

原生的Ajax技术

在**JavaScript**的方法里使用这个技术（在不重新加载整个页面的情况下）来完成异步的与服务器交换数据并更新部分网页的功能。

**XMLHttpRequest**：是javascript技术

➤ 方法：

- \* **open**(请求方式, 请求路径, 是否异步) : 异步去向服务器发送请求. //创建连接
- \* **send**(请求参数) : 发送请求.

➤ 属性：

\* **readyState**: 表示XMLHttpRequest的状态

\* **status** : 获得响应的状态码

获取数据的属性：

\* **responseText** : 获得响应的文本数据.

/\* **responseXML** : 获得响应的XML的数据. document对象中

➤ 事件：

\* **onreadystatechange** : 当XMLHttpRequest状态改变的时候触发一个函数

1、创建XMLHttpRequest对象

```
var xmlhttp;  
if (window.XMLHttpRequest) {  
    xmlhttp = new XMLHttpRequest();  
} else {  
    xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");  
}
```

2、创建连接

```
xmlhttp.open("get","/day17_ajax/ckNameServlet?username="+username);
```

3、发送请求

```
xmlhttp.send();
```

4、通过事务处理结果

```
xmlhttp.onreadystatechange = function(){  
    //判断请求是否一切正常  
    if(xmlhttp.readyState==4){  
        //服务器响应一切正常  
        if(xmlhttp.status==200){  
            //获得响应正文  
            document.getElementById("s1").innerHTML = xmlhttp.responseText;  
        }  
    }  
}
```

# JQuery的AJAX

JQuery的AJAX部分方法:

\* \$.get(路径,请求参数,回调函数,数据类型);

\* \$.post(路径,参数,回调函数,数据类型); 数据类型: 默认是字符串

```
<script type="text/javascript">
    //页面加载事件
    $(function(){
        //得到用户名文本框, 并注册事件
        $("#username").blur(function(){
            var username = $(this).val();
            //执行ajax操作
            /*
            参数1 url: 访问路径
            参数2 data:请求参数
            参数3 function: 回调函数
            参数4 : 数据的返回类型
            */
            var url = "/Ajax/onblurs";
            var data = "username="+username;
            $.post(url,data,function(result){
                //处理结果
                //var b = eval(result);
                if(result){
                    $("#s1").html("用户名已存在").css("color","red");
                }else{
                    $("#s1").html("用户名可以使用").css("color","lime");
                }
            }, "json");
        });
    });
</script>
```

```

    });
});
</script>

```

## 案例1核心代码：

```

1 <input type="text" id="username" name="username" /><span id="s1"></span><br/>
  <input type="button" onclick="ckName()" value="提交"/>

<!-- 以下是 ajax 的方法 -->
<script type="text/javascript">
    function ckName(){
        //得到用户名
        var username = document.getElementById("username").value;
        //1、创建XMLHttpRequest对象
        var xmlhttp;
        if (window.XMLHttpRequest) {
            xmlhttp = new XMLHttpRequest();
2        } else {
            xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
        }
        //2、创建连接
        xmlhttp.open("get", "/Ajax/onclicks?username="+username);
        //3、发送请求
        xmlhttp.send();

        //4、通过事务处理结果
        xmlhttp.onreadystatechange = function(){
            //判断请求是否一切正常 服务器响应一切正常
            if(xmlhttp.readyState==4 && xmlhttp.status==200){
                // document.getElementById("s1").innerHTML = xmlhttp.responseText;
                // }
            }
        }
        /* 服务器响应回来的是Boolean类型 而web获取的响应数据是文本类型
           eval(xmlhttp.responseText)方法可以将字符转为表达式 */
        var b = eval(xmlhttp.responseText);
        if(b){
            document.getElementById("s1").innerHTML = "<font style='color:red;'>用户名已存在</font>";
        }else{
            document.getElementById("s1").innerHTML = "<font style='color:lime;'>用户名可以使用</font>";
        }
    }
}
</script>

//处理请求数据
String username = request.getParameter("username");
username = new String(username.getBytes("iso-8859-1"), "utf-8");
//调用业务
UserService us = new UserService();
代 Boolean conditionSelect = us.ConditionSelect(username);
码 //响应给WEB
response.setContentType("text/html; charset=UTF-8"); //响应给客户端的数据类型
response.setCharacterEncoding("utf-8"); //设置响应给客户端的数据 字符编码
PrintWriter out = response.getWriter(); //字符流 向客户端响应方式的另外一种写法
out.println(conditionSelect); //向客户端响应方式的另外一种写法

}

```

# jQuery的Ajax技术

```
<!-- 以下是 jQuery 选择器 的方法 -->
<script type="text/javascript">
    //页面加载事件
    $(function(){
        //得到用户名文本框，并注册事件
        $("#username").blur(function(){
            var username = $(this).val();
            /*      执行ajax操作
            参数1 url: 访问路径          参数2 data:请求参数
            参数3 function: 回调函数      参数4: 数据的返回类型    */
            var url = "/Ajax/onblurs";
            var data = "username="+username;
            $.post(url,data,function(result){
                //处理结果 var b = eval(result);
                if(result){
                    $("#s1").html("用户名已存在").css("color","red");
                }else{
                    $("#s1").html("用户名可以使用").css("color","lime");
                }
            },"json");
        });
    });
</script>
```