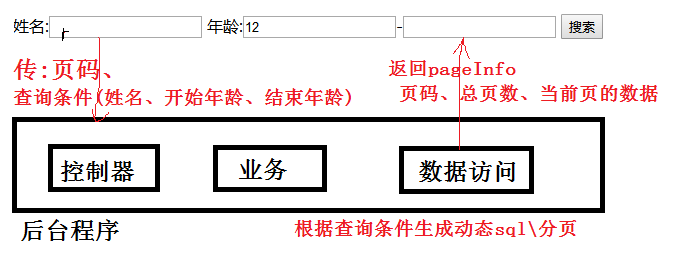
目标:

1. 带条件查询分页
2. 连接查询
3. 介绍bootstrap前端框架
4. 带条件分页展示

　分析:



1. 创建条件实体 放到工具包中util

|  |
| --- |
| **public class** PageParmeter { //分页实体  **private** Integer **page**=1; *//页码*  **private** Integer **pageSize**=5; *//页大小*  } |

|  |
| --- |
| *//学生查询条件 放到实体包中* **public class** StudentCondition **extends** PageParmeter {  *//一个条件一个属性* **private** String **name**;  **private** Integer **startAge**;  **private** Integer **endAge**;  setter和getter } |

1. 编写Dao层

List<Students> selectByExample(StudentsExample example);

支持条件查询:无需编写dao

1. 编写业务层

接口方法:

|  |
| --- |
| */\*\*  \* 带条件的分页查询  \** ***@param studentCondition*** *查询条件  \* 包含查询条件和分页参数  \** ***@return*** *pageInfo  \*/* **public** PageInfo<Student> getStudentByCondition(StudentCondition studentCondition); |

实现:

|  |
| --- |
| @Override **public** PageInfo<Student> getStudentByCondition(StudentCondition studentCondition) {  *//1.开启分页* PageHelper.*startPage*(studentCondition.getPage(),studentCondition.getPageSize());  *//2.调用dao层条件查询所有信息(存在单表的学生查询)* StudentExample studentExample=**new** StudentExample();  StudentExample.Criteria criteria=studentExample.createCriteria();  *//根据条件动态生成查询语句 null "a"* **if**(!StringUtils.*isEmpty*(studentCondition.getName()))  criteria.andNameLike(**"%"**+studentCondition.getName()+**"%"**);  **if**(!StringUtils.*isEmpty*(studentCondition.getStartAge()))  criteria.andAgeGreaterThan(studentCondition.getStartAge().byteValue());  **if**(!StringUtils.*isEmpty*(studentCondition.getEndAge()))  criteria.andAgeLessThan(studentCondition.getEndAge().byteValue());  List<Student> list=**studentMapper**.selectByExample(studentExample);  **return new** PageInfo<Student>(list); } |

1. 编写控制器

|  |
| --- |
| *//条件分页查询* @RequestMapping(**"/searchStudent"**) **public** String searchStudent(StudentCondition studentCondition,Model model){  *//调用业务查询* PageInfo<Student> pageInfo=**this**.**studentService**.getStudentByCondition(studentCondition);  *//填充返回的数据* model.addAttribute(**"pageInfo"**,pageInfo); *//分页信息* model.addAttribute(**"condition"**,studentCondition); *//查询的条件信息* **return "indexPage"**; } |

1. 前端展示，回显，分页

实现思路:

1. 显示当前页的数据
2. 实现搜索及回显示条件
3. 实现搜索条件的分页导航:

给表单设置跳转的页面(利用隐藏域)、提交表单(传递当前页码和查询条件)

|  |
| --- |
| <%@ **page contentType**="**text/html; charset=utf-8**" **pageEncoding**="**UTF-8**" **language**="**java**" %> <%@**taglib prefix**="**c**" **uri**="**http://java.sun.com/jsp/jstl/core**" %> <**html**> <**body**> <**h2**>所有信息如下:</**h2**> <**form name="form1" id="form1" method="post" action="searchStudent"**>  <!--利用隐藏域传递页码-->  <**input type="hidden" name="page" id="setPage" value="1"**>  姓名:<**input type="text" name="name" value="${**condition.name**}"** />  年龄:<**input type="text" name="startAge" value="${**condition.startAge**}"**/>-<**input type="text"value="${**condition.endAge**}" name="endAge"**/>  <**input type="submit" value="搜索" name="ss"**/> </**form**> <**table border="2"**>  <**tr**>  <**td**>学号</**td**>  <**td**>姓名</**td**>  <**td**>年龄</**td**>  <**td**>性别</**td**>  <**td**>生日</**td**>  <**td**>地址</**td**>  <**td**>年级</**td**>  <**td**>操作</**td**>  </**tr**>  <**c:forEach items="${**pageInfo.list**}" var="s"**>  <**tr**>  <**td**>**${**s.xh**}**</**td**>  <**td**>**${**s.name**}**</**td**>  <**td**>**${**s.age**}**</**td**>  <**td**>**${**s.sex**}**</**td**>  <**td**>**${**s.birthday**}**</**td**>  <**td**>**${**s.address**}**</**td**>  <**td**>**${**s.gid**}**</**td**>  <**td**><**a href="javascript:if(***confirm***('确定想删除吗?')) *location*.href='delStudent?xh=${**s.xh**}';"**>删除</**a**>  <**a href="getStudent?xh=${**s.xh**}"**>修改</**a**></**td**>  </**tr**>  </**c:forEach**> </**table**> <**div**>  <**div**>每页**${**pageInfo.pageSize**}**条,当前是第**${**pageInfo.pageNum**}**页,共**${**pageInfo.pages**}**页,共**${**pageInfo.total**}**条记录</**div**>  <**div**>  <**a href="javascript:***goPage***(1)"**>首页</**a**>  <**a href="javascript:***goPage***(${**pageInfo.prePage==0?1:pageInfo.prePage**})"**>上一页</**a**>  <**a href="javascript:***goPage***(${**pageInfo.nextPage==0?pageInfo.pages:pageInfo.nextPage**})"**>下一页</**a**>  <**a href="javascript:***goPage***(${**pageInfo.pages**})"**>尾页</**a**>  </**div**>  <**div**>  <**select onchange="***goPage*(**this**.**value**)**"**>  <**option value=""**>请选择页码</**option**>  <**c:forEach begin="1" end="${**pageInfo.pages**}" var="i" step="1"**>  <**option ${**i==pageInfo.pageNum?**"selected"**:**""} value="${**i**}"**>第**${**i**}**页</**option**>  </**c:forEach**>  </**select**>  </**div**> </**div**> </**body**> </**html**> <**script language="JavaScript" type="text/javascript" src="js/jquery-1.8.3.min.js"**></**script**> <**script language="javaScript" type="text/javascript"**>  //跳转分页的方法  **function** goPage(pageNum){  //alert("我要显示第"+pageNum+"页");  //1.设置当前的页码  **$**(**"#setPage"**).val(pageNum);  //2.提交搜索的表单 分页导航传递搜索条件  **$**(**"#form1"**).submit(); //提交表单  } </**script**> |

1. 查询学生的同时关联年级名称

Mybatis执行连接查询(针对多表)

首先: 修改学生实体添加年级名称属性，以便实体传递数据

|  |
| --- |
| */添加额外年级名称* **private** String **gname**;  **public** String getGname() {  **return gname**; } **public void** setGname(String gname) {  **this**.**gname** = gname; } |

1. 编写dao层

接口:

|  |
| --- |
| *//带条件查询所有* List<Student> **getStudentByCondition**(**StudentCondition condition**); |

持久化操作:

|  |
| --- |
| *<!--搜索学生信息-->* <**select id="getStudentByCondition" resultType="Student" parameterType="com.kgc.grade.util.StudentCondition"**>  SELECT student.*\**,gname FROM student LEFT JOIN grade ON  student.gid=grade.`gid` where 1=1  <**if test="name !=null"**>  and name like concat('%',#{name},'%')  </**if**>  <**if test="startAge !=null"**>  and age > #{startAge}  </**if**>  <**if test="endAge !=null"**>  and age **&lt;** #{endAge}  </**if**> </**select**> |

1. 编写业务层

接口方法:

|  |
| --- |
| */\*\*  \* 条件查询分页  \** ***@param studentCondition*** *学生查询条件+分页参数  \** ***@return*** *学生信息  \*/* **public** PageInfo<Students> getStudentByCondition (StudentCondition studentCondition); |

实现:

|  |
| --- |
| @Override **public** PageInfo<Students> getStudentByCondition(StudentCondition studentCondition) {  *//1.开启分页* PageHelper.*startPage*(studentCondition.getPage(),studentCondition.getPageSize());  *//2.利用selectByExample实现条件查询 或者 自己编写持久化方法  //注意 selectByExample只是基于单表查询  //2.调用持久化操作:持久化执行的是连接查询* List<Students> list= **studentsMapper**. **getStudentByCondition** (studentCondition);  **return new** PageInfo<Students>(list); *//获取分页信息* } |

1. 控制器层:

略

1. 表示:

略

扩展:

1. 对springmvc web项目中的静态资源放行 ,在springmvc配置文件中添加以下配置

*<!--对项目中的静态资源放行-->*<**mvc:resources mapping="/js/\*\*" location="/js/"**></**mvc:resources**>