

# day35-jquery3高级

## 学习目标：

1. 能够使用jQuery对象的遍历方法
2. 能够使用jQuery全局的遍历方法
3. 能够使用jQuery3.0的for-of遍历方法
4. 能够使用jQuery的DOM操作的方法
5. 能够理解jQuery的插件机制及其相关API
6. 能够使用jQuery的表单校验插件进行表单校验

## 案例一使用jQuery完成省市联动效果

### 一,需求分析

籍贯:

- 请选择-
- 深圳
- 广州
- 东莞
- 惠州

### 二,技术分析

#### 1.遍历

##### 1.2 jquery对象方法遍历

- 语法

```
jquery对象.each(function(index,element){});
```

其中:

index:就是元素在集合中的索引

element :就是集合中的每一个元素对象

```

<body>
</body>
<script>
    var array=[1,2,3,4,5,6];
    $(array).each(function (a,b) {
        document.write(b);
    })
</script>

```

### 1.3 jquery的全局方法遍历

- 语法

```
$.each(jquery对象,function(index,element){});
```

其中，

index:就是元素在集合中的索引

element：就是集合中的每一个元素对象

```

<body>
</body>
<script>
    var array=[1,2,3,4,5,6];
    $.each($(array),function (a,b) {
        document.write(b);
    })
</script>

```

### 1.4 jQuery3.0新特性for of语句遍历

- 语法

```

for(变量 of jquery对象){
    变量;
}

```

其中，

变量:定义变量依次接受jquery数组中的每一个元素

jquery对象：要被遍历的jquery对象

```

<body>
</body>
<script>
    var array=[1,2,3,4,5,6];
    for(n of $(array)){
        document.write(n);
    }
</script>

```

## 1.5遍历二维数组

```
<script>
var array=[["武汉","荆州","仙桃"],["桂林","阳朔","世外桃源"],["北海","百色","贵阳"]];
$(array).each(function (a,b) {
    $(b).each(function (c,d) {
        document.write(d);
    });
})
</script>
```

## 2.使用JQ操作DOM

### 2.1jQuery对DOM树中的文本和值进行操作

- API

API方法	解释
val()	获得标签里面value属性的值 value属性的封装
val([value])	给标签value属性设置值
html()	获得标签的内容，如果有标签，一并获得。
html([value])	设置html代码，如果有标签，将进行解析。支持标签
text()	获得标签的内容，如果有标签，忽略标签。
text([value])	设置文本，如果含有标签，把标签当做字符串.不支持标签

- val()和val(...)

```
<body>
<input type="text" value="hello" id="vid"/>
<input type="submit" value="获取" id="btn1"/>
<input type="submit" value="设置" id="btn2"/>
</body>
<script>
    $("#btn1").click(function () {
        document.write($("#vid").val());
    });
    $("#btn2").click(function () {
        $("#vid").val("嘿嘿");
    });
</script>
```

- text()和html()

<body>

```

<p id="pId"><font>hello...</font></p>
<input type="button" value="text()" id="btn1" />
<input type="button" value="html()" id="btn2" />
</body>

<script>
    $("#btn1").click(function(){
        var data= $("#pId").text();
        console.log("data="+data);//结果:hello...
    });

    $("#btn2").click(function(){
        var data= $("#pId").html();
        console.log("data="+data);结果:<font>hello...</font>
    });

</script>

```

- text(...)和html(...)

```

<p id="pId"></p><br />
<input type="button" id="btn1" value="html(...)" />
<input type="button" id="btn2" value="text(...)" />

<script>
    $("#btn1").click(function(){
        $("#pId").html("<font color='red'>hello...</font>");
    });

    $("#btn2").click(function(){
        $("#pId").text("<font color='red'>hello...</font>");
    });

</script>

```

## 2.2jQuery创建,插入对象

- API

API方法	解释
\$("#A")	创建A元素对象
append(element)	a.append(c), 将c插入到a的内部的后面; 添加最小的孩子
prepend(element)	a.prepend(c),将c插入到a的内部的前面; 添加最大的孩子
appendTo(element)	a.appendTo(c), 将a插入到c的内部的后面; 最小的孩子认干爹
prependTo(element)	a.prependTo(c),将a插入到c的内部的前面 ; 最大的孩子认干爹
before(element)	a.before(c);弟弟找哥哥
after(element)	a.after(c); 哥哥找弟弟

```
<body>
<p id="pid"><font>八戒</font></p>
<input type="button" value="append(...)" id="bt"/>
<input type="button" value="prepend(...)" id="bt1"/>
</body>
<script>
$("#bt").click(function () {
    $("#<font color='orange'>沙悟净</font>").appendTo("#pid");
});
$("#bt1").click(function () {
    $("#<font color='yellow'>孙大圣</font>").prependTo("#pid");
})
</script>
```

### 2.3jQuery移除节点(对象)

- API

API方法	解释
remove()	删除指定元素(自己移除自己)
empty()	清空指定元素的所有子元素

## 三,思路分析

## 四,代码实现

- js代码

```
<body>
籍贯:
```

```

<select id="pId">
  <option value="-1">-请选择-</option>
  <option value="0">广东</option>
  <option value="1">湖北</option>
  <option value="2">山东</option>
</select>
<select id="cId">
  <option>-请选择-</option>
</select>
</body>
<script>
  var array = new Array(3);
  array[0] = ["深圳", "广州", "东莞", "惠州"];
  array[1] = ["武汉", "黄冈", "黄石", "鄂州"];
  array[2] = ["济南", "青岛", "烟台"];
  $("#pId").change(function () {
    var index=this.value;
    var addcity=$("#cId");
    if(index>-1){
      addcity.html("<option>-请选择-</option>");
      var citys=array[index];
      $(citys).each(function (i,city) {
        addcity.append("<option>-"+city+"-</option>")
      })
    }
  });
</script>

```

## 案例二QQ表情案例

### 一,需求分析



请发言: 🤔 🤔 🤔 🤔 🤔

### 二, 思路分析

- 给每一个表情图片设置点击事件
- 把点击的每一个表情img添加到下面的发言区域(段落)

### 三,代码实现

```

<script src="../../js/jquery-3.3.1.js"></script>
<script>
    $(function () {
        $(".emoji li img").click(function () {
            //把当前的小表情克隆，再追加到p元素中
            $(this).clone().appendTo(".word");
        });
    });
</script>

```

## 案例三使用jQuery完成表单的校验

### 一,需求分析

在学习JavaScript时，我们手动的完成过表单数据的校验，此功能在开发中非常常见，属于通用功能范畴，实际开发一般使用都是第三方工具。本案例我们将使用jQuery插件validation进行表单的校验。

#### 员工信息录入

真实姓名(不能为空,没有其他要求)	<input type="text"/> 真实姓名必填
登录名(登录名不能为空,长度应该在5-8之间,可以包含中文字符(一个汉字算一个字符)):	<input type="text"/> 登录名必填
密码(不能为空,长度6-12位):	<input type="password"/> 密码不能为空
重复密码密码(不能为空,两次密码需要一致):	<input type="password"/> 两次密码不一致
性别(必选其一)	<input type="radio"/> 男 <input type="radio"/> 女 性别必填
年龄(必填18-60):	<input type="text" value="1000"/> 年龄超出范围
你的学历:	<input type="text" value="- 请选择你的学历 -"/>
出生日期(日期格式):	<input type="text"/>
兴趣爱好:	<input type="checkbox"/> 乒乓球 <input type="checkbox"/> 羽毛球 <input type="checkbox"/> 上网 <input type="checkbox"/> 旅游 <input type="checkbox"/> 购物
电子邮箱:	<input type="text"/>
身份证:	<input type="text"/>
	<input type="button" value="保存"/>

## 二,技术分析

### 1.JSON

#### 1.1概念

json是一种容易生成和解析的数据的交换格式

#### 1.2格式

- json对象{},1.以key:value的形式存在.

2. key是字符串类型(标准JSON),value可以是任何合法数据类型.

3.多个字段(key)之间通过,隔开,最后一个,不用写

```
{ "name": "张三", "age": 18 }
```

- 数组 []

```
[ { "name": "张三", "age": 18 }, { "name": "李四", "age": 20 } ]
```

- 对象和数组嵌套

```
{ "p1": { "name": "张三", "age": 18 }, "p23": [ { "name": "李四", "age": 19 }, { "name": "王五", "age": 20 } ] };
```

```
<script>
    var p = { "name": "zs", "age": 18 };
    alert(p.name);
    alert(p.age);
    var n = [ { "name": "zs", "age": 18 }, { "name": "lisi", "age": 19 } ]
    alert(n[0].name);
    alert(n[0].age);
    alert(n[1].name);
    alert(n[1].age);
    var zu = { "p1": { "name": "ww", "age": 20 }, "p23": [ { "name": "zs", "age": 18 }, { "name": "ls", "age": 19 } ] }
    alert(zu.p23[1].name);
</script>
```

## 2.jQuery插件

### 2.1什么是jq插件

插件就是以jquery为基础进行扩展具有特定功能的代码，就是jquery插件。例如：图片轮播、选项卡等。

### 2.2jQuery的插件机制

jQuery插件的机制说白了就是插件实现原理. 其实就是利用jQuery提供的. *fn.extend()*和.extend()方法，扩展jQuery的功能。

### 2.3 jQuery插件机制语法

语法	解释
\$.fn.extend(object)	对jQuery对象进行方法扩展
\$.extend(object)	对jQuery全局进行方法扩展

### 2.4自定义jq插件

- 自定义插件check.js 使用上面方法,参数object使用json语句



```
$.fn.extend({
  checked:function () {
    this.prop("checked",true);
  }
});
$.extend({max:function (a,b) {
  return a>b?a:b;
}}
);
```

- 对jQuery进行扩展

```
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
  <script src="../../js/jquery-3.3.1.min.js"></script>
  <script src="../../js/check.js"></script>
</head>
<body>
<input type="button" value="全选" id="bton"/><br/>
<input type="button" value="全不选" id="btun"/><br/>
<input type="checkbox"/>奈落<input type="checkbox"/>犬夜叉
<input type="checkbox"/>阿离
</body>
<script>
  $("#bton").click(function () {
    $("input[type='checkbox']").checked();
  });
  alert($.max(5,6));
</script>
</html>
```

## 3.表单校验插件validation

### 3.1validation概述

网络上有许多成熟的插件供使用者参考，插件就是将jquery实现的常用操作进行封装，封装后的“小框架”就可以叫做插件，按照插件的语法去操作可以实现很多复杂的功能，而我们需要做的是学会该插件的使用语法即可。表单校验插件是按照一定的语法编写代码，可以使用简单的代码完成复杂的表单校验工作

### 3.2validation下载

### 3.3validation使用步骤

- 将该插件（也就是一个js文件）导入到我们的工程中
- 在要使用校验插件的html中引入该js文件

```
<!--依赖的jQuery库-->
<script src="../../js/jquery-1.8.3.js" type="text/javascript" charset="utf-8" />
<!--validation校验库-->
<script src="../../js/jquery.validate.js" type="text/javascript" charset="utf-8" />
```

- 编写表单校验的代码（语法）

```
<script type="text/javascript">
    $(function() {
        $("#formId").validate(
            { //以key:value的形式 }
        );
    });
</script>
```

## 2.4 检验方式：js 代码方式

语法：

```
$(...).validate({
    rules:{},          // 规则
    messages:{}         //提示信息
});
```

rules 规则语法：

```
rules:{
    字段名(标签的name的属性值):校验器,
    字段名(标签的name的属性值):校验器
}
```

校验器语法：

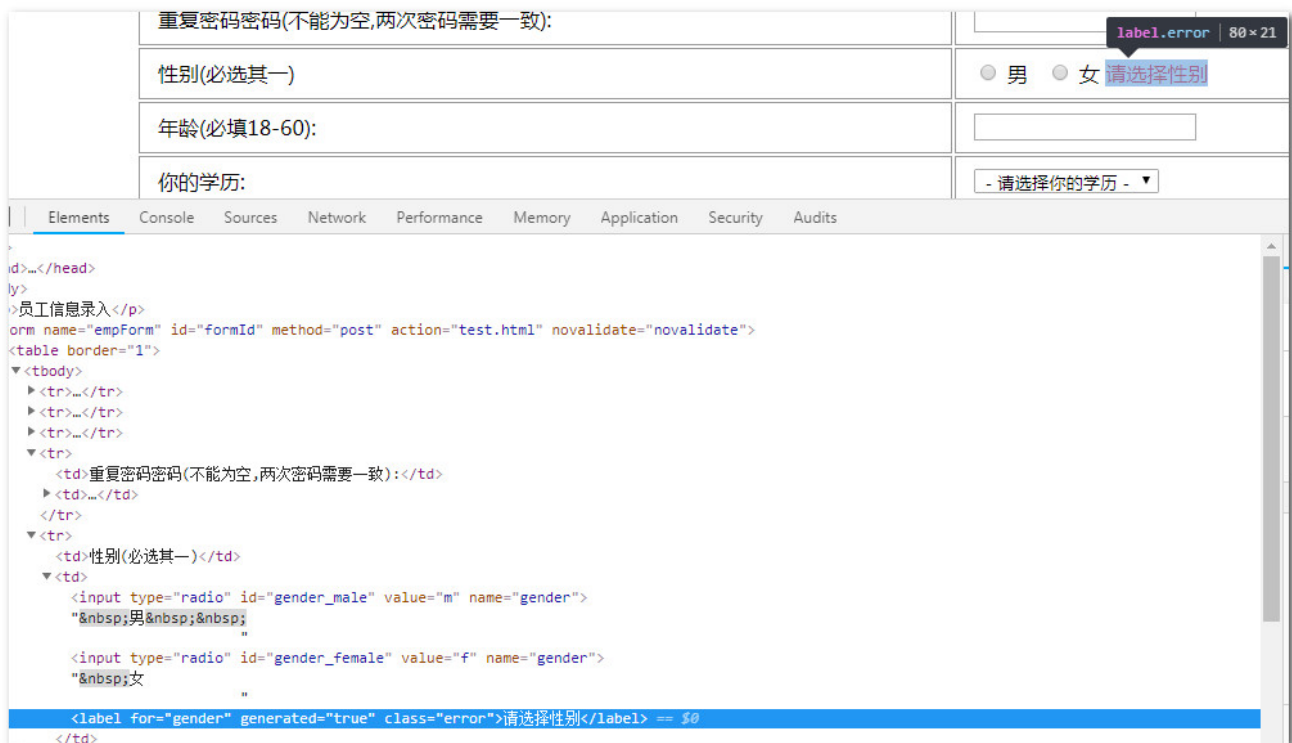
```
语法：{校验器:值,校验器:值,...}
```

message 提示语法：

```
messages:{
    字段名:{校验器:"提示",校验器:"提示",...}
}
```

序号	规则	描述
1	required:true	必须输入的字段。
2	remote:"check.php"	使用 ajax 方法调用 check.php 验证输入值。
3	email:true	必须输入正确格式的电子邮件。
4	url:true	必须输入正确格式的网址。
5	date:true	必须输入正确格式的日期。日期校验 ie6 出错，慎用。
6	dateISO:true	必须输入正确格式的日期（ISO），例如：2009-06-23，1998/01/22。只验证格式，不验证有效性。
7	number:true	必须输入合法的数字（负数，小数）。
8	digits:true	必须输入整数。
9	creditcard:	必须输入合法的信用卡号。
10	equalTo:"#id名"	输入值必须和 #id对应的标签对象的值 相同。
11	accept:	输入拥有合法后缀名的字符串（上传文件的后缀）。
12	maxlength:5	输入长度最多是 5 的字符串（汉字算一个字符）。
13	minlength:10	输入长度最小是 10 的字符串（汉字算一个字符）。
14	rangelength:[5,10]	输入长度必须介于 5 和 10 之间的字符串（汉字算一个字符）。
15	range:[5,10]	输入值必须介于 5 和 10 之间。
16	max:5	输入值不能大于 5。
17	min:10	输入值不能小于 10。

### 三,思路分析



## 四,代码实现

- html

```
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title>我的jquery表单校验页面</title>
  <style type="text/css">
    p{text-align: center;font-size:24px;}
    table{margin: 0 auto;border: 0;}
    table tr{height:40px;border:0;}
    table tr td{padding:0 14px;border:1px solid #999}
    .error{color:red}
  </style>
  <script src="../js/jquery-3.3.1.min.js"></script>
  <script src="../js/jquery.validate.min.js"></script>
</head>
<body>
  <p>员工信息录入</p>
  <form name="empForm" id="formId" method="post" action="test.html">
    <table border=1>
      <tr>
        <td>真实姓名(不能为空 ,没有其他要求)</td>
        <td><input type="text" id="realname" name="realname" />
        </td>
      </tr>
      <tr>
        <td>登录名(登录名不能为空,长度应该在5-8之间,可以包含中文字符(一个汉字算一个字符)):
      </td>
        <td><input type="text" id="username" name="username" /></td>
      </tr>
```



```
        range:[18,60]
    }
},
messages:{
    realname:{required:"用户名不能为空"},
    username:{required:"登录名错误",rangelength:"用户名长度错误"},
    psw:{required:"密码未填写",rangelength:"密码长度错误"},
    psw2:{
        required:"密码未填写",equalTo:"密码不一致"
    },
    gender:{
        required:"请选择性别",
    },
    age:{
        required:"请输入年龄",
        range:"年龄不符"
    }
}
});
</script>
```