

程序说明书

目录

1、程序使用手册.....	3
2、代码结构.....	12

程序说明书

一、程序使用手册

1、首页：



功能说明：

(1) 导航总共四个选项，点击不同的选项进入对应页面。

2、学生信息

返回首页 添加 删除 修改 请输入学生姓名或学号 <input type="text"/> 搜索						
学生列表						
<input type="checkbox"/> 学号	姓名	性别	入学年龄	入学年份	班级	
<input type="checkbox"/> 002	李滢	女	18	1997	中文三班	
<input type="checkbox"/> 003	李邱	男	17	2018	软工五班	
<input type="checkbox"/> 3017000015	华章宇	男	19	2017	软工二班	
<input type="checkbox"/> 3017000017	薛致	男	18	2017	软工二班	
<input type="checkbox"/> 3017000018	苏莹	女	19	2017	软工二班	
<input type="checkbox"/> 3017000020	马伯干	男	18	2017	软工二班	
<input type="checkbox"/> 3017000021	陈理	女	19	2017	软工二班	
<input checked="" type="checkbox"/> 3017000023	王子异	男	18	2017	软工二班	

功能说明：

(1) 展示学生列表。显示学生的学号、姓名、性别、入学年龄、入学年份、班级，并按学号的大小进行排列。

(2) 添加

点击左上“添加”的小图标，弹出添加窗口，输入相关信息完成添加操作。

+

添加学生

×

学号:

001

姓名:

张三

性别:

男

▼

入学年龄:

18

入学年份:

2019

班级:

软工四班

添加学生

点击“添加学生”，页面自动刷新。出现新添加的学生信息。

返回首页

添加

删除

修改

请输入学生姓名或学号

搜索

学生列表						
<input type="checkbox"/>	学号	姓名	性别	入学年龄	入学年份	班级
<input type="checkbox"/>	001	张三	男	18	2019	软工四班
<input type="checkbox"/>	002	李滢	女	18	1997	中文三班
<input type="checkbox"/>	003	李邱	男	17	2018	软工五班
<input type="checkbox"/>	3017000015	华章宇	男	19	2017	软工二班
<input type="checkbox"/>	3017000017	薛致	男	18	2017	软工二班
<input type="checkbox"/>	3017000018	苏莹	女	19	2017	软工二班
<input type="checkbox"/>	3017000020	马伯千	男	18	2017	软工二班
<input type="checkbox"/>	3017000021	陈理	女	19	2017	软工二班
<input type="checkbox"/>	3017000024	陈立农	女	19	2017	软工二班
<input type="checkbox"/>	3017000026	炎言	男	18	2017	软工二班

(3) 删除。

选中某一列（以上一张图举例），点击左上图标，进行删除操作

127.0.0.1:8080 显示

删除成功

确定

	性别	入学年龄	入学年份
	女	18	1997
	男	17	2018
	男	19	2017
	男	18	2017

删除成功

返回首页

添加

删除

修改

请输入学生姓名或学号

搜索

学生列表					
<input type="checkbox"/> 学号	姓名	性别	入学年龄	入学年份	班级
<input type="checkbox"/> 002	李滢	女	18	1997	中文三班
<input type="checkbox"/> 003	李邱	男	17	2018	软工五班
<input type="checkbox"/> 3017000015	华章宇	男	19	2017	软工二班
<input type="checkbox"/> 3017000017	薛致	男	18	2017	软工二班
<input type="checkbox"/> 3017000018	苏莹	女	19	2017	软工二班
<input type="checkbox"/> 3017000020	马伯干	男	18	2017	软工二班
<input type="checkbox"/> 3017000021	陈理	女	19	2017	软工二班
<input type="checkbox"/> 3017000024	陈立农	女	19	2017	软工二班

列表中不再显示学号为“3017000023”学生的相关信息

注：删除该信息的同时也删除在选课信息中该学号所对应的选课信息。

(4) 修改

可以修改学生的学号、姓名、性别、入学年龄、入学年份、班级。

例：选择学号为“001”的学生，点击修改，弹出对话框（dialog），上一次保存的数据自动显示在文本框中，用户在文本框中修改信息。此处选择修改学号、姓名、班级，点击“确定”保存修改。

127.0.0.1:8080 显示

修改成功!

确定

学号 005

修改成功

返回首页

添加

删除

修改

请输入学生姓名或学号

搜索

学生列表					
<input type="checkbox"/> 学号	姓名	性别	入学年龄	入学年份	班级
<input type="checkbox"/> 002	李滢	女	18	1997	中文三班
<input type="checkbox"/> 003	李邱	男	17	2018	软工五班
<input type="checkbox"/> 005	张七	男	18	2019	软工二班
<input type="checkbox"/> 3017000015	华章宇	男	19	2017	软工二班
<input type="checkbox"/> 3017000017	薛致	男	18	2017	软工二班
<input type="checkbox"/> 3017000018	苏莹	女	19	2017	软工二班

学生列表自动刷新为已修改的信息。

（5）根据学生姓名与学号搜索学生信息。

例：在搜索框中输入：005，点击搜索。

返回首页

添加

删除

修改

请输入学生姓名或学号

005

搜索

学生列表

<input type="checkbox"/>	学号	姓名	性别	入学年龄	入学年份	班级
<input type="checkbox"/>	005	张七	男	18	2019	软工二班

出现搜索结果。

搜索张七同理



返回首页



添加



删除



修改

请输入学生姓名或学号

张七

 搜索

学生列表

<input type="checkbox"/> 学号	姓名	性别	入学年龄	入学年份	班级
<input type="checkbox"/> 005	张七	男	18	2019	软工二班

2、课程信息

返回首页

添加

删除

修改

请输入课程编号或课程名称

搜索

课程列表

<input type="checkbox"/> 课程编号	课程名称	教师名称	学分	适合年级	取消年份
<input type="checkbox"/> 00000012	课程三	刘武	1	2018级	
<input type="checkbox"/> 0000003	计算机基础	张武	1	2018级	
<input type="checkbox"/> 0000006	计算机网络	毕可	3.5	2017级	
<input type="checkbox"/> 0000007	计算机组成原理	李萌	4	2017级	
<input type="checkbox"/> 0000010	课程一	李丽	3	2017级	2018年
<input type="checkbox"/> 0000011	课程二	毕可	3.5	2017级	
<input type="checkbox"/> 001	计算机	张力	2	2018级	2018

（1）展示课程列表。显示课程的编号、名称、教室名称、学分、适合年级、取消年份（其中，取消年份为非必填，其余为必填）

（2）添加

点击左上“添加”的小图标，弹出添加窗口，输入相关信息完成添加操作。

例：输入如下信息

添加课程

课程编号: 0000001

课程名称: 课程五

教师姓名: 李林

学分: 3

适合年级: 2018级

取消年份:

添加课程

点击“添加课程”

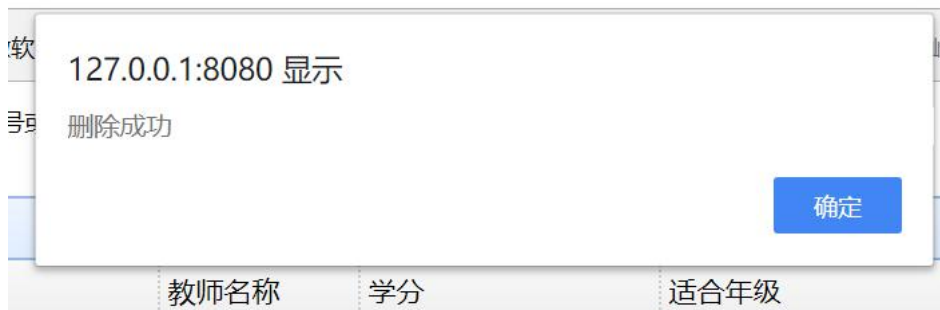
[返回首页](#) [添加](#) [删除](#) [修改](#) 请输入课程编号或课程名称 [搜索](#)

课程列表					
<input type="checkbox"/> 课程编号	课程名称	教师名称	学分	适合年级	取消年份
<input type="checkbox"/> 0000001	课程五	李林	3	2018级	NULL
<input type="checkbox"/> 00000012	课程三	刘武	1	2018级	
<input type="checkbox"/> 0000003	计算机基础	张武	1	2018级	
<input type="checkbox"/> 0000006	计算机网络	毕可	3.5	2017级	

页面自动刷新，出现新添加的内容。

(3) 删除

选中某一行内容，点击删除，删除表及数据库中的相关信息。



[返回首页](#) [添加](#) [删除](#) [修改](#) 请输入课程编号或课程名称 [搜索](#)

课程列表					
<input type="checkbox"/> 课程编号	课程名称	教师名称	学分	适合年级	取消年份
<input type="checkbox"/> 00000012	课程三	刘武	1	2018级	
<input type="checkbox"/> 0000003	计算机基础	张武	1	2018级	
<input type="checkbox"/> 0000006	计算机网络	毕可	3.5	2017级	
<input type="checkbox"/> 0000007	计算机组成原理	李萌	4	2017级	
<input type="checkbox"/> 0000010	课程一	李丽	3	2017级	2018年

课程已删除。

注：删除该课程的同时删除选课信息中该课程名称对应的选课信息。

(4) 修改

点击左上角小图标“修改”，可以修改课程编号、名称、教师姓名、学分、适合年级、取消年份（打开 dialog 时，会自动显示原先的信息）

例：

修改课程编号、名称、取消年份，点击“保存修改”




课程编号

课程名称

教师姓名

学分

适合年级

取消年份

保存修改






<input checked="" type="checkbox"/>	0000005	课程五	刘武	1	2018级	2018
-------------------------------------	---------	-----	----	---	-------	------

信息修改成功。

（5）根据课程名称或课程编号搜索课程信息。






在搜索框中输入课程名称或课程编号，将会显示对应课程的信息。

例：输入课程编号





 请输入课程编号或课程名称


课程列表						
<input type="checkbox"/>	课程编号	课程名称	教师名称	学分	适合年级	取消年份
<input type="checkbox"/>	0000005	课程五	刘武	1	2018级	2018

输入课程名称





 请输入课程编号或课程名称


课程列表						
<input type="checkbox"/>	课程编号	课程名称	教师名称	学分	适合年级	取消年份
<input type="checkbox"/>	0000005	课程五	刘武	1	2018级	2018

3、选课信息

（1）展示选课列表。显示所有选课信息，包括学号、课程编号、选课年份、课程成绩。

 返回首页  添加  删除  修改

选课列表				
<input type="checkbox"/>	学号 ▾	课程编号	选课年份	课程成绩
<input type="checkbox"/>	3017000015	0000005	2017	99
<input type="checkbox"/>	3017000017	0000005	2017	66
<input type="checkbox"/>	3017000018	0000005	2019	78
<input type="checkbox"/>	3017000021	0000005	2016	85
<input type="checkbox"/>	3017000024	0000005	2017	77

(2) 添加

点击左上添加图标，弹出对话框，输入基本信息（所有信息均为必填），点击“添加选课”，完成添加。

例：

 添加选课 

学号：

课程编号：

选课年份：

课程成绩：

添加选课

127.0.0.1:8080 显示

添加成功!

确定

返回首页
 添加
 删除
 修改

选课列表				
<input type="checkbox"/>	学号	课程编号	选课年份	课程成绩
<input type="checkbox"/>	002	0000005	2017	98
<input type="checkbox"/>	3017000015	0000005	2017	99
<input type="checkbox"/>	3017000017	0000005	2017	66
<input type="checkbox"/>	3017000018	0000005	2018	78

系统自动更新

注：无法添加学号不在学生信息列表，或课程编号不在课程信息列表的选课信息

(3) 删除

功能同学生信息和课程信息相同。

(4) 修改

点击左上角“修改”小图标，可以修改学号、课程编号、选课年份、课程成绩。

修改信息

学号:

课程编号:

选课年份:

课程成绩:

保存修改

4、选课查询

(1) 根据学生的姓名或学号，查询该学生的选课情况（学号、姓名、课程编号、课程名称、选课年份、成绩）

例：

根据学生信息		根据课程信息				
返回首页		请输入学生姓名或学号		3017000015		搜索
<input type="checkbox"/> 学号	姓名	课程编号	课程名称	选课年份	成绩	
<input type="checkbox"/> 3017000015	华章宇	0000005	课程五	2017	99	

(2) 根据课程的名或课程编号，查询该课程的选课情况（学号、姓名、课程编号、课程名称、选课年份、成绩）

根据学生信息		根据课程信息				
返回首页		请输入课程名称或编号		0000005		搜索
<input type="checkbox"/> 学号	姓名	课程编号	课程名称	选课年份	成绩	
<input type="checkbox"/> 3017000015	华章宇	0000005	课程五	2017	99	
<input type="checkbox"/> 3017000017	薛致	0000005	课程五	2017	66	
<input type="checkbox"/> 3017000018	苏莹	0000005	课程五	2019	78	
<input type="checkbox"/> 3017000021	陈理	0000005	课程五	2016	85	
<input type="checkbox"/> 3017000024	陈立农	0000005	课程五	2017	77	

例：

5、成绩查询

(1) 输入学生姓名或学号和课程名称或编号，查询该学生此课程的成绩。

并计算出该学生所有课程的平均成绩（若只输入姓名，则计算出该学生所有科目的总平均分）

例：

学生成绩		班级成绩		课程成绩						
返回首页		请输入学号或学生姓名		华章宇		请输入课程编号或课程名称		0000005		搜索
学生总平均分：99										
<input type="checkbox"/> 学号	姓名	课程编号	课程名称	选课年份	成绩					
<input type="checkbox"/> 3017000015	华章宇	0000005	课程五	2017	99					

(2) 输入班级名称和课程编号或课程名称，查询该班级某课程的平均成绩；

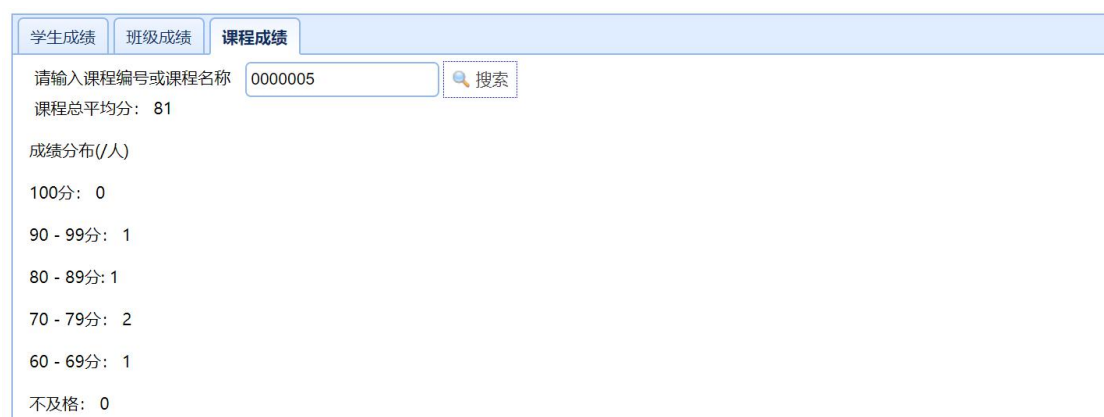
并计算出该班级所有课程的平均成绩（若只输入班级名称，则计算出该班级所有课程的总平均分）

例：



（3）输入课程的编号或名称，计算该课程的总平均分和课程成绩分布

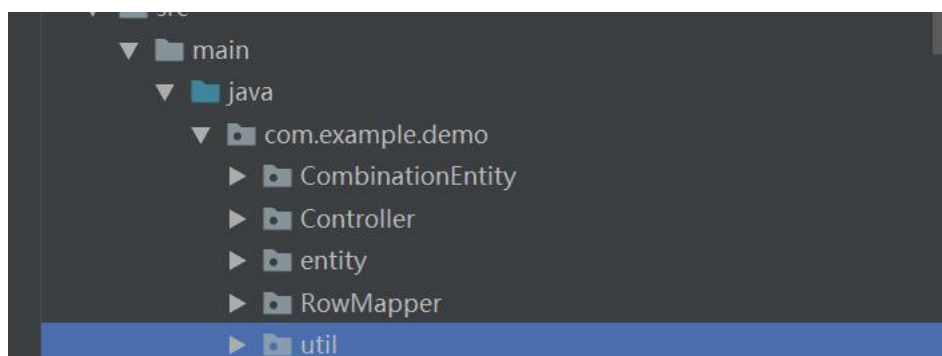
例：

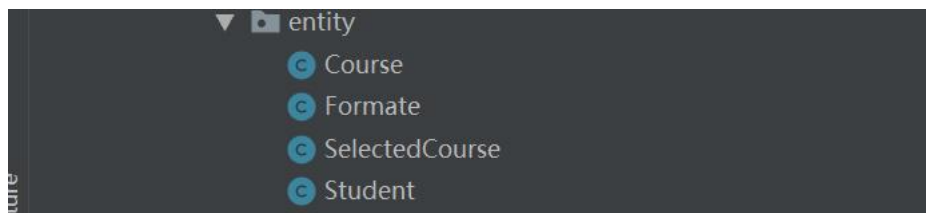


二、代码结构

本系统使用了 **springboot**（后端）+**easyui**（前端）框架，

后端结构：

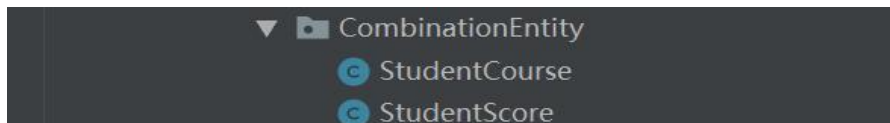




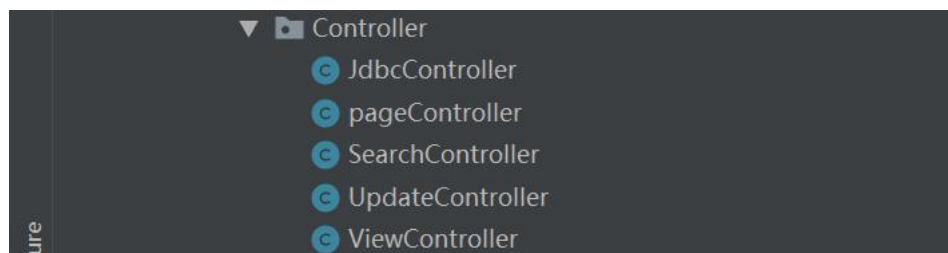
1、entity

实体类，主要存放学生类、课程类、选课类、以及一个用于格式转换的类（用于将数据库查找到的数据转换为 json 格式）

2、CombinationEntity



实体类，类似于 entity，用于存放学生成绩类、学生选课类。



2、Controller

后台控制类，

（1）JdbcController

主要用于数据库连接测试

（2）ViewController

主要用于查找数据库中所有学生/课程/选课的基本信息

例：查找数据库中所有学生信息，并返回前端所需要的 json 格式

```
private JdbcTemplate jdbcTemplate;
//显示所有学生基本信息
@RequestMapping(value = "/StudentView", method = RequestMethod.GET)
@ResponseBody
public Object StudentView () {
    String sql = "select * from student";
    List<Student> studentList = jdbcTemplate.query(sql, new StudentRowMapper());
    Formate<Student> dataformat = new Formate<>();
    dataformat.setTotal(studentList.size());
    dataformat.setRows(studentList);
    return dataformat;
}
```

(3) SearchController

通过输入特定条件，查找对应的学生/课程/选课/成绩信息

例：根据学生学号或姓名，查找对应学生的选课信息，并返回前端所需要的 json 格式

```
//查询选课情况 (根据学生姓名, 学号)
@RequestMapping(value = "/SearchSelectedCourse", method = RequestMethod.GET)
public <value> Object SearchSelectedCourse(@RequestParam(value = "info") String s){
    String sql = "select distinct s.sname, s.sid, c.cid, c.cname, sc.sel_year, sc.score from selecte
        + s + "OR c.cname = '" + s + "'" ) AND s.sid = sc.sid AND c.cid = sc.cid";
    List<StudentCourse> studentCoursesList = jdbcTemplate.query(sql, new StudentCourseRowMapper());
    Formate<StudentCourse> dataformat = new Formate<>();
    dataformat.setTotal(studentCoursesList.size());
    dataformat.setRows(studentCoursesList);
    return dataformat;
}
```

(3) UpdataController

添加、删除、修改等更新操作

例：添加某课程：

```
//插入课程
@RequestMapping(value = "/InsertCourse", method = RequestMethod.GET)
public Object InsertCourse (@RequestParam(value = "cid") String cid,
    @RequestParam(value = "cname") String cname,
    @RequestParam(value = "tname") String tname,
    @RequestParam(value = "credit") String credit,
    @RequestParam(value = "suit_grade") String suit_grade,
    @RequestParam(value = "canc_year",required = false, defaultValue = "NULL") String cance_year) {
    Course course = new Course();
    String sql = "INSERT INTO course (cid, cname, tname, credit, suit_grade, cance_year) VALUES (?, ?, ?, ?, ?, ?)";
    jdbcTemplate.update(sql, cid, cname, tname, credit, suit_grade, cance_year);
    return "插入成功";
}
```

(4) pageController

与前端交互实现页面之间的跳转

```
@Controller
@RequestMapping("/page")
public class pageController {

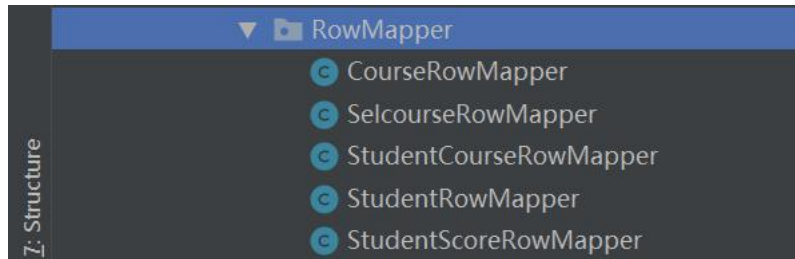
    @RequestMapping(value = "/user")
    public String user () { return "user"; }

    @RequestMapping(value = "/student")
    public String student() { return "student"; }

    @RequestMapping(value = "/search")
    public String search() { return "search"; }

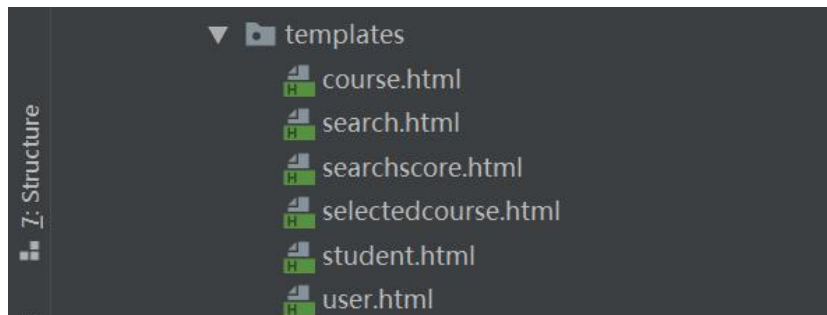
    @RequestMapping(value = "/searchscore")
    public String searchscore() { return "searchscore"; }
}
```

3、RowMapper



主要用于实现 RowMpper 接口。

前端结构：



编写前端页面