、自定义动画效果 9](#_Toc475345304)

[5、习题：实现一组图片的淡入淡出功能 10](#_Toc475345305)

[6、说说事件编程中的this关键字 10](#_Toc475345306)

[7、习题：实现一款后台折叠菜单 11](#_Toc475345307)

[三、jQuery中的文档处理 13](#_Toc475345308)

[1、内部插入 13](#_Toc475345309)

[2、外部插入 13](#_Toc475345310)

[3、删除操作 14](#_Toc475345311)

[4、复制操作（克隆操作） 15](#_Toc475345312)

[5、替换操作 15](#_Toc475345313)

[6、包裹操作 16](#_Toc475345314)

[7、查找操作 16](#_Toc475345315)

[四、jQuery中的插件机制 17](#_Toc475345316)

[1、jQuery插件机制 17](#_Toc475345317)

[2、习题：实现后台全选、全不选、反选功能 18](#_Toc475345318)

[五、jQuery课程总结 20](#_Toc475345319)

[1、事件编程 20](#_Toc475345320)

[2、jQuery动画效果 20](#_Toc475345321)

[3、jQuery文档操作 21](#_Toc475345322)

[4、插件机制 21](#_Toc475345323)

# 一、事件编程

## 1、回顾JS中的三种绑定事件

① 行内绑定

② 动态绑定

③ 事件监听

## 2、jQuery中的事件绑定

* bind(type,[data],fn) ：为某个对象的某个事件绑定相应的事件程序

参数说明：

type ：事件类型，不带’on’前缀的，如click,mouseover,mouseout

[data] ：可选参数，事件发生时所传递的数据

fn ：事件的处理程序

* bind({type:fn,type:fn}) ：为某个对象同时绑定多个事件

参数说明：

特别注意：使用此方法，其参数必须是json格式的数据

type ：事件的类型

fn ：事件的处理程序

* one(type,[data],fn) ：为某个对象的事件绑定相应事件处理程序，但是只触发一次

参数说明：

type ：事件类型，不带’on’前缀的，如click,mouseover,mouseout

[data] ：可选参数，事件发生时所传递的数据

fn ：事件的处理程序

* unbind([type]) ：移除事件

参数说明：

[type] ：可选参数，不写代表移除当前对象的所有事件。

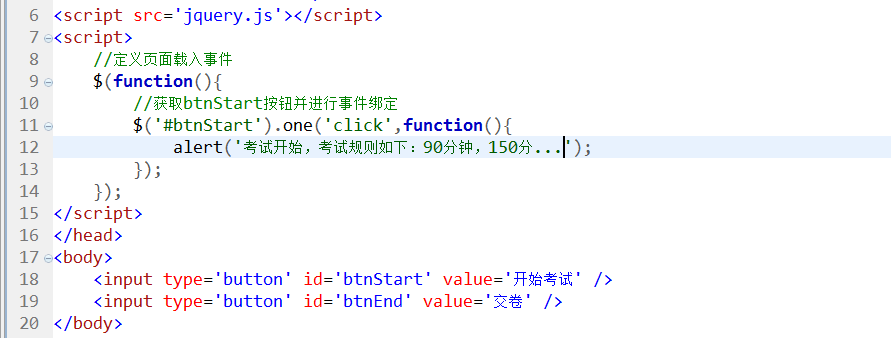
demo01\_bind.html 示例代码：



demo02\_bind.html 示例代码：

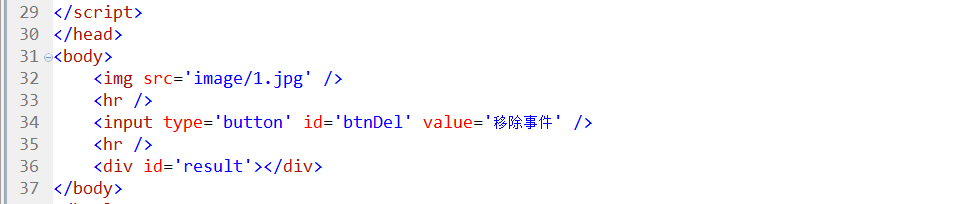


demo03\_one.html 示例代码：



demo04\_ unbind.html 示例代码：





## 3、jQuery中的事件冒泡

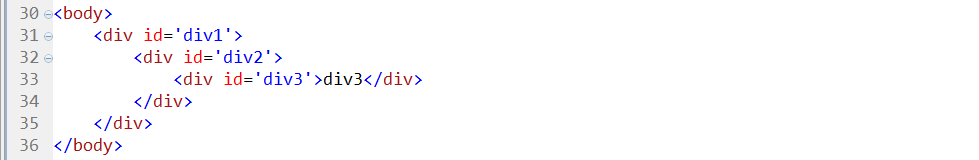
什么是事件冒泡？

所谓的事件冒泡是指事件的响应像水泡一样上升至最顶级对象，我们把这个过程就称之为“事件冒泡”。



demo05\_bubble.html 示例代码：





说明：有些情况下，事件冒泡是人为设计的，但是大多数情况，冒泡需要被禁止！

jQuery中禁止事件冒泡的基本语法：

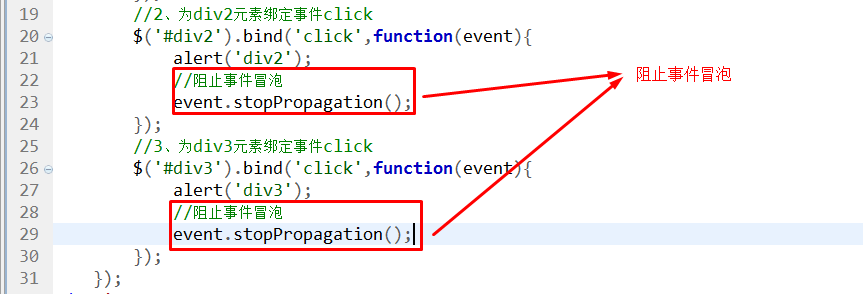
jQuery对象.bind(‘事件’,function(event){

//禁止事件冒泡

event.stopPropagation();

});

示例代码：



## 4、jQuery中的默认行为

在HTML标签中，有些元素拥有自己的默认行为。如：

a标签：单击后会自动跳转到href所指向的页面

submit按钮：单击后会自动提交表单

但是有些情况下，我们并不需要元素的默认行为，我们可以使用如下方法进行禁止！

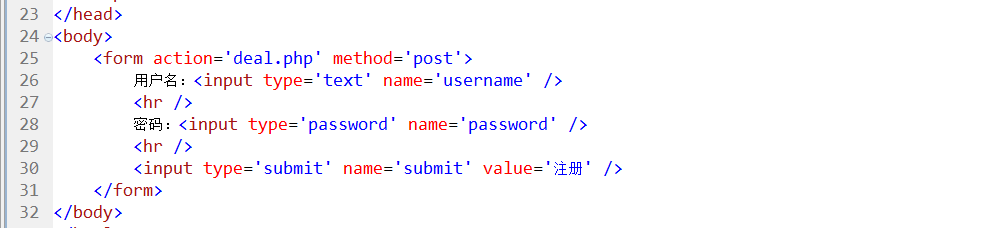
jQuery对象.bind(‘事件’,function(event){

event.preventDefault();

});

demo06\_default.html 示例代码：





# 二、jQuery中的动画效果

## 1、基本效果（显示与隐藏）

* show() ：显示
* show(speed,[callback]) ：以动画效果显示

参数说明：

speed ：动画的持续时间，可以是"slow", "normal", "fast"；也可以是具体的时间

[callback] ：当动画完成时所触发的回调函数

* hide() ：隐藏
* hide(speed,[callback]) ：以动画效果隐藏

参数说明：

speed ：动画的持续时间，可以是"slow", "normal", "fast"；也可以是具体的时间

[callback] ：当动画完成时所触发的回调函数

* toggle() ：显示或隐藏
* toggle(switch) ：显示或隐藏，true代表一直显示，false代表一直隐藏
* toggle(speed,[callback]) ：以动画效果显示或隐藏

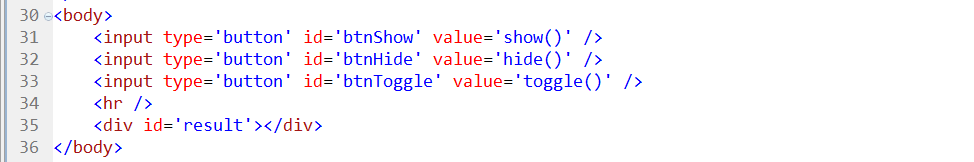
参数说明：

speed ：动画的持续时间，可以是"slow", "normal", "fast"；也可以是具体的时间

[callback] ：当动画完成时所触发的回调函数

demo07\_jiben.html 示例代码：





## 2、滑动效果（向上或向下滑动）

* slideDown(speed,[callback]) ：以动画效果向下滑动
* slideUp(speed,[callback]) ：以动画效果向上滑动
* slideToggle(speed,[callback]) ：以动画效果向上或向下滑动

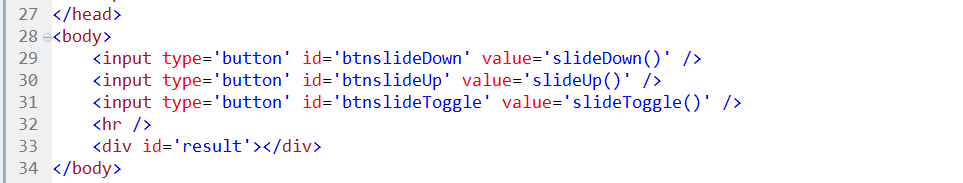
参数说明：

speed ：动画的持续时间

[callback] ：可选参数，动画完成时所触发的回调函数

demo08\_slide.html 示例代码：





## 3、淡入淡出效果

* fadeIn(speed,[callback]) ：以动画效果淡入
* fadeOut(speed,[callback]) ：以动画效果淡出

参数说明：

speed ：动画的持续时间

[callback] ：可选参数，动画完成时所触发的回调函数

* fadeTo(speed,opacity,[callback]) ：以动画效果设置元素的透明度

参数说明：

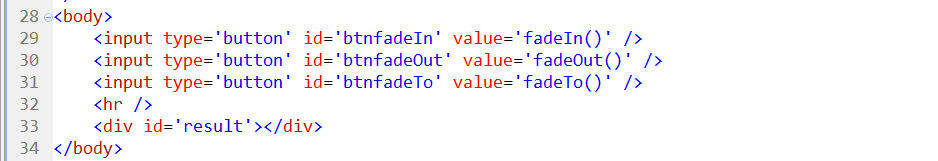
speed ：动画的持续时间

opacity ：元素的透明度，从0到1（0代表全透明，1代表不透明，0-1之间代表半透明）

[callback] ：动画完成时所触发的回调函数

demo09\_fade.html 示例代码：





## 4、自定义动画效果

* animate(params,[speed]) ：自定义动画

参数说明：

params ：要求是一个json对象，动画效果

speed ：动画的持续时间

demo10\_animate.html 示例代码：



## 5、习题：实现一组图片的淡入淡出功能

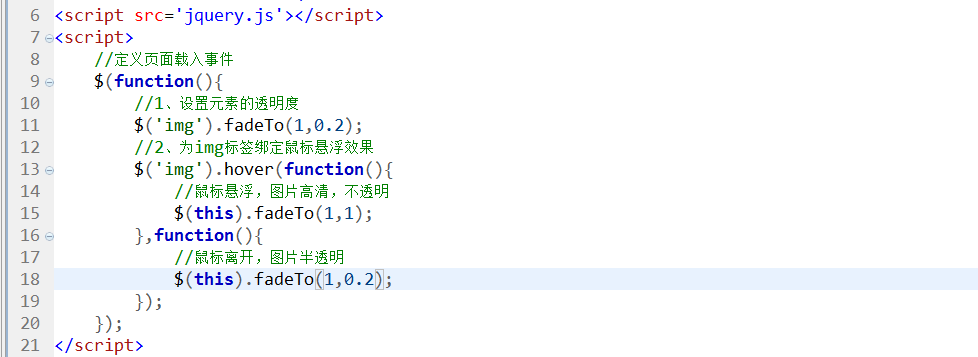
习题要求：

① 当进入Web页面之前，所有都图片都必须是半透明的

② 当鼠标悬浮在某张图片上时，要求图片变成不透明效果（高清）

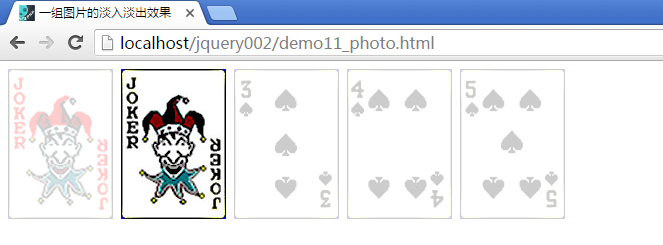
③ 当鼠标离开时，要求图片恢复成半透明效果

demo11\_photo.html 示例代码：





运行结果：



## 6、说说事件编程中的this关键字

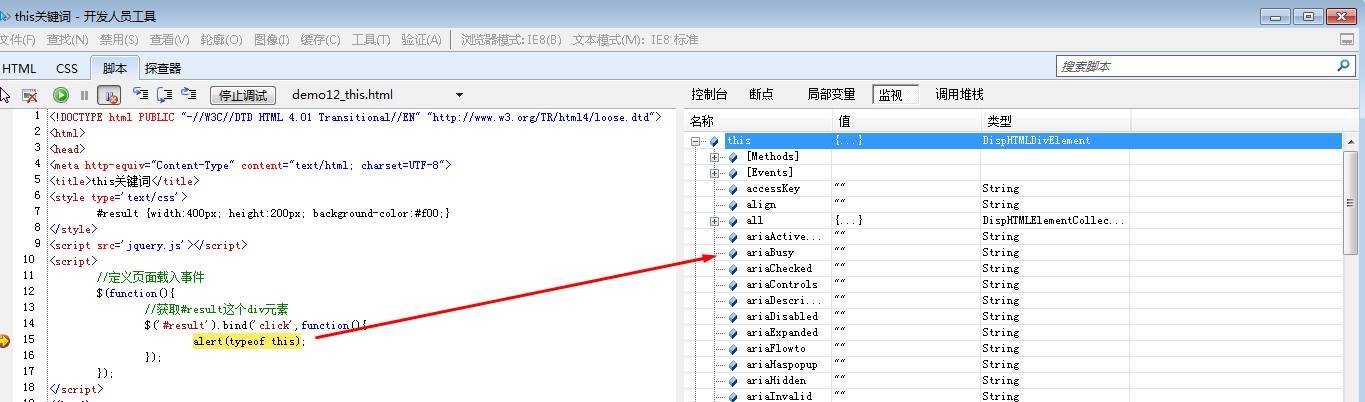
问题：在动态绑定中，this关键字指向当前正在操作的DOM对象，那么在jQuery中，其事件处理程序内部的this指向何方呢？

demo12\_this.html 示例代码：



运行结果：弹出object，所以可以证明，在jQuery事件处理程序中的确存在一个this关键词，那么this关键词代表什么对象么？① DOM对象 ② jQuery对象

使用调试工具，调试，如下图所示：



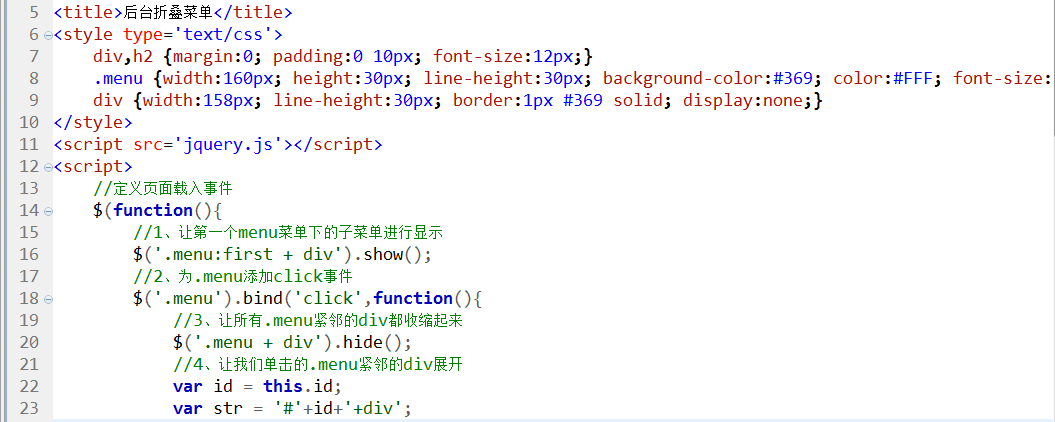
由此可知，jQuery中事件处理程序内部的确存在一个this，其指向了当前正在操作的DOM对象。

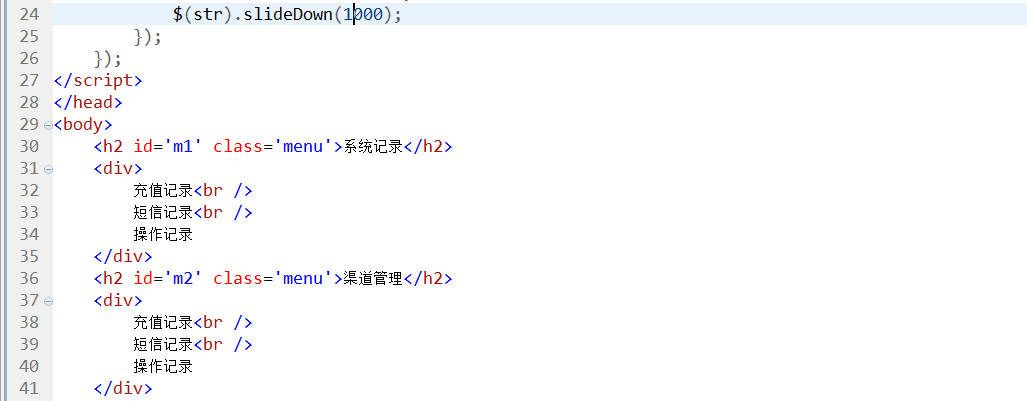
示例代码：

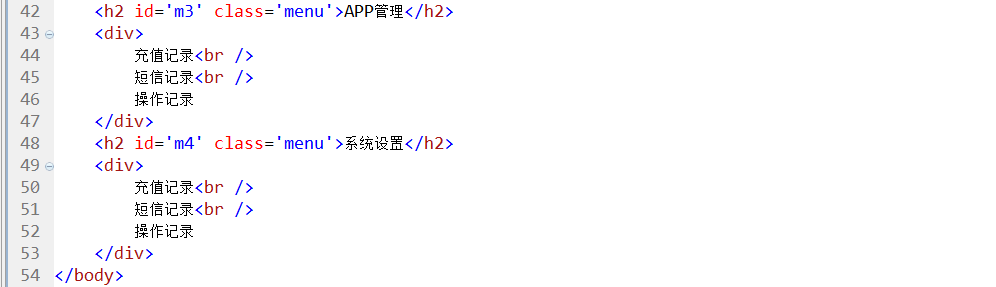


## 7、习题：实现一款后台折叠菜单

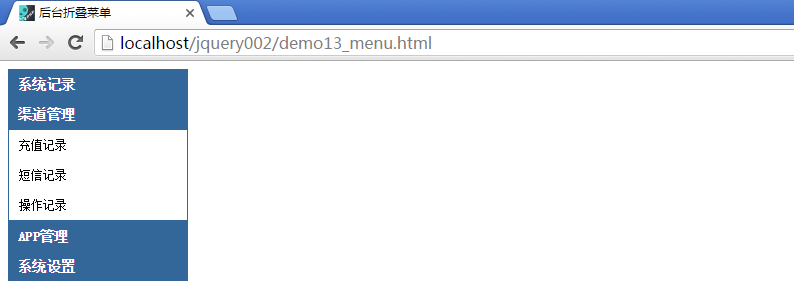
demo13\_menu.html 示例代码：







运行结果：



# 三、jQuery中的文档处理

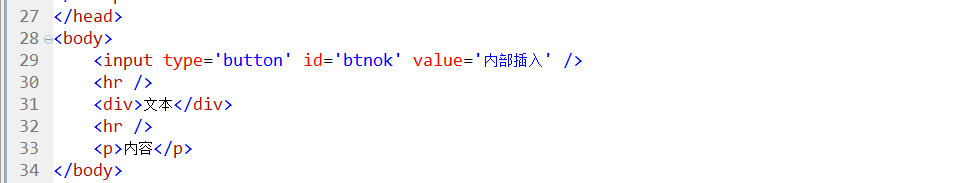
## 1、内部插入

例：<div>content文本content</div>

* append(content) ：把content内容追加到元素内容的尾部
* appendTo(content) ：把匹配到的内容插入到content内容的尾部
* prepend(content) ：把content内容插入到元素内容的头部
* prependTo(content) ：把匹配到的内容插入到content内容的头部

demo14\_neicha.html 示例代码：





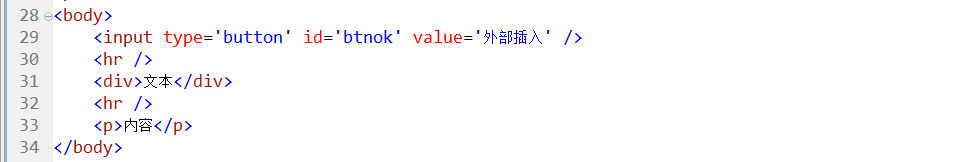
## 2、外部插入

例：content <div>文本</div>content

* after(content) ：把content内容插入到元素的尾部
* before(content) ：把content内容插入到元素的头部
* insertAfter(content) ：把匹配到的内容插入到content元素的尾部
* insertBefore(content) ：把匹配到的内容插入到content元素的头部

demo15\_waicha.html 示例代码：





## 3、删除操作

* empty() ：清空元素的内容，但是不删除元素自身
* remove() ：清空元素的内容同时删除元素自身

demo16\_del.html 示例代码：

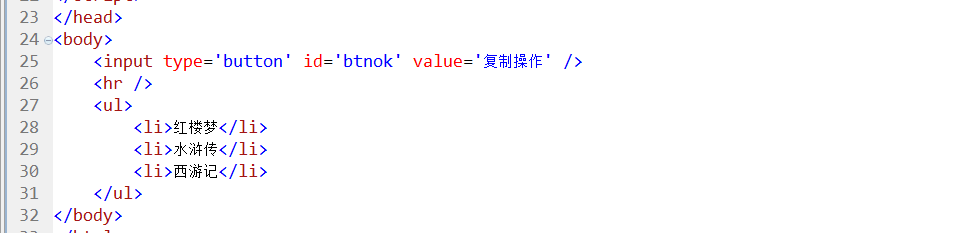


## 4、复制操作（克隆操作）

* clone() ：复制元素本身
* clone(true) ：复制元素本身还会复制元素自身的事件

demo17\_clone.html 示例代码：

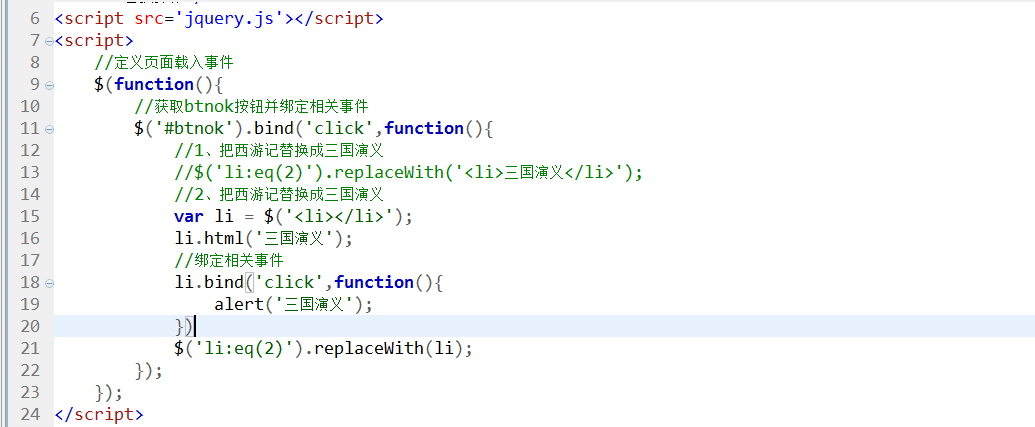


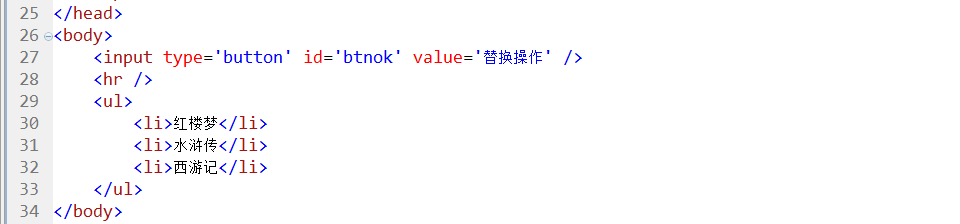


## 5、替换操作

* replaceWith() ：元素的替换操作

demo18\_replace.html 示例代码：





## 6、包裹操作

* wrap() ：对所有匹配的元素进行包裹操作
* wrapAll() ：对所有匹配的元素进行一次包裹操作
* wrapInner() ：对所有匹配的元素内容进行包裹操作

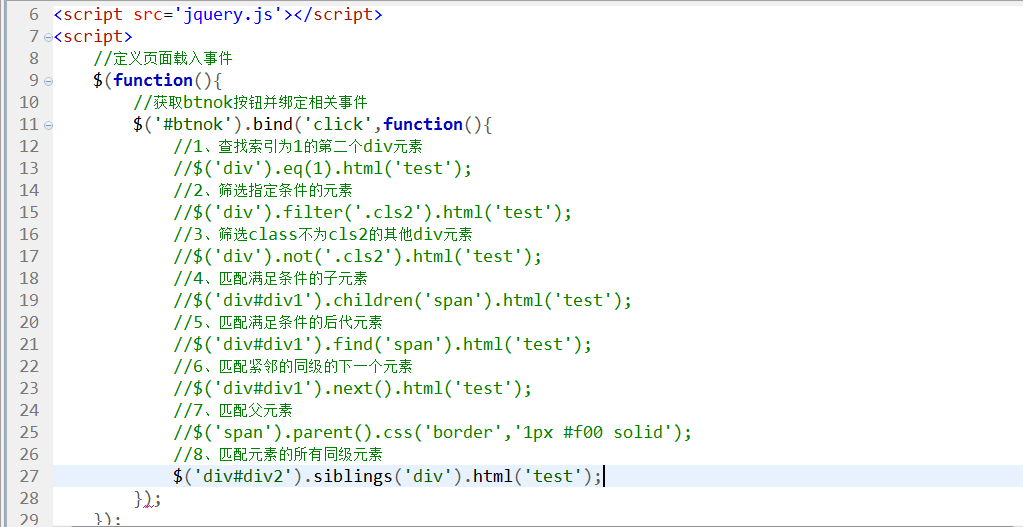
demo19\_wrap.html 示例代码：

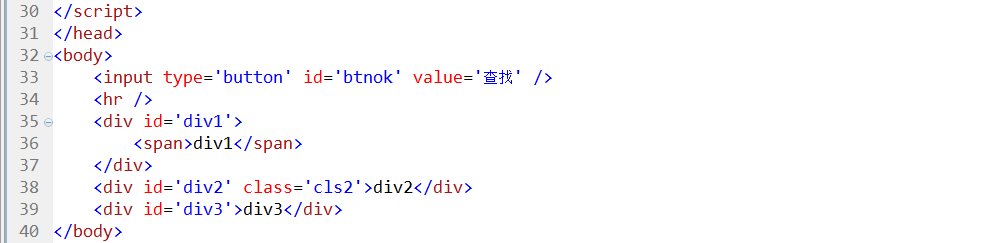


## 7、查找操作

* eq(index) ：根据元素的index索引来匹配元素，默认从0开始
* filter(expr) ：使用过滤器来筛选元素
* not(expr) ：匹配除指定选择器以外的其他元素
* children([expr]) ：匹配满足条件的子元素（一级关系）
* find(expr) ：匹配满足条件的所有后代元素（多级关系）
* next([expr]) ：匹配紧邻的同级的下一个元素
* prev([expr]) ：匹配紧邻的同级的上一个元素
* parent([expr]) ：匹配父元素
* siblings([expr]) ：匹配所有同级元素（无论上下）

demo20\_chazhao.html 示例代码：





# 四、jQuery中的插件机制

## 1、jQuery插件机制

基本语法：

jQuery.fn.extend(object) 或 $.fn.extend(object);

参数说明：

object ：要求数据格式为json对象，格式如下：

jQuery.fn.extend({

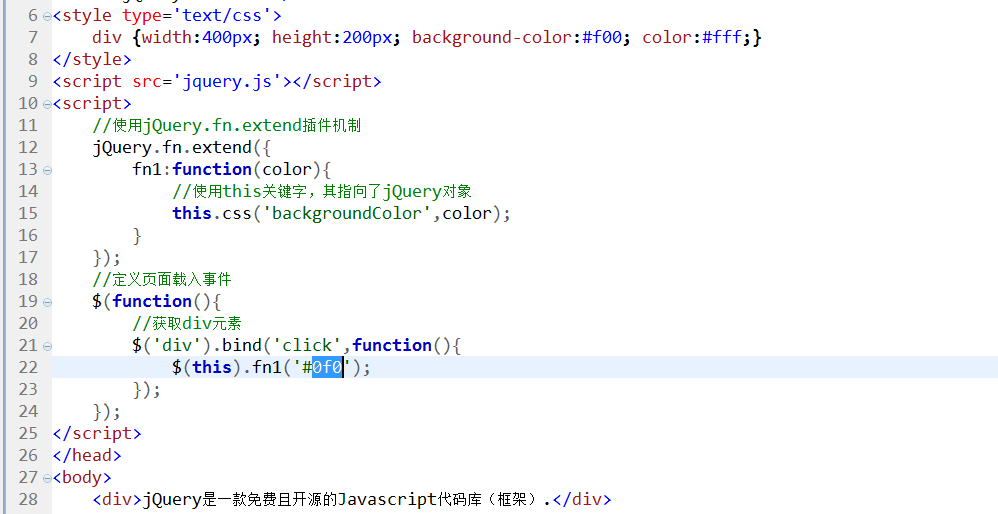
fn1:function(){},

fn2:function(){},

......

});

demo21\_extend.html 示例代码：



## 2、习题：实现后台全选、全不选、反选功能

demo22\_anli.php 示例代码：

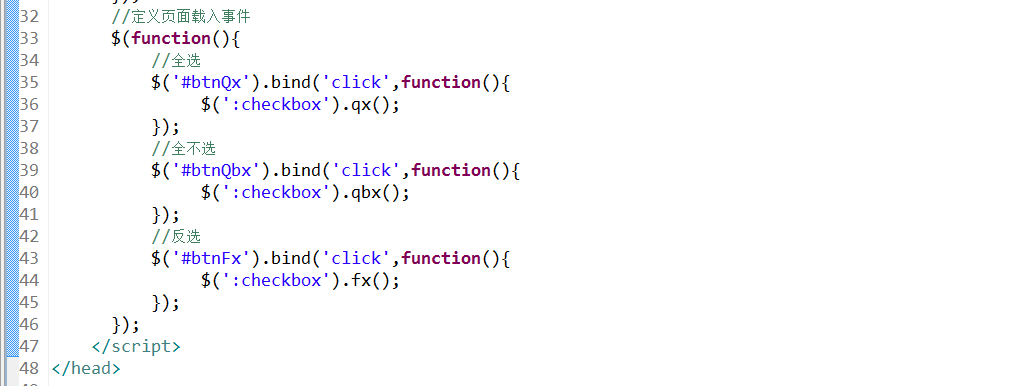


使用php+mysql读取数据表中的数据:



编写jQuery代码，如下图所示：





运行结果：



# 五、jQuery课程总结

## 1、事件编程

① 行内绑定 ② 动态绑定 ③ 事件监听

bind方法进行事件绑定，其是采用事件监听的方式进行实现的。

one绑定：一次绑定

unbind取消绑定

事件冒泡

event.stopPropagation(); 取消事件冒泡

默认行为

event.preventDefault(); 取消默认行为

## 2、jQuery动画效果

基本效果，show/hide/toggle

滑动效果，slideUp/slideDown/slideToggle

淡入淡出效果，fadeIn/fadeOut/fadeTo

自定义动画，animate

事件处理程序中都有一个关键字this，其指向了当前正在操作的dom对象，如果想使用jquery中的方法必须使用$(dom)封装

## 3、jQuery文档操作

内部插入 append/prepend

外部插入 after/before

删除操作 empty/remove

复制操作 clone

替换操作 replaceWith

包裹操作 wrap/wrapAll/wrapInner

查找操作 eq/next/prev/children/find/not/filter/siblings

## 4、插件机制

jQuery.fn.extend或$.fn.extend

window.jQuery = window.$