****

《数据库》

课程设计报告书

**题目： 教务管理系统**

**学 院 计算机科学与工程学院**

**专 业 计算机科学与技术1班**

**学生姓名**

**学生学号**

**指导教师**

**课程编号**

**课程学分**

**起始日期**

|  |  |
| --- | --- |
| 教  师  评  语 | 教师签名：  日期： |
| 成  绩  评  定 |  |
| 备  注 |  |

**教务管理系统**

一、选题背景

教务管理系统是基本每个高校都有的一个系统，教务系统管理系统充分利用互联网络B/S管理系统模式，以网络为平台，为各个学校教务系统的管理提供一个平台，帮助学校管理教务，用一个账号解决学校教务教学管理，并且学校可以自由选择学校需要的教务管理系统，灵活地定制符合学校自己实际情况的教务系统。

二、方案论证(设计理念)

由于我们设计的是教务管理系统，虽然我们是使用ORACLE数据库来存储系统的各种信息，但是对于前端界面的设计和显示都需要预先知道一定的处理数据量，这样才比较好安排各种界面的交互以及展示。所以我们首先要知道需要处理的数据量到低有多大，所以需要一定的前期调研。由于教务系统面向的主要对象是学生（查课表查成绩等）和老师（录成绩查名单查课表等），所以我们重点调研一个学校中老师，学生的人数，以及学院和专业的人数。

以我们学校（华南理工大学）为例，通过查阅华南理工大学2017年《本科招生资讯册》.pdf，和《华南理工大学2017年本科招生专业情况总表》，发现我们学校2017年招生的学院数是26个学院，招收专业数是101个专业。招收的总学生人数大概在6300人左右，那么这里我们可以估计本科4个年级的总人数6300x4=25200人（在每年的招生计划变化都不大的情况下）。

再仔细考察招生专业情况表，发现每个专业所招的人数各不相同，比如，招收人数最多的专业“软件工程”会招230人，招收人数最少的专业“音乐学”只招8个人。可见，每个专业招收的人数各不相同，并且，招收人数比较多的（大于60~70人）的专业很可能会分成2个或多个班级。

再考虑我们学校老师的人数，每个学院老师的人数也是各不相同。这里我没有想到很好的办法去获取我们学校每个学院的老师人数，只能通过查阅每个学院的官网去获取老师的人数，先以我的学院（计算机科学与工程学院）为例，我们学院一共有93名老师，这个人数应该属于比较普遍的人数的水平。然后我查到的最多老师的学院是322人，是机械与汽车工程学院。其它的学院的教职工人数普遍在90~120的区间。

得到了这些关于总数统计的数据，我们能更加科学和结合实际情况来设计如何去组织这些数据，从而使我们的系统更加科学。

在本程序的设计里面，我们选择了用面向对象进行编写程序，我们编写的教务管理系统里面有如下的类：

**1．学生类涉及的信息包括：**

①基本信息（ID（学号）、姓名、性别、登录密码、身份证号、专业、联系方式）；

②学生对应本专业的课表；

③个人成绩表；

④关于该学生年级和专业的通知展示；

**2.老师类涉及的信息包括：**

①基本信息（ID（工号）、姓名、性别、登录密码、身份证号、任教科目、联系方式）；

②上课安排表；

③所教专业班级和学生的名单；

④所教专业班级和学生的成绩单；

⑤关于这位老师的通知的展示；

⑥给任教的班级的所有学生发布通知；

**3.学校管理人员类涉及的信息包括：**

1. 基本信息（ID（帐号）、姓名、性别、登录密码、身份证号、学院、联系方式）；
2. 所有学生、老师的信息表；
3. 整个学校的课程列表；
4. 所有专业的课程表；
5. 各专业班级的GPA（绩点）表。
6. 所有学院和专业列表；
7. 通知表；

**4.用户登录类涉及的信息包括：**

里面没有任何信息，只是用作登录界面的代码编写。

面向类对象设计这个程序能够使我们更好地分类编写程序，避免重复编写同样功能的函数。

下面简述不同类实现的功能：

**（1）对于整个程序的login类：**

①提供登录入口；

②每个类型的人可以凭ID和密码登录；

③提供通过验证身份证号和预留的手机号来重置帐号密码的功能；

**（2）对于Student类：**

①提供个人课表查询和课表导出；

②成绩查询和成绩单导出（单科排名等）；

③开始界面展示和该名学生相关的通知；

**（3）对于Teacher类：**

①提供各教学班级的上课节次安排；

②所教各专业班级学生名单的查询与导出；

③所教班级成绩的设置与导出，成绩的处理（最高分、最低分、平均分、及格率等）；

④开始界面展示和这位老师相关的通知；

⑤给自己的任教班级的学生发通知；

**（4）对于学校管理人员类：**

后台函数：

通过所有学生的个人成绩单，生成不同专业的GPA文件。

前台界面函数：

1. 用户数据管理：
   1. 通过一个外部特定格式的csv（文本）文件来初始化系统的二进制数据库，并且提供
   2. 添加一名新用户，用户类型可选学生、老师或管理员。
   3. 通过特定关键字搜索用户，可以对搜索出来的结果执行查看，删除或者修改；
2. 课程管理：
   1. 通过一个外部特定格式的csv导入课程列表；
   2. 通过输入关键字或者留空搜索栏选择查看所有课程或者查看特定的课程，并可以对搜索结果执行删除操作；
3. 学生课表管理：
   1. 通过某个文件夹下面放置的所有专业的课程表csv文件，来导入全部学生的课程表；
   2. 可以按年级+专业分类查看课程表，可以通过搜索框搜索特定的年级或者专业查看指定的课程表；
4. 学生绩点及成绩表管理：
   1. 可以生成、查看所有学生的绩点（GPA），可以清空所有学生的成绩信息；
5. 学院管理：
   1. 可以添加单条学院和专业的记录；
   2. 可以通过一个外部的csv文件来批量添加学院和专业；
   3. 可以查询、修改、删除某条学院专业记录；
6. 通知管理：
   1. 管理员可以选择发送通知的对象（全体成员，某个学院的老师，某个年级某个专业的学生）；
   2. 可以查询和删除通知；

另外，每个用户的界面的最后一个Tab都是修改密码的界面，用户可以通过验证旧密码的方式来修改密码。

为了方便用户打印查看的数据，我们队每一张表格的旁边都设计了“导出为csv文件”的按钮，用户可以把在系统里面查询到的结果输出为csv文件，用excel可以直接打开，方便用作其它用途（e.g.老师可以把学生名单导出，并用excel打印出来当做点名册）。

以上为我们设计的教务系统的全部功能。

除此之外，我们设计的程序安全性较高，每次提供用户选择操作都会有明确的提示以及错误输入的应对方法，输出错误提示并让用户重新输入或者选择，对于数据库的修改也有一定的维护措施，如果数据库的操作有错，我们会弹窗告诉用户数据返回的错误信息，进一步方便用户对出错位置的判断。

系统的运行是在Windows系统下，以windows程序运行界面展示的，只要数据库前期把相应的表格正确建立就能够完整地运行。至于性能方面，虽然数据量比较大，但是我们使用的是oracle的数据库，是一款比较成熟的数据库管理系统，所以在运行速度和稳定性方面都有一定的保障。

三、过程论述

这一次数据库大作业，要做的第一件事情自然就是设计数据库了，那我们也已经做过中期汇报，不过那个时候我们的数据库模型还不够完整，在编码的过程中，我们发现需要增删不少表项，方便我们对数据进行查询、插入、删除等操作。

那么现在，我先分析一下数据库的设计过程：

1.需求分析：

关于需求分析，我们主要需要知道的是这个教务系统应该有些什么功能，下面开始分析教务系统应有的功能。

对于功能的设计，我们分为几个模块，分别是学生、教师、管理员（另外我想加一个通知功能，就是说管理员可以发通知，然后学生和教师登陆后可以看到通知的标题，点进去可以看详细内容）

对于学生来说，教务系统的功能应有：

① 查询自己所修的科目；

② 查询课程表（课程名、时间、地点）；

③ 查询各科的成绩及单科排名；

④ 查询绩点和班（专业）排名

⑤ 选课（这应该是我们这次想要实现的东西里面最麻烦的）；

⑥ 当然还要能查看自己的信息 ⑦还要能看通知，能登录

对于老师来说，教务系统的功能应有：

① 查询自己所教的课程（课程名、时间、地点、班级），并能导出课程表；

② 查询所教的所教班级各班级的成绩；

③ 能修改所教科目各班级的成绩；

④ 能查看自己的信息（所在院系、办公室地点之类的）

⑤ 能看通知、能登录

对于管理员来说，教务系统的功能有：

① 批量录入学生的信息（导入csv文件）

② 单个录入学生的信息（在程序中输入）

③ 批量录入老师的信息（导入csv文件）

④ 单个录入老师的信息（在程序中输入）

⑤ 输出每个老师的上课安排表

⑥ 计算每个学生的GPA

⑦ 能够发布通知、读取通知、能登录

2.概念结构设计（E-R图）：

这是我设计的初稿：

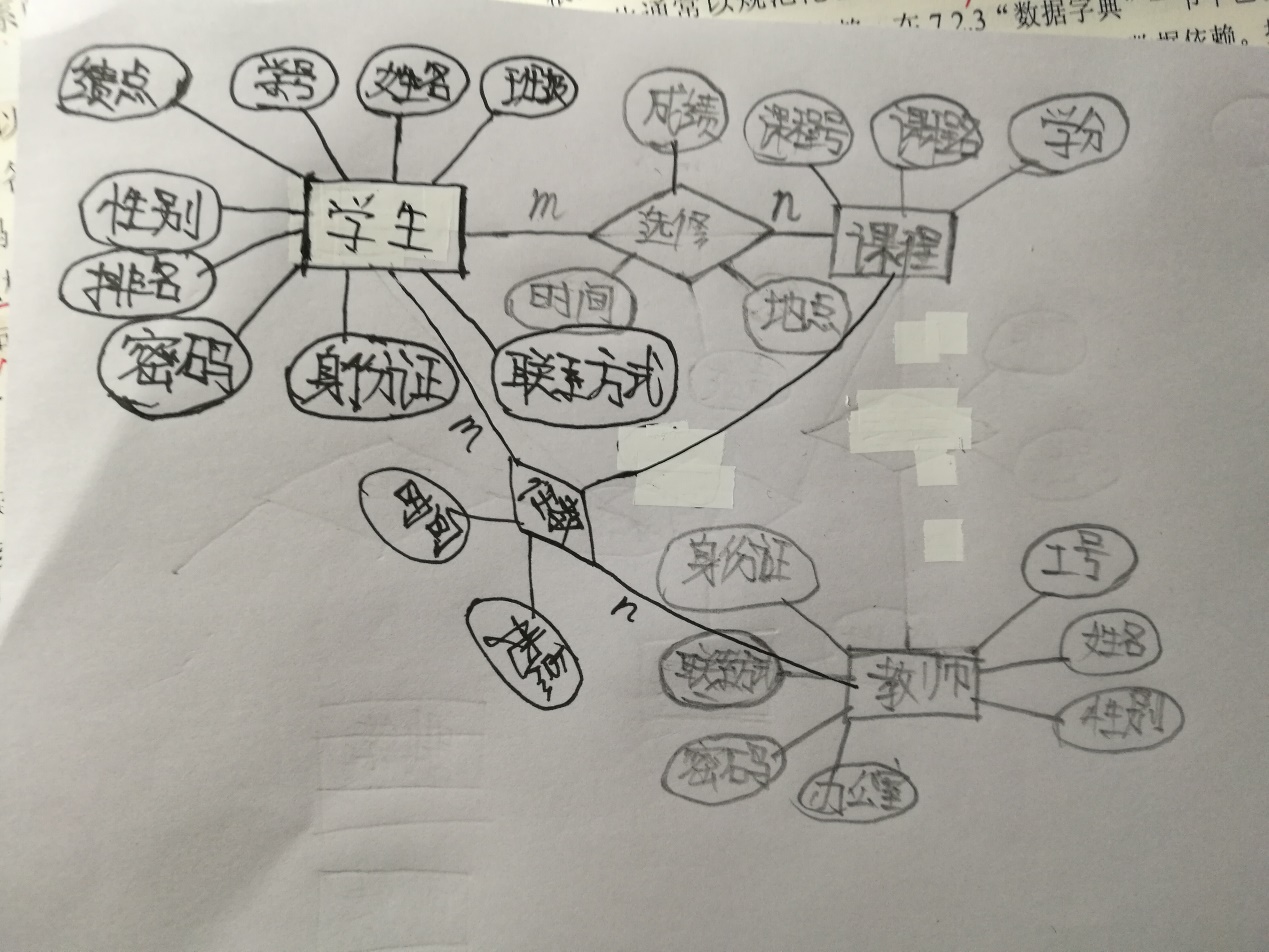


图1

这是我们组在编写程序完善后得到的终稿：

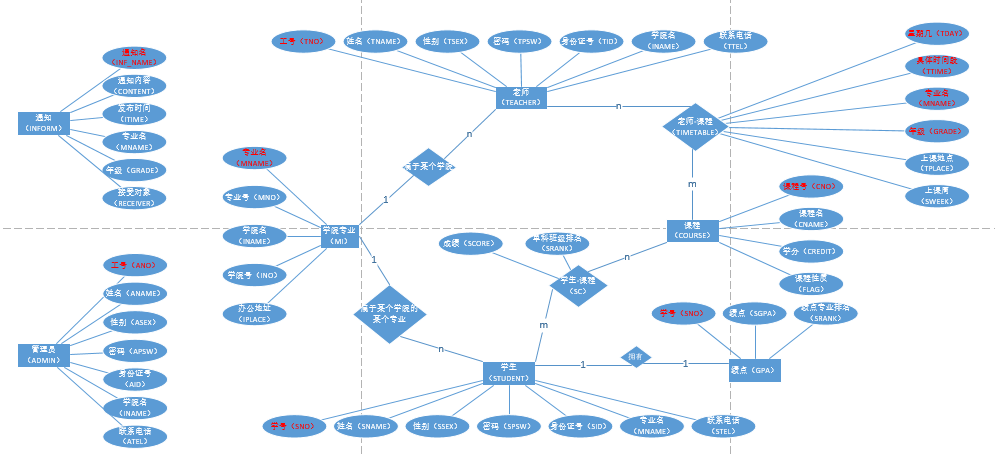


图2

由于图比较大，所以没办法看得很清楚，具体可以打开教务管理系统ER图.vsdx文件来看。

3.逻辑结构设计：

我们使用的是oracle数据库，根据初始的E-R图我们设计出如下关系表：

1. STUDENT（SNO（学号），SNAME（学生姓名）、SSEX（性别）、SPSW（密码）、SID（身份证）、MNAME（专业名）、STEL（联系方式））
2. TEACHER（TNO（工号）、TNAME（教师姓名）、TSEX（性别）、TPSW（密码）、TID（身份证）、INAME（教师所属学院）TTEL（联系方式）
3. 管理员表格，和教师表类似。（ANO（工号）、ANAME（教师姓名）、ASEX（性别）、APSW（密码）、AID（身份证）、INAME（教师所属学院）ATEL（联系方式））
4. COURSE（CNO（课程号）(4位)、CNAME（课程名称）、CREDIT（学分数）、FLAG（标志必修、选修））
5. SC（SNO(学号)、CNO(课程号) (4位)、SCORE（学科分数）、SRANK（单科排名）
6. INFORM（INF\_NAME（通知名）、CONTENT（内容）、ITIME（发布时间）、RECEIVER（接受对象）（0为全部人接收，1为学生接收，2位老师接收））
7. GPA（SNO（学号）、SGPA（学生绩点）、SRANK（学生排名）)
8. TIMETABLE（TDAY(星期几)、TTIME(时间)（几点~几点）、CNO（课程号(4位)）、TNO(工号)、MNAME（专业名）、GRADE（年级）,INAME（学院名），TPLACE(上课地点)、SWEEK（上课周数））
9. MI（MNO（专业号）,MNAME(专业名)，INO（学院号），INAME（学院名），IPLACE（办公地址）)
10. RECYCLE（DNO），用来回收删除的学号，工号，在增加用户的时候到垃圾桶找找看有没有可以用的，可以用的话就取出来用，不可以用再分配新的。
11. 年级表GLIST（GRADE（年级，2015-2018））

4.物理结构设计

这次我们在数据库框架的设计方面，我们考虑到很多时候表的读写和连接需要反复读写内存，所以我们的表会有一些冗余项，事实上这些冗余项是我们在设计的时候遇到了问题，发现数据量大的时候这样的表结构会使查询操作比较慢，所以才加上去的。而事实上我们改变了这些以后发现确实快上了那么一秒左右的时间，主要原因是不用进行大量表项的连接。事实上，我们也有考虑过用视图，但是后来觉得视图并不是那么科学，因为视图其实并不是真正存在的一张表，它只是构建出来给没有权限获取全部信息的用户使用的，查询的时候视图还是要进行视图消解，所以耗时并不会减少多少。

5.qt和oracle数据库的连接：

作为面向C++语言GUI设计软件的qt可以和oracle进行连接，具体地说就是要进行以下操作：



图3

在qt中要对数据库进行操作，使用的是sql\_query对象，具体来说，我们对数据库的查询都是通过这个对象来完成的，我们用prepare语句可以把sql语言的语句准备好，里面的参数我们用‘?’来代替，这样做的目的是防止有人用字符串连接攻击对对数据库的内容进行窃取。然后这些参数我们用addBindValue函数进行填充，具体的代码片段如下：

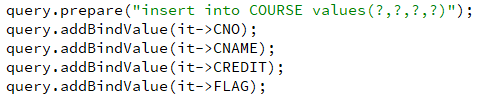


图4：插入数据到COURSE表中的实例

6.数据库的使用和程序编码：

下面我来介绍一下自己负责的部分：

由我负责编写的功能有：

1. 课程列表的导入和显示以及删除：

插入方面，我们直接使用qt文件路径获取的窗口，然后对课程列表文件进行读取，考虑到一个大学每一个学期课表一旦确定下来就不会有太多的改变，所以这里我只写了删除操作，这个删除操作可以删除一门课程，比如说在学期初选课的时候，可能有一些课程是不够人开课的，这时候就可以把这么课程删掉。具体来说，我们用一个csv文件（一种可以用word进行编辑的文件）存放课表，然后直接读入这个csv文件就可以把文件中相关的课程信息全部导入到数据库里面。

显示方面，我用QTableWidget这个qt的表格类型来显示信息，此外我设计了一个查询框供用户输入关键字，只要关键字能够匹配的上课程列表里面任何一个表项的内容，该条记录就会被显示出来，既可以按专业查询，也可以按课程号或课程名来查询，我认为这种模糊查询比起全字匹配查询更方便用户使用，因为当管理员查询一门课程，很多时候它不知道这门课的完整名字，这时候模糊搜索的作用就比较大了。此外，我搜索得到的结果还可以点击标题头部，实现按某标题的字符串顺序来显示课程信息。

删除方面，我做了一个右键菜单，其实就是获取鼠标点击的焦点，对应表格的第几行第几列，然后获取到对应的课程号，在数据库进行删除即可。

然后我还实现了到处查询结果为CSV文件的功能，这个实现起来使用的是文件操作，这里就不详细叙述了。

2.课程表的导入和查询操作：

导入方面，具体逻辑是：获取文件夹路径，读取里面所有的文件名，对各个文件路径进行构造，并存放在一个数组里面，然后逐个文件读入并放入数据库中。这个过程涉及到比较多的磁盘I/O操作，可能要等上那么两三秒的时间。读入数据以后，我会显示第一个课程表在数据库的界面上（只是为了让用户看看效果而已），然后点击“确认导入”按钮就能够获取插入数据，插入错误的时候会返回相应的提示。

要注意一点，插入课程表后，相应学生修了什么课程也已经知道了，所以同时构造出SC表（学生-课程 表）。因此确认导入后需要的时间会比较长。

查询课程表方面，我做了一个下拉框用来给用户选择年级，然后旁边有一个搜索框，给用户作模糊搜索用；另外一种查询方式是，我设置了另外一个下拉框，专门用来获取全部的专业列表，用户选择什么专业就会显示什么专业的课程。显示方面的话比较麻烦的是，要根据上课星期数和上课时间来决定这门课放在QTableWidget的什么地方，这样显示就能做到和我们平时的课程表一模一样，而不是一条普通的记录。

当然我也做了删除全部课程表的功能，这个就比较简单了，直接执行SQL语句：DELETE FROM TIMETABLE就可以了。

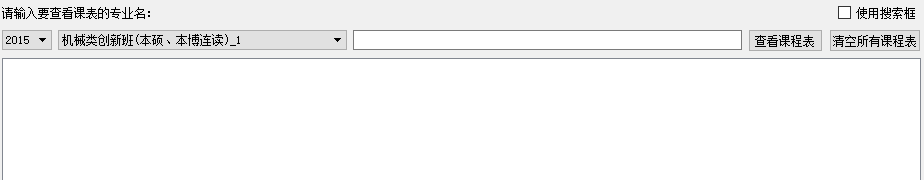


图5：课程表查询界面

3.学生绩点的生成、查询操作：

学生GPA的生成设计到数据量比较多，GPA表结构为：GPA（SNO（学号）、SGPA（学生绩点）、SRANK（学生排名）)因为获取学生的GPA，首先要有学生的成绩，而要计算排名，就要把同一个专业的学生的GPA计算出来才能够进行排名，因此需要进行STUDENT表、SC表（学生课程表）、COURSE表（课程表）这三个表的连接，由于学生有37000多个学生，所以这个操作耗费的时间也比较多，大概需要30s左右。

具体计算步骤：先连接三张表，获取所有学生的所有科目的成绩，然后遍历该表，遍历这个表的时候需要注意的是一条记录只是一个学生一门科目的成绩，当读完这个学生的所有成绩以后才可以算出该学生的gpa，但是这时候该学生的GPA信息里面还没有排名，只有当把一个专业的所有学生的GPA都计算出来以后才能进行排序，排序之后再把该专业各个学生的GPA记录存到一个vector向量里面。直到把全校所有学生的GPA信息都统计完成后，我们再把这些信息入库。

至于显示GPA信息的窗口，我也提供了一个搜索框，这个框提供模糊搜索，能够实现含通配符字符串的查找，比如输入：“计算机%\_1”就会显示出所有专业名中有“计算机”，并且以“\_1”结尾的专业学生的GPA。然后标题也是有正序反序排序的功能，比如一个专业的学生的gpa，我可以按gpa排名来顺序显示gpa信息，也可以按照这个班学生的学号递增顺序来显示gpa信息。

另外，我也提供了清空所有学生GPA，和清空所有学生成绩的按钮，方便学期末的时候管理员清理数据库中的数据。

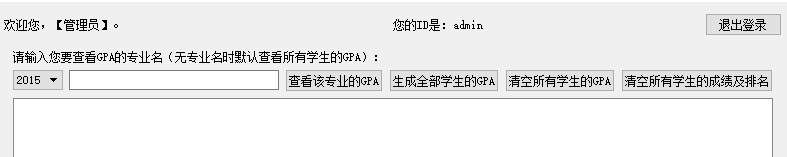


图6：GPA查询界面

四、结果分析

这是我写的内容有这些：

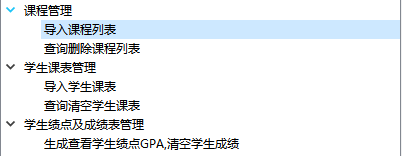


图7：目录树

下面演示一下导入课程列表的效果：

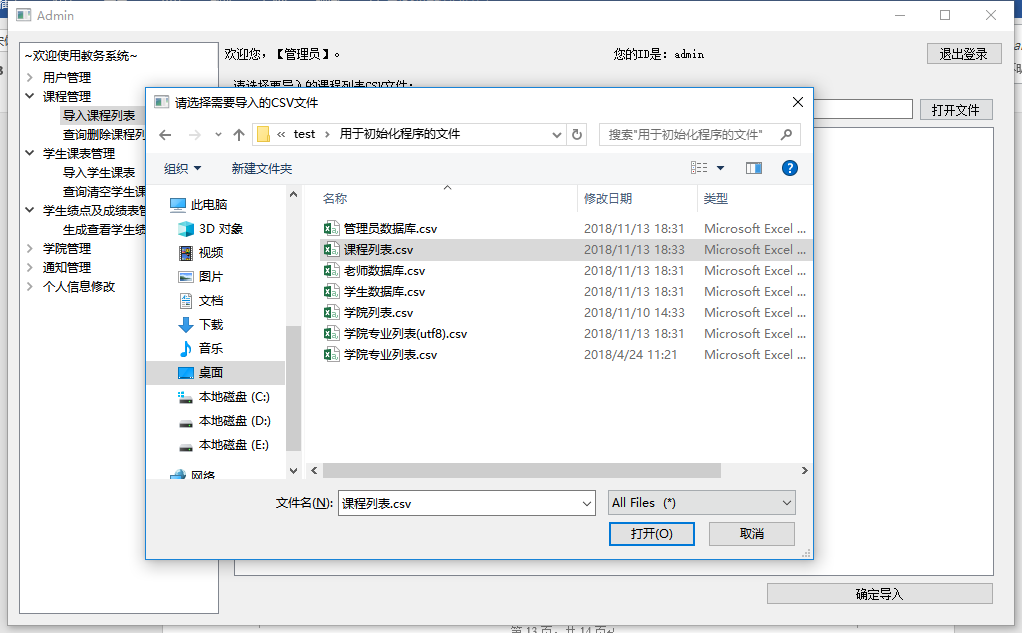


图8：课程列表导入

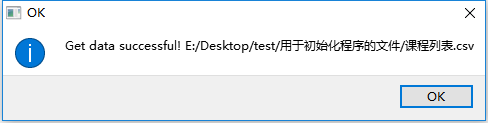


图9：系统提示



图10：读取文件后显示读取到的课程信息

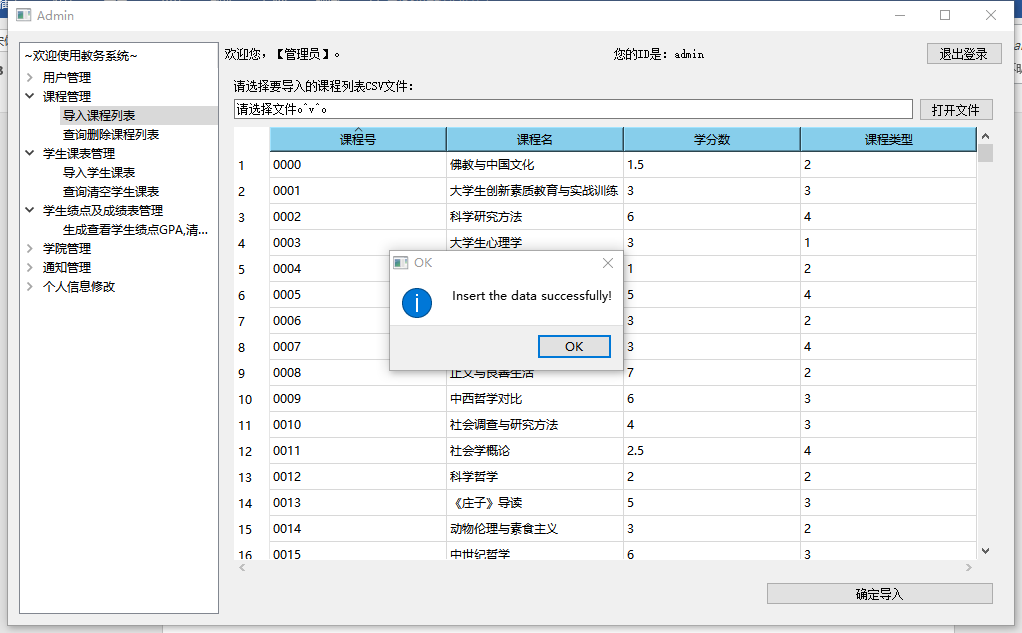


图11：导入课程列表成功

然后是查询课程列表的效果图：



图12：查询全部课程信息



图13：模糊搜索得到的结果（显示全部有‘现’字的课程）



图14：查看学分数为3.5分的全部课程



图15：按课程类型的顺序来显示



图16：按课程号升序来显示

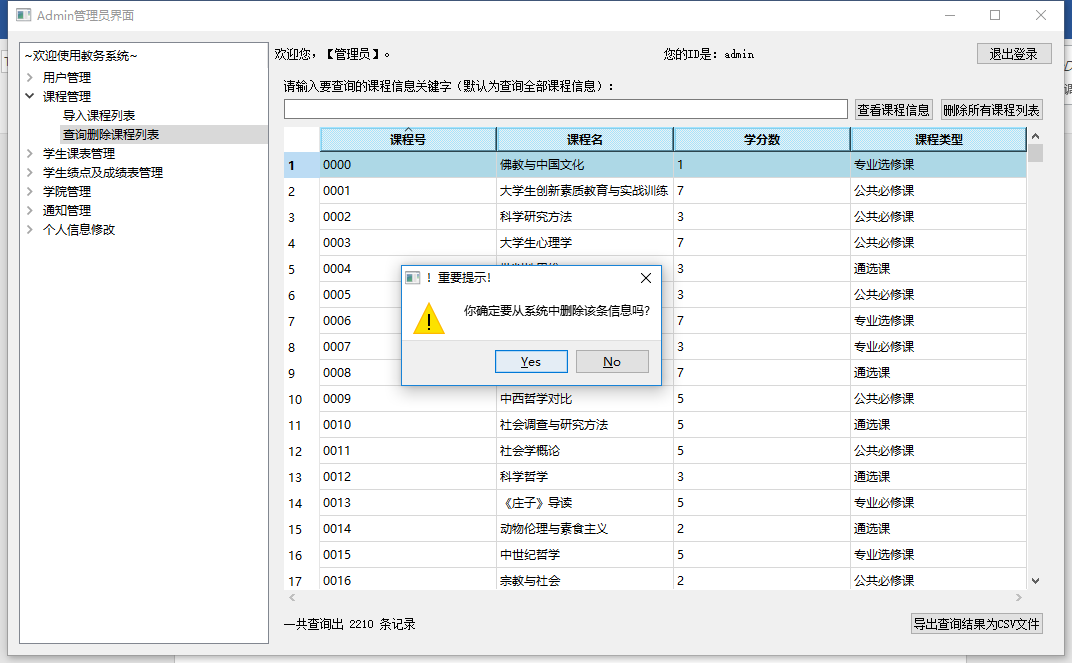


图17.1：删除课表的某门课



图17.2：可以看到删除成功（0000课程号的课程已经删掉了）

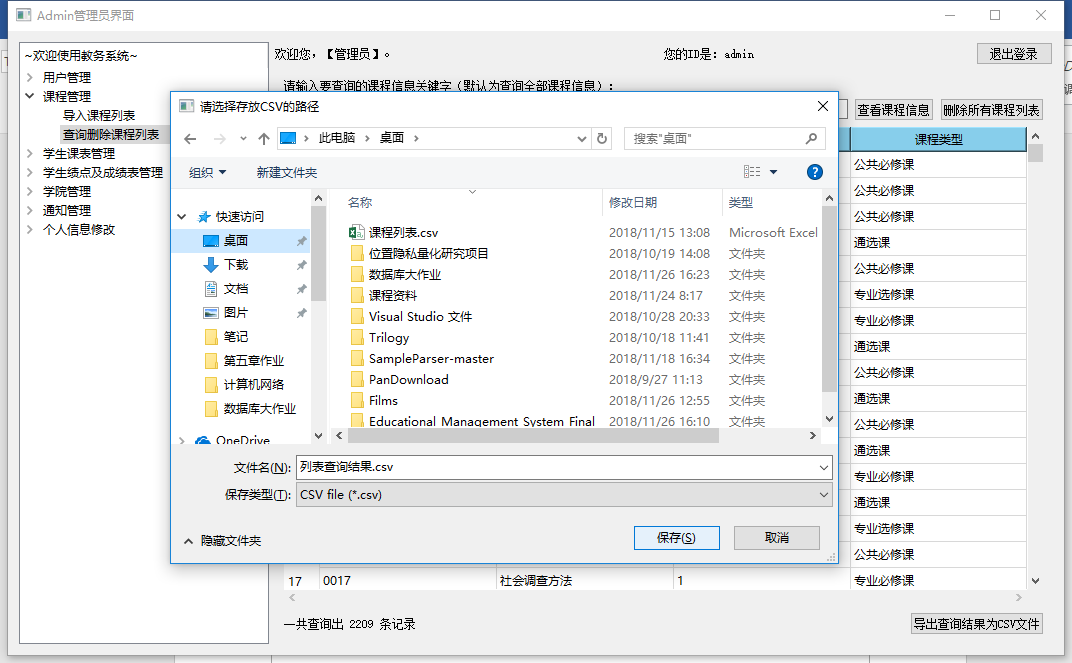


图18：导出文件效果(1)

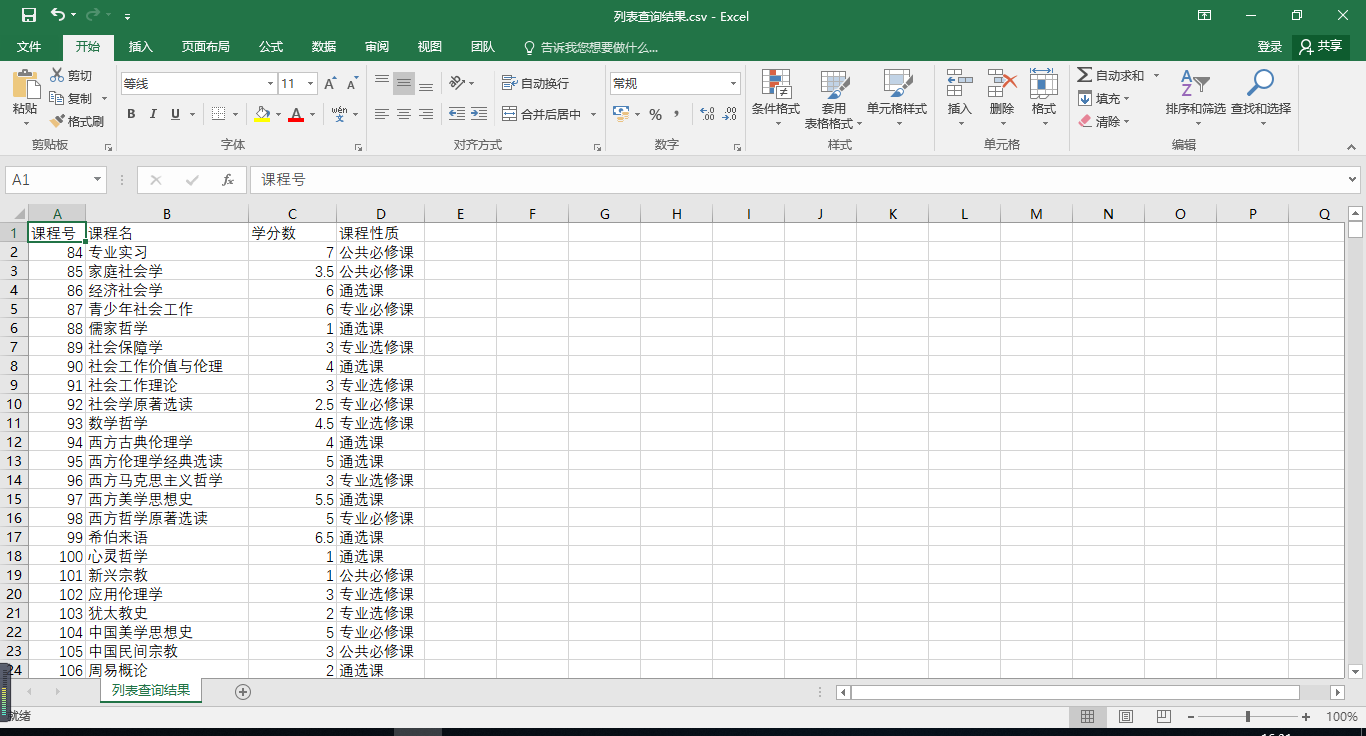


图19：导出文件的效果（2）

然后我们来看看导入学生课表的效果：

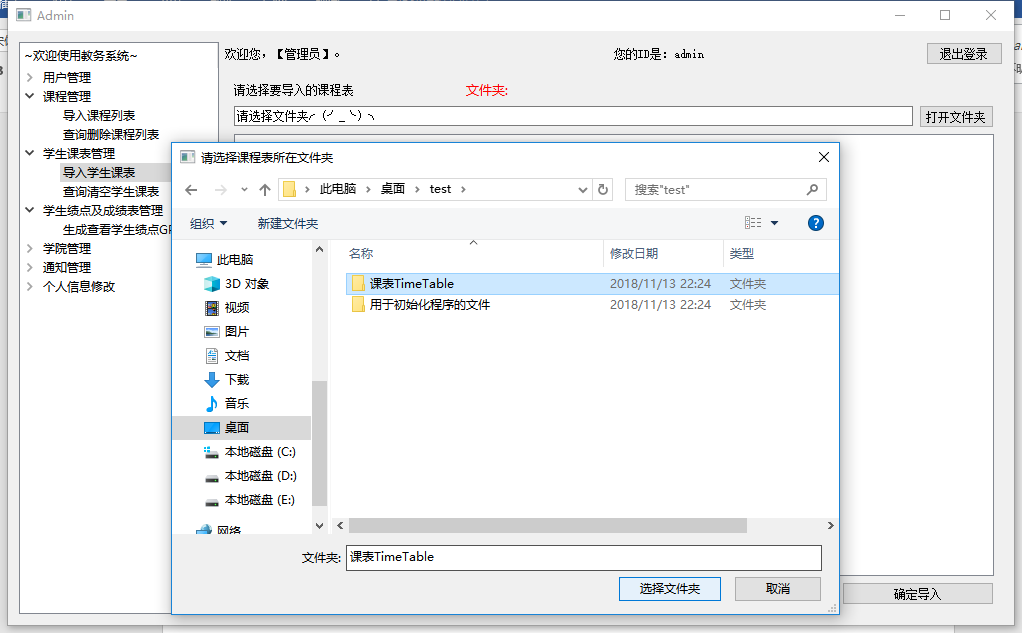


图20：导入各专业课程表



图21：读取各个课程表文件以后显示读取到的第一张课表的信息

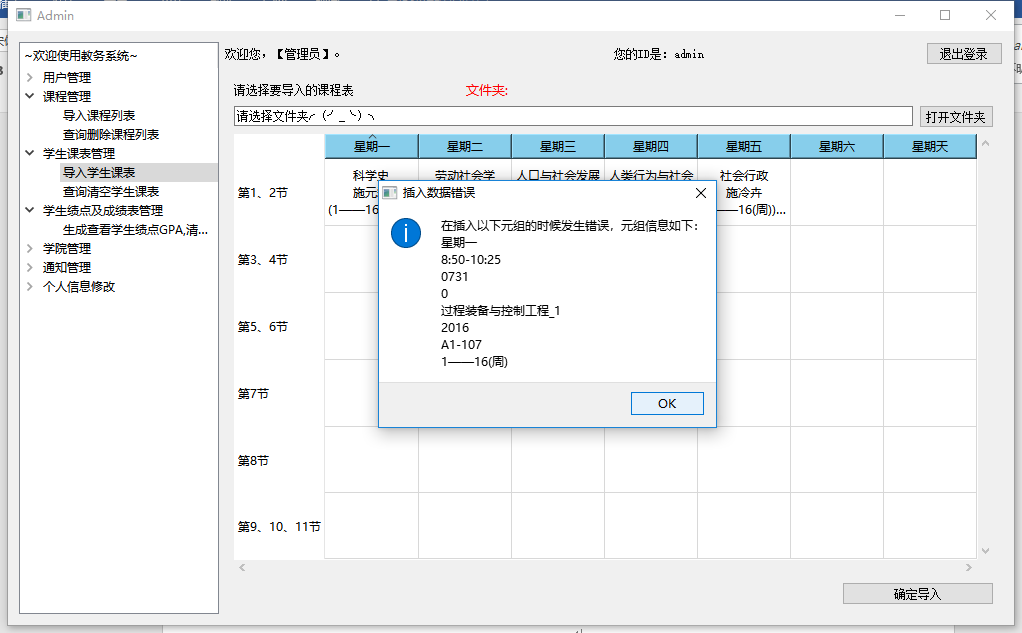


图22：插入时的错误提示

PS：这是我故意多准备了一条相同的记录，为了测试错误信息



图23：导入课程表成功，开始生成SC表



图24：SC表创建成功

接下来是查询课程表，查询课程表可以选择用搜索框查询，还是用下拉框选择专业来查询课程表：



图25：使用下拉框查询课程表的效果（和我们平时的课表一模一样）



图26：使用搜索框实现课程搜索

最后演示一下生成GPA和查询GPA的页面效果：

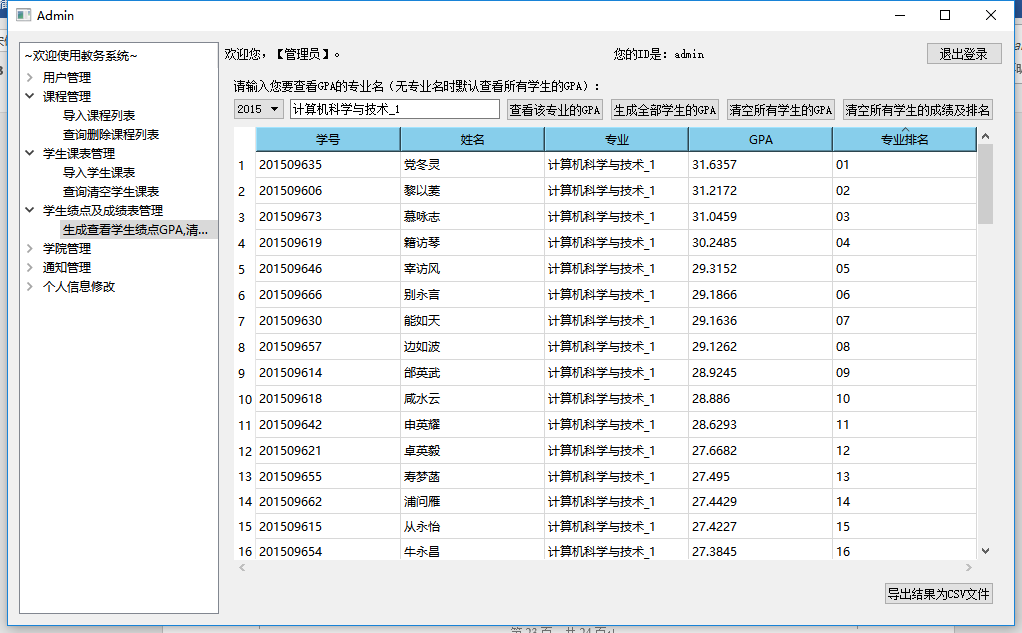


图27：生成全部学生GPA，并进行查询（按照排名显示）

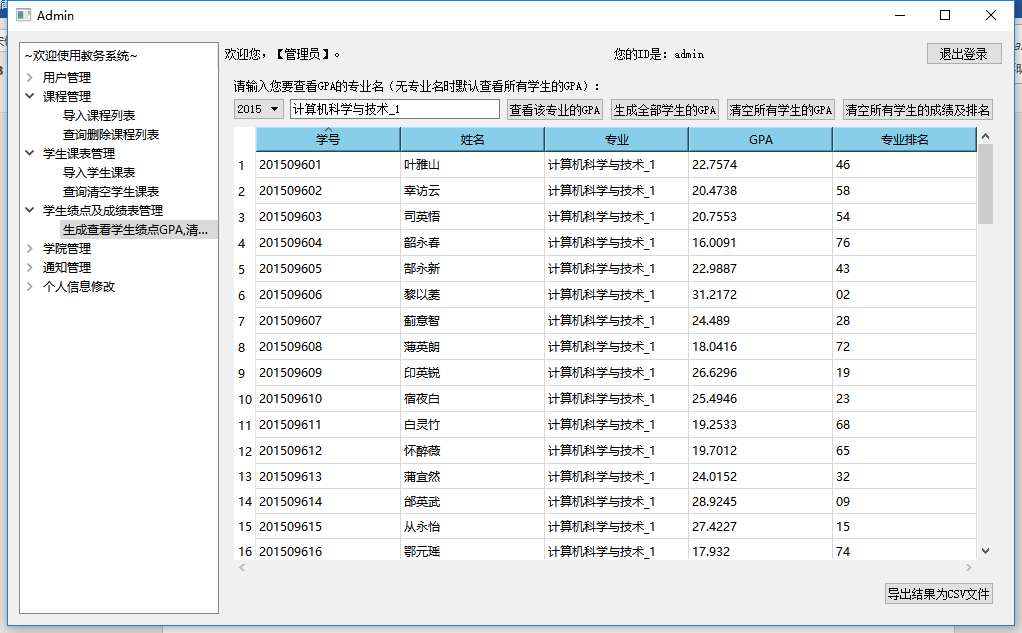


图28：按照学号升序显示gpa信息



图29：进行模糊搜索，会显示不同专业的GPA信息

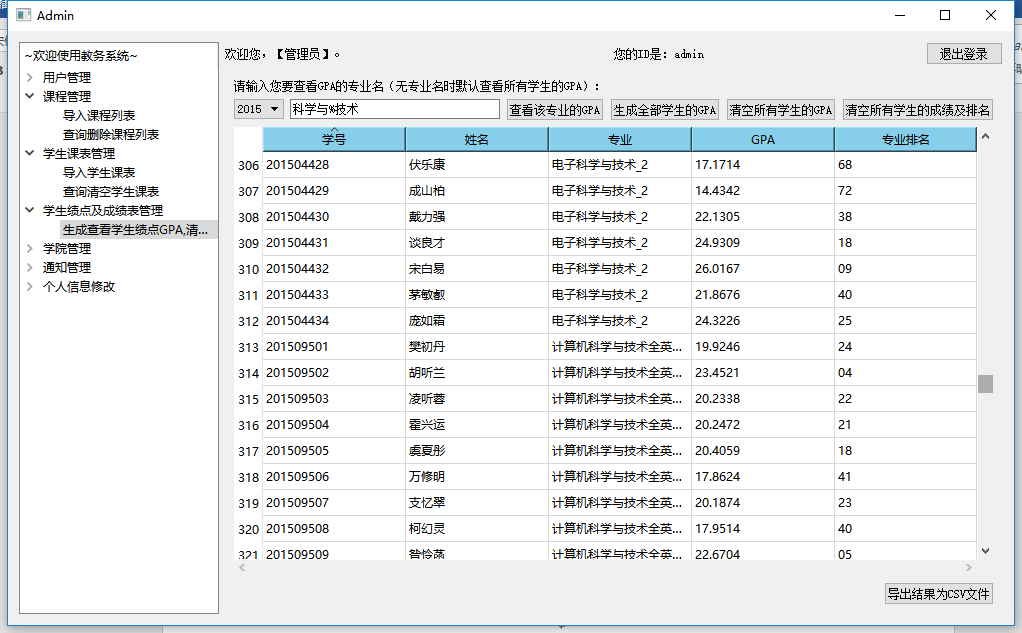


图30：用含通配符字符串进行模糊搜索

五、课程设计总结

这次数据库大作业，让我更加清楚地认识到了数据库的重要性。大二上学期，我们也做过一个类似的系统，但是我们当时对数据操作都是以文本文件或者二进制文件来存储的，读取起来真的非常的慢。而且当时我们为了求出某一项数据，还必须要将很多个文件连接起来，而这些文件又放在不同的目录里面，所以我们要查找一个文件还要先查找目录名。

现在回想起来，那个目录名其实就是数据库的表名，如果我们要生成某个学生的GPA，我们要先查找这个学生修的都有些什么课程，进而获取这些课程的学分，还要获取这个学生的各科的成绩，因此要对很多个文件进行读写操作。如下图所示：

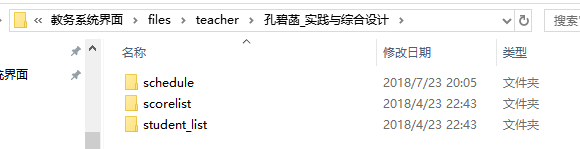


图30：例子

在这种情况下，要实现一个数据的查询，确实需要有非常清晰的文件读取逻辑才能完成文件的读写，我们当时可以说是用文件系统实现了一个数据库。

但是有了数据库以后，一切都变得不一样了，简单的几个表，就可以把所有的信息记录起来，而对于某项数据的查询，只需要进行表连接即可。甚至数据库还能帮我们完成一些简单的排序，运算操作。

关于表的设计，之前在学习数据库概念模型设计时，我不明白为什么一些数据库的表会有一些冗余项；然后到我们现在自己去设计数据库结构的时候，我们就会发现，一个表里面，有些表项可能确实会破坏关系规范性，插入了这项数目，这个表可能就不是BCNF或者3NF了，但是出于对数据库运行速度的考虑，我们为了减少查询时对表的连接次数，我们很有可能会选择将两张3NF的表合成为1张2NF的表，虽然有可能会有冗余数据，但是我认为现在我们的存储空间相对来说都是比较大的，所以有一部分冗余数据也是可以接受的。但是读写硬盘的时间就不一样了，我们学习过操作系统就知道，I/O是最花时间的，对表进行连接，需要直接对内存进行操作，数据量大的时候甚至还要对磁盘进行操作。因此表中的冗余项其实就是“用空间冗余换取时间缩减”。

此外，我觉得用户需求真的是非常重要，因为只有知道了用户想要实现什么功能，我们才可以确定自己需要什么数据，才可以确定数据库的结构。就比如这次我们开始设计的时候虽然有需求分析，但是我们没有想清楚实现这些需求的时候需要什么数据，这就导致我们在编写代码的过程中需要重新构建E-R图，再次修改各个关系表的结构，因此程序代码也要作出相应的调整。所以说，设计数据库之前理解好客户需求是非常重要的。

其次的话就是编码了，大一大二的大作业中，我编写的大多是后台函数，没有怎么接触用户界面的编写，这次我尝试用qt写了一下界面，感觉不算太困难，不过由于初次使用，所以做出来的界面会比较简陋，不过如果再有机会使用qt，我觉得我会把界面设计的更加好看。

总的来说，这次数据库大作业我还是有挺多的收获的，希望下次大作业能够学习到其他更多的知识。

**参考文献：**

[1]周爱武，汪海威，肖云．数据库课程设计［M］．北京： 机械工业出版社． 2012

[2]王珊，萨师煊．数据库系统概论(第五版) ［M］．北京： 高等教育出版社． 2014

[3]杨冬青等译． 数据库系统概念(第六版) ［M］．北京： 机械工业出版社． 2012

[4] Qt中文手册 之 QTreeWidget - CSDN博客<https://blog.csdn.net/u010168781/article/details/51565317>

[5] Qt中文手册 之 QTreeWidgetItem - CSDN博客https://blog.csdn.net/u010168781/article/details/51579526

[6] Qt UserRole详解 - fanyun的博客 - CSDN博客https://blog.csdn.net/fanyun\_01/article/details/78640277

[7] 强大的QVariant - u010335285的专栏 - CSDN博客https://blog.csdn.net/zhanghuiyi222/article/details/8949828

[8] QTreeWidget的使用方法\_百度经验https://jingyan.baidu.com/article/22fe7ced24e7033002617fbe.html

[9] Qt之QStackedWidget多界面切换 - Lance - CSDN博客https://blog.csdn.net/u011731378/article/details/79975782

[10] QFileDialog-CSDN论坛

https://bbs.csdn.net/topics/390375238

[11] Qt获取文件夹下文件夹，文件列表 - Seven - CSDN博客

https://blog.csdn.net/qq\_33854260/article/details/70299353

[12] QString替换多个字符 - 各位-请不吝赐教 - CSDN博客

https://blog.csdn.net/u013776188/article/details/73492349?utm\_source=blogxgwz5

[13] QTableWidget之右键菜单的实现！ - hitxiaya的专栏 - CSDN博客

https://blog.csdn.net/hitxiaya/article/details/7193732

[14] QTableWidget怎么得到鼠标位置所在的单元格-CSDN论坛

https://bbs.csdn.net/wap/topics/360266015

[15] QT实现鼠标右键快捷菜单(QTableWidget) - TanChengkai的博客 - CSDN博客

https://blog.csdn.net/tanchengkai/article/details/52104924