# Day04笔记-JavaScript&jquery（入门）

# 1、案例1：完善注册表单校验

## 1.1、需求说明及分析



之前注册校验通过警告框来提示信息，这使得信息的提示很不友好。

所以我们对信息的提示进行改进，使错误信息的提示以图示方式进行展示。

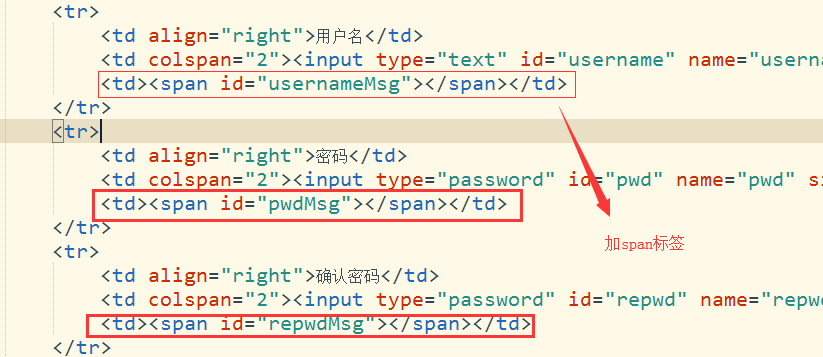
## 1.2、JavaScript知识讲解(一)

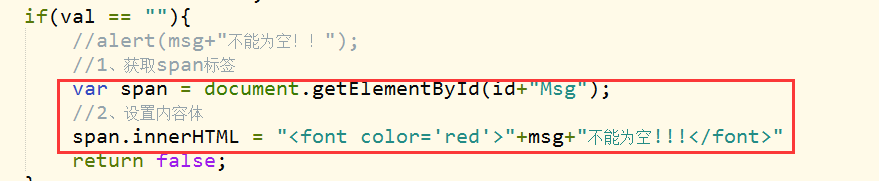
### 1.2.1、操作元素内容体-innerHTML

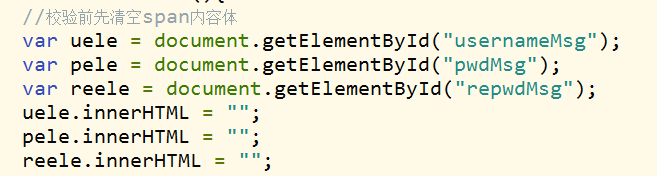
|  |  |
| --- | --- |
| 属性名 | 属性说明 |
| **元素对象.innerHTML** | 获取某个元素的内容体信息 |
| **元素对象.innerHTML=值** | 设置某个元素的内容体信息 |
|  |  |

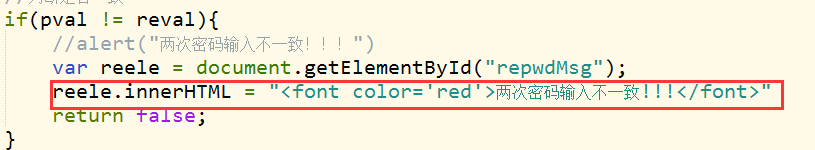


## 1.3、案例代码实现









# 2、案例2：表格隔行换色

## 2.1、需求说明及分析

开发中，需要使用表格陈列数据，数据过多不易查看，通常使用隔行换色进行表示。



分析：

1. 选中偶数行 (其他的选择器)
2. 返回值 js中的数组

## 2.2、JavaScript知识讲解（二）

### 2.2.1、JS数组

回顾：

Java中数组和集合的区别：

1. 数组的长度是固定的，集合的长度是可变的
2. 数组可以存储基本数据类型和引用数据类型，集合只可以存储引用数据类型
3. 数组只能存储同一种数据类型(Object)，集合可以存储任意数据类型

JS中的数据相当于java中的集合，长度是可变的，并且可以存储任意类型

数组创建方式：

|  |  |
| --- | --- |
| 方式 | 方式说明 |
| **Var arr= [**'a','b','c',true,10**];** | 常用的JS数组创建方式。 数组长度为5  数组的默认值是：undefined |
| **var arr = new Array();** | 常用的JS数组创建方式。创建一个数组对象，数组长度默认为0 |
| var arr = new Array(4); | 数组长度是4，相当于开辟了度长为4的空间，每个元素是undefined。 |
| var arr = new Array(“a”,”b”,true); | 创建了一个数组，长度3, 数组元素是”a”,”b”,true |
|  |  |

数组中的方法：



### 2.2.2、获取元素对象的四种方式

通常情况下，我们使用document，即DOM对象，就可以获取并操作页面元素标签，进而改变页面效果。

Doucument.getElementById(id)

准备代码

<span id="s1">span1111</span>

<hr />

<input type="radio" name="sex" value="man"/>男

<input type="radio" name="sex" value="woman"/>女

<hr />

<div>div1111</div>

<div>div2222</div>

<div>div3333</div>

<hr />

<font class="redF">红色字体1</font>

<font class="redF">红色字体2</font>

<font class="redF">红色字体3</font>

<br />

<input type="button" value="点击展示span内容体" onclick="run1()" />

<input type="button" value="点击展示两个单选框的value" onclick="run2()" />

<input type="button" value="点击展示三个div的内容体" onclick="run3()" />

<input type="button" value="点击展示三个font的内容体" onclick="run4()" />

CSS中的三个选择器：

Id 、 类 、元素选择器

document对象方法-DOM

|  |  |
| --- | --- |
| 方法名 | 方法说明 |
| getElementById(id值) | 通过元素的ID值 获取到某个元素对象  返回值是单个对象 |
| getElementsByName(name值) | 通过元素的name值 获取到所有符合要求的元素对象  返回值是数组 |
| getElementsByTagName(标签名) | 通过元素的标签名 获取到所有符合要求的元素对象  返回值是数组 |
| getElementsByClassName(class值) | 通过元素的class值 获取到所有符合要求的元素对象  返回值是数组 |



## 2.3、案例代码实现



# 3、JavaScript知识讲解(三)

## 3.1、DOM对象

### 3.1.1、DOM对象是什么？有什么作用？

是什么：DOM（Document Object Model） 文档对象模型 document

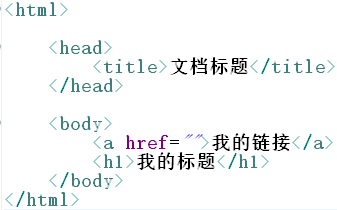
作用：DOM是将标记型文档(HTML)中所有内容（标签、文本、属性）都封装成对象，用于操作网页中的元素

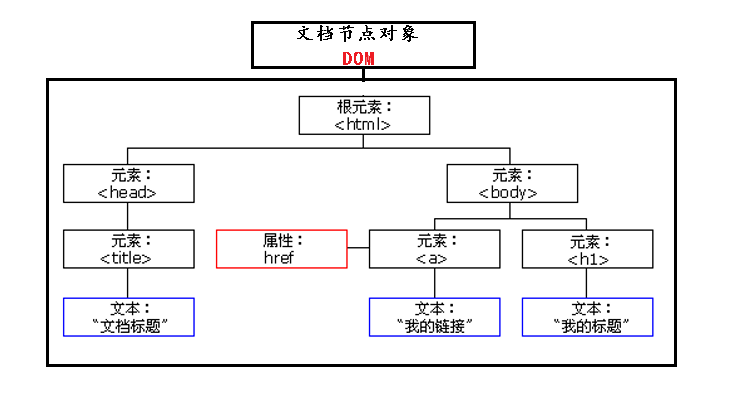
每个HTML页面在被浏览器解析的时候都会在内存中创建一棵DOM树，我们通过编写JS代码就可以访问这棵树上任

何一个节点，并且对节点进行操作。通过 DOM模型，可以访问所有的 HTML 元素，连同它们所包含的文本和属

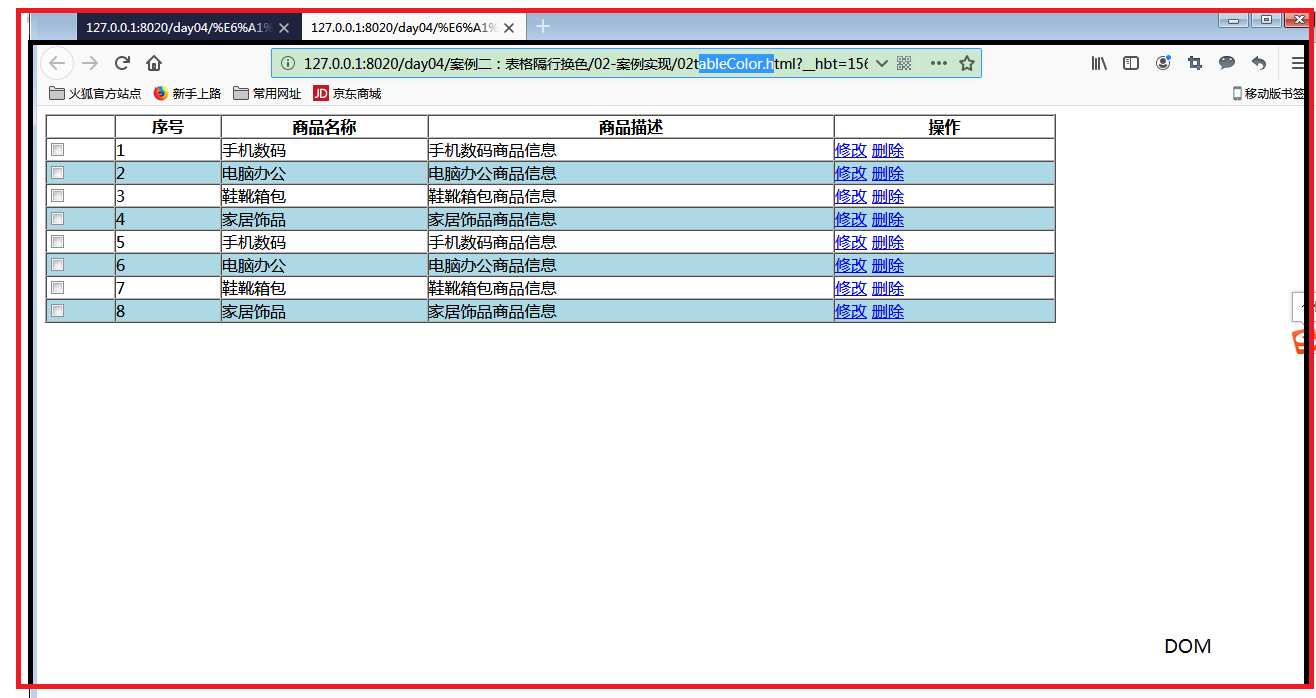
性。可以对其中的内容进行修改和删除，同时也可以创建新的元素。新创建的元素对象，要挂到DOM树上才可以

在网页上显示出来。





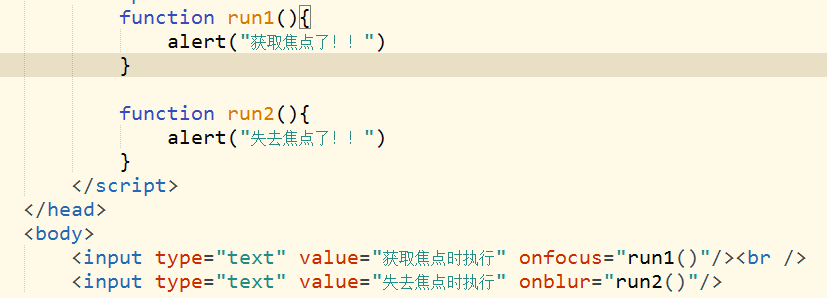
### 3.1.2、DOM和BOM关系



## 3.2、JS事件总结

* 常见事件

|  |  |
| --- | --- |
| **事件名** | **描述** |
| **onload** | 页面加载完成时执行 |
| **onsubmit** | 表单提交触发 |
| **onclick** | 点击事件 |
| ondblclick | 双击事件 |
| **onblur** | 失去焦点事件  前提：必须先获取焦点 |
| **onfocus** | 获取焦点事件 |
| **onchange** | 值发生改变事件（案例4中讲解） |



鼠标键盘事件：

|  |  |
| --- | --- |
| **事件名** | **描述** |
| **onkeydown** | 某个键盘的键被按下 |
| onkeypress | 某个键盘的键被按下或按住 |
| **onkeyup** | 某个键盘的键被松开 |
| onmousedown | 某个鼠标按键被按下 |
| onmouseup | 某个鼠标按键被松开 |
| **onmouseover** | 鼠标被移到某元素之上  鼠标移入 |
| **onmouseout** | 鼠标从某元素移开  前提：必须先移入  鼠标移出 |

示例代码：

<script>

function run1(){

alert("键位被按下了");

}

function run2(){

alert("键位被弹起了");

}

function run3(){

alert("鼠标移入了");

}

function run4(){

alert("鼠标移出了");

}

</script>

<input value="某个键位被按下时会弹出窗口" onkeydown="run1()" />

<hr />

<input value="某个键位被弹起时会弹出窗口" onkeyup="run2()" />

<hr />

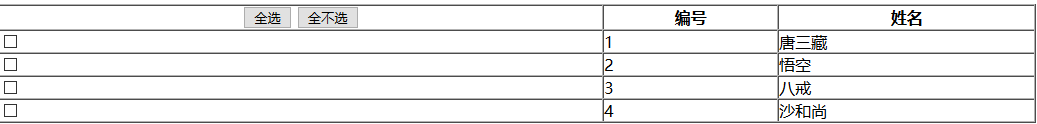
<input value="鼠标移入时会弹出窗口" onmouseover="run3()" />

<hr />

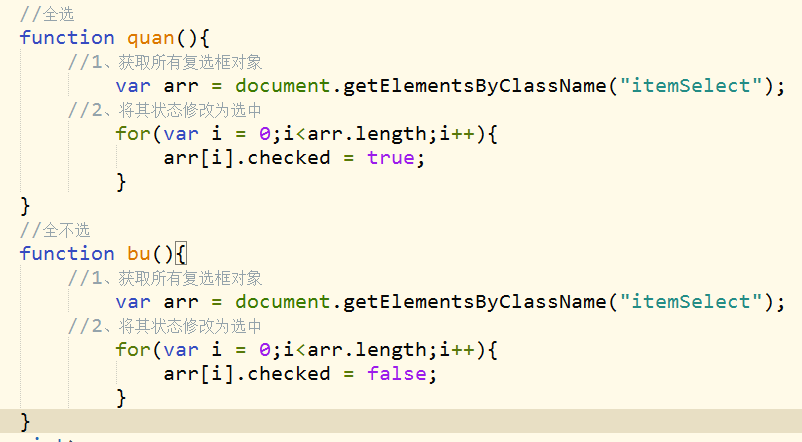
<input value="鼠标移出时会弹出窗口" onmouseout="run4()" />

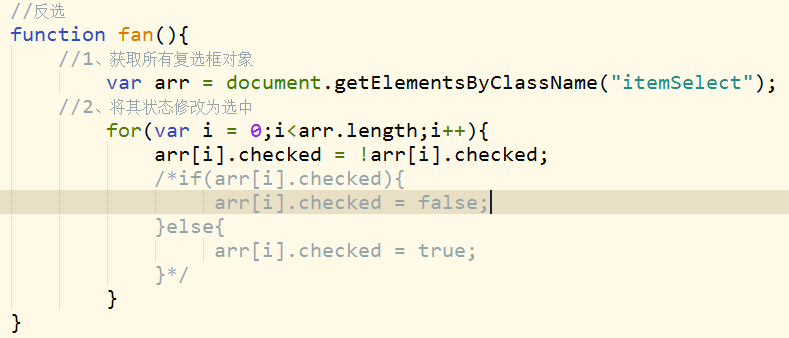
# 4、案例3：全选-全不选

## 4.1、需求说明及分析



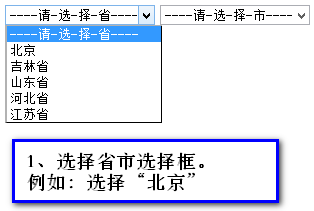
## 4.2、案例代码实现

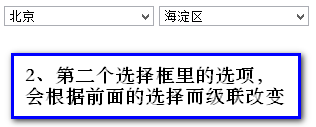


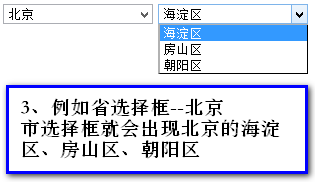


# 5、案例4：省市二级联动

## 5.1、需求说明及分析







## 5.2、JavaScript知识讲解(四)

Onchange事件：值发生改变时触发

This:当前被选中的元素对象

This.value 当前被选中的元素对象的value值

## 5.3、案例代码实现



# 6、jquery入门

## 6.1、jQuery简述



核心理念：Write Less,Do More（写更少，做更多）

Jquery是js的框架，底层封装了js代码

压缩版(.min)和非压缩版的区别

功能上没有任何区别，区别是压缩版将回车、换行、注释...内容删除掉了

学习使用：非压缩版

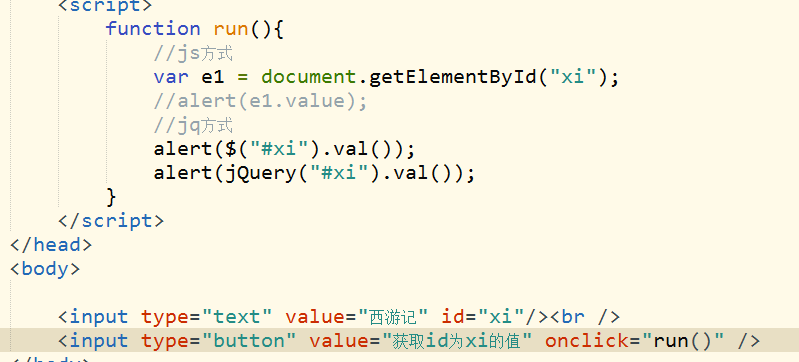
开发使用：压缩版

## 6.2、jQuery的引入及使用

jQuery版本有很多，不同版本的jQuery之间方法和功能有一定改变。

* 引入



* 基本语法
* jQuery(选择器);
* **$(选择器);（建议）**
* 

如果jquery没有引入成功，则控制台报以下异常：



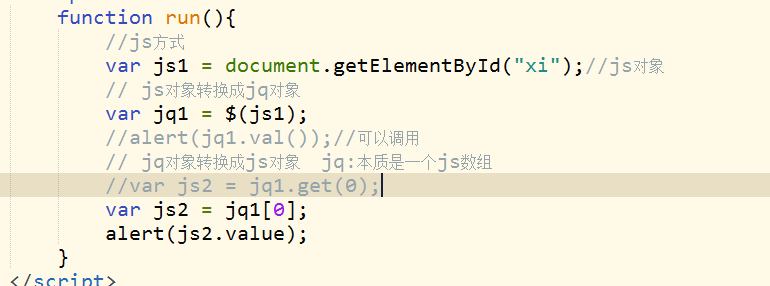
## 6.3、jQuery对象和JS对象的互转

Jquery对象不可以调用js中的方法和属性

同样的js对象也不能调用jquery中的属性和方法

JS对象转jq：var jq = $(js对象);

JQ对象转js： var js = jq[0];或者jq.get(0);

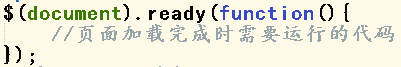


## 6.4、jQuery页面加载完成时执行代码

jQuery对onload进行了包装，变为了更为方便的代码：

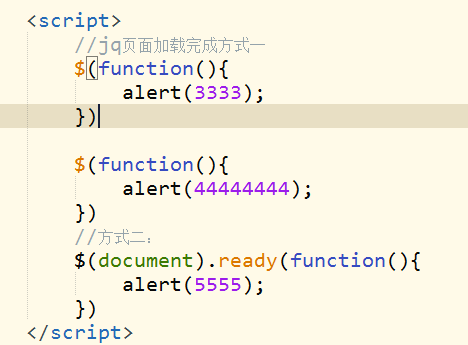
不会覆盖

* 格式1：



* **格式2：**





## 6.5、jQuery基本选择器

和CSS中基本选择器一样

ID选择器：

类选择器：

元素选择器：

