**中华女子学院计算机系**

课程实训报告

2019 – 2020 学年 第 1 学期

**学生综合测评系统设计**

**课程名称： 数据库设计与应用 综合实训**

**学 号： 170811008**

**姓 名： 于越**

**日 期： 2019年11月10日**

**成 绩：**

摘要

综合测评系统现今已被广泛应用于高校教学，其一般设有学生和管理员两个角色。其中，学生可以申报并管理已申请的综测加分项，管理员可以管理学生、课程、成绩、评分标准信息，审核学生申报的加分项和汇总综测总体情况等。为了能够更好地实现以上功能，本课程实训项目结合《数据库原理》和《数据库应用开发》课程所学知识，根据实际需求，通过使用数据库建模工具对数据库进行概念和逻辑结构的设计，应用SQL语言和数据库编程实现主要业务功能，使用Java语言对数据库进行操作。本课程实训项目目前可利用JDBC API实现学生加减分项的申请和管理员对学生信息的管理功能，以及创建数据库对象存储过程、游标和事务实现管理员审核学生加减分项申请和汇总总体综测情况的功能。

关键词：综合测评系统，数据库应用开发，存储过程，事务，游标

**Abstract**

Comprehensive evaluation system has been widely used in college teaching, and it generally has two roles: student and administrator. Among them, students can declare and manage the points applied, administrators can manage the information of students, courses, grades and scoring standards, review the points declared by students and summarize the overall situation of the comprehensive test, etc. In order to better achieve the above functions, this course training program combined with the curriculum knowledge in database principle and database application development, according to the actual demand, through the use of the database modeling tool to design the concept and logic structure of database, application of SQL language and database programming to achieve the main business functions, use of Java language to operate in a database. The training program of this course can now utilize the JDBC API to realize the application of applying the points for students and the administrator's management function of student information, as well as the function of creating database object which are stored procedures, cursors and transactions to make administrator be able to review the students’ application and summarize the results.

**Keywords:** Comprehensive evaluation system，database application development

目录

[摘要 I](#_Toc24446572)

[Abstract I](#_Toc24446573)

[1. 系统需求分析 1](#_Toc24446574)

[1.1学生的功能需求分析 1](#_Toc24446575)

[1.1.1申请加分项 1](#_Toc24446576)

[（1）选择申请的类别与加分项明细 1](#_Toc24446577)

[（2）提交分数与说明 1](#_Toc24446578)

[1.1.2查看已申请的加分项 1](#_Toc24446579)

[（1）修改已申请加分项 1](#_Toc24446580)

[（2）删除已申请加分项 2](#_Toc24446581)

[1.2管理员的功能需求分析 2](#_Toc24446582)

[1.2.1添加学生信息 2](#_Toc24446583)

[（1）删除学生信息 2](#_Toc24446584)

[（2）修改学生信息 2](#_Toc24446585)

[（3）查询学生信息 3](#_Toc24446586)

[1.2.2添加课程信息 3](#_Toc24446587)

[（1）删除课程信息 3](#_Toc24446588)

[（2）修改课程信息 3](#_Toc24446589)

[（3）查询课程信息 3](#_Toc24446590)

[1.2.3添加学生成绩 3](#_Toc24446591)

[（1）删除某学生某科信息 3](#_Toc24446592)

[（2）修改某学生某科成绩 3](#_Toc24446593)

[（3）查询学生某科成绩 3](#_Toc24446594)

[1.2.4浏览学生申报的加分项 3](#_Toc24446595)

[（1）审核加分项 4](#_Toc24446596)

[1.2.5浏览综测评分标准 4](#_Toc24446597)

[（1）添加评分标准 4](#_Toc24446598)

[（2）修改评分标准 4](#_Toc24446599)

[（3）删除评分标准 4](#_Toc24446600)

[1.2.6浏览学生综测情况 4](#_Toc24446601)

[2. 数据库概念结构设计 4](#_Toc24446602)

[3. 数据库逻辑结构设计 6](#_Toc24446603)

[3.1数据库的PDM模型 6](#_Toc24446604)

[3.2由PDM得到关系模式 7](#_Toc24446605)

[3.3根据应用需求对关系模式进行优化 7](#_Toc24446606)

[4. 创建数据库 11](#_Toc24446607)

[4.1建立数据库 11](#_Toc24446608)

[4.2建立数据表 11](#_Toc24446609)

[4.3生成数据库关系图 11](#_Toc24446610)

[5. 系统功能实现 12](#_Toc24446611)

[5.1数据库编程 12](#_Toc24446612)

[5.1.1 事务：实现管理员对学生已申请加分项进行审核的操作 12](#_Toc24446613)

[5.1.2 游标：实现学生智育测评中学习成绩的计算 12](#_Toc24446614)

[5.1.3 存储过程：实现各类加减分项总分的计算 12](#_Toc24446615)

[5.2 Java编程 13](#_Toc24446616)

[5.2.1 学生申报加分项 13](#_Toc24446617)

[5.2.2 学生查看已申报的加分项 13](#_Toc24446618)

[6. 总结 13](#_Toc24446619)

[6.1 设计过程中遇到的问题及解决方式 13](#_Toc24446620)

[6.2 设计难点和亮点 14](#_Toc24446621)

[6.3 需改进和完善之处 14](#_Toc24446622)

[7. 参考文献 14](#_Toc24446623)

1. 系统需求分析

用户在打开综合测评系统的页面后，可以通过选择用户身份、输入学号/工号进入学生综合测评系统。如果系统中没有该用户的信息，则提示为无效账号，无法进入系统。

1.1学生的功能需求分析

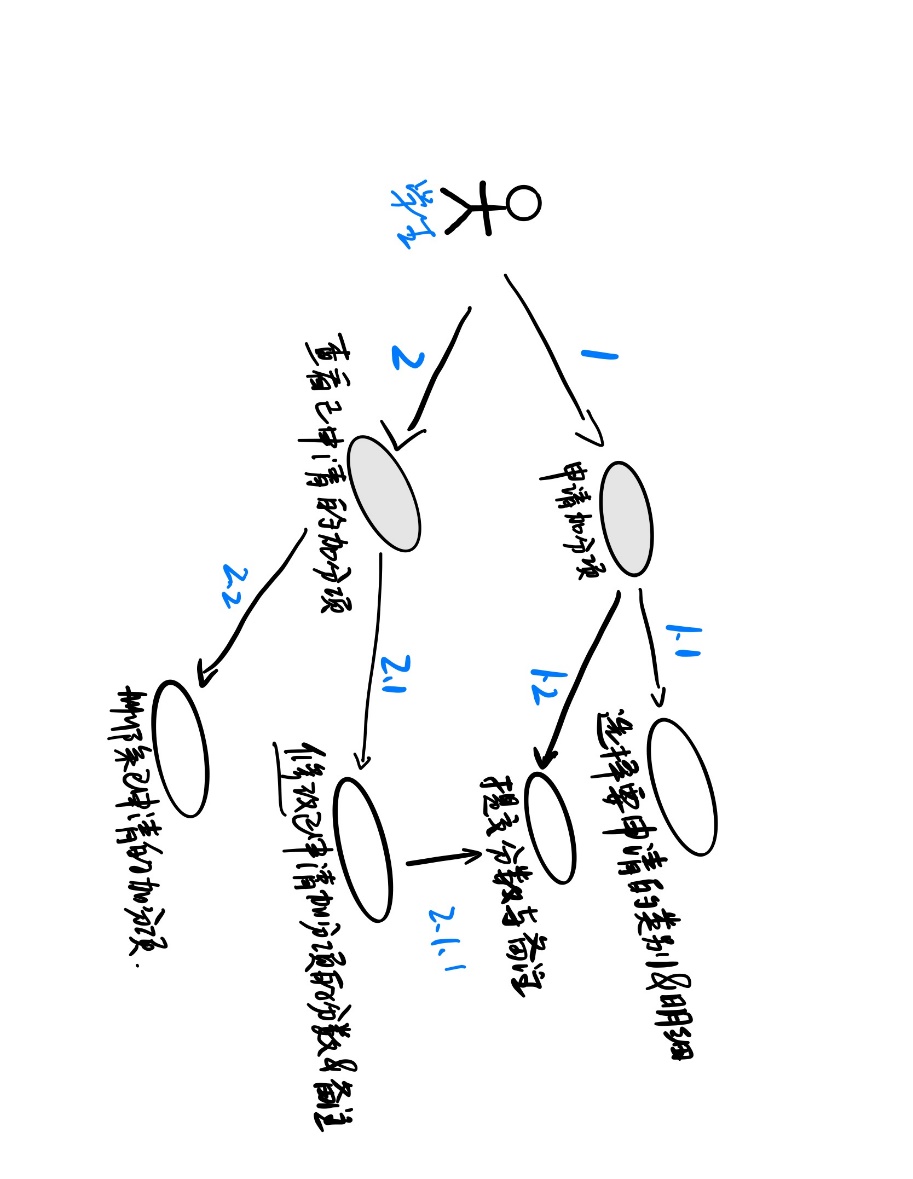


图1.1 学生的功能需求分析

1.1.1申请加分项

（1）选择申请的类别与加分项明细

浏览各类别名称，选择要申请的类别；根据选择的类别，浏览该类别各加分项的名称，选择要申请的加分项名称。

（2）提交分数与说明

浏览该加分项的具体信息，输入申请分数与说明。若申请分数超出上下限，则无法提交；若申请分数不是数字，也无法提交。

1.1.2查看已申请的加分项

按年份、类别查看已申请的加分项。

（1）修改已申请加分项

选择要修改的加分项类别与名称序号，显示待修改的加分项各项信息；选择要修改的项目，输入要修改的内容并提交。

（2）删除已申请加分项

选择要删除的加分项类别与名称序号，删除并提交。

1.2管理员的功能需求分析

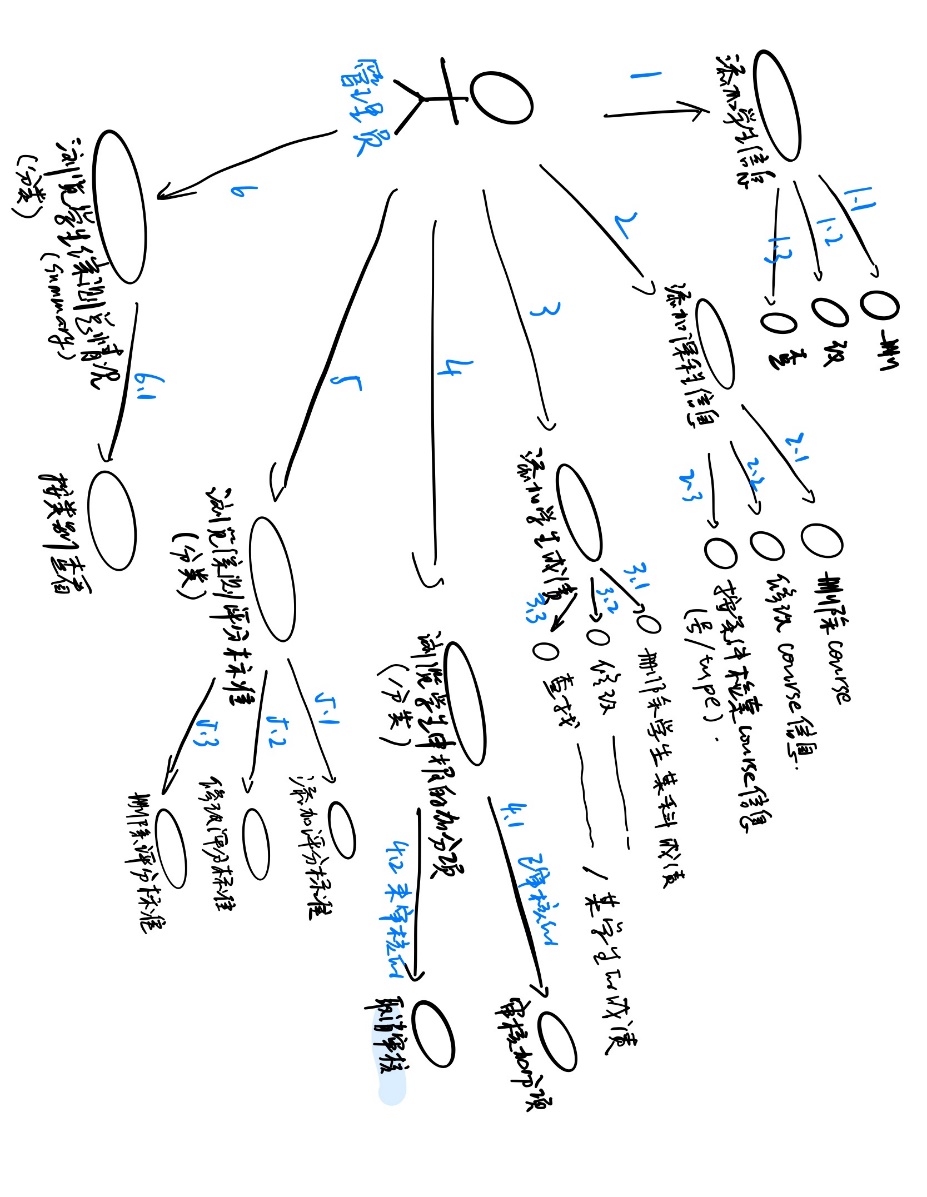


图1.2 管理员的功能需求分析

1.2.1添加学生信息

可按照“学号，姓名，班级，系别专业，属性”的格式（批量）添加学生信息。

（1）删除学生信息

输入并提交需要删除学生信息的学号。

（2）修改学生信息

输入需要修改的学生学号，显示该学生的各项信息；选择需要修改的项目，输入更改的内容并提交。

（3）查询学生信息

可按照学号、姓名、班级或系别专业等条件查询学生信息。

1.2.2添加课程信息

可按照“学号，姓名，班级，系别专业，属性” 的格式（批量）添加课程信息。

（1）删除课程信息

输入并提交需要删除课程信息的课程号。

（2）修改课程信息

输入需要修改的课程号，显示该课程各项信息；选择需要修改的项目，输入更改的内容并提交。

（3）查询课程信息

可按照课程号、课程名、课程分数或课程类型条件查询课程信息。

1.2.3添加学生成绩

选择年份，输入学号、课程号和成绩，系统将对输入信息进行判断，即所输入学号及课程号应分别存在于学生表和课程表中，所输入成绩值应在0~100之间。

（1）删除某学生某科信息

输入该学生学号，显示该学生所有科目成绩，选择删除该学生所有科目成绩或某一科成绩，若只删除某一科成绩，需要输入该科目的课程号。

（2）修改某学生某科成绩

输入该学生学号，显示该学生的所有科目成绩；输入并提交该学生需要修改科目的课程号和成绩。

（3）查询学生某科成绩

输入需要查询的学号或课程号，显示该学生所有科目的成绩或该课程所有学生的成绩，成绩默认按照学号/课程号进行排序。

1.2.4浏览学生申报的加分项

选择年份和班级，浏览该年份该班级所有学生申报的加分项。默认按照学号、类别、加分项序号逐级进行排序。

（1）审核加分项

在按年份和班级浏览学生申报的加分项的基础之上，筛选未审核的加分项，逐条显示学生所申请的加分项的具体内容，进行审核。如果学生申请分数合理，则予以通过，选择并审核下一条记录；如果学生申请分数分值不合理，管理员可以修改该分数并予以通过，修改后的分数即为管理员的审核分数，选择显示下一条记录；如果学生申请说明不合理，则不予以通过，选择显示下一条记录。

1.2.5浏览综测评分标准

按照类别浏览综合测评所有评分标准，即评分标准的编号、加分项名称和分值范围。

（1）添加评分标准

选择该评分标准所属类别，输入加分项名称、分值上限和下限；若最高分无上限，则无需输入。

（2）修改评分标准

选择需要修改的评分标准所属类别，浏览该类别所有的评分标准，选择要修改的评分标准编号级项目，输入更改的内容并提交。

（3）删除评分标准

选择需要删除的评分标准所属类别，浏览该类别所有的评分标准，输入要删除的评分标准的评分标准编号，进行删除。

1.2.6浏览学生综测情况

可按照年份和班级浏览学生综测总体情况，也可以在此基础上分类查看学生各个分类的评分情况。

1. 数据库概念结构设计



图2.1 数据库概念结构设计图

模型中联系类型（基数）的语义：

学生与测评总表之间的联系类型为：一对一。即一个学生只拥有一条综测总评记录，而一条综测总评记录也一定属于且仅属于一个学生。

学生与加减分明细表之间的联系类型为：一对多。即一个学生可以申请多项加减分项，而一条申请记录一定属于且仅属于一个学生。

加减分明细表与体育得分表之间的联系类型为：多对一。即一条体育得分由多条加分项记录组成，且多条加分项记录组成且仅组成一条体育得分记录。

加减分明细表与德育得分表之间的联系类型为：多对一。即一条德育得分由多条加减分项记录组成，且多条加减分项记录组成且仅组成一条德育得分记录。

加减分明细表与智育得分表之间的联系类型为：多对一。即一条智育得分由多条加分项记录组成，且多条加分项记录组成且仅组成一条智育得分记录。

加减分名称表与加减分明细表之间的联系类型为：一对多。即一条加减分名称记录对应多条加减分项记录，而一条加减分项记录属于且仅属于一条加减分名称记录。

智育得分表与学习成绩总分表之间的联系类型为：一对一。即一条学习成绩总分记录组成一条智育得分记录，而一条智育得分仅对应一条学习成绩总分记录。

学习成绩总分表与学生各科成绩表之间的联系为：一对多。即一条学习成绩记录总分由多条学科成绩记录组成，而一个学生多条学科成绩记录仅组成一条学习成绩总分记录。

学生各科成绩表与课程表之间的联系为：多对一。即一条学科成绩记录只对应一门课程，而一门课程可以有多条成绩记录记录。

学生表与学生各科成绩表之间的联系为：一对多。即一个学生可以拥有多条成绩记录，而一条成绩记录一定属于且仅属于一个学生。

3. 数据库逻辑结构设计

3.1数据库的PDM模型



图3.1 数据库逻辑结构设计图

3.2由PDM得到关系模式

student(**s\_no**,s\_name,s\_class,s\_pro,s\_feature)

department(**d\_no**,department,major)

course(**no**, year,s\_no,s\_me,s\_ie,s\_pe,s\_total,s\_grade)

moral(**no**, year,s\_no,me\_basic,me\_extra,me\_minus,me\_total)

Intellectual(**no**, year,s\_no,s\_studyscore,ie\_extra,ie\_total)

physical(**no**, year,s\_no,pe\_basic,pe\_test,pe\_extra,pe\_total)

extra\_name(no, extra, detail, name, max, min)

extra(**no**, year,s\_no,extra,detail,remarks,s\_point,g\_point,status,pass)

grade\_by\_course(**no**, year,s\_no,crs\_no,crs\_grade)

grade(**no**,yar,s\_no,s\_test\_total,s\_test\_avg,s\_exam\_total,t\_exam\_avg,s\_weight\_mean)

3.3根据应用需求对关系模式进行优化

学生表(student)的设计结果如表3.1所示。

表3.1 学生表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名 | 含义 | 数据类型 | 长度 | 允许空 | 约束 | 备注 |
| s\_no | 学号 | 字符串 | 10 | 否 | 主码 |  |
| s\_name | 姓名 | 字符串 | 10 | 否 |  |  |
| s\_class | 班级 | 字符串 | 5 | 否 |  |  |
| s\_pro | 系别专业 | 字符串 | 6 | 否 |  |  |
| s\_feature | 属性 | 字符串 | 50 | 否 |  |  |

系部表(department)的设计结果如表3.2所示。

表3.2 系部表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名 | 含义 | 数据类型 | 长度 | 允许空 | 约束 | 备注 |
| d\_no | 编号 | 字符串 | 10 | 否 | 主码 |  |
| department | 系别/学院 | 字符串 | 25 | 否 |  |  |
| major | 专业 | 字符串 | 25 | 否 |  |  |

课程表(course)的设计结果如表3.3所示。

表3.3 课程表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名 | 含义 | 数据类型 | 长度 | 允许空 | 约束 | 备注 |
| crs\_no | 课程号 | 字符串 | 15 | 否 | 主码 |  |
| crs\_name | 课程名 | 字符串 | 20 | 否 |  |  |
| crs\_point | 课程学分 | 字符串 | 5 | 否 |  |  |
| crs\_type | 课程类型 | 字符串 | 5 | 否 |  |  |

测评总表(summary)的设计结果如表3.4所示。

表3.4 测评总表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名 | 含义 | 数据类型 | 长度 | 允许空 | 约束 | 备注 |
| no | 序号 | 自增整型 |  | 否 | 主码 |  |
| year | 年份 | 字符串 | 10 | 否 |  |  |
| s\_no | 学号 | 字符串 | 10 | 否 | 外码 | 参照于学生表 |
| s\_me | 德育 | 字符串 | 5 | 否 |  |  |
| s\_ie | 智育 | 字符串 | 5 | 否 |  |  |
| s\_pe | 体育 | 字符串 | 5 | 否 |  |  |
| s\_total | 总分 | 字符串 | 5 | 否 |  |  |
| s\_grade | 学习成绩 | 字符串 | 5 | 否 |  |  |

德育表(moral)的设计结果如表3.5所示。

表3.5 德育表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名 | 含义 | 数据类型 | 长度 | 允许空 | 约束 | 备注 |
| no | 序号 | 自增整型 |  | 否 | 主码 |  |
| year | 年份 | 字符串 | 10 | 否 |  |  |
| s\_no | 学号 | 字符串 | 10 | 否 | 外码 | 参照于extra表 |
| me\_basic | 基础分 | 整型 |  | 否 |  |  |
| me\_extra | 加分项总分 | 字符串 | 5 | 否 |  |  |
| me\_minus | 减分项总分 | 字符串 | 5 | 否 |  |  |
| me\_total | 总分 | 字符串 | 5 | 否 |  |  |

智育表(intellectual)的设计结果如表3.6所示。

表3.6 智育表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名 | 含义 | 数据类型 | 长度 | 允许空 | 约束 | 备注 |
| no | 序号 | 自增整型 |  | 否 | 主码 |  |
| year | 年份 | 字符串 | 10 | 否 |  |  |
| s\_no | 学号 | 字符串 | 10 | 否 | 外码 | 参照于extra表 |
| s\_studyscore | 学习成绩 | 字符串 | 5 | 否 |  |  |
| ie\_extra | 学术成果加分 | 字符串 | 5 | 否 |  |  |
| ie\_total | 总分 | 字符串 | 5 | 否 |  |  |

体育表(physical)的设计结果如表3.7所示。

表3.7 体育表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名 | 含义 | 数据类型 | 长度 | 允许空 | 约束 | 备注 |
| no | 序号 | 自增整型 |  | 否 | 主码 |  |
| year | 年份 | 字符串 | 10 | 否 |  |  |
| s\_no | 学号 | 字符串 | 10 | 否 | 外码 | 参照于extra表 |
| pe\_basic | 基础分 | 整型 |  | 否 |  |  |
| pe\_test | 体质健康测量 | 字符串 | 5 | 否 |  |  |
| pe\_extra | 体育加分项 | 字符串 | 5 | 否 |  |  |
| pe\_total | 总分 | 字符串 | 5 | 否 |  |  |

加减分名称表(extra\_name)的设计结果如表3.8所示。

表3.8 加减分名称表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名 | 含义 | 数据类型 | 长度 | 允许空 | 约束 | 备注 |
| no | 序号 | 自增整型 |  | 否 | 主码 |  |
| extra | 类别 | 字符串 | 10 | 否 |  |  |
| detail | 加分项明细 | 字符串 | 2 | 否 |  |  |
| name | 加分项名称 | 字符串 | 50 | 否 |  |  |
| max | 最高分 | 字符串 | 5 | 允许 |  |  |
| min | 最低分 | 字符串 | 5 | 否 |  |  |

加分项明细表(extra)的设计结果如表3.9所示。

表3.9 加分项明细表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名 | 含义 | 数据类型 | 长度 | 允许空 | 约束 | 备注 |
| no | 序号 | 自增整型 |  | 否 | 主码 |  |
| year | 年份 | 字符串 | 10 | 否 |  |  |
| s\_no | 学号 | 字符串 | 10 | 否 | 外码 | 参照于学生表 |
| extra | 类别 | 字符串 | 10 | 否 | 外码 | 参照于加减分名称表 |
| detail | 加分项明细 | 字符串 | 2 | 否 |  |  |
| remarks | 备注 | 文本 |  | 允许 |  |  |
| s\_point | 申请分数 | 字符串 | 5 | 否 |  |  |
| g\_point | 审查小组通过分数 | 字符串 | 5 | 否 |  |  |
| status | 审核状态 | 字符串 | 2 | 否 |  |  |
| pass | 通过状态 | 字符串 | 2 | 否 |  |  |

学生各科成绩表(grade\_by\_course)的设计结果如表3.10所示。

表3.10 学生各科成绩表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名 | 含义 | 数据类型 | 长度 | 允许空 | 约束 | 备注 |
| no | 序号 | 自增整型 |  | 否 | 主码 |  |
| year | 年份 | 字符串 | 10 | 否 |  |  |
| s\_no | 学号 | 字符串 | 10 | 否 | 外码 | 参照于学生表 |
| crs\_no | 课程号 | 字符串 | 15 | 否 | 外码 | 参照于课程表 |
| crs\_grade | 成绩 | 字符串 | 5 | 否 |  |  |

学习成绩总表(grade)的设计结果如表3.11所示。

表3.11 学习成绩总表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名 | 含义 | 数据类型 | 长度 | 允许空 | 约束 | 备注 |
| no | 序号 | 自增整型 |  | 否 | 主码 |  |
| year | 年份 | 字符串 | 10 | 否 |  |  |
| s\_no | 学号 | 字符串 | 10 | 否 | 外码 | 参照于学生表 |
| s\_test\_total | 考试课总分 | 字符串 | 5 | 否 |  |  |
| s\_test\_avg | 考试课平均分 | 字符串 | 5 | 否 |  |  |
| s\_exam\_total | 考察课总分 | 字符串 | 5 | 否 |  |  |
| s\_exam\_avg | 考查课平均分 | 字符串 | 5 | 否 |  |  |
| s\_weight\_mean | 分值权重 | 字符串 | 5 | 否 |  |  |

4. 创建数据库

4.1建立数据库

创建数据库scoring\_system。

4.2建立数据表

加载ScoringSystem.sql文件。

4.3生成数据库关系图

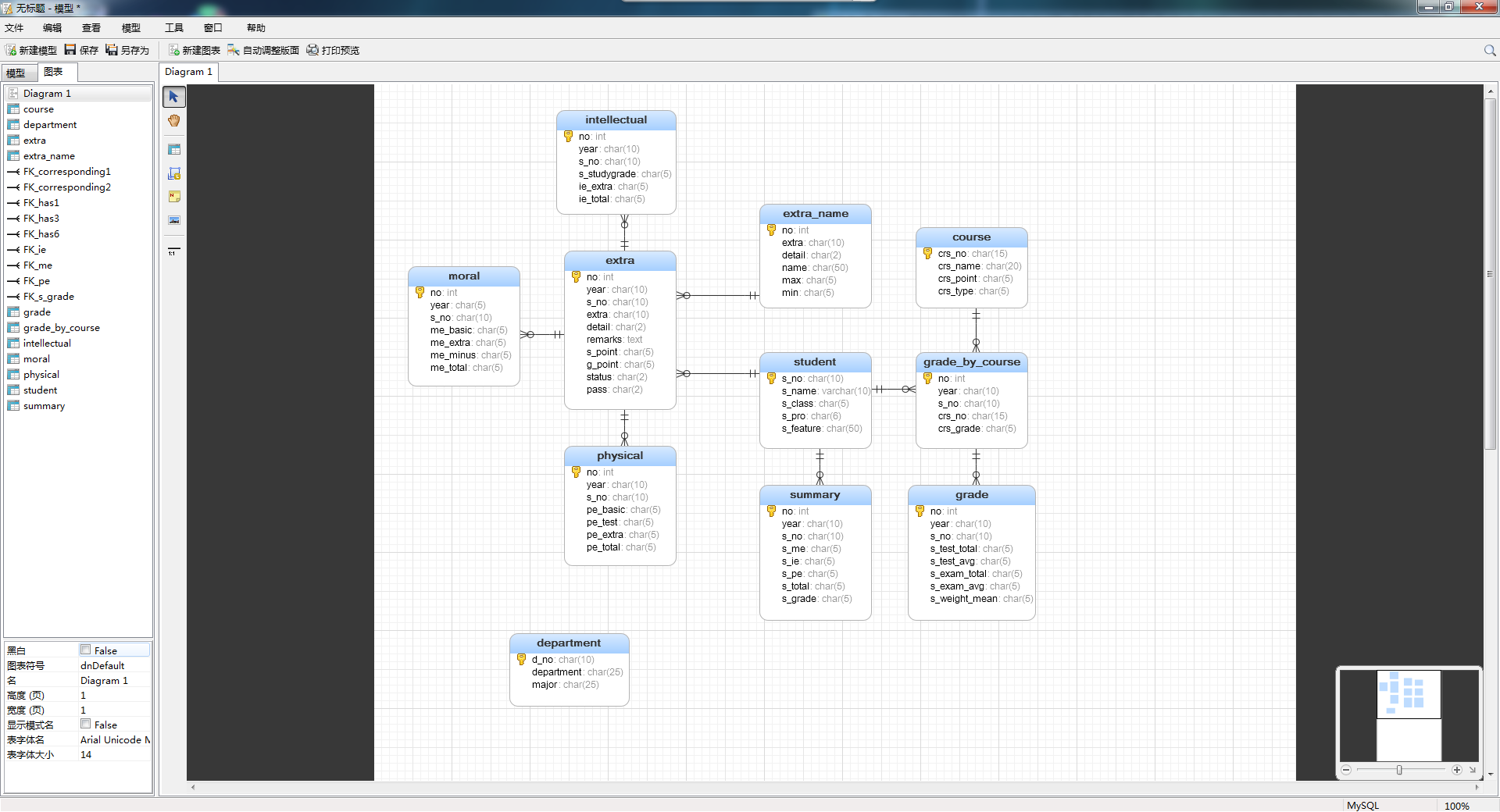


图4.1 数据库关系图

5. 系统功能实现

5.1数据库编程

5.1.1 事务：实现管理员对学生已申请加分项进行审核的操作

proc\_admin\_check\_one\_extra()存储过程中，在获得到管理员用户对该条加分项纪录的审核操作后，判断并修改该条记录的通过状态、更新该条记录的审核分数，修改该条记录的审核状态是一个整体操作，如果其中任意一条update语句执行失败，则所有的update语句都应该撤销，从而确保该条记录审核状态的真实情况。

5.1.2 游标：实现学生智育测评中学习成绩的计算

创建名为proc\_admin\_collect\_studygrade()的存储过程，该存储过程接收学生学号作为输入参数，该参数决定了计算的是哪位同学的学习成绩，返回学生学习成绩作为输出参数。存储过程中声明了一个名为sg\_cursor的游标。select结果集是该学生学科成绩与该科目对应的学分，通过fetch语句提取其中数据，计算各科成绩与对应学分乘积之和以及学分之和，从而得出该学生的学习成绩。

5.1.3 存储过程：实现各类加减分项总分的计算

创建名为proc\_admin\_collect\_me\_etr()的存储过程，该存储过程接收学生学号、加减分项类别作为输入参数，这两个参数决定了计算的是哪位同学哪个类别的加减分项总和，返回加减分项总分为输出变量。在该存储过程中，使用select语句从加减分项明细表中，先筛选出该学生该类别下已审核并通过的记录，再对其审核分数进行求和。

5.2 Java编程

5.2.1 学生申报加分项

在Stu类中，将包含学生用户基本信息以及其所选择的加减分项类别与名称，所填写的申请分数与说明等信息的Extra类对象作为输入参数，调用stuDeclareDAO类中的insertExtra()方法将该条加分项信息插入到数据库中的加减分项明细表中。该方法通过JDBC API，调用了数据库中的proc\_insert\_extra()存储过程，如果插入成功，则能得到标记成功的返回数值。

5.2.2 学生查看已申报的加分项

在Stu类中，将学生用户基本信息对象，类别名称作为输入参数的s\_getExtrasBy()方法实现了按类别查看该学生已申报的加分项操作。该方法中，调用了stuDeclareDAO类中的queryExtrasBy()方法，返回了一个符合查看条件，包含加减分项基本信息的Map对象。queryExtrasBy()方法通过JDBC API，调用的是数据中的proc\_get\_extras\_by\_n\_extra()存储过程。

6. 总结

6.1 设计过程中遇到的问题及解决方式

1、德育、智育和体育评分的组成结构略有不同，若将各项评分设计为一个表，则表的结构会比较复杂。因此根据各个评分的组成特性，设计符合的表结构，建立德育、智育和体育评分表。

2、为了确保德育、智育和体育评分表内的记录必须是由加减分明细表中所对应的记录自动生成的，因此设置德育、智育和体育评分表中学生学号为外键，即评分表中记录的学生学号，必须在加减分表中有申请记录。

6.2 设计难点和亮点

1、学生申请加分项时，根据学生选择的类别获取对应的加分项名称。

学生每申请一个加分项，都会生成一个Extra类对象，根据学生选择获取该对象的extra属性值，并将其作为调用stuDeclareDAO类的queryDetails()方法的输入参数。该方法通过JDBC API调用proc\_get\_extra\_detail\_name()存储过程，其返回的是Map对象。这个对象的键为String类型，代表的是加分项的编号，而其对应的值则是ExtraName类型，代表的是该项加分项的所有基本信息。之所以这样设计，是为了后续学生用户在申请加分项时能过对其申请的分数是否在分值范围内进行判断，以及能够将该项加分项的所有信息一次性全部插入到数据库对应的表中，而无需再连接数据库获取所需的数据。

2、实现管理员汇总总体综测情况功能时，数据库对象的设计。

总体综测情况由德育、智育、德育各项评分之和组成。创建存储过程proc\_admin\_collect\_summary()，其输入参数为年份和学生学号。在该存储过程中，分别调用计算德育、智育和德育评分的存储过程，求和并将各项评分和学生基本信息添加到测评总表中。创建存储过程proc\_admin\_collect()，其中使用了游标，对按照年份和班级筛选出来的学生学号结果集的每一条记录调用proc\_admin\_collect\_summary()存储过程，生成该学生的综测情况。

6.3 需改进和完善之处

1、程序各功能之间的限制关系还不够完善。由于数据库的设计，无法通过约束实现在存有未被审核的学生加分项申请记录时，管理员无法修改评分标准中加分项名称的功能。应该在管理员想要对加分项名称进行修改时，通过数据库对象判断加减项明细表是否存有该项加分项的申请记录。

2、管理员管理课程信息、学生成绩信息和综测评分标准信息的功能还需进一步完善。

7. 参考文献

[1] 孔祥盛.MySQL数据库基础与实例教程[M].北京：人民邮电出版社，2014