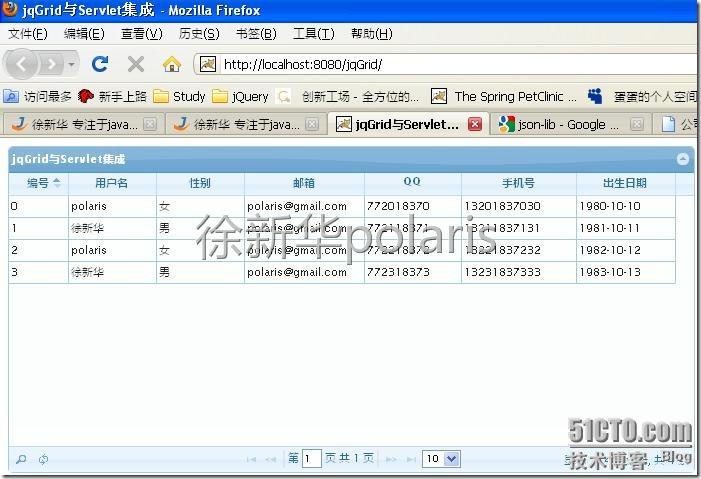
jQuery学习之：jqGrid表格插件——从servlet获得数据

在我的jqGrid第一个demo中，没有涉及到任何与服务端的交互：数据是从自己定义的一个固定数组中得到的。当然，实际应用中不可能会这么 做。在 jqGrid官方网站上提供的demo，服务器端全部是PHP，在网上找了很久，也没有找到写的完整详细的从JSP/Servlet获得数据的例子。下面 是我写的一个从Servlet获得数据的例子，希望对初学者有点帮助。   
jqGrid支持的好几种数据类型，现在使用比较多的是JSON。本文中也采用这种数据类型。另外，本文使用了json-lib，因而您也需要下载json-lib.jar，可以到<http://json-lib.sourceforge.net/>上下载，且该网站上有关于json-lib的详细介绍与使用。感兴趣的可以阅读。

下面开始我们的jqGrid与Servlet之旅吧。

1、效果图：

[](http://img1.51cto.com/attachment/201001/12/1146394_1263258477MHke.jpg)

这个跟第一个Demo看起来没有什么区别。的确没啥区别。只是这次的数据是从服务器端获取的。

2、代码及解释

HTML代码：

<body>  
         <table id="gridTable"></table>  
         <div id="gridPager"></div>  
</body>

html代码跟之前一样，很简单。

JavaScript代码：

$(function()  
{  
        $("#gridTable").jqGrid({  
                url:'jqGridServlet',  
                datatype: "json",  
                height: 250,  
                colNames:['编号','用户名', '性别', '邮箱', 'QQ','手机号','出生日期'],  
                colModel:[  
                        {name:'id',index:'id', sorttype:"int"},  
                        {name:'userName',index:'userName',  
                        {name:'gender',index:'gender',  
                        {name:'email',index:'email', ;string"},  
                        {name:'QQ',index:'QQ', ;              
                        {name:'mobilePhone',index:'mobilePhone', ;              
                        {name:'birthday',index:'birthday', sorttype:"date"}  
                ],  
                sortname:'id',  
                sortorder:'asc',  
                viewrecords:true,  
                rowNum:10,  
                rowList:[10,20,30],  
                jsonReader:{  
                        repeatitems : false  
                },  
                pager:"#gridPager",  
                caption: "jqGrid与Servlet集成"  
        }).navGrid('#gridPager',{edit:false,add:false,del:false});  
})

JavaScript的代码跟之前有几个地方不一样。其中，url:'jqGridServlet'的jqGridServlet是 Servlet的 url-pattern；datatype: "json"表示服务器端需要返回json数据；jsonReader:{repeatitems : false}则是为了后台处理数据方便而设置，具体意思见下面Java代码的注释。

Java代码：

package com.polaris.jqgrid.servlet;  
  
import java.io.IOException;  
  
import javax.servlet.ServletException;  
import javax.servlet.http.HttpServlet;  
import javax.servlet.http.HttpServletRequest;  
import javax.servlet.http.HttpServletResponse;  
  
import net.sf.json.JSONArray;  
import net.sf.json.JSONObject;  
/\*\*  
\* 该Servlet向客户端返回一个json对象。为了简便，数据不是从数据库获得的。  
\* jqGrid默认期望返回的json对象格式要求如下：  
\* {"page":"1","total":"2","records":"13",  
\* "rows":[  
\*                 {id:"1",cell:["1","polaris","男","polaris@gmail.com","772618379","18329382732","1985-10-2"]},  
\*                 {id:"2",cell:["2","张三","女","zhangsan@163.com","272618382","15329382732","1986-10-12"]},  
\*                 {id:"3",cell:["3","王五","女","wangwu@yahoo.com","172635372","13329389832","1987-12-21"]},  
\*                 {id:"4",cell:["4","赵六","男","zhaoliu@sina.com","372618332","18929343731","1988-09-22"]}  
\*         ]  
\* }  
\* 当然，在js中，可以通过jqGrid的jsonReader属性来修改默认格式  
\* 因为默认的格式，rows的数据要求顺序不能变，且每个字段都得有值（空也得有"")。因而，  
\* 在jsonReader中定义repeatitems : false。这样，rows就变成了：  
\* "rows":[  
\*                 {id:"1",userName:"polaris",gender:" 男",email:"polaris@gmail.com",QQ:"772618379",mobilePhone:"18329382732",birthday:"1985-10-2"]},  
\*                 {id:"2",userName:"徐新华",gender:" 男",email:"xh.xu@163.com",QQ:"272618382",mobilePhone:"15329382732",birthday:"1986-10-12"]},  
\*                 {id:"3",userName:"王五",gender:" 女",email:"wangwu@yahoo.com",QQ:"172635372",mobilePhone:"13329389832",birthday:"1987-12-21"]},  
\*                 {id:"4",userName:"赵六",gender:" 女",email:"zhaoliu@sina.com",QQ:"372618332",mobilePhone:"18929343731",birthday:"1988-09-22"]}  
\*         ]  
\* @author xuxinhua  
\*  
\*/  
public class JqGridForJSONServlet extends HttpServlet  
{  
        private static final long serialVersionUID = 132383828833L;  
  
        @Override  
        public void doGet(HttpServletRequest req, HttpServletResponse resp)  
                        throws ServletException, IOException {  
                // 定义返回的数据类型：json，使用了json-lib  
                JSONObject jsonObj = new JSONObject();  
                // 根据jqGrid对JSON的数据格式要求给jsonObj赋值  
                jsonObj.put("page", 1);                // 当前页  
                jsonObj.put("total", 1);        // 总页数  
                jsonObj.put("records", 4);        // 总记录数  
                // 定义rows，存放数据  
                JSONArray rows = new JSONArray();  
                // 放入4条数据  
                for(int i=0;i<4;i++)  
                {  
                        // 存放一条记录的对象  
                        JSONObject cell = new JSONObject();  
                        cell.put("id", i);  
                        if(i%2==0)  
                        {  
                                cell.put("userName", "polaris");  
                                cell.put("gender", "女");  
                        }  
                        else  
                        {  
                                cell.put("userName", "徐新华");  
                                cell.put("gender", "男");  
                        }  
                        cell.put("email", "polaris@gmail.com");  
                        cell.put("QQ", "772"+i+"1837"+i);  
                        cell.put("mobilePhone", "132"+i+"1837"+i+"3"+i);  
                        cell.put("birthday", "198"+i+"-10-"+"1"+i);  
                        // 将该记录放入rows中  
                        rows.add(cell);  
                }  
                // 将rows放入json对象中  
                jsonObj.put("rows", rows);  
                // 自控制台打印输出，以检验json对象生成是否正确  
                System.out.println("要返回的json对象：\n" + jsonObj.toString());  
                // 设置字符编码  
                resp.setCharacterEncoding("UTF-8");  
                // 返回json对象（通过PrintWriter输出）  
                resp.getWriter().print(jsonObj);  
        }  
        @Override  
        public void doPost(HttpServletRequest req, HttpServletResponse resp)  
                        throws ServletException, IOException {  
                doGet(req,resp);  
        }  
}

请注意看该类的详细注释。

web.xml

<servlet>  
            <servlet-name>jqGridServletForJson</servlet-name>  
            <servlet-class>com.polaris.jqgrid.servlet.JqGridForJSONServlet</servlet-class>  
</servlet>  
<servlet-mapping>  
            <servlet-name>jqGridServletForJson</servlet-name>  
            <url-pattern>/jqGridServlet</url-pattern>  
</servlet-mapping>

web.xml中Servlet的配置

3、部署运行

导入必须的Jar包（记得导入json-lib依赖的jar包）。将工程打包部署到tomcat下，运行tomcat，在浏览器中访问写的jsp 页面,如：http://localhost:8080/jqGrid/jqGrid\_servlet.jsp，回车会出现效果图的效果。

总结：

可以看出，jqGrid集成Servlet还是很简单的，就jqGrid而言，它只需要请求一个Servlet，并设置期望返回的数据类型即可。而对于 Servlet而言，这需要根据jqGrid期望返回的数据类型而定，通常都会选择json数据格式，因而，Servlet主要就是构造json数据，我 们可以根据json的语法要求构建，然而，那样比较繁琐，会有很多的“{}"之类的，且不清晰，容易出错。而json-lib则是一个不错的选择，因为这 时操作的只是Java对象，构建json很容易，结构清晰不容易出错。因而推荐采用。