# 评分页面

1. 添加评分控件

|  |
| --- |
| <button onClick=*"lookActivity(*${res.RId}*)"* type=*"button"* class=*"btn btn-primary btn-xs"*>  <span>评分</span>  </button> |

|  |  |
| --- | --- |
| **类** | **描述** |
| .btn | 为按钮添加基本样式 |
| .btn-default | 默认/标准按钮 |
| .btn-primary | 原始按钮样式（未被操作） |
| .btn-success | 表示成功的动作 |
| .btn-info | 该样式可用于要弹出信息的按钮 |
| .btn-warning | 表示需要谨慎操作的按钮 |
| .btn-danger | 表示一个危险动作的按钮操作 |
| .btn-link | 让按钮看起来像个链接 (仍然保留按钮行为) |
| .btn-lg | 制作一个大按钮 |
| .btn-sm | 制作一个小按钮 |
| .btn-xs | 制作一个超小按钮 |
| .btn-block | 块级按钮(拉伸至父元素100%的宽度) |
| .active | 按钮被点击 |
| .disabled | 禁用按钮 |

1. 添加按钮的响应事件

|  |
| --- |
| **function** lookActivity(id) {  **var** url = "passStu2.jsp?actid=" + id;  window.location.href = url;  } |

1. 添加passStu2,jsp页面
2. 外面先嵌套了一层栅格系统

|  |
| --- |
| <div class=*"row"*>  <div class=*"col-sm-12"*>  </div>  </div> |

1. 书写导航栏



|  |
| --- |
| <div class=*"tabs-container"*>  <ul class=*"nav nav-tabs"*>  <li class=*"active"*><a data-toggle=*"tab"* href=*"#tab-1"*  aria-expanded=*"true"*> 良好的学生</a></li>  <li class=*""*><a data-toggle=*"tab"* href=*"#tab-2"*  aria-expanded=*"false"*>优秀的学生</a></li>  <li class=*""*><a data-toggle=*"tab"* href=*"#tab-3"*  aria-expanded=*"false"*>不及格的学生</a></li>  </ul>  </div> |

|  |
| --- |
| 这里用了 Bootstrap 标签页（Tab）插件用法：  1. 通过data属性：你需要添加data-toggle=”tab”或data-toggle=”pill”到文本链接中。   例如：<ul class=”nav nav-tabs”>  <li><a href=”#” data-toggle=”tab”></a></li>  <li><a href=”#” data-toggle=”tab”></a></li>  <li><a href=”#” data-toggle=”tab”></a></li>  <li><a href=”#” data-toggle=”tab”></a></li>  </ul> |

1. 添加不同按钮的响应

|  |
| --- |
| <div class=*"tab-content"*>  <div id=*"tab-1"* class=*"tab-pane active"*>  <div class=*"panel-body"*>  <table class=*"table table-bordered"* id=*"datatable2"*>  <thead>  <tr>  <th>序号</th>  <th>学号</th>  <th>姓名</th>  <th>班级类别</th>  <th>联系方式</th>  <th>是否迟到</th>  <th>是否早退</th>  <th>是否迟到</th>  <th>是否早退</th>  </tr>  </thead>  </table>  <div class=*"form-group"*>  <button class=*"btn btn-sm btn-primary m-t-n-xs "* id=*"removeSR1"*>  <strong>移至优秀</strong>  </button>  <button class=*"btn btn-sm btn-primary m-t-n-xs "* id=*"removeSR2"*>  <strong>移至不及格</strong>  </button>  </div>  </div>  </div>  </div>  只需要重复再 <div></div>里面添加你的类容 然后就是修改第二个div的id 和 table里的id信息 |

1. 像签到管理一样给table添加信息

|  |
| --- |
| **var** url = window.location;  **function** getUrlParam(url, name) {  **var** pattern = **new** RegExp("[?&]" + name + "\=([^&]+)", "g");  **var** matcher = pattern.exec(url);  **var** items = **null**;  **if** (matcher != **null**) {  **try** {  items = decodeURIComponent(decodeURIComponent(matcher[1]));  } **catch** (e) {  **try** {  items = decodeURIComponent(matcher[1]);  } **catch** (e) {  items = matcher[1];  }  }  }  **return** items;  }  **var** deleteid = getUrlParam(url, 'actid'); |
| **var** right = $('#datatable2').DataTable(  {  // 及格  destory : **true**,  searching : **true**,  bLengthChange : **true**,  ordering : **false**,  bScrollInfinite : **true**,  bScrollCollapse : **true**,  ajax : "oneActRecord.action?actid=" + deleteid,  aoColumns : [  {  "data" : "id"  },  {  "data" : "xuehao"  },  {  "data" : "name"  },  {  "data" : "classleibie"  },  {  "data" : "dianhua"  },  {  "data" : "ischidao"  },  {  "data" : "iszaotui"  },  {  "data" : "ischidao2"  },  {  "data" : "iszaotui2"  }  ],  }  ); |

1. 在struts.xml添加对应的action

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

<result type=*"json"* name=*"success"*>

<param name=*"root"*>s</param>

</result>

</action>

1. 在WebAction添加oneActRecod方法

|  |
| --- |
| // 活动成绩  **public** String oneActRecord() {  activitingslist = actRecordDao.findActRecord(actid);  s = JSONTools.*createJsonObject*("data", activitingslist);  **return** "success";  } |

1. 在ActRecordDao添加findActtRecord(int id)方法

|  |
| --- |
| List<Map<String, Object>> findActRecord(**int** actid); |

1. 在ActRecordDaoImp实现方法findActRecord(int id)

|  |
| --- |
| @Override  **public** List<Map<String, Object>> findActRecord(**int** actid) {  List<Activiting> list = getHibernateTemplate().find("from Activiting a where a.id.ARId=?", actid);  **int** len = list.size();  List<Map<String, Object>> ll = **new** ArrayList<Map<String, Object>>();  **if** (len > 0 && list != **null**) {  List<Students> l = **null**;  List<Activity> aclist = **null**;  Object[] values = **new** Object[len];  Object[] values2 = **new** Object[len];  StringBuilder sb = **new** StringBuilder();  StringBuilder sb2 = **new** StringBuilder();  **for** (**int** j = 0; j < len; j++) {  values[j] = list.get(j).getId().getASId();  values2[j] = list.get(j).getId().getARId();  **if** (j != len - 1) {  sb.append("?,");  sb2.append("?,");  } **else** {  sb.append("?");  sb2.append("?");  }  }  l = getHibernateTemplate().find("FROM Students t WHERE t.id.SStudentId IN (" + sb.toString() + ")", values);  aclist = getHibernateTemplate().find("FROM Activity t WHERE t.RId IN (" + sb2.toString() + ")", values2);  **for** (Activiting a : list) {  **if** (a.getId().getASId().equals("") || a.getId().getARId() == 0 || a.getId().getASId() == **null**) {  **this**.getHibernateTemplate().delete(a);  **break**;  }  **for** (Students st : l)  **if** (a.getId().getASId().equals(st.getId().getSStudentId())) {  a.setStudent(st);  **break**;  }  **for** (Activity ac : aclist)  **if** (a.getId().getARId().equals(ac.getRId())) {  a.setActivity(ac);  **break**;  }  }  **int** k = 1;  Map<String, Object> map;  **for** (Activiting a : list) {  map = **new** HashMap<String, Object>();  map.put("id", k);  **if** (a.getId().getASId() != **null**)  map.put("xuehao", a.getId().getASId());  **else**  map.put("xuehao", "空");  **if** (a.getStudent() != **null**) {  map.put("name", a.getStudent().getSName());  map.put("classleibie", a.getStudent().getSClass());  map.put("dianhua", a.getStudent().getSPhone());  } **else** {  map.put("name", "空");  map.put("classleibie", "空");  map.put("dianhua", "空");  }  **if** (a.getAFirsttime() == **null**) {  map.put("ischidao", "未签到");  } **else** **if** (a.getAFirsttime().getTime() > a.getActivity().getRStartTime().getTime()) {  map.put("ischidao", "是");  } **else** {  map.put("ischidao", "否");  }  **if** (a.getASecondtime() == **null**) {  map.put("iszaotui", "未签到");  } **else** **if** (a.getASecondtime().getTime() < a.getActivity().getREndTime().getTime()) {  map.put("iszaotui", "是");  } **else** {  map.put("iszaotui", "否");  }  **if** (a.getAThirdtime() == **null**) {  map.put("ischidao2", "未签到");  } **else** **if** (a.getAThirdtime().getTime() > a.getActivity().getRStartTimeTwo().getTime()) {  map.put("ischidao2", "是");  } **else** {  map.put("ischidao2", "否");  }  **if** (a.getAFourtime() == **null**) {  map.put("iszaotui2", "未签到");  } **else** **if** (a.getAFourtime().getTime() < a.getActivity().getREndTimeTwo().getTime()) {  map.put("iszaotui2", "是");  } **else** {  map.put("iszaotui2", "否");  }  k++;  ll.add(map);  // System.out.println("ll:：" + ll);  }  }  **return** ll;  } |

效果：



