

工程作业报告

1. 项目信息（10 分）

学号	2011763	姓名	黄天昊	专业	计算机科学与技术
项目名称	教务信息管理系统				
必备环境	<p>操作系统： Windows 10</p> <p>后台数据库管理系统： MySQL 8.0</p> <p>IDE： Visual Studio 2019</p> <p>UI 设计： MFC</p>				
系统主要功能简介（4 分）	本教务信息管理系统面向数据库管理员能够实现教师、学生、课程、成绩等信息的数据的插入、删除、更新以及查询等操作，并在成绩的统计与分析方面增加了大量功能，通过可视化的操作界面，大大简化了教务信息管理员的各项操作。				
系统主要页面截图（6 分）					

教师信息 学生信息 课程信息 成绩信息

添加

删除

修改

查询

[illegible]

欢迎来到卡塞尔学院教务系统

教师信息学生信息课程信息成绩信息

添加

删除

修改

查询

c_id	c_name	c_credit	c_type	c_major	
1	炼金术	4.0	A	炼金	
2	龙族史	4.0	A	历史	
3	屠龙实践	8.0	A	修改	
4	游泳	2.0	C	体育	
5	汽车驾驶	2.0	D	无	

欢迎来到卡塞尔学院教务系统

教师信息学生信息课程信息成绩信息

添加

删除

修改

查询

总分排名

s_id	c_id	t_id	score	
1	2	3	80.00	
1	3	1	100.00	
1	4	2	85.00	
2	1	2	95.00	
2	3	1	98.00	
3	2	2	100.00	
3	3	1	100.00	
3	4	2	99.00	
4	3	1	90.50	
4	4	2	100.00	
5	1	1	100.00	
5	2	3	100.00	
5	3	1	95.00	
6	1	2	65.00	
6	2	1	70.00	
6	3	3	65.00	
7	1	1	100.00	
7	2	1	95.00	
7	3	1	92.00	

说明		(2 分) 请说明系统配置情况 (后台数据库, 高级语言); (8 分) 请使用连接串连接高级语言和数据库, 并分析字符串的各个部分。
配置步骤 2	DBMS	1. 后台数据库使用 MySQL
		2. 搭配 Navicat 平台实现数据库的搭建
	高级语言	1. C++
		2. 开发平台为 Visual Studio, 主要使用 MFC 基础类库

分					
连接串分析 (6分)	序号	名称	功能说明	取值	
	1	MYSQL *STDCALL mysql_init()	初始化数据库句柄	参数为 MYSQL *mysql，即 MYSQL 类类型的指针	
	2	MYSQL *STDCALL mysql_real_connect()	连接数据库	MYSQL *STDCALL mysql_real_connect(MYSQL *mysql, const char *host, const char *user, const char *passwd, const char *db, unsigned int port, const char *unix_socket, unsigned long clientflag); 参数分别为ip地址，用户名，密码，数据库名，端口	
	3	int STDCALL mysql_query();	数据库查询语句	参数分别为MYSQL *mysql, const char *q 即连接到数据库的MYSQL类类型的指针以及查询语句	
	4	MYSQL_RES *STDCALL mysql_store_result();	查询结果返回语句，与mysql_query搭配使用	参数为连接了数据库的指针MYSQL *mysql，返回一个MYSQL_RES类类型的指针，其中包含查询结果	
	5	MYSQL_ROW STDCALL mysql_fetch_row();	获取数据库查询结果的一行结果	参数为连接了数据库的指针 MYSQL *mysql，返回一个 MYSQL_RES 类类型的指针，其中包含查询结果	
连接串代码 (截屏) (2分)	<pre>//初始化数据库句柄 mysql_init(mysql); //连接数据库 参数为ip 用户名 密码 数据库名 端口 mysql_real_connect(mysql, "127.0.0.1", "root", "hth20020315", "educationaladministratorsystem", 0, NULL, 0);</pre>				
备注					

3. 数据库设计 (14分)

说明	(10分) 按照数据表的创建顺序, 依次给出所涉及数据表的信息, 其中参照字段以“(字段1, 字段2, ……, 字段n)”的形式给出, 被参照字段以“表名(字段1, 字段2, ……, 字段n)”的形式给出; (4分) 一般 DBMS 都可以为数据库生成关系图, 请将该图片截屏并粘贴到表格中。				
数据表 (10)	创建顺序	数据表名称	主键	参照属性	被参照表及属性
	1	教师表(teacher)	t_id	无	无
	2	学生表(student)	s_id	无	无
	3	课程表(course)	c_id	无	无

	4	成绩表(score)	s_id, c_id	s_id	student(s_id)
				c_id	course(c_id)
				t_id	teacher(t_id)
	5	系统管理员表 (administrator)	a_use	无	无
关系图 4分					
备注					

4. 含有事务应用的删除操作（13分）

说明	（1分）简要说明该操作所要完成的功能； （2分）该操作会涉及的表（必须含有两张或两张以上的关系表，同时以“表名”的形式给出） （1分）表连接涉及字段描述（描述方式为“表1.属性=表2.属性”） （1分）删除条件涉及的字段描述（以“表名.属性=? ”形式给出） （4分）实现该操作的关键代码（高级语言、SQL），截图即可；（其中如果删除语句中不包含任何形式的事务应用将扣除3分） （4分）如何执行该操作，按所述方法能够正常演示程序则给分。	
功能描述 (1分)	如果一个学生毕业或退学，需要删除他的信息。具体为从学生表中删除该学生的数据，同时，该学生在成绩表中的所有记录也应该被删除，以保证实体完整性和参照完整性，在删除过程中应当采用事务操作，以保证数据库的原子性、一致性、隔离性、持久性。	
涉及的表 (2分)	student(s_id, s_name, s_sex, s_score, s_major, s_pho, s_credit) score(s_id, c_id, t_id, socre)	
表连接涉及字段 (1分)	score.s_id=student.s_id	
删除条件	字段	规则

字段描述 (1 分)	<table><tr><td>student.s_id</td><td>当 student 表中某一条数据被删除之后，触发器会删除该数据在</td></tr><tr><td>score.s_id</td><td>score 表中的 s_id=student.s_id 的所有数据</td></tr></table>	student.s_id	当 student 表中某一条数据被删除之后，触发器会删除该数据在	score.s_id	score 表中的 s_id=student.s_id 的所有数据
student.s_id	当 student 表中某一条数据被删除之后，触发器会删除该数据在				
score.s_id	score 表中的 s_id=student.s_id 的所有数据				
代码 (4 分)	<pre>//删除选中的一行数据 //首先得到点击的位置 POSITION pos = m_listcontrol.GetFirstSelectedItemPosition(); if (pos == NULL) { MessageBox(L"请至少选择一项", L"提示", MB_ICONEXCLAMATION); return; } //得到行号，通过POSITION转化 int row = (int)m_listcontrol.GetNextSelectedItem(pos); //获取第row第0列的内容，并保存到str中 CString str = m_listcontrol.GetItemText(row, 0); char query[200]; USES_CONVERSION; sprintf_s(query, "DELETE FROM student WHERE s_id = %s", T2A(str)); if (mysql_query(mysql, query)) { CString error; error.Format(L"删除失败, 错误信息:\n%hs", (LPCTSTR) (LPCTSTR)mysql_error(mysql)); MessageBox(error); } else { MessageBox(L"删除成功"); } sprintf_s(query, "CALL delete_student(%s)", T2A(str)); if (mysql_query(mysql, query)) { CString error; error.Format(L"成绩表更新失败, 错误信息:\n%hs", (LPCTSTR) (LPCTSTR)mysql_error(mysql)); MessageBox(error); } else { MessageBox(L"成绩表更新成功"); } 1 CREATE DEFINER=`root`@`localhost` PROCEDURE `delete_student`(IN sid INT) 2 BEGIN 3 DECLARE error INTEGER DEFAULT 0; 4 DECLARE CONTINUE HANDLER FOR SQLEXCEPTION SET error = 1;-- 初始化事务 5 START TRANSACTION;-- 创建事务过程 6 DELETE 7 FROM 8 score 9 WHERE 10 score.s_id = sid; 11 IF 12 error = 1 THEN-- 撤销事务 13 ROLLBACK; 14 ELSE -- 提交事务 15 COMMIT; 16 END IF; 17 END</pre>				
程序演示 (4 分)	删除前，可以看到，id 为 10 的学生信息，以及其对应的 3 条成绩记录				

欢迎来到卡塞尔学院教务系统

×

教师信息 学生信息 课程信息 成绩信息

添加

删除

修改

查询

s_id	s_name	s_sex	s_age	s_major	s_pho	s_GPA
1	路明非	男	20	屠龙学	00000000000	
2	凯撒·加图索	男	21	屠龙学	14987875664	
3	楚子航	男	21	屠龙学	19878564234	
4	诺诺	女	21	艺术		
5	零	女	19	舞蹈		
6	芬格尔	男	26	摆烂	14897987777	
10	啦啦啦	男	56	未知		

欢迎来到卡塞尔学院教务系统

×

教师信息 学生信息 课程信息 成绩信息

添加

删除

修改

查询

统计分析

总分排名

s_id	c_id	t_id	score
1	3	1	100.00
1	4	2	85.00
2	3	1	98.00
3	2	2	100.00
3	3	1	100.00
4	3	1	90.50
4	4	2	100.00
5	2	3	100.00
10	1	1	100.00
10	2	2	20.00
10	3	4	100.00

删除后，可以看到不仅学生表中该学生数据被删除，其在成绩表中的数据也一并被删除

欢迎来到卡塞尔学院教务系统

×

教师信息 学生信息 课程信息 成绩信息

添加

删除

修改

查询

s_id	s_name	s_sex	s_age	s_major	s_pho	s_GPA
1	路明非	男	20	屠龙学	00000000000	
2	凯撒·加图索	男	21	屠龙学	14987875664	
3	楚子航	男	21	屠龙学	19878564234	
4	诺诺	女	21	艺术		
5	零	女	19	舞蹈		
6	芬格尔	男	26	摆烂	14897987777	

欢迎来到卡塞尔学院教务系统

教师信息

学生信息

课程信息

成绩信息

添加

删除

修改

查询

统计分析

总分排名

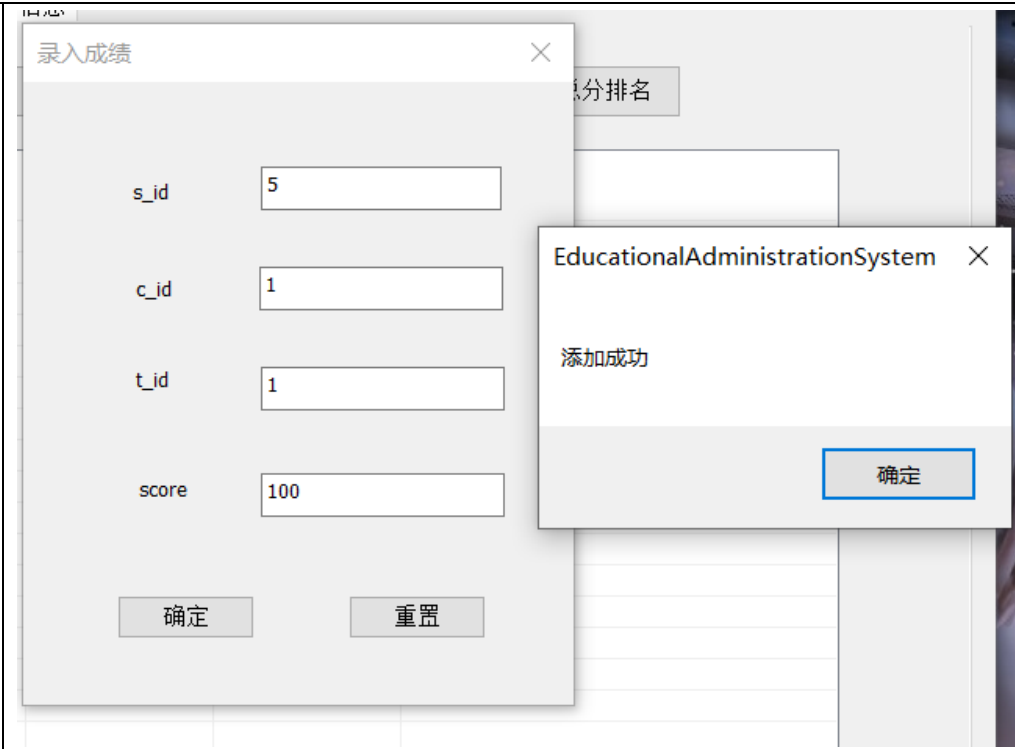
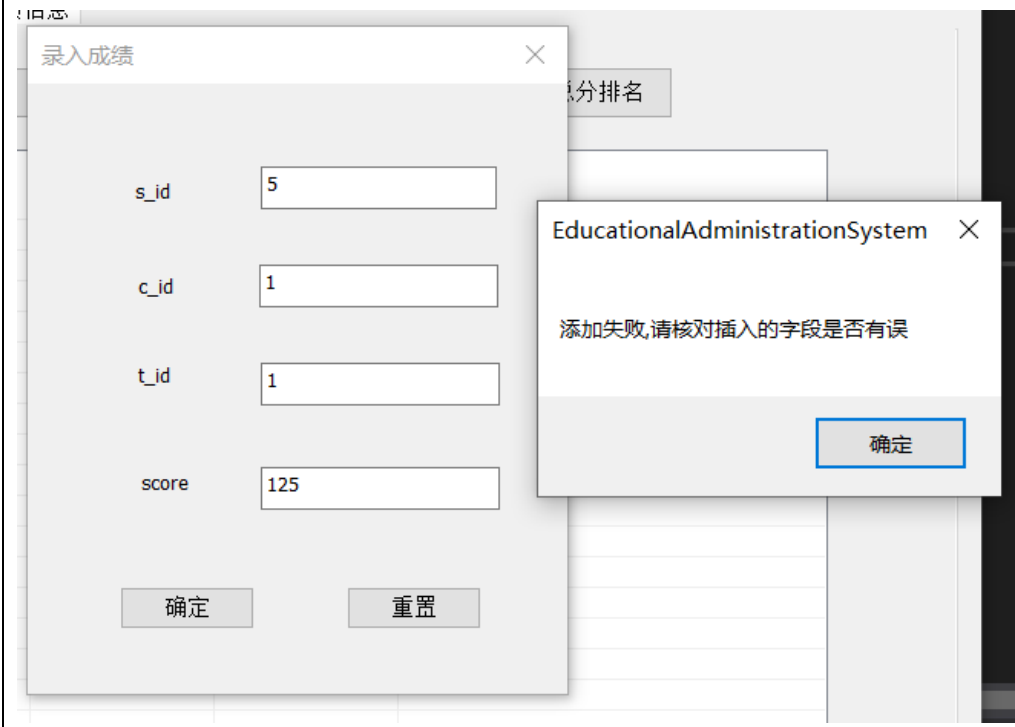
s_id	c_id	t_id	score	
1	3	1	100.00	
1	4	2	85.00	
2	3	1	98.00	
3	2	2	100.00	
3	3	1	100.00	
4	3	1	90.50	
4	4	2	100.00	
5	2	3	100.00	

备注

5. 触发器控制下的添加操作（20 分）

说明	(1 分) 简要说明该操作所要完成的功能; (2 分) 简要说明该触发器所要完成的功能 (1 分) 该操作会涉及的表 (以 “表名” 的形式给出)。 (2 分) 该操作输入数据以及输入数据应该满足的条件, 如: 数值范围、是否为空; (6 分) 实现该操作的关键代码 (高级语言、SQL), 截图即可; (8 分) 如何执行该操作, 按所述方法能够正常演示程序则给分。	
功能描述 (1 分)	录入学生成绩时, 学生的成绩应该在 0-100 分之间, 该触发器可以防止成绩录入出现错误的情况	
触发器描述 (2 分)	当系统需要录入学生成绩时, 每插入一条成绩的数据之前, 触发器会先检查其 score 的值, 如果在 0-100 之间, 则插入数据, 否则不予插入, 并弹出故障提示	
涉及的表 (1 分)	score(s_id, c_id, t_id, socre)	
输入数据 (2 分)	字段	规则
	s_id, c_id, t_id, score	由用户输入, 如果 score 在 0-100 之间则插入成功

插入 操作 源码 (3 分)	<pre> //更新输入的数据 UpdateData(TRUE); if (update == false) { //增加操作 CString str[5]; str[0] = m_sid; str[1] = m_cid; str[2] = m_tid; str[3] = m_score; char query[200]; USES_CONVERSION; sprintf_s(query, "INSERT INTO score VALUES (%s, %s, %s, %s)", T2A(str[0]), T2A(str[1]), T2A(str[2]), T2A(str[3])); mysql_query(mysql, "SET NAMES GBK");//设置编码格式, 否则cmd下中文乱码 if (mysql_query(mysql, query)) { CString error; error.Format(L"添加失败, 请核对插入的字段是否有误, 错误信息:\n%s", (LPCTSTR) (LPCTSTR)mysql_error(mysql)); MessageBox(error); } else { MessageBox(L"添加成功"); } } </pre>
触发 器源 码 (3 分)	<pre> CREATE TRIGGER trigger_t3 BEFORE INSERT ON score FOR EACH ROW]BEGIN DECLARE msg VARCHAR (100); IF] new.score < 0 OR new.score > 100 THEN SET msg = CONCAT('输入的成绩为 ', new.score, ' 不符合实际'); SIGNAL SQLSTATE 'HY000' SET MESSAGE_TEXT = msg; END IF; END </pre>
程序 演示 (4 分)	<p>说明：不违背触发器能够执行插入操作。</p> <pre> INSERT INTO score VALUES(3,4,2,99) > Affected rows: 1 > 时间: 0.005s </pre>

	
程序演示 (4分)	<p>说明：违背触发器要求，不能够执行插入操作，系统报错。</p> <pre>INSERT INTO score VALUES(3,4,2,123)</pre> <p>> 1644 - 输入的成绩为 123.00 不符合实际 > 时间：0.002s</p> 
备注	

6. 存储过程控制下的更新操作（18 分）

说明	<p>（1 分）简要说明该操作所要完成的功能；</p> <p>（1 分）简要说明该存储过程所要完成的功能；</p> <p>（2 分）说明该操作涉及操作的表（必须包含两张或两张以上的关系表，以“表名形式”描述）</p> <p>（1 分）表连接涉及字段描述（描述方式为“表 1. 属性=表 2. 属性”）</p> <p>（2 分）该操作会修改字段（以“表名. 字段名”的形式给出），以及修改规则，如新数值的计算方法、在何种条件下予以修改等；</p> <p>（6 分）实现该操作的关键代码（高级语言、SQL），截图即可；</p> <p>（5 分）如何执行该操作，按所述方法能够正常演示程序则给分。</p>	
功能描述 （1 分）	当修改学生成绩时，需要更新学生已修学分，如果该学生此课程得分修改后超过 60，则已修学分增加该课程的学分，采用存储过程自动更新	
存储过程功能描述 （1 分）	在 score 表增添或更新数据时，student 表的 s_credit 字段也需要更新，否则会造成数据不一致的问题，存储过程接收 score 表数据的参数，实现学生表已修学分的自动更新	
涉及的关系表 （2 分）	student(s_id, s_name, s_sex, s_score, s_major, s_pho, s_credit) score(s_id, c_id, t_id, socre)	
表连接涉及字段 （1 分）	score.s_id=student.s_id	
更改字段 （2 分）	字段	规则
	s_cretit	等于该学生的所有修过的课程的所有学分

<p>更新 代码 (3 分)</p>	<pre> else { //修改操作 CString str[6]; str[0] = m_sid; str[1] = m_cid; str[2] = m_tid; str[3] = m_score; char query[300]; USES_CONVERSION; sprintf_s(query, "UPDATE score SET t_id = %s, score = %s WHERE s_id = %s AND c_id = %s", T2A(str[2]), T2A(str[3]), T2A(str[0]), T2A(str[1])); if (mysql_query(mysql, query)) { CString error; error.Format(L"更新失败, 错误信息:\n %hs", (LPCTSTR) (LPCTSTR)mysql_error(mysql)); MessageBox(error); } else { MessageBox(L"更新成功"); } } </pre>
<p>创建 存储 过程 源码 (3 分)</p>	<pre> 1 CREATE DEFINER=`root`@`localhost` PROCEDURE `update_credit2` 2 (IN id INT, IN inscore INT, IN inscoreold INT, IN credit INT) 3 BEGIN 4 IF 5 inscore >= 60 AND inscoreold < 60 THEN 6 UPDATE student 7 SET student.s_credit = student.s_credit + credit 8 WHERE 9 s_id = id; 10 11 END IF; 12 13 END </pre>

存储
过程
执行
源码
(1
分)

```
char query[200];
sprintf_s(query, "SELECT c_credit FROM course WHERE c_id = %s", T2A(str[1]));
mysql_query(mysql, query);

MYSQL_RES* res;
MYSQL_ROW myrow;

res = mysql_store_result(mysql);
myrow = mysql_fetch_row(res);
CString credit;
credit = myrow[0];

sprintf_s(query, "CALL update_credit2(%s, %s, %s, %s)", T2A(str[0]), T2A(caa.m_score),
if (mysql_query(mysql, query)) {
    CString error;
    error.Format(L"更新失败, 请核对插入的字段是否有误");
    MessageBox(error);
}
else {
    MessageBox(L"学生表更新成功");
}

int count = m_listcontrol.GetItemCount();
for (int i = 0; i < count; i++) {
    m_listcontrol.DeleteItem(0);
}

row_num = 0;
Exhibition();
```

程序
演示
(2
分)

说明：不违背存储过程，能够执行更新操作
更新前，姓名为凯撒的学生已修学分为 12，他修了三门课，但是有一门成绩不及格，因此没有获得相应的学分

欢迎来到卡塞尔学院教务系统

教师信息 学生信息 课程信息 成绩信息

添加 删除 修改 查询

s_id	s_name	s_sex	s_age	s_major	s_pho	s_credit
1	路明非	男	20	屠龙学	00000000000	14.00
2	凯撒·加图索	男	21	屠龙学	14987875664	12.00
3	楚子航	男	21	屠龙学	19878564234	14.00
4	诺诺	女	21	艺术		18.00
5	零	女	19	舞蹈		16.00
6	芬格尔	男	26	摆烂	14897987777	18.00

欢迎来到卡塞尔学院教务系统

教师信息 学生信息 课程信息 成绩信息

添加 删除 修改 查询 总分排名

s_id	c_id	t_id	score
1	2	3	80.00
1	3	1	100.00
1	4	2	85.00
2	1	2	95.00
2	2	3	50.00
2	3	1	98.00
3	2	2	100.00
3	3	1	100.00
3	4	2	99.00
4	1	1	90.00
4	2	1	65.00
4	3	1	90.50
4	4	2	100.00
5	1	1	100.00
5	2	3	100.00
5	3	1	95.00
6	1	2	65.00
6	2	1	70.00
6	3	3	65.00

修改学生凯撒不及格科目的成绩到 94 分，可以看到，他的已修学分也更新了

欢迎来到卡塞尔学院教务系统

教师信息 学生信息 课程信息 成绩信息

添加 删除 修改

s_id	c_id	t_id	score
1	2	3	80.00
1	3	1	100.00
1	4	2	85.00
2	1	2	95.00
2	2	3	50.00
2	3	1	98.00
3	2	2	100.00
3	3	1	100.00
3	4	2	99.00
4	1	1	90.00
4	2	1	65.00
4	3	1	90.50
4	4	2	100.00
5	1	1	100.00
5	2	3	100.00
5	3	1	95.00
6	1	2	65.00
6	2	1	70.00
6	3	3	65.00

增加课程

c_id(可不填) 2

c_name 2

c_credit 3

c_type 94

c_major

确定 重置

EducationalAdministrationSystem

更新成功

确定

欢迎来到卡塞尔学院教务系统

教师信息 学生信息 课程信息 成绩信息

添加 删除 修改 查询

s_id	s_name	s_sex	s_age	s_major	s_pho	s_credit
1	路明非	男	20	屠龙学	0000000000	14.00
2	凯撒·加图索	男	21	屠龙学	14987875664	16.00
3	楚子航	男	21	屠龙学	19878564234	14.00
4	诺诺	女	21	艺术		18.00
5	零	女	19	舞蹈		16.00
6	芬格尔	男	26	摆烂	14897987777	18.00

说明：违背存储过程，系统报错；

欢迎来到卡塞尔学院教务系统

教师信息 学生信息 课程信息 成绩信息

添加 删除 部分排名

s_id	c_id
1	3
1	4
2	3
3	2
3	3
3	4
4	3
4	4
5	1
5	2
6	1
6	2

录入成绩

s_id6

c_id1

t_id2

score20

确定 重置

EducationalAdministrationSystem

添加失败,请核对插入的字段是否有误

确定

程序演示
(2分)

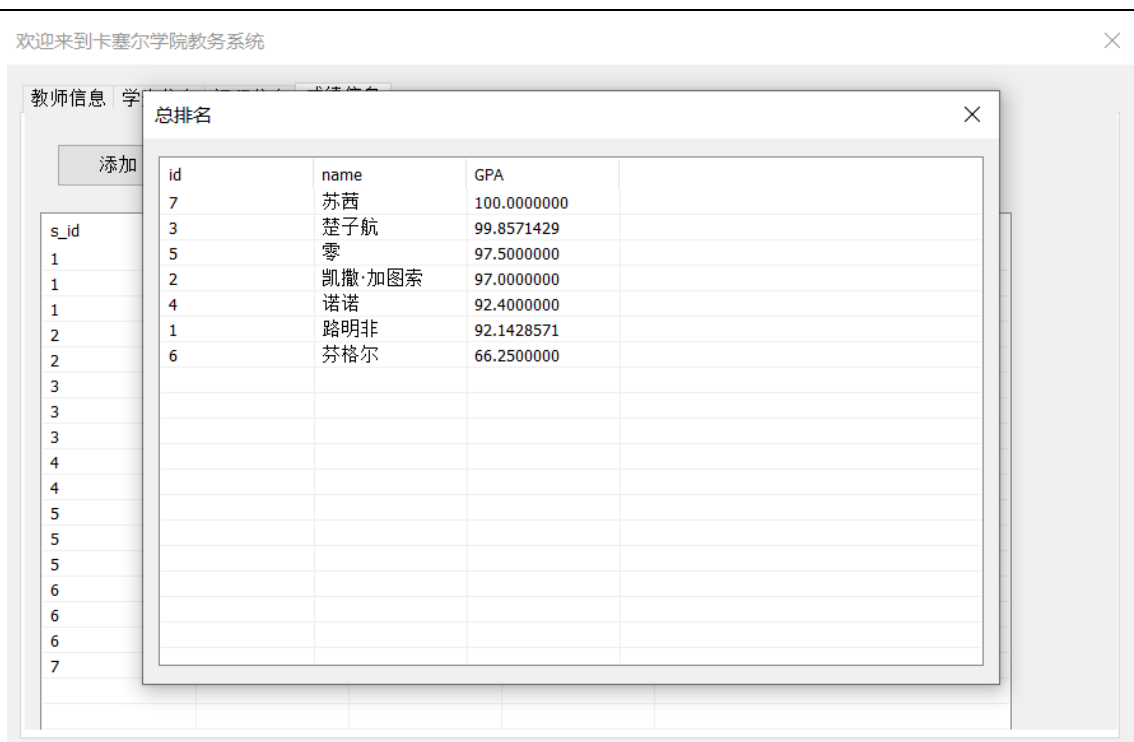
备注

7. 含有视图的查询操作 (15 分)

说明	<p>(1 分) 简要说明该操作所要完成的功能;</p> <p>(1 分) 简要说明建立的该视图的功能;</p> <p>(2 分) 简要说明该操作涉及的关系数据表 (以 “表名” 的形式给出)</p> <p>(1 分) 简要说明表连接涉及的字段 (以 “表 1. 属性=表 2. 属性”)</p> <p>(6 分) 实现该操作的关键代码 (高级语言、SQL), 截图即可;</p> <p>(4 分) 如何执行该操作, 按所述方法能够正常演示程序则给分。</p>
操作功能描述 (1 分)	学生的总排名可能会经常查看, 因此为所有学生 GPA 的排名建立视图, 方便操作
视图功能描述 (1 分)	计算所有学生的 GPA 并且为其单独设立一个视图来存放总排名
涉及的关系表 (2 分)	<p>student(<u>s_id</u>, <u>s_name</u>, s_sex, s_score, s_major, s_pho, s_credit)</p> <p>score(<u>s_id</u>, <u>c_id</u>, t_id, socre)</p> <p>course(<u>c_id</u>, c_name, c_credit, c_major)</p>
表连接字段 (1 分)	<p>student.s_id=score.s_id</p> <p>course.c_id=score.c_id</p>

<p>创建视图代码 (3分)</p>	<pre> 1 CREATE OR REPLACE VIEW view_grade 2 AS 3 SELECT 4 student.s_id, 5 student.s_name, 6 SUM(course.c_credit * score.score)/ SUM(course.c_credit) AS GPA 7 FROM 8 student 9 NATURAL JOIN course 10 NATURAL JOIN score 11 #student NATURAL JOIN score NATURAL JOIN course 12 13 GROUP BY 14 student.s_id 15 ORDER BY 16 GPA DESC 17 </pre>
<p>查询代码 (3分)</p>	<pre> char query[300]; USES_CONVERSION; sprintf_s(query, "%s", "SELECT * FROM view_grade"); mysql_query(mysql, "SET NAMES GBK");//设置编码格式, 否则cmd下中文乱码 if (mysql_query(mysql, query))return FALSE;//查询语句, 如果发生错误, 返回FALSE MYSQL_RES* res; MYSQL_ROW row; int row_num = 0; res = mysql_store_result(mysql); int field_num = res->field_count;//获取字段数 //设置列名 CString s[3]; s[0] = "id"; s[1] = "name"; s[2] = "GPA"; int i = 0; while (i < field_num) { m_listcontrol.InsertColumn(i, s[i], LVCFMT_LEFT, 120, i); i++; } //填充数据 while (row = mysql_fetch_row(res)) { CString cs; cs = row[0]; m_listcontrol.InsertItem(row_num, cs); for (int j = 1; j < field_num; j++) { cs = row[j]; m_listcontrol.SetItemText(row_num, j, cs); } row_num++; } </pre>

程序演示
(4分)



点击排名汇总按钮，可以看到总的 GPA 排名

备注