# 成绩管理

### 一：修改原来的界面样式

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1. 标题栏：  |  | | --- | | <h2 style="text-align:*center*">活动管理</h2>  <div style="text-align:*center*;color:*#60CCC6*;">  <span class=*"glyphicon glyphicon-lock"* style=*"font-weight:bold"*>只有模块管理员可以修改删除</span></div>  效果： |   2.button按钮的颜色   | **按钮** | **class=""** | | --- | --- | | 默认 | btn | | 主要 | btn btn-primary | | 信息 | btn btn-info | | 成功 | btn btn-success | | 警告 | btn btn-warning | | 危险 | btn btn-danger | | 反向 | btn btn-inverse | | 链接 | btn btn-link |   IMG_256 |

### 二:添加成绩管理响应事件

|  |
| --- |
| <button onClick=*"lookGrade(*${res.RId}*)"* type=*"button"*  class=*"btn btn-primary btn-xs"*>  <span>成绩管理</span></button>  //成绩管理  **function** lookGrade(id){  **var** url="gradelist2.jsp?actid="+id;  window.location.href=url;  } |

### 三:添加gradelist.jsp

|  |
| --- |
| 这里用了  <link href=*"jules/css/dataTables.bootstrap.css"* rel=*"stylesheet"*>  <script src=*"jules/js/dataTables.bootstrap.js"*></script>  Jquery的datatable控件，官网上有不同的制作案例[http://datatables.net/](http://datatables.net/" \t "https://blog.csdn.net/tyyytcj/article/details/_blank) |

##### 3.1

|  |
| --- |
| 这里我先模仿Datatable来访问本地文件的数据，然后在表格上显示出来   1. 新建数据文件放在同级的文件位置   里面的数据：  {  "draw": 1,  "recordsTotal": 200,  "recordsFiltered": 200,  "data":[  {  "id": "1",  "name": "Tiger Nixon",  "position": "System Architect",  "salary": "$320,800",  "start\_date": "2011/04/25",  "office": "Edinburgh",  "extn": "5421"  }  ]  }(以这样的格式)   1. 样式   <div id=*"tab-2"* >  <table id=*"example"* class=*"table table-bordered"* style="width:*100%*">  <thead>  <tr>  <th>Name</th>  <th>Position</th>  <th>Office</th>  <th>Extn.</th>  <th>Start date</th>  <th>Salary</th>  </tr>  </thead>  </table>  </div>   1. Jquery实现   $('#example').DataTable( {  serviceSize:**true**,  ajax:{  url:'objects.json',  type:'POST',  dataSrc:'data'  },  columns: [  { "data": "name" },  { "data": "position" },  { "data": "office" },  { "data": "extn" },  { "data": "start\_date" },  { "data": "salary" }  ]  } );  效果： |

很明显我用的这方法，在记事本里面添加新数据很麻烦，而且这样的方式读取的都是以json结尾的文件，无法操作excle文件

##### 3.2datatable操作excle文件

|  |  |
| --- | --- |
| 1.学生表格的绘制  <div id=*"tab-1"* class=*"tab-pane active"* >  <div class=*"panel-body"*>  <table class=*"table table-bordered"* id=*"datatable1"*>  <thead>  <tr>  <th>学号</th>  <th>姓名</th>  <th>学院</th>  <th>班级</th>  <th>电话</th>  <th>成绩</th>  </tr>  </thead>  </table>  </div>   1. jequery获取数据   $(document).ready(**function**() {  $('#datatable1').DataTable(  {  ordering : **true**,  bScrollInfinite : **true**,  bScrollCollapse : **true**,  ajax:{  url:"allActRecord.action?actid="+deleteid,  type : "Get",  async : **false**,  dataType : "json",  dataSrc:'datas'  },  aoColumns : [  {"data" : "id"},  {"data" : "name"},  {"data" : "college"},  {"data" : "clasz"},  {"data" : "iphone"},  {"data" : "grade"}  ],  }  );  });   1. 写数据来源：在struts.xml添加allActRecord.action   <action name=*"allActRecord"* class=*"webAction"* method=*"allActRecord"*>  <result type=*"json"* name=*"success"*>  <param name=*"root"*>s</param>  </result>  </action>   1. 添加方法allActRecord方法   // 学生活动成绩列表+  @SuppressWarnings("unchecked")  **public** String allActRecord() {  activitingslist = actRecordDao.allStuRecord(actid);  s = JSONTools.*createJsonObject*("datas", activitingslist);  **return** "success";  }   |  | | --- | | **public** List<Map<String, Object>> allStuRecord(**int** id) {  Session ssn = getSession();  Query query = ssn.createQuery("from Activiting a where a.id.ARId=" + id);  List<Activiting> l = query.list();  ssn.close();  List<Map<String, Object>> listg = **new** ArrayList<>();  **if** (l != **null** && l.size() > 0) {  List<Students> list = **null**;  Map<String, Object> map;  **int** len = l.size();  Object[] values = **new** Object[len];  StringBuilder sb = **new** StringBuilder();  **for** (**int** j = 0; j < len; j++) {  values[j] = l.get(j).getId().getASId();  **if** (j != len - 1) {  sb.append("?,");  } **else** {  sb.append("?");  }  }  list = getHibernateTemplate().find("FROM Students t WHERE t.id.SStudentId IN (" + sb.toString() + ")",  values);  **if** (list != **null** && list.size() > 0) {  **for** (Activiting a : l) {  **for** (Students stu : list) {  // for (List<Object> a : l) {  **if** (stu.getId().getSStudentId().equals(a.getId().getASId())) {  map = **new** HashMap<String, Object>();  map.put("id", stu.getId().getSStudentId());  map.put("name", stu.getSName());  map.put("college", stu.getSCollege());  map.put("clasz", stu.getSClass());  map.put("iphone", stu.getSPhone());  **if** (a.getAGrade() != **null**) {  **if** (a.getAGrade() == 1)  map.put("grade", "优秀");  **else** **if** (a.getAGrade() == 2)  map.put("grade", "良好");  **else** **if** (a.getAGrade() == 3)  map.put("grade", "不合格");  } **else**  map.put("grade", "未评价");  listg.add(map);  }  }  }  }  }  **return** listg;  } | |

### 四：下载功能

1.添加按钮的响应事件

|  |
| --- |
| $("#mingdan").click(**function**() {  alert("下载名单,请等待生成");  $.ajax({  url: "exportStuActRecord.action?id="+deleteid,  type: "POST",  async: **false**,  dataType: "json",  success: **function**(data) {  **if**(data.data=="OK"){  window.location.href = "https://sdsy.zzjc.edu.cn/SDSYw/excleFile/grade.xls";  }  }  });  });  这里的href要改成本地的：http://127.0.0.1:8080/shangde/excleFile/grade.xls |

2.添加exportStuActRecord.action

|  |
| --- |
| <action name=*"exportStuActRecord"* method=*"exportStuActRecord"* class=*"excle"*>  <result type=*"json"* name=*"success"*>  <param name=*"root"*>s</param>  </result>  </action> |

3.添加方法

|  |
| --- |
| // 一个活动成绩导出  **public** String exportStuActRecord() {  **try** {  System.*out*.println("成绩名单导出");  WritableWorkbook wwb = **null**;  // 默认保存在d盘  String path = ServletActionContext.*getServletContext*().getRealPath("/excleFile");  File file = **new** File(path + "/grade.xls");  // System.out.println(file);  **if** (!file.exists()) {  file.createNewFile();  }  wwb = Workbook.*createWorkbook*(file);  WritableSheet ws = wwb.createSheet("sheet0", 0);  List<Map<String, Object>> list = actRecordDao.allStuRecord(id);  Label labelSId = **new** Label(0, 0, "学号");//列号 行号 内容  Label laelName = **new** Label(1, 0, "姓名");  Label labelSchool = **new** Label(2, 0, "学院");  Label labelClass = **new** Label(3, 0, "班级");  Label labelPhone = **new** Label(4, 0, "联系方式");  Label lablegrade = **new** Label(5, 0, "成绩");  ws.addCell(labelSId);  ws.addCell(laelName);  ws.addCell(labelSchool);  ws.addCell(labelClass);  ws.addCell(labelPhone);  ws.addCell(lablegrade);  **int** i = 1;  **for** (Map<String, Object> actR : list) {  ws.addCell(**new** Label(0, i, (String) actR.get("id")));  ws.addCell(**new** Label(1, i, (String) actR.get("name")));  ws.addCell(**new** Label(2, i, (String) actR.get("college")));  ws.addCell(**new** Label(3, i, (String) actR.get("clasz")));  ws.addCell(**new** Label(4, i, (String) actR.get("iphone")));  ws.addCell(**new** Label(5, i++, (String) actR.get("grade")));  }  wwb.write();  wwb.close();  } **catch** (IOException e) {  // **TODO** Auto-generated catch block  e.printStackTrace();  } **catch** (RowsExceededException e) {  // **TODO** Auto-generated catch block  e.printStackTrace();  } **catch** (WriteException e) {  // **TODO** Auto-generated catch block  e.printStackTrace();  }  s = JSONTools.*createJsonObject*("data", "OK");  **return** "success";  } |

完成