# 尚德书院制作过程

目录

[一．菜单栏的制作](#_Toc17477_WPSOffice_Level1) [2](#_Toc17477_WPSOffice_Level1)

[一:搭建数据库](#_Toc20028_WPSOffice_Level2) [2](#_Toc20028_WPSOffice_Level2)

[1. 建立数据库:](#_Toc20028_WPSOffice_Level3) [2](#_Toc20028_WPSOffice_Level3)

[2. 添加数据](#_Toc179_WPSOffice_Level3) [2](#_Toc179_WPSOffice_Level3)

[二：创建工程：](#_Toc179_WPSOffice_Level2) [2](#_Toc179_WPSOffice_Level2)

[1. 创建shangde web工程：](#_Toc23273_WPSOffice_Level3) [2](#_Toc23273_WPSOffice_Level3)

[2. 导入lib包](#_Toc8420_WPSOffice_Level3) [2](#_Toc8420_WPSOffice_Level3)

[3. 搭建ssh支持：](#_Toc5325_WPSOffice_Level3) [2](#_Toc5325_WPSOffice_Level3)

[三：工程实施：](#_Toc23273_WPSOffice_Level2) [4](#_Toc23273_WPSOffice_Level2)

[1. 连接数据库的时候还有一个关键一步，加上这句话，不然跑不起来，](#_Toc23671_WPSOffice_Level3) [4](#_Toc23671_WPSOffice_Level3)

[2. 对数据库的数据方向工程](#_Toc31863_WPSOffice_Level3) [5](#_Toc31863_WPSOffice_Level3)

[3. 导入jsp页面需要的页面布局配置：](#_Toc21490_WPSOffice_Level3) [5](#_Toc21490_WPSOffice_Level3)

[4. 在src下面新建一个放接口的包web.dao](#_Toc4544_WPSOffice_Level3) [5](#_Toc4544_WPSOffice_Level3)

[5. 在src下新建一个是实现接口的包web.dao.imp](#_Toc2953_WPSOffice_Level3) [5](#_Toc2953_WPSOffice_Level3)

[6. 建立页面index.jsp](#_Toc17606_WPSOffice_Level3) [6](#_Toc17606_WPSOffice_Level3)

[二：地址管理的实现](#_Toc20028_WPSOffice_Level1) [13](#_Toc20028_WPSOffice_Level1)

[1. 实现点击地址管理，跳到地址页面](#_Toc8420_WPSOffice_Level2) [13](#_Toc8420_WPSOffice_Level2)

[① 数据库里已经定义了地址管理对应的跳转页面的标识项](#_Toc9395_WPSOffice_Level3) [13](#_Toc9395_WPSOffice_Level3)

[② struts.xml增加这项标识：](#_Toc9135_WPSOffice_Level3) [13](#_Toc9135_WPSOffice_Level3)

[③ 在WebAction里面增加方法 addAddress](#_Toc17763_WPSOffice_Level3) [13](#_Toc17763_WPSOffice_Level3)

[④ 新增跳转页面addresslist.jsp](#_Toc21369_WPSOffice_Level3) [13](#_Toc21369_WPSOffice_Level3)

[2. 数据库的建立：（address）](#_Toc5325_WPSOffice_Level2) [13](#_Toc5325_WPSOffice_Level2)

[3. 给数据库添加数据](#_Toc23671_WPSOffice_Level2) [14](#_Toc23671_WPSOffice_Level2)

[4. 对address表反向工程](#_Toc31863_WPSOffice_Level2) [14](#_Toc31863_WPSOffice_Level2)

[5. 新建实现接口 AddressDao.java](#_Toc21490_WPSOffice_Level2) [14](#_Toc21490_WPSOffice_Level2)

[6. 新建实现AddressDao.java接口的类](#_Toc4544_WPSOffice_Level2) [14](#_Toc4544_WPSOffice_Level2)

[7. 新建地址页面addresslist.jsp](#_Toc2953_WPSOffice_Level2) [15](#_Toc2953_WPSOffice_Level2)

[8. 解释addresslist里面的数据是怎么拿到的：](#_Toc17606_WPSOffice_Level2) [18](#_Toc17606_WPSOffice_Level2)

[9. 给这些值赋值（在WebAction里面）](#_Toc9395_WPSOffice_Level2) [18](#_Toc9395_WPSOffice_Level2)

[10. 效果](#_Toc9135_WPSOffice_Level2) [19](#_Toc9135_WPSOffice_Level2)

[三． 添加地址：](#_Toc179_WPSOffice_Level1) [20](#_Toc179_WPSOffice_Level1)

[2. 给接口AddressDao添加 添加地址的方法](#_Toc17763_WPSOffice_Level2) [20](#_Toc17763_WPSOffice_Level2)

[3. 在AddressDaoImp类里实现这方法](#_Toc21369_WPSOffice_Level2) [20](#_Toc21369_WPSOffice_Level2)

[4. 添加地址后面对应的连接：addAddress.jsp](#_Toc8722_WPSOffice_Level2) [20](#_Toc8722_WPSOffice_Level2)

[5. 添加addAddress.jsp页面](#_Toc5223_WPSOffice_Level2) [21](#_Toc5223_WPSOffice_Level2)

[6. 这个表单对应的action是：](#_Toc2294_WPSOffice_Level2) [22](#_Toc2294_WPSOffice_Level2)

[7. 在WebAction.java里面添加 addAddress方法](#_Toc22225_WPSOffice_Level2) [22](#_Toc22225_WPSOffice_Level2)

[① 先声明网页里用到的变量：](#_Toc8722_WPSOffice_Level3) [22](#_Toc8722_WPSOffice_Level3)

[② 添加addAddress方法](#_Toc5223_WPSOffice_Level3) [22](#_Toc5223_WPSOffice_Level3)

[四：修改界面](#_Toc23273_WPSOffice_Level1) [23](#_Toc23273_WPSOffice_Level1)

[1. 添加修改按钮的响应事件](#_Toc29_WPSOffice_Level2) [23](#_Toc29_WPSOffice_Level2)

[2. 在WebAction.java类里添加变量 addressId](#_Toc19285_WPSOffice_Level2) [23](#_Toc19285_WPSOffice_Level2)

[private int addressId;// 地址ID 然后生成get set方法](#_Toc2294_WPSOffice_Level3) [23](#_Toc2294_WPSOffice_Level3)

[3. 在structs.xml添加oneAddress.action](#_Toc21833_WPSOffice_Level2) [23](#_Toc21833_WPSOffice_Level2)

[4. 在AddressDaoImp实现 查找指定Address的方法](#_Toc29123_WPSOffice_Level2) [24](#_Toc29123_WPSOffice_Level2)

[5. 在WebAction.java添加oneAddress方法](#_Toc13760_WPSOffice_Level2) [24](#_Toc13760_WPSOffice_Level2)

[1. 先定义Address addre;//地址信息类 并生成 get set方法](#_Toc1083_WPSOffice_Level2) [24](#_Toc1083_WPSOffice_Level2)

[6. 添加updateAddress.jsp页面](#_Toc21860_WPSOffice_Level2) [24](#_Toc21860_WPSOffice_Level2)

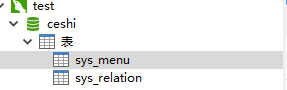
[7. 在添加struts.xml添加 这个action](#_Toc18510_WPSOffice_Level2) [26](#_Toc18510_WPSOffice_Level2)

[8. 在WebAction.java添加updateAddress方法](#_Toc10195_WPSOffice_Level2) [26](#_Toc10195_WPSOffice_Level2)

# 一．菜单栏的制作

一:搭建数据库

1. 建立数据库:



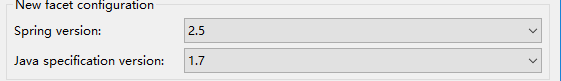
1. 添加数据

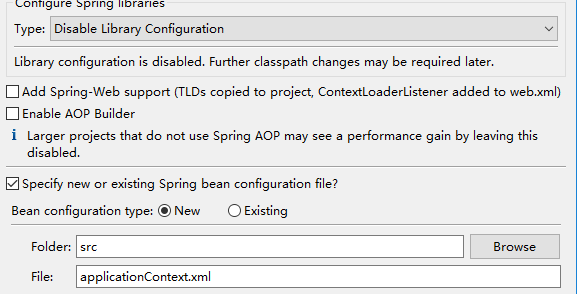
二：创建工程：

1. 创建shangde web工程：

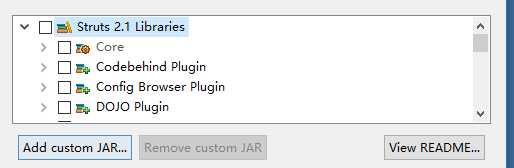


1. 导入lib包
2. 搭建ssh支持：
3. 添加spring支持



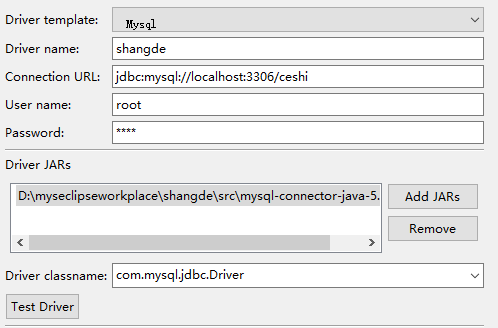


1. 添加structs支持

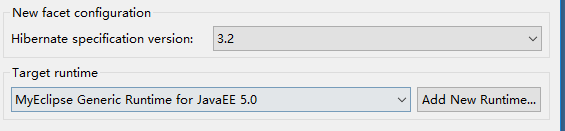


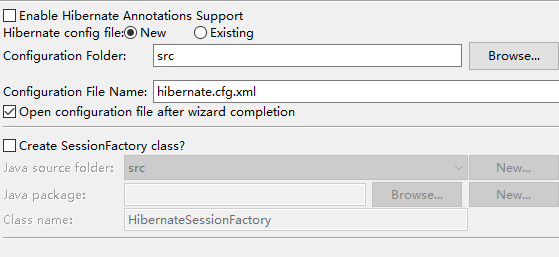
勾勾都去掉

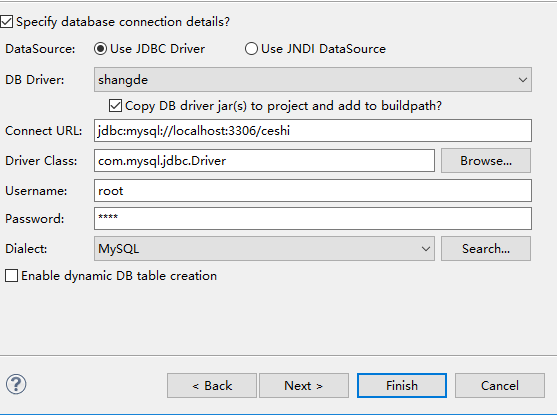
1. 添加数据库

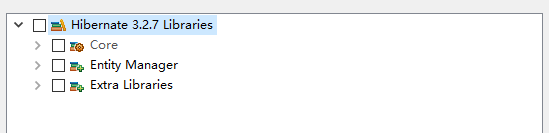


1. 添加Hibernate支持







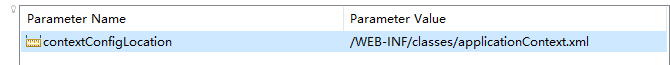


1. 建立spring struts联系
2. 在src下新建一个文件及并添加这项数据：struts.objectFactory=spring



1. 在web.xml配置

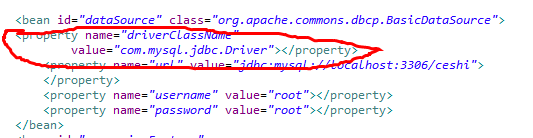
添加：org.springframework.web.context.ContextLoaderListener

添加：

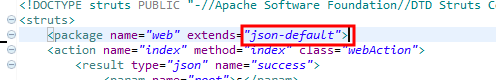
1. 工程建立完成

三：工程实施：

1. 连接数据库的时候还有一个关键一步，加上这句话，不然跑不起来，

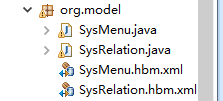


还有struts.xml也有修改

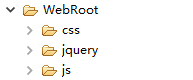


写成json-default

1. 对数据库的数据方向工程



1. 导入jsp页面需要的页面布局配置：



1. 在src下面新建一个放接口的包web.dao

|  |
| --- |
| public interface RelationDao{  List<String> findPmenu(int id);  List<SysMenu0> findRelation(int id)l;  }  这个接口有两个方法：   1. 用来查找所有的主菜单名字   List<String> findPmenu(int id)   1. 用来查找所有的主菜单所有信息   List1<SysMenu> findRelation(int id); |

1. 在src下新建一个是实现接口的包web.dao.imp

|  |
| --- |
| Public class RelationDaoImp extends ActionSupport implements RelationDao{  Public List<String> findPmenu(int id){  List<object []> list=this.getHibernateTemolate().find(“from Sysmenu m.SysRelation r where m.id=r.mid and r.tid=?”+”order by m.fsign”,id);  List<String> listm=new Arraylist<string>();  If(list!=null&&list.size()>0){  For(int i=0;i<list.size();i++){  Listm.add(((System)list.get(i)[0]).getName());  }  }  return listm;  }  Public List<SysMenu> findRelation(int id){  List<Object[]>list=this.getHibernateTemplate().find(“from SysMenu m,SysRelation r where m.id=r.mid and r.tid=? order by m.sign ”,id);  List<SysMenu> listm=new Arraylist<>();  If(list!=null&&list.size()>0){  For(int i=0;i<list.size();i++){  Listm.add((System)list.get(i)[0]);  }  }  Return listm;  }  }  } |

1. 建立页面index.jsp

|  |  |  |
| --- | --- | --- |
| 1. 先导入外部支持文件 css js jsquary   <link rel=*"stylesheet"* href=*"css/bootstrap.min.css"* />  <link rel=*"stylesheet"* href=*"css/bootstrap-responsive.min.css"* />  <link rel=*"stylesheet"* href=*"css/matrix-style.css"* />  <link rel=*"stylesheet"* href=*"css/matrix-media.css"* />  <link href=*"font-awesome/css/font-awesome.css"* rel=*"stylesheet"* />  <script src=*"js/jquery.min.js"*></script>  <script src=*"http://apps.bdimg.com/libs/html5shiv/3.7/html5shiv.min.js"*></script>  <script src=*"http://apps.bdimg.com/libs/respond.js/1.4.2/respond.min.js"*></script>  <script src=*"js/excanvas.min.js"*></script>  <script src=*"js/jquery.ui.custom.js"*></script>   1. 页面初始化方法：   <script type=*"text/javascript"*>  $(document).ready(**function**() {  //index.action  **var** url = "index.action?id=" +  <%=id%>;  login(url);    });  **function** login(url) {  $.ajax({  url : url,  type : "Get",  async : **false**,  dataType : "json",  success : **function**(daa) {  **for** (**var** i = 0; i < daa.dataa.length; i++) {  **var** str = "<li class='submenu'><a herf='#'><span>"  + daa.dataa[i].namee + "</span></a><ul>"    **for** (**var** j = 0; j < daa.dataa[i].listt.length; j++) {  str += '<li><a class="menu\_a" link="'+daa.dataa[i].listt[j].url+'"> '  + daa.dataa[i].listt[j].functions  + '</a></li>'  }  str += "</ul></li>"    document.getElementById("indexxx").innerHTML += str;  }  }  });  }  </script>   1. 这里有个index.action 所以要在struts.xml添加它的相关事件，   <struts>  <package name=*"web"* extends=*"json-default"*>  <action name=*"index"* method=*"index"* class=*"webAction"*>  <result type=*"json"* name=*"success"*>  <param name=*"root"*>s</param>  </result>  </action>  </package>  </struts>   1. 给webAction添加依赖注入（在applicationContext.xml）   <bean id=*"webAction"* class=*"web.action.WebAction"*>  <property name=*"relationDao"* ref=*"relationDaoImp"*></property>  </bean>   1. WebAction里面定义了relationDao,所以也要添加它的依赖注入   <bean id=*"relationDaoImp"* class=*"web.dao.imp.RelationDaoImp"*>  <property name=*"sessionFactory"* ref=*"mysessionFactory"*></property>  </bean>   1. 在WebAction类里面添加实现方法 index  |  | | --- | | 1. RelationDao relationDao; 2. **private** **int** id;//从网页上获取数据，所以名字要和index.jsp上定义得id一样 3. **private** JSONObject s;//页面上的ajas需要一个JSONObject对象传来的东西   生成get set方法   1. 实现index方法   **public** String index() **throws** Exception {  List<String> listp = relationDao.findPmenu(id);//存放的都是父菜单名  List<SysMenu> list = relationDao.findRelation(id);//获取一整个父类的信息  List<Map<String, Object>> lista = **new** ArrayList<Map<String, Object>>();    **if** (list != **null** && list.size() > 0) {  **for** (String str : listp) {  Map<String, Object> map = **new** HashMap<String, Object>();  List<Map<String, Object>> listf = **new** ArrayList<Map<String, Object>>();  map.put("namee", str);//这里存的是父名  **for** (SysMenu me : list) {  //menu = relationDao.findMenu(re.getMid());  **if** (str.equals(me.getName())) {//这里的作用是取出父类名下的小类  Map<String, Object> mapr = **new** HashMap<String, Object>();  mapr.put("functions", me.getFunction());  mapr.put("url", me.getUrl());  listf.add(mapr);  }  }  map.put("listt", listf);//这个"list"里面包括了父类里的小项所有信息  lista.add(map);  }  }  s = JSONTools.*createJsonObject*("dataa", lista);//这个dataa是ajax里面获取数据的标志  **return** "success";  } | | 1. 完整的的index.jsp 页面   <%@ page contentType=*"text/html; charset=UTF-8"*%>  <%@ taglib uri=*"/struts-tags"* prefix=*"s"*%>  <%  Integer id = 18;  %>  <%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>  <%@ page isELIgnored=*"false"*%>  <%@ taglib uri=*"http://java.sun.com/jsp/jstl/fmt"* prefix=*"fmt"*%>  <!DOCTYPE html>  <html lang=*"zh"*>  <head>  <title>“尚德书院”管理系统平台</title>  <meta charset=*"UTF-8"* />  <meta name=*"viewport"* content=*"width=device-width, initial-scale=1.0"* />  <meta http-equiv=*"X-UA-Compatible"* content=*"IE=edge,Chrome=1"* />  <meta http-equiv=*"X-UA-Compatible"* content=*"IE=9"* />  <link rel=*"stylesheet"* href=*"css/bootstrap.min.css"* />  <link rel=*"stylesheet"* href=*"css/bootstrap-responsive.min.css"* />  <link rel=*"stylesheet"* href=*"css/matrix-style.css"* />  <link rel=*"stylesheet"* href=*"css/matrix-media.css"* />  <link href=*"font-awesome/css/font-awesome.css"* rel=*"stylesheet"* />  <script src=*"js/jquery.min.js"*></script>  <script src=*"http://apps.bdimg.com/libs/html5shiv/3.7/html5shiv.min.js"*></script>  <script src=*"http://apps.bdimg.com/libs/respond.js/1.4.2/respond.min.js"*></script>  <script src=*"js/excanvas.min.js"*></script>  <script src=*"js/jquery.ui.custom.js"*></script>  <script type=*"text/javascript"*>  $(document).ready(**function**() {  //index.action  **var** url = "index.action?id=" +  <%=id%>;  login(url);  });  **function** login(url) {  $.ajax({  url : url,  type : "Get",  async : **false**,  dataType : "json",  success : **function**(daa) {    **for** (**var** i = 0; i < daa.dataa.length; i++) {  **var** str = "<li class='submenu'><a herf='#'><span>"  + daa.dataa[i].namee + "</span></a><ul>"    **for** (**var** j = 0; j < daa.dataa[i].listt.length; j++) {  str += '<li><a class="menu\_a" link="'+daa.dataa[i].listt[j].url+'"> '  + daa.dataa[i].listt[j].functions  + '</a></li>'  }  str += "</ul></li>"    document.getElementById("indexxx").innerHTML += str;  }    }  });  }  </script>  </head>  <body >  <div id=*"header"*>  <h1>  <a>管理平台</a>  </h1>  </div>  <!--top-Header-menu-->  <div id=*"user-nav"* class=*"navbar navbar-inverse"*>  <ul class=*"nav"*>  <li class=*"dropdown"* id=*"profile-messages"*><a title=*""* href=*"#"*  data-toggle=*"dropdown"* data-target=*"#profile-messages"*  class=*"dropdown-toggle"*> <i class=*"icon icon-user"*></i>&nbsp; <span  class=*"text"*>欢迎你  </span>&nbsp; <b class=*"caret"*></b>  </a>  <ul class=*"dropdown-menu"*>  <li><a class=*"menu\_a"* link=*"allActivity.action"*><i  class=*"icon-check"*></i> 我的发布</a></li>  <li class=*"divider"*></li>  <li><a class=*"menu\_a"* onclick="unLogin(this)"><i  class=*"icon-key"*></i> 退出系统</a></li>  </ul></li>  </ul>  </div>  <!--close-top-Header-menu-->  <div id=*"search"*></div>  <div id=*"sidebar"* style="OVERFLOW-Y: *auto*; OVERFLOW-X:*hidden*;">  <ul id=*"indexxx"*>    <li class=*"submenu"*><a href=*"#"*> <i class=*"icon icon-lock"*></i>  <span>权限管理</span>  </a>  <ul>  <li><a class=*"menu\_a"* link=*"relationList.jsp"*><i  class=*"icon icon-caret-right"*></i>权限列表</a></li>  </ul></li>  </ul>  </div>  <div id=*"content"*>  <div id=*"content-header"*>  <div id=*"breadcrumb"*>  <a href=*"newIndex.jsp"* title=*"返回主页"* class=*"tip-bottom"*><i  class=*"icon-home"*></i> Home</a>  </div>  </div>  <iframe src=*"allActivity.action?"* id=*"iframe-main"* style="width:*100%*;"></iframe>  </div>  <!--end-main-container-part-->  <script src=*"js/bootstrap.min.js"*></script>  <script src=*"js/matrix.js"*></script>  <script type=*"text/javascript"*>  //初始化相关元素高度  **function** init() {  $("body").height($(window).height() - 80);  $("#iframe-main").height($(window).height() - 90);  $("#sidebar").height($(window).height());  }  /\*\*/  $(**function**() {  init();  $(window).resize(**function**() {  init();  });  });  **function** goPage(newURL) {  **if** (newURL != "") {  **if** (newURL == "-") {  } **else** {  document.location.href = newURL;  }  }  }  **function** resetMenu() {  document.gomenu.selector.s // if url is empty, skip the menu dividers and reset the menu selection to default  electedIndex = 2;  }  </script>  </body>  </html> |   按照以上方法，就可以出现菜单栏 |

# 二：地址管理的实现

1. 实现点击地址管理，跳到地址页面
2. 数据库里已经定义了地址管理对应的跳转页面的标识项



1. struts.xml增加这项标识：

<action name=*"allAddress"* class=*"webAction"* method=*"allAddress"*>

<result name=*"success"*>/addresslist.jsp</result>

</action>

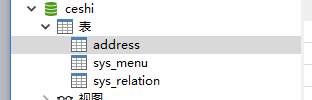
1. 在WebAction里面增加方法 addAddress

|  |
| --- |
| **public** String allAddress() {  **return** "success";  } |

1. 新增跳转页面addresslist.jsp



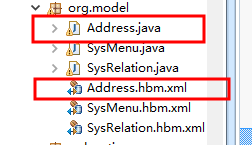
1. 数据库的建立：（address）



1. 给数据库添加数据

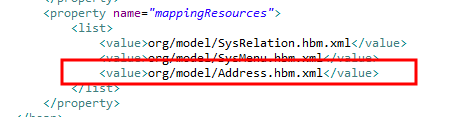


1. 对address表反向工程



因为这是中途添加的表，所以要在applicationContext.xml里面看看有没有他的映射文件

我第一次中途添加表的时候，就是因为applicationContext.xml里面没有自动添加他的映射，而导致我的工程运行不起来

（确保有映射）

1. 新建实现接口 AddressDao.java

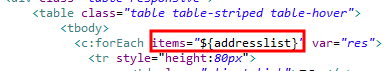
|  |
| --- |
| **public** **interface** AddressDao {  **public** List<Address> allAddress(**int** i);//用来查找第i条到第几条的类容，作用是给分页时提供不同的内容  **public** **double** pageSum();//用来查找数据的总量，然后在做分页的时候，用来得到页数  } |

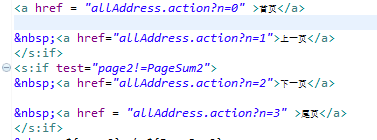
1. 新建实现AddressDao.java接口的类

|  |
| --- |
| **public** **class** AddressDaoImp **extends** HibernateDaoSupport **implements** AddressDao {  **public** List<Address> allAddress(**int** i) {  Session ssn = getSession();  Query query = ssn.createQuery("from Address");  query.setFirstResult(i);//从第i条数据开始查  query.setMaxResults(7);//最大可以查的数据量 7 范围就是i~i+7 List<Address> l = query.list();//query.list()的作用是获得query查找出来的量  ssn.close();  **return** l;  }  **public** **double** pageSum() {  Session session = getSession();  Query query = session.createQuery("select count(\*) from Address");  Long count = (Long) query.uniqueResult();//uniqueResult的用法是，当确定返回值这有一个的时候。用uniqueresult  **double** a = count;  session.close();  **return** a;  }  } |

1. 新建地址页面addresslist.jsp

|  |
| --- |
| <%@page contentType=*"text/html; charset=UTF-8"*%>  <%@taglib uri=*"/struts-tags"* prefix=*"s"*%>  <%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"* %>  <%@taglib uri=*"http://java.sun.com/jsp/jstl/fmt"* prefix=*"fmt"*%>  <%@ page isELIgnored=*"false"*%>  <!DOCTYPE html>  <html>  <head>  <title>地址管理</title>  <script src=*"jules/js/jquery.min.js?v=2.1.4"*></script>  <script src=*"jules/js/bootstrap.min.js?v=3.3.6"*></script>  <script src=*"jules/js/plugins/slimscroll/jquery.slimscroll.min.js"*></script>  <script src=*"jules/js/content.min.js?v=1.0.0"*></script>  <link href=*"jules/css/bootstrap.min14ed.css?v=3.3.6"* rel=*"stylesheet"*>  <link href=*"jules/css/font-awesome.min93e3.css?v=4.4.0"* rel=*"stylesheet"*>  <link href=*"jules/css/animate.min.css"* rel=*"stylesheet"*>  <link href=*"jules/css/style.min862f.css?v=4.1.0"* rel=*"stylesheet"*  <script type=*"text/javascript"*>  var goodid;  function deleteaddress(id){  var url = "oneAddress.action?addressId="+id;  window.location.href=url;  }  <!--用来做修改用的 -->  </script>  </head>  <s:debug/>  <body class=*"gray-bg"*>  <div class=*"wrapper wrapper-content animated fadeInRight"*>  <div class=*"row"*>  <div class=*"col-sm-12"*>  <div class=*"ibox"*>  <div class=*"ibox-content"*>  <h2>所有地址</h2>  <div class=*"clients-list"*>  <div class=*"tab-content"*>  <div id=*"tab-1"* class=*"tab-pane active"*>  <div class=*"full-height-scroll"*>  <div class=*"table-responsive"*>  <table class=*"table table-striped table-hover"*>  <tbody>  <c:forEach items=*"*${addresslist}*"* var=*"res"*>  <tr style="height:*80px*">  <td class=*"client-link"*>序号：</a>  <small>  ${res.id}  </small>  </td>  <td class=*"project-title"*>  <a href=*""*>教室名称：</a>    <small>  ${res.name}&nbsp;  </small>  </td>  <td class=*"project-title"*>  <a href=*""*>教室经度：</a>    <small>  ${res.longitude}&nbsp;  </small>  </td>    <td class=*"project-title"*>  <a href=*""*>教室纬度：</a>    <small>  ${res.latitude}&nbsp;  </small>  </td>  <td>    <button onClick=*"deleteaddress(*${res.id}*)"* type=*"button"* class=*"btn btn-danger btn-xs"*><span>修改</span></button>    </td>  </tr>    </c:forEach>  </td>  </tbody>  </table>  <td style="text-align:*center*">  <div style="text-align: *center*;">  <s:if test=*"page2!=1"*>  <a href = *"allAddress.action?n=0"* >首页</a>  &nbsp;<a href=*"allAddress.action?n=1"*>上一页</a>  </s:if>  <s:if test=*"page2!=PageSum2"*>  &nbsp;<a href=*"allAddress.action?n=2"*>下一页</a>  &nbsp;<a href = *"allAddress.action?n=3"* >尾页</a>  </s:if>  &nbsp;第${page2}页/共${PageSum2}页  </div>  </td>  </div>  </div>  </div>  </body>  </html> |

1. 解释addresslist里面的数据是怎么拿到的：
2. 
3. 这个是在WebAction.java类里申明了一个量 **private** List<Address> addresslist;
4. 后面的var = “res”是一个标识列，用来遍历addresslist



这里n的作用是：标识点击的是哪个。 所以等下WebAction.java类里也要声明一个

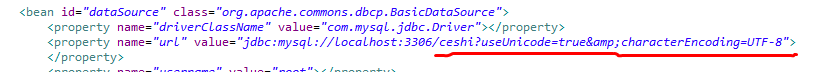
1. 给这些值赋值（在WebAction里面）

|  |  |
| --- | --- |
| |  | | --- | | ApplicationContext.xml添加配置  <bean id=*"addressDaoImp"* class=*"web.dao.imp.AddressDaoImp"*>  <property name=*"sessionFactory"* ref=*"mysessionFactory"*>  </property>  </bean>  <bean id=*"webAction"* class=*"web.action.WebAction"*>  <property name=*"relationDao"* ref=*"relationDaoImp"*></property>  <property name=*"addressDao"* ref=*"addressDaoImp"*></property>  </bean> |   AddressDao addressDao;  Map session;  **private** **int** n;  **private** **int** pageSum, page;// 活动页码  **private** **int** pageSum1, page1;// 活动内别页码  **private** **int** pageSum2, page2;// 地址页码  **private** **int** pageSum3, page3;// 公告页码  **private** **int** pageSum6, page6;// 学生cj列表页码  **private** List<Address> addresslist;  然后生成get set方法  实现allAddress（）方法  **public** String allAddress() {  Integer pa = (Integer) session.get("addresspag");//原来这句话的用法是，在用户登录的时候，就给addresspag赋值了1,现在我没有登录界面所以我只能在 inddex()方法里 进行赋值 session.set(“addresspag”,1);  pageSum2 = (**int**) (addressDao.pageSum() / 7 + 0.99);//用来获取总页数  **if** (pageSum2 == 0)  pageSum2 = 1;  **if** (n == 0)  pa = 1;  **else** **if** (n == 1 && pa > 0)  pa = pa - 1;  **else** **if** (n == 2 && pa < pageSum2)  pa = pa + 1;  **else** **if** (n == 3 && pa < pageSum2)  pa = pageSum2;  n = 5;// 保持原位置  page2 = pa;  addresslist = addressDao.allAddress((pa - 1) \* 7);  session.put("addresspag", pa);  //pageSum2 n 的值都是在页面中自己获得的  **return** "success";  }  注意的是：在给session.set(“addresspag”,1)赋值的时候，要在index（）方法里，如果在set方法里，将导致每次session.set()方法都被执 |

1. 效果

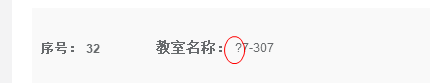


# 添加地址：

1. 在applicationContext.xml里面给数据里添加中文支持

|  |
| --- |
| <property name=*"url"* value=*"jdbc:mysql://localhost:3306/ceshi?useUnicode=true&amp;characterEncoding=UTF-8"*> |

不添加的结果就是，在网页里输入中文时：数据库里添加的数据变成问号，如下：



1. 给接口AddressDao添加 添加地址的方法

|  |
| --- |
| **public** **interface** AddressDao {  **public** List<Address> allAddress(**int** i);  **public** **double** pageSum();  **public** **void** newAddress(Address address);//添加新地址  } |

1. 在AddressDaoImp类里实现这方法

|  |
| --- |
| **public** **void** newAddress(Address address) {  **this**.getHibernateTemplate().save(address);  } |

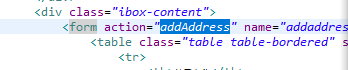
1. 添加地址后面对应的连接：addAddress.jsp



1. 添加addAddress.jsp页面

|  |
| --- |
| <%@page contentType=*"text/html; charset=UTF-8"*%>  <%@taglib uri=*"/struts-tags"* prefix=*"s"*%>  <%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>  <%@ page isELIgnored=*"false"*%>  <%@taglib uri=*"http://java.sun.com/jsp/jstl/fmt"* prefix=*"fmt"*%>  <!DOCTYPE html>  <html>  <head>  <meta charset=*"UTF-8"*>  <title>Title</title>  <link href=*"jules/css/bootstrap.min14ed.css?v=3.3.6"* rel=*"stylesheet"*>  <link href=*"jules/css/font-awesome.min93e3.css?v=4.4.0"* rel=*"stylesheet"*>    <link href=*"jules/css/animate.min.css"* rel=*"stylesheet"*>  <link href=*"jules/css/style.min862f.css?v=4.1.0"* rel=*"stylesheet"*>  <link rel=*"stylesheet"* type=*"text/css"* href=*"jules/css/iconfont.css"*>  <script src=*"jules/js/jquery.min.js?v=2.1.4"*></script>  <style>  **th** {  background-color: *#F5F5F6*;  }  *#container* {  width: *500px*;  height: *300px*;  border: *1px solid gray*;  margin: *0 auto*;  }  </style>  </head>  <body class=*"gray-bg"*>  <div class=*"wrapper wrapper-content animated fadeInRight article"*>  <div class=*"row"*><div class=*"col-lg-10 col-lg-offset-1"*>  <div class=*"ibox"*><div class=*"ibox-content"*>  <div class=*"ibox float-e-margins"*>  <div class=*"text-center"*>  <h3>新增地址</h3>  </div>  <div class=*"ibox-content"*>  <form action=*"addAddress"* name=*"addaddress"* method=*"post"* enctype=*"multipart/form-data"*>  <table class=*"table table-bordered"* style="height: *auto*;"><tr><th>教室名称</th><td>  <input class=*"form-control "* type=*"text"* name=*"address"* required=*"required"*>  </td></tr><tr><th>教室经度</th><td>  <input class=*"form-control "* type=*"text"* name=*"longitude"* id=*"jd"* required=*"required"*></td></tr><tr><th>教室纬度</th><td>  <input class=*"form-control "* type=*"text"* name=*"latitude"* id=*"wd"* required=*"required"*></td></tr><tr>  <th>扫码范围&nbsp;<i class=*"iconfont icon-wenhao"* id=*"displayInput"* required=*"required"*></th><td>  input class=*"form-control "* type=*"text"* name=*"range"* placeholder=*"推荐范围10~20"*>  </td>  </tr>  </table>  <div class=*"form-group"* style="height: *auto*">  <input class=*"btn btn-sm btn-primary m-t-n-xs pull-right"* style="margin-left: *20px*" type=*"submit"* value=*"取消"*>  <input id=*"submit"* class=*"btn btn-sm btn-primary m-t-n-xs pull-right"* type=*"submit"* value=*"确定"*>  </div></form></div></div></div></div></div></div></div></body>  </html> |

1. 这个表单对应的action是：

則在sturcts.xml添加对他的响应



1. 在WebAction.java里面添加 addAddress方法
2. 先声明网页里用到的变量：

**private** Double latitude;// 地址纬度

**private** Double longitude;//教师经度

**private** **int** range;//范围

**private** String address;//地点名称

生成get set 方法

1. 添加addAddress方法

|  |
| --- |
| // 添加地址  **public** String addAddress() {  Address add = **new** Address();  add.setName(**this**.getAddress());  add.setLatitude(**this**.getLatitude());  add.setLongitude(**this**.getLongitude());  add.setARange(**this**.getRange());  **if** (add.getARange() == **null**) {  add.setARange(10);  }  addressDao.newAddress(add);  **return** "success";  } |

效果：

# 四：修改界面

1. 添加修改按钮的响应事件

<button onClick=*"deleteaddress(*${res.id}*)"* type=*"button"* class=*"btn btn-danger btn-xs"*>

添加javascript语句

|  |
| --- |
| <script type=”text/javascript”>  Function deleteaddress(id){  Var url=”oneAddress.action?addressId=”+id;  Window.location.href=url;  }  </script> |

1. 在WebAction.java类里添加变量 addressId

**private** **int** addressId;// 地址ID 然后生成get set方法

1. 在structs.xml添加oneAddress.action

<action name=*"oneAddress"* class=*"webAction"* method=*"oneAddress"*>

<result name=*"success"*>/updateAddress.jsp</result>

</action>

1. 在AddressDaoImp实现 查找指定Address的方法

|  |
| --- |
| **public** Address findAddress(**int** addressId) {  List<Address> list = getHibernateTemplate().find("from Address where id=?", addressId);  **return** list.get(0);  } |

1. 在WebAction.java添加oneAddress方法
2. 先定义Address addre;//地址信息类 并生成 get set方法

|  |
| --- |
| **public** String oneAddress() {  // System.out.println(itemId);  addre = addressDao.findAddress(addressId);  // System.out.println(acItem.getAname());  **return** "success";  } |

1. 添加updateAddress.jsp页面

|  |
| --- |
| <%@page contentType=*"text/html; charset=UTF-8"*%>  <%@taglib uri=*"/struts-tags"* prefix=*"s"*%>  <%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>  <%@ page isELIgnored=*"false"*%>  <%@taglib uri=*"http://java.sun.com/jsp/jstl/fmt"* prefix=*"fmt"*%>  <!DOCTYPE html>  <html>  <head>  <meta charset=*"utf-8"*>  <meta name=*"viewport"* content=*"width=device-width, initial-scale=1.0"*>  <title>修改地址</title>  <link href=*"jules/css/bootstrap.min14ed.css?v=3.3.6"* rel=*"stylesheet"*>  <link href=*"jules/css/font-awesome.min93e3.css?v=4.4.0"* rel=*"stylesheet"*>  <link href=*"jules/css/animate.min.css"* rel=*"stylesheet"*>  <link href=*"jules/css/style.min862f.css?v=4.1.0"* rel=*"stylesheet"*>  <style type=*"text/css"*>  </style>  <script type=*"text/javascript"* src=*"js/jquery.min.js"*>  </script>  <script src=*"js/qrcode.min.js"*>  </script>  </head>  <body class=*"gray-bg"*>  <div id=*"showLargeImage"* style="display:*none*;z-index:*9999*;width:*500px*;height:*400px*;position:*absolute*;left:*50%*;">  <img id=*"ShowImageBox"* src=*"http://img.52fuqing.com/upload/news/20160125/201601251216541264.jpg"* width=*"500px;"* height=*"380px;"* />  </div>  <div class=*"wrapper wrapper-content animated fadeInRight article"*>  <div class=*"row"*>  <div class=*"col-lg-10 col-lg-offset-1"*>  <div class=*"ibox"*>  <div class=*"ibox-content"*>  <form action=*"updateAddress"* name=*"cform"* method=*"post"* enctype=*"multipart/form-data"* id=*"upDetailFrom"*>  <div class=*"text-center "*>  <h1>${addre.name}</h1>  <!-- >input style="display:none" type="text" value="${acItem.AId}" name="itemId" /-->  <div class=*"ibox float-e-margins"*>  <div class=*"ibox-content"*>  </div>  <div class=*"form-group"* style="height: *95px*">  <label class=*"col-sm-2"* style="a">地址名：</label>  <textarea class=*"form-control "* rows=*"2"* type=*"text"* name=*"name"*>${addre.name}</textarea>  </div>  <div class=*"form-group"* style="height: *95px*">  <label class=*"col-sm-2"* style="a">经度：</label>  <textarea class=*"form-control "* rows=*"2"* type=*"text"* name=*"longitude"*>${addre.longitude}</textarea>  </div>  <div class=*"form-group"* style="height: *95px*">  <label class=*"col-sm-2"* style="a">纬度：</label>  <textarea class=*"form-control "* rows=*"3"* type=*"text"* name=*"latitude"*>${addre.latitude}</textarea>  </div>  <div class=*"form-group"* style="height: *auto*">  <input id=*"submit"* class=*"btn btn-sm btn-primary m-t-n-xs pull-right"* type=*"submit"* value=*"确定 "* />  </div></div></div></form></div></div></div></div></div>  </div>  </div>  <script src=*"jules/js/jquery.min.js?v=2.1.4"*></script>  <script src=*"jules/js/bootstrap.min.js?v=3.3.6"*></script>  <script src=*"jules/js/content.min.js?v=1.0.0"*></script>  </body>  </html> |

1. 在添加struts.xml添加 这个action



<action name=*"updateAddress"* class=*"webAction"* method=*"updateAddress"*>

<result name=*"success"* type=*"chain"*>allAddress</result>

</action>

1. 在WebAction.java添加updateAddress方法

|  |
| --- |
| // 修改地址  **public** String updateAddress() {  Address address = addressDao.findAddress(addressId);  address.setName(name);  address.setLatitude(**this**.getLatitude());  address.setLongitude(**this**.getLongitude());  address.setARange(**this**.getRange());  addressDao.updateAddress(address);  **return** "success";  } |

效果：



