**培训要求**

**1）掌握jQuery常用AJAX-API**

**2）掌握Java调用MySQL / Oracle过程与函数**

**一）jQuery常用AJAX-API**

目的：简化客户端与服务端进行局部刷新的异步通讯

（1）取得服务端当前时间

简单形式：jQuery对象.load(url)

返回结果自动添加到jQuery对象代表的标签中间

如果是Servlet的话，采用的是GET方式

复杂形式：jQuery对象.load(url,sendData,function(backData,textStatus,ajax){... ...})

sendData = {"user.name":"jack","user.pass":"123"};

以JSON格式文本方式发送，使用POST方式发送，服务端能收到数据

注意：对于load方法而言，如果请求体无参数发送的话，load方法

采用GET方式提交

注意：对于load方法而言，如果请求体有参数发送的话，load方法

采用POST方式提交

注意：使用load方法时，自动进行编码，无需手工编码

|  |
| --- |
| <script type="text/javascript">  $(":button").click(function(){  var url = "${pageContext.request.contextPath}/loadTimeRequest?time"+new Date().getTime();  var sendData = null;  $.load(url,sendData,function(a,b,ajax){  var jsonJAVA = ajax.responseText;  var jsonJS = eval("("+jsonJAVA+")");  var strTime = jsonJS.strTime;  $("span:first").html(strTime).css("color","red");  $("span:last").html(strTime).css("color","blue");  });  //$("span").load(url);  });  </script> |

load()方法参数解释：

参数一：url发送到哪里去

参数二： sendData发送请求体中的数据，符合JSON格式，例如：{key:value,key:value}

参数三：function处理函数，类似于传统方式ajax.onreadystatechange = 处理函数

其中参数三为function处理函数最多可以接收三个参数，含义如下

第一个参数：服务端返回的数据，例如：backData

第二个参数：服务端状态码的文本描述，例如：success、error、

第三个参数：ajax异步对象，即XMLHttpRequest对象

以上所有参数的名字可以任意，但必须按顺序书写，尽量做到见名知意

（2）检查注册用户名和密码是否存在

$.get(url,sendData,function(backData,textStatus,ajax){... ...})

$.post(url,sendData,function(backData,textStatus,ajax){... ...})赵君提倡

注意：使用get或post方法时，自动进行编码，无需手工编码

（3）jQuery对象.serialize()

作用：自动生成JSON格式的文本

注意：为每个jQuery对象设置一个name属性，因为name属性会被认为请求参数名

注意：必须用<form>标签元素

适用：如果属性过多，强烈推荐采用这个API

|  |
| --- |
| <script type="text/javascript">  //定位按钮，同时添加单击事件  $(":button").click(function(){  //获取用户名和密码  var username = $(":text:first").val();  var password = $(":text:last").val();  //去空格  username = $.trim(username);  password = $.trim(password);  //异步发送到服务端  var url = "${pageContext.request.contextPath}/checkRequest?time="+new Date().getTime();  /\*手工书写JSON文本  var sendData = {  "user.username":username,  "user.password":password  };  \*/  /\*工具生成JSON文本\*/  var sendData = $("form").serialize();  $.post(url,sendData,function(backData,textStatus,ajax){  //backData是一个JSON文本/对象，你得通过.号访问其属性值  var tip = backData.tip;  //创建img节点  var $img = $("<img src='" + tip + "' height='18px' widht='18px'/>")  //清空span节点中的内容  $("span").text("");  //将img节点添加到span节点中  $("span").append( $img );  });  });  </script> |

|  |
| --- |
| **public** **class** RegisterAction **extends** ActionSupport{  **private** User user;  **public** User getUser() {  **return** user;  }  **public** **void** setUser(User user) {  **this**.user = user;  }  /\*\*  \* 检查注册用户名和密码是否存在  \*/  **public** String checkMethod() **throws** Exception {  tip = "images/MsgSent.gif";  **if**("帅帅".equals(user.getUsername()) && "123".equals(user.getPassword())){  tip = "images/MsgError.gif";  }  **return** "ok";  }  **private** String tip;  **public** String getTip() {  **return** tip;  }  /\*\*  \* var backData = {  \* "tip":"images/MsgError.gif"  \* }  \*/  } |

（4）jQuery解析XML

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <root>  <city>广州</city>  <city>中山</city>  <city>深圳</city>  <city>佛山</city>  <city>珠海</city>  </root> |

|  |
| --- |
| <script type="text/javascript">  $(":button").click(function(){  var url = "${pageContext.request.contextPath}/03\_city.xml";  var sendData = null;  $.get(url,sendData,function(backData,textStatus,ajax){  /\*  //将xml文件转成jquery对象，目的是用jquery api解析xml文件  var $xml = $(backData);  var $city = $xml.find("city");  $city.each(function(){  //获取每一个city节点  var city = $(this).text();  alert(city);  });  \*/  var xml = ajax.responseXML;  var $xml = $(xml);  var $city = $xml.find("city");  $city.each(function(){  //获取每一个city节点  var city = $(this).text();  alert(city);  });  });  });  </script> |

（5）省份-城市，基于jQuery的AJAX二级联动，用Struts2整合AJAX【非数据库版】

$.ajax(

{

type:"POST",

url:"${pageContext.request.contextPath}/findCityByProvinceRequest?id="+new Date().getTime(),

data:{"province":province},

success:function(backData,textStatus,ajax){}

}

);

注意：这里每个key值不能乱写

|  |
| --- |
| <select id="province">  <option>选择省份</option>  <option>广东</option>  <option>湖南</option>  </select>  <select id="city">  <option>选择城市</option>  </select> |

|  |
| --- |
| <script type="text/javascript">  $("#province").change(function(){  //删除原城市下拉框中的内容，除第一项外  $("#city option:gt(0)").remove();  var province = $("#province option:selected").text();  if("选择省份" != province){  $.ajax({  "type":"POST",  "url":"${pageContext.request.contextPath}/findCityByProvince?time="+new Date().getTime(),  "sendData":{"province":province},  "success":function(backData,textStatus,ajax){  //js对象  var city = backData.cityList;  //jquery对象  var $city = $(city);  //each()  $city.each(function(){  //this表示每个城市  var $option = $("<option>" + this + "</option>");  $("#city").append( $option );  });  }  });  }  });  </script> |

|  |
| --- |
| **public** **class** ProvinceCityAction **extends** ActionSupport{  **private** String province;//广东  **public** **void** setProvince(String province) {  **this**.province = province;  }  **public** String findCityByProvince() **throws** Exception {  cityList = **new** ArrayList<String>();  **if**("广东".equals(province)){  cityList.add("湛江");  cityList.add("汕头");  }**else** **if**("湖南".equals(province)){  cityList.add("邵阳");  cityList.add("浏阳");  }  **return** "ok";  }  **private** List<String> cityList;  **public** List<String> getCityList() {  **return** cityList;  }  /\*\*  \* var backData = {  \* "cityList":["邵阳","浏阳"]  \* }  \*/  } |

|  |
| --- |
| <struts>  <package name="timePackage" extends="json-default" namespace="/">    <global-results>  <result name="ok" type="json"/>  </global-results>  <!-- 获取服务端时间 -->  <action  name="loadTimeRequest"  class="cn.itcast.javaee.js.time.TimeAction"  method="loadTimeMethod"/>  <!-- 检查用户名和密码是否存在 -->  <action  name="checkRequest"  class="cn.itcast.javaee.js.register.RegisterAction"  method="checkMethod"/>    <!-- 根据省份查询城市 -->  <action  name="findCityByProvince"  class="cn.itcast.javaee.js.provincecity.ProvinceCityAction"  method="findCityByProvince"/>    </package>  </struts> |

**二）掌握Java调用Oracle过程与函数**

写一个计算个人所得税的应用

|  |
| --- |
| --定义过程  create or replace procedure get\_rax(salary in number,rax out number)  as  --需要交税的钱  bal number;  begin  bal := salary - 3500;  if bal<=1500 then  rax := bal \* 0.03 - 0;  elsif bal<=4500 then  rax := bal \* 0.1 - 105;  elsif bal<=9000 then  rax := bal \* 0.2 - 555;  elsif bal<=35000 then  rax := bal \* 0.25 - 1005;  elsif bal<=55000 then  rax := bal \* 0.3 - 2755;  elsif bal<=80000 then  rax := bal \* 0.35 - 5505;  else  rax := bal \* 0.45 - 13505;  end if;  end;  /  --调用过程  declare  --交税  rax number;  salary number := &salary;  begin  get\_rax(salary,rax);  dbms\_output.put\_line(salary||'元工资需要交'||rax||'元税');  end;  / |

|  |
| --- |
| **public** **class** TestCallOracleProc {  **public** **static** **void** main(String[] args) **throws** Exception{  String sql = "{call get\_rax(?,?)}";  Connection conn = JdbcUtil.*getConnection*();  CallableStatement cstmt = conn.prepareCall(sql);  cstmt.setInt(1,10000);  cstmt.registerOutParameter(2,Types.*INTEGER*);  cstmt.execute();  Integer rax = cstmt.getInt(2);  System.*out*.println("10000元需要交" + rax + "元税");  JdbcUtil.*close*(cstmt);  JdbcUtil.*close*(conn);  }  } |

查询7788号员工的的姓名，职位，月薪

|  |
| --- |
| --定义函数  create or replace function findEmpNameAndJobAndSal(pempno in number,pjob out varchar2,psal out number)  return varchar2  as  pename emp.ename%type;  begin  select ename,job,sal into pename,pjob,psal from emp where empno = pempno;  return pename;  end;  /  --调用函数  declare  pename emp.ename%type;  pjob emp.job%type;  psal emp.sal%type;  begin  pename := findEmpNameAndJobAndSal(7788,pjob,psal);  dbms\_output.put\_line('7788'||'--'||pename||'--'||pjob||'--'||psal);  end;  / |

|  |
| --- |
| **public** **class** TestCallOracleFunc {  **public** **static** **void** main(String[] args) **throws** Exception{  String sql = "{?=call findEmpNameAndJobAndSal(?,?,?)}";  Connection conn = JdbcUtil.*getConnection*();  CallableStatement cstmt = conn.prepareCall(sql);  cstmt.registerOutParameter(1,Types.*VARCHAR*);  cstmt.setInt(2,7788);  cstmt.registerOutParameter(3,Types.*VARCHAR*);  cstmt.registerOutParameter(4,Types.*INTEGER*);  cstmt.execute();  String ename = cstmt.getString(1);  String job = cstmt.getString(3);  Integer sal = cstmt.getInt(4);  System.*out*.println(ename+":"+job+":"+sal);  JdbcUtil.*close*(cstmt);  JdbcUtil.*close*(conn);  }  } |

作业：

1. 基于jQuery的AJAX应用----三级联动【数据库版或是非数据库版不限制】

省/市/区

广东/广州/天河，越秀

广东/深圳/罗湖，福田

湖南/长沙/雨花，天心

湖南/株洲/荷塘，天元

湖北/武汉/江夏，洪山

湖北/宜昌/西陵，夷陵