# jQuery简介

## jQuery简介

### 什么是jQuery?

jQuery：JavaScript Query(JS查询), 是一个轻量的, 免费开源的JS函数库, 能够极大的简化JS代码。

### jQuery的优势

（1）可以极大简化JS代码

（2）可以像CSS那样来获取页面元素

例如:为所有的div设置背景颜色

在CSS中：div{ background-color:red }

在jQuery中：$(“div”).css(“background-color”, “red”);

在CSS中：#d1{ background-color:red }

在jQuery中：$(“#d1”).css(“background-color”, “red”);

（3）可以操作CSS属性来控制页面的效果

（4）可以兼容常用的浏览器

在JS中少数内容在不同的浏览器中不兼容。比如：innerText、removeNode()、replaceNode()等在火狐中无法使用.

### jQuery的版本支持

1.x – 支持常用浏览器, 以及IE6+以上的版本

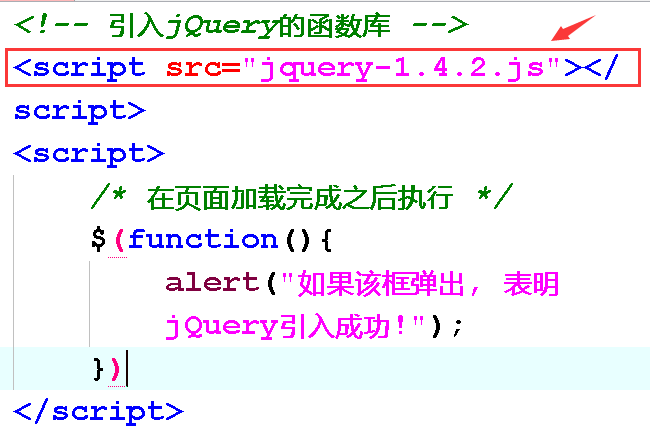
2.x – 支持常用浏览器, 以及IE9+以上的版本

3.x – 支持常用浏览器, 以及IE9+以上的版本

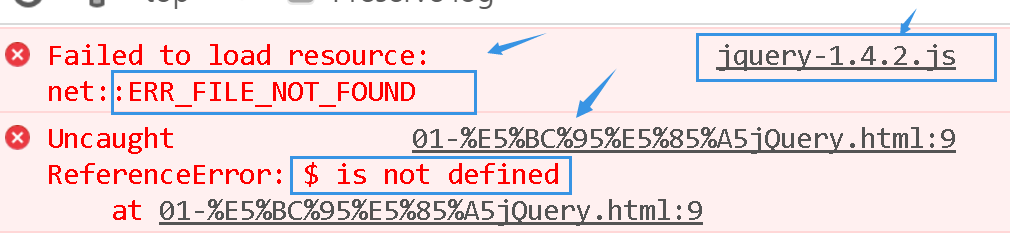
jQuery的缺点：jQuery的高版本不兼容低版本。这是因为jQuery在升级时除了会做一些内部优化之外，还会增加或删除一些方法。如果升级到高版本，可能会造成之前的部分代码无法执行!

### jQuery引入

1.jQuery函数库本身是一个js文件, 所以引入jQuery和引入一个普通的JS文件一样



2.FAQ常见问题：在引入的过程中，经常会出现文件引入失败的错误:

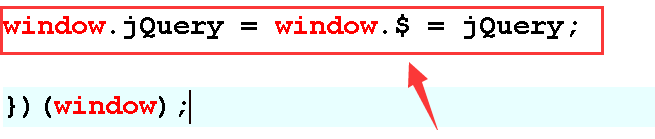


文件引入失败是往往是由于jQuery库文件的路径错误导致的!!

# jQuery语法

## $符号介绍

$符号等价于jQuery, 参见jQuery源码:



调用$()等价于调用jQuery(), 该函数会返回一个jQuery对象, 该对象包含了若干个html元素, 可以调用jQuery中提供的方法或属性来操作这些元素.

$(“div”).css(“background-color”, “red”) – 为所有的div设置背景颜色为red

## 文档就绪事件

在jQuery中提供了一个文档就绪事件函数:

$(document).ready(function(){

//在整个html文档加载完成之后立即触发执行…

});

等价于window.onload事件：在整个html文档加载完成之后立即触发

其简写形式为:(推荐!)

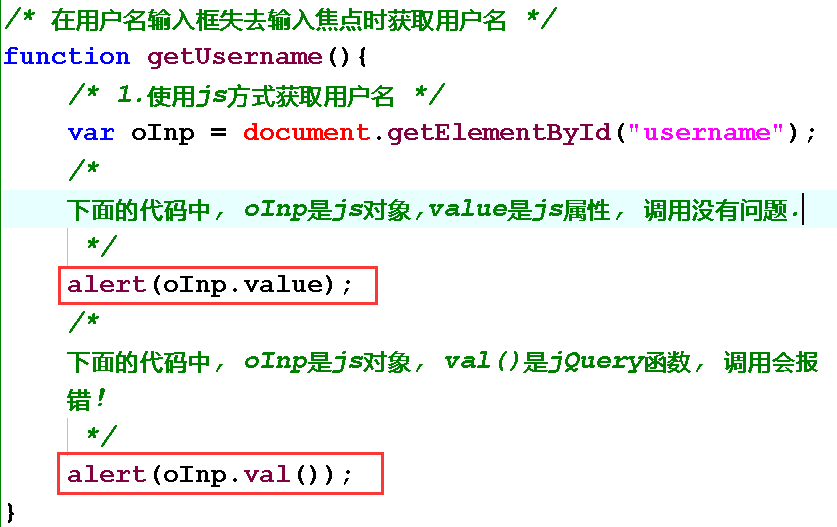
$(function(){

//在文档加载完之后立即触发执行…

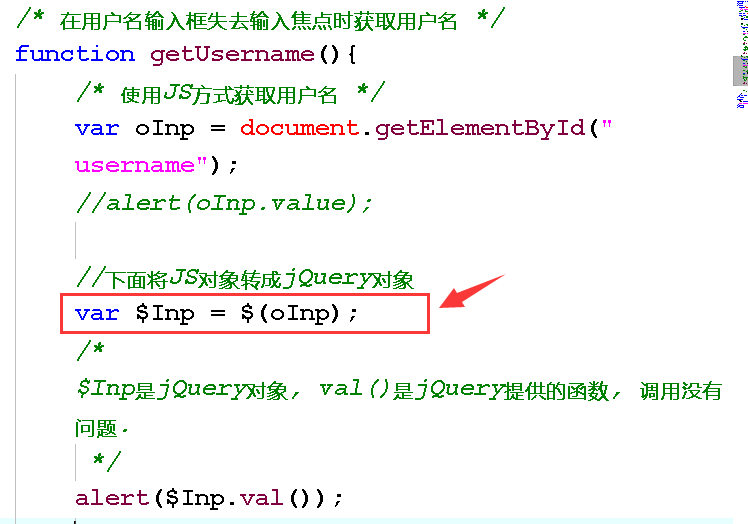
});

* 1. **jQuery对象和JS对象的相互转换**
     1. **为什么需要相互转换**

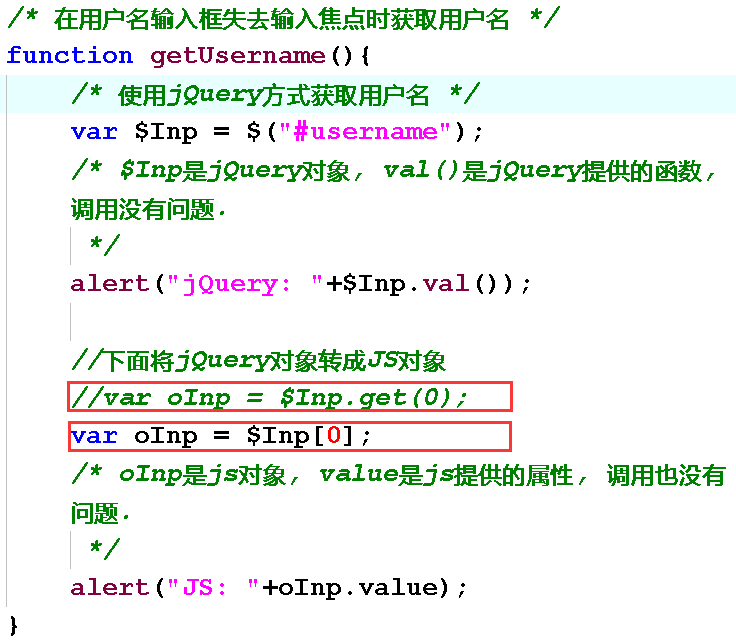
jQuery是js的简单框架, 在使用时通过$()可以返回一个jQuery对象, 可以通过jQuery提供的属性或方法来操作其中的内容, 但是, 不可以通过jQuery对象来调用js的属性或者js的方法. 反之, 也不可以通过JS对象来调用jQuery的属性或者jQuery方法, 否则调用会报错!!



### JS对象转成jQuery对象



* + 1. **jQuery对象转成JS对象**

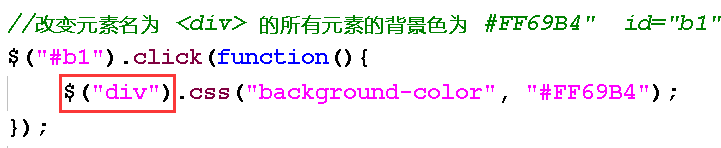


# jQuery选择器(重点)

## 基本选择器

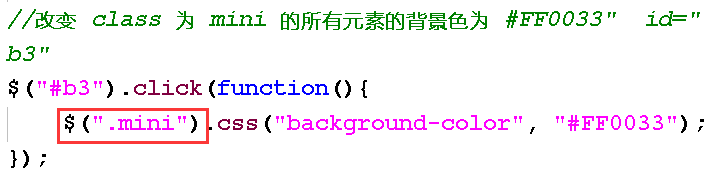
1、元素名(标签名)选择器：根据元素的名称选中指定名称的元素

格式：$("元素名称")



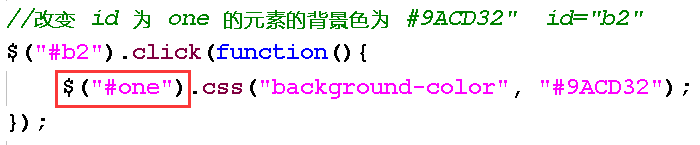
2、类选择器：根据元素的class属性, 匹配所有class值相同的元素

格式：$(".class值")



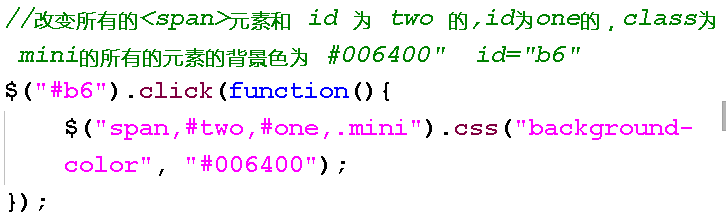
3.ID选择器：根据元素的id属性值, 匹配具有特定id的元素

格式：$("#id值")



4.多元素选择器：匹配多个选择器选中的元素

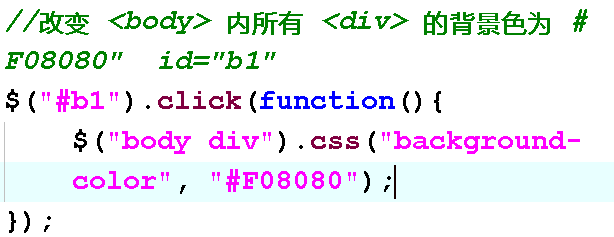
格式：$("选择器1,选择器2,…选择器n")



## 层级选择器

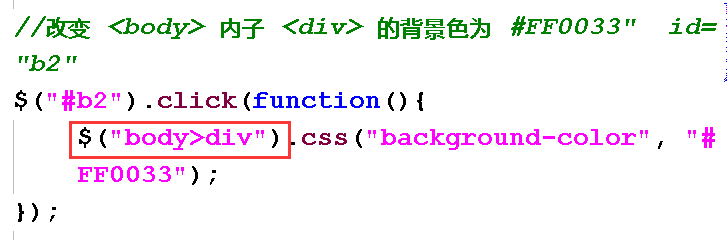
1、后代选择器

格式：$("祖先元素 后代元素")



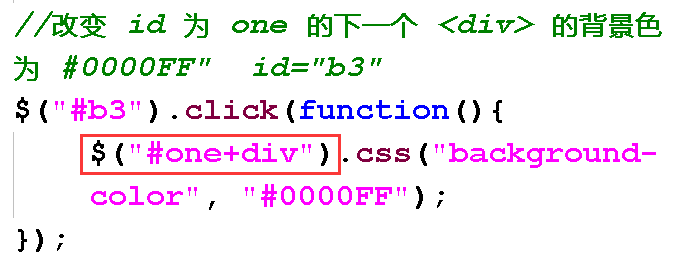
2.子元素选择器

格式：$("祖先元素>子元素")

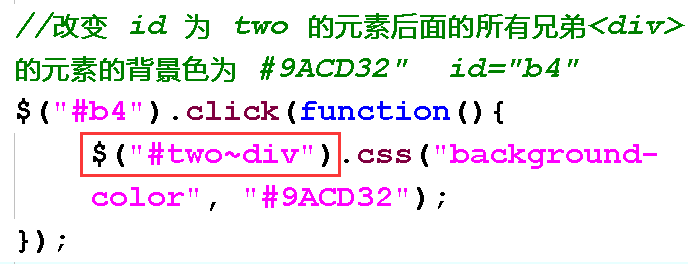


3.相邻兄弟选择器

格式：$("大哥+小弟")



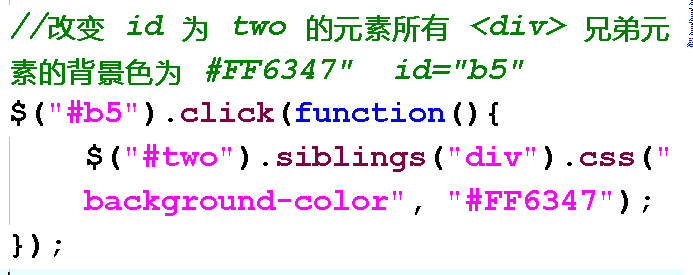
4.$(“#two~div”)：匹配id为two元素后面所有的兄弟div元素



**相关方法介绍：**

（1）siblings()方法：返回所匹配元素的所有兄弟元素

示例：



（2）next()方法，示例：

$(“#d1”).next(“span”) – 匹配id为d1元素后面紧邻的span兄弟元素

（3）nextAll()方法，示例：

$(“#d1”).nextAll(“span”) – 匹配id为d1元素后面所有的span兄弟元素

（4）prev()方法，示例：

$(“#d1”).prev(“span”) – 匹配id为d1元素前面紧邻的span兄弟元素

（5）prevAll()方法，示例：

$(“#d1”).prevAll(“span”) – 匹配id为d1元素前面所有的span兄弟元素

### 基本过滤选择器

1. :first

$(“div:first”) – 匹配所有div中的第一个div元素

2. :last

$(“div:last”) – 匹配所有div中的最后一个div元素

3. :not

$(“div:not(.one)”) – 匹配所有div中class值不为one的div元素

4. :eq

$(“div:eq(3)”) – 匹配所有div中的索引值为3的div元素

5. :lt

$(“div:lt(3)”) – 匹配所有div中的索引值小于3的div元素

6. :gt

$(“div:gt(3)”) – 匹配所有div中的索引值大于3的div元素

7. :even(偶数)

$(“div:even”) – 匹配所有div中的索引值为偶数的div元素

8. :odd

$(“div:odd”) – 匹配所有div中的索引值为奇数的div元素

## 内容选择器

1. contains

$(“div:contains(‘xxx’)”) – 匹配所有div中包含xxx文本的div元素

1. empty

$(“div:empty”) – 匹配所有div中内容为空的div元素

1. has

$(“div:has(span)”) – 匹配所有div中包含span子元素的div元素

1. parent

$(“div:parent”) – 匹配所有div中内容不为空的div元素

## 可见选择器

1.hidden

$(“div:hidden”) – 匹配所有隐藏的div元素

2.visible

$(“div:visible”) – 匹配所有可见的div元素

## 属性选择器

1.$(“div[id]”) – 匹配所有具有id属性的div元素

2.$(“div[id=’d1’]”) – 匹配具有id属性并且值为d1的div元素

3.$(“div[id!=’d1’]”) – 匹配具有id属性并且值不等于d1的div元素

4.$(“div[id^=’d1’]”) – 匹配具有id属性并且值是以d1开头的div元素

5.$(“div[id$=’d1’]”) – 匹配具有id属性并且值是以d1结尾的div元素

## 子元素选择器

1.$(“div span:first-child”) – 匹配div内部的第一个span元素

2.$(“div>span:first-child”) – 匹配div内部的第一个span子元素

3.$(“div span:nth-child(n)”) – 匹配div内部的第n个span元素

4.$(“div>span:nth-child(n)”) – 匹配div内部的第n个span子元素

## 子元素选择器

1.$(“div span:first-child”) – 匹配div内部的第一个span元素

2.$(“div>span:first-child”) – 匹配div内部的第一个span子元素

3.$(“div span:nth-child(n)”) – 匹配div内部的第n个span元素

4.$(“div>span:nth-child(n)”) – 匹配div内部的第n个span子元素

## 表单选择器

1. :input

$(“:input”) – 匹配所有的表单项元素(包括input,select,textarea等)

1. :text

$(“:text”) – 匹配所有的单行文本输入框元素

1. :password

$(“:password”) – 匹配所有的密码输入框

1. :radio

$(“:radio”) – 匹配所有的单选框

1. :checkbox

$(“:checkbox”) – 匹配所有的复选框

1. :checked

$(“:checked”) – 匹配所有被选中的单选框, 复选框, option选项

# 文档操作(练习案例)

## html元素的增删改查

1.创建新元素

$(“<div></div>”) – 创建一个div元素

$(“<div>hello…</div>”) – 创建一个包含文本内容的div元素

$(“<div><span></span></div>”) – 创建一个包含span子元素的div元素

2.添加子元素

$(“div”).append($span); -- 为div添加一个span子元素

3.删除元素本身

$(“div”).remove(); -- 删除匹配的所有div元素

4.替换元素

$(“div”).replaceWith($span) – 将匹配的所有div元素替换为span元素

## html属性/值操作

1.html() – 获取或设置元素的html内容

$(“div”).html() – 获取所匹配元素中的第一个元素的html内容

$(“div”).html(“<span>xxx</span>”) – 为所有匹配的div元素设置html内容

2.text() – 获取或设置元素的文本内容

$(“div”).text() – 获取所匹配元素的所有文本内容

$(“div”).text(“xxx”) – 为所有匹配的div元素设置文本内容

3.attr() – 获取或设置元素的属性

$(“div”).attr(“id”) – 获取所匹配元素中的第一个元素的id属性值

$(“div”).attr(“id”, “div1”) – 为所匹配的div元素设置id属性值

## 操作CSS样式

1.css()函数

$(“div”).css(“background-color”) – 获取所匹配的div元素中的第一个div元素的背景颜色

$(“div”).css(“background-color”, “red”) – 为所匹配的div元素设置背景颜色

## 效果

1.hide() — 设置所有匹配的元素的为隐藏

2.show() – 设置所有匹配的元素为显示

3.toggle() – 切换元素的显示状态, 如果显示则切换为隐藏, 如果是隐藏则切换为显示.

# 事件

## 常用事件介绍

1.click – 点击事件函数, 在所匹配的元素被点击时触发执行

2.blur – 输入框失去输入焦点事件, 在输入框失去输入焦点时触发执行

3.focus – 输入框获得输入焦点事件, 在输入框获得输入焦点时触发执行

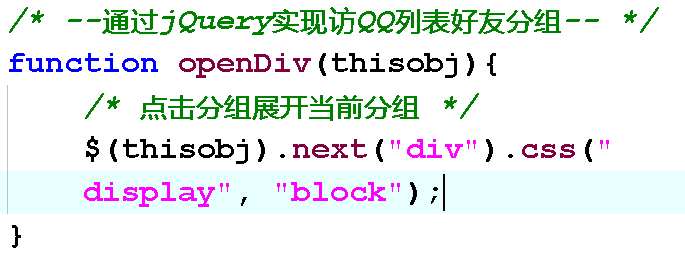
4.change – 下拉选框选项切换事件, 在选项切换时触发执行

5.ready – 文档就绪事件, 在整个html文档加载完成后立即触发执行

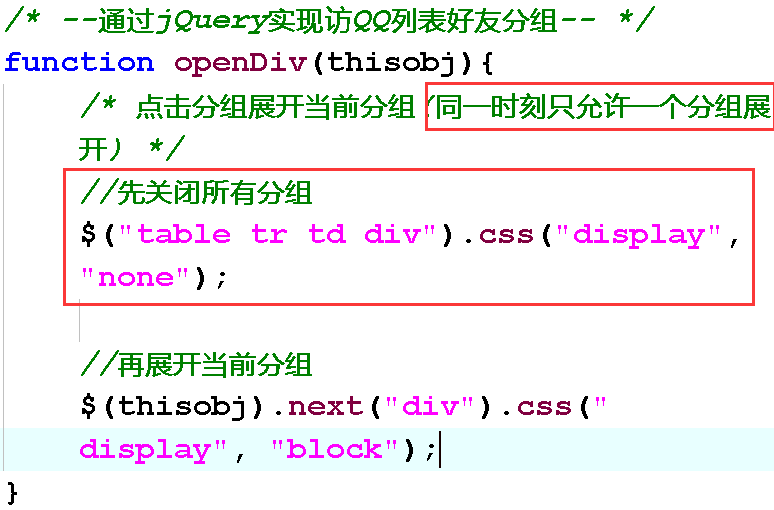
# 综合练习

## 仿QQ好友列表

**1、version\_1:** 实现点击分组名称, 展开当前分组

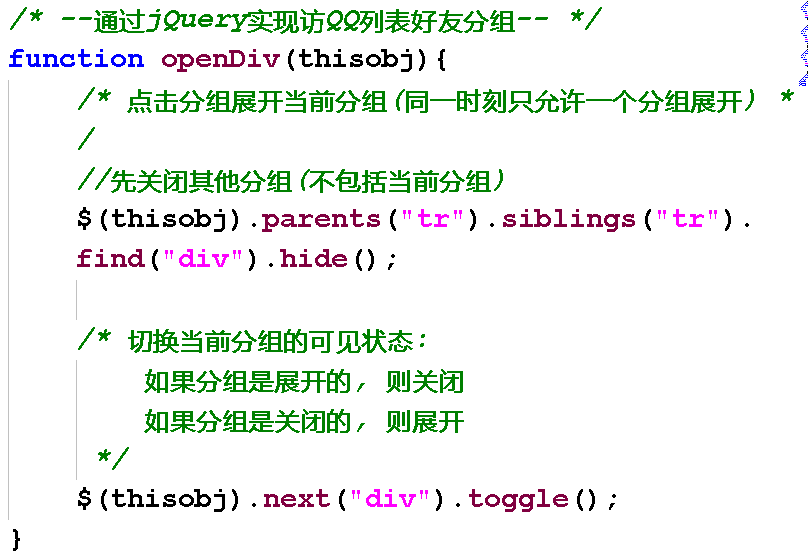


**2、version\_2:** 再展开当前分组之前, 先关闭所有的分组, 再展开当前分组

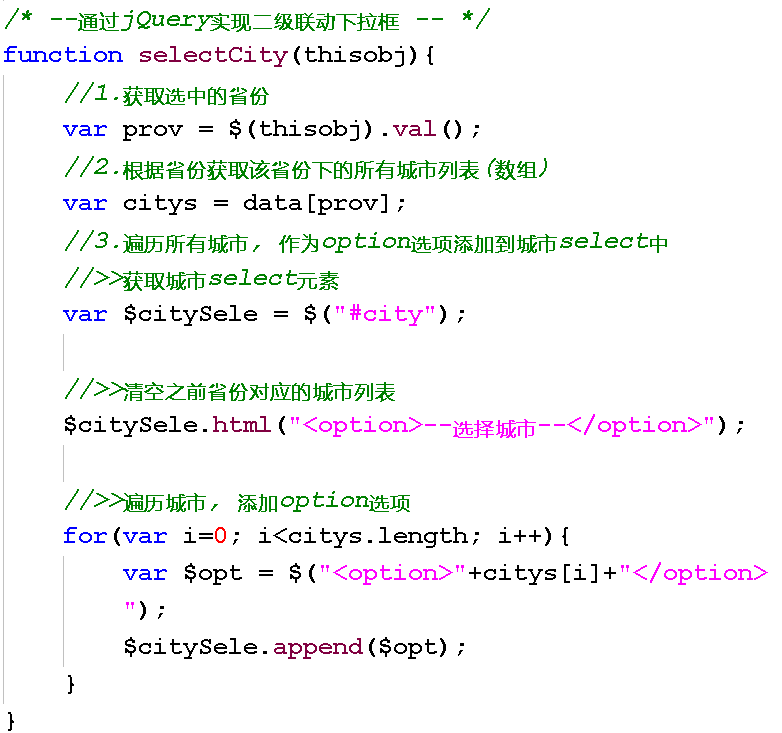


**3、version\_3:** 如果当前分组是展开的, 点击后则关闭分组

如果当前分组是关闭的, 点击后先关闭其他分组, 再展开当前分组

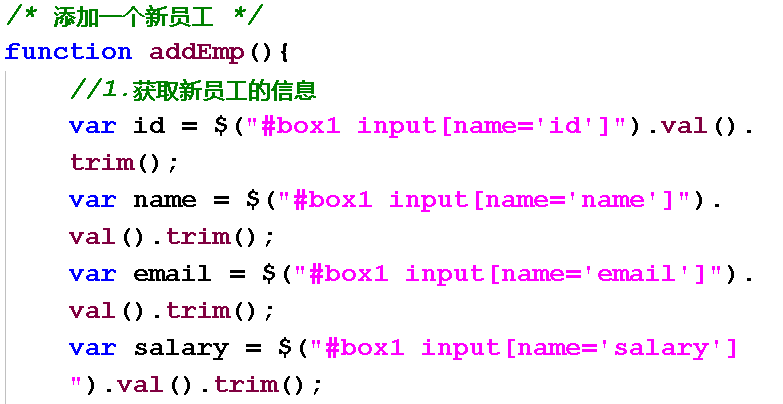


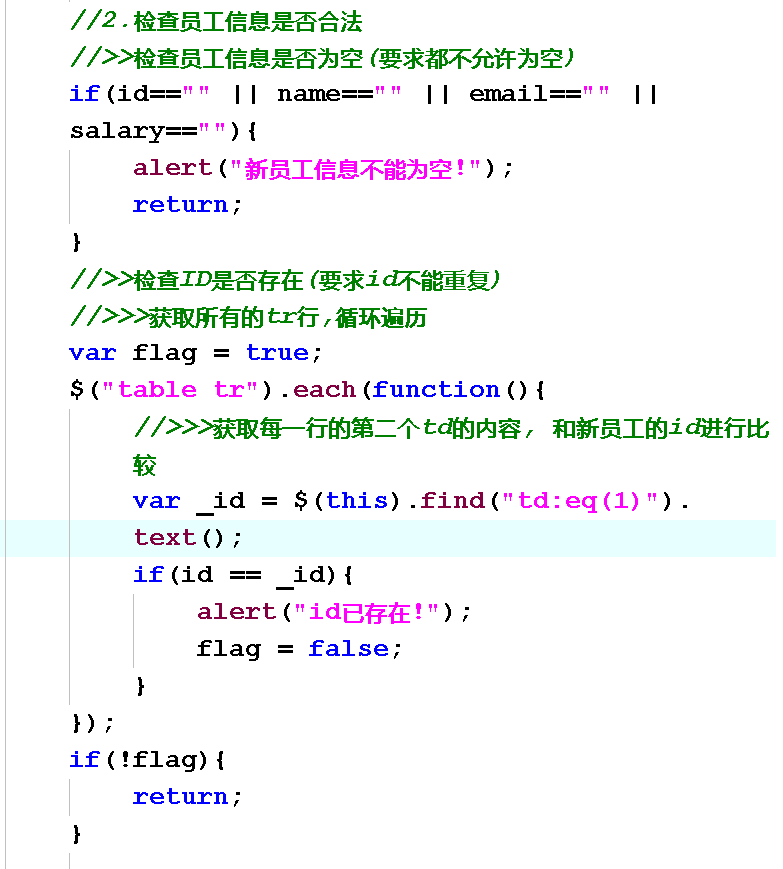
* 1. **二级联动下拉框**

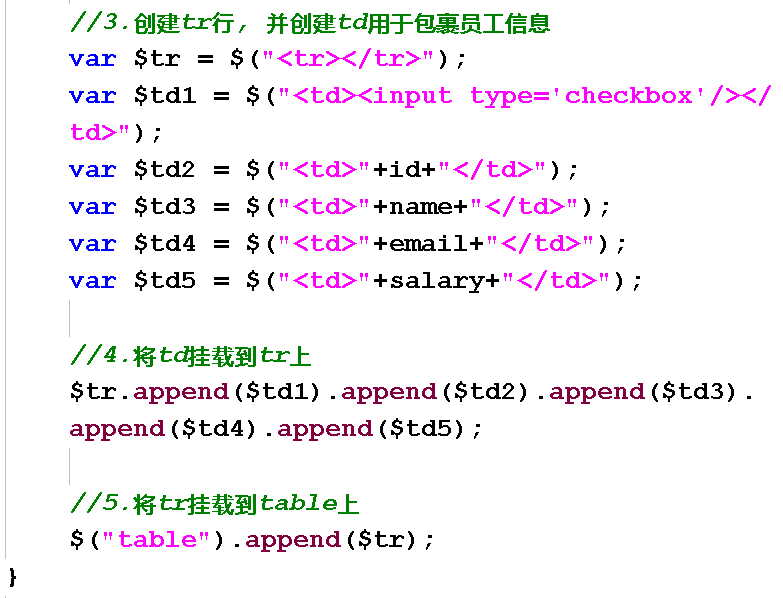


* 1. **模拟员工信息管理系统**

### 添加员工

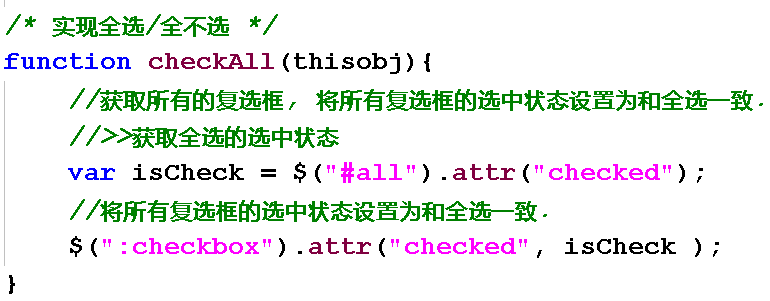




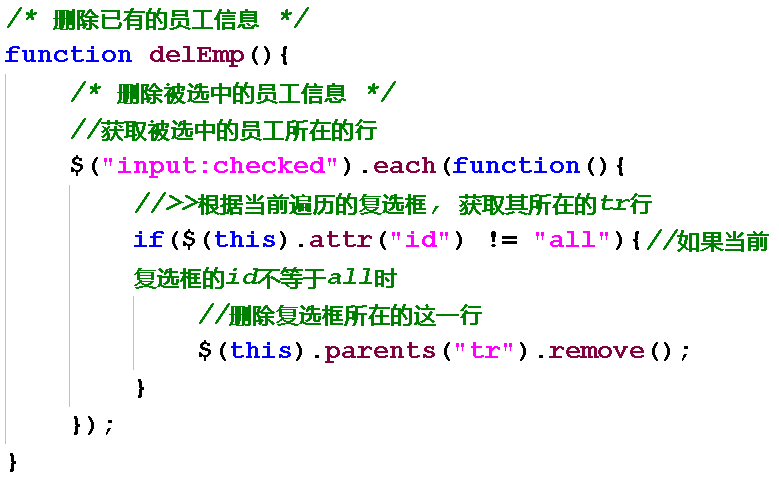


* + 1. **删除员工**

1、实现全选/全不选



2、删除被选中的员工信息



* + 1. **修改员工信息**

