**学年论文**

**2023-2024学年第一学期**

**《Linux系统管理与编程环境》**

**题　　目：****基于Linux和c++的班级管理系统及示例设计**

**作　　者： 梁浩铂**

**所在学院： 计算机科学与技术学院**

**专业年级： 计算机21-1**

**指导教师： 李莉**

**2023 年 12 月 26日**

计算机科学与技术学院学生“Linux系统管理与编程环境”课程设计报告

**一、基本信息**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 学生姓名 | 学号 | 班级 | 课程名 | 任课教师 | 学期 |
| 梁浩铂 | 20211401206 | 计算机21-1 | Linux系统管理与编程环境 | 李莉 | 2023-2024学年第一学期 |

**二、设计目的**

通过课程学习，使学生熟练掌握Linux操作系统常用命令，Linux系统维护的原理与方法，理解Linux操作系统管理基础，培养并提高学生Linux系统的操作能力、维护能力和管理能力。

通过课程设计培养学生学习能力及查阅参考书籍、网络资源和文献资料的能力，并能熟练掌握Linux系统的基本概念和基本操作方法，能够在计算机上安装Linux操作系统及各种软件包，具有搭建基于Linux系统实验环境的能力。熟练掌握Linux常用命令操作方法，熟练运用Linux系统的常用管理工具，具有维护Linux系统的能力。

通过课程学习和设计开拓学生视野，深入体会并理解Linux系统的多种应用领域。更好地理解操作系统、文件系统、网络和安全等关键概念，提升学生的计算机科学知识和技能。

**三、设计要求**

1.综合应用Linux操作系统知识，根据选题搭建基于Linux系统的环境，在完成选题基本功能基础上，能够探究选题涉及到的解决方案，或程序开发涉及到的相关技术等。

2.撰写报告和设计文稿（学年论文），清晰阐述与Linux系统相关的管理、维护、设计和开发等相关技术。

**四、题目及任务书**

题目：基于Linux和c++的班级管理系统及示例设计

任务：

1. 掌握Linux操作系统相关基本知识
2. 掌握c++语言在Linux系统中的应用
3. 掌握MySQL数据库在Linux系统中的应用
4. 掌握使用c++语言连接MySQL数据库技术
5. 撰写课程设计论文

## 

## 摘 要

本文介绍了如何在Linux系统中使用C++语言连接MySQL数据库开发班级管理系统。班级管理系统是一种用于管理学生、教师和班级等信息的软件应用程序，它提供了学生信息管理、班主任信息管理、班级信息管理等功能。在本文中，我们首先介绍了MySQL数据库的基本概念和准备工作，包括在MySQL中创建相应的数据库和表。然后，我们讨论了如何使用C++语言编写程序来连接MySQL数据库，并执行常见的数据库操作，如查询、插入、更新和删除数据。最后，编写Makefile文件，使用make命令对源文件进行编译并生成可执行文件。本文提供了一个基本的指导，帮助读者了解如何在Linux环境下使用C++和MySQL进行数据库开发，为进一步开发更复杂的班级管理系统奠定了基础。

**关键词**：linux系统，c++，MySQL，班级管理系统，Makefile

**目 录**

[摘 要 4](#_Toc14406)

[1. MySQL数据库准备工作 7](#_Toc10093)

[1.1 准备工作 7](#_Toc32262)

[1.1.1 创建数据库和数据表 7](#_Toc24638)

[1.1.2 创建触发器 8](#_Toc20730)

[1.1.3 增添预设数据 8](#_Toc7239)

[1.1.4 c++连接数据库测试 10](#_Toc30416)

[2. 代码实现 11](#_Toc13080)

[2.1 mysql.h 11](#_Toc22276)

[2.2 mysql.cpp 11](#_Toc27559)

[2.3 class.h 12](#_Toc19576)

[2.4 class.cpp 12](#_Toc11781)

[2.5 teacher.h 15](#_Toc7372)

[2.6 teacher.cpp 15](#_Toc30690)

[2.7 student.h 19](#_Toc2897)

[2.8 student.cpp 19](#_Toc21023)

[2.9 management.h 23](#_Toc27862)

[2.10 management.cpp 23](#_Toc14331)

[2.11 main.cpp 26](#_Toc2720)

[2.12 Makefile文件 27](#_Toc9376)

[3. 运行结果分析 27](#_Toc22086)

[3.1 make编译 27](#_Toc25111)

[3.2 查看数据 28](#_Toc15688)

[3.3 添加数据 29](#_Toc19700)

[3.4 修改数据 31](#_Toc8867)

[3.5 删除数据 33](#_Toc6705)

[3.6 退出系统 34](#_Toc22803)

[总结 35](#_Toc30991)

[参考文献 36](#_Toc8379)

# MySQL数据库准备工作

## 准备工作

### 创建数据库和数据表

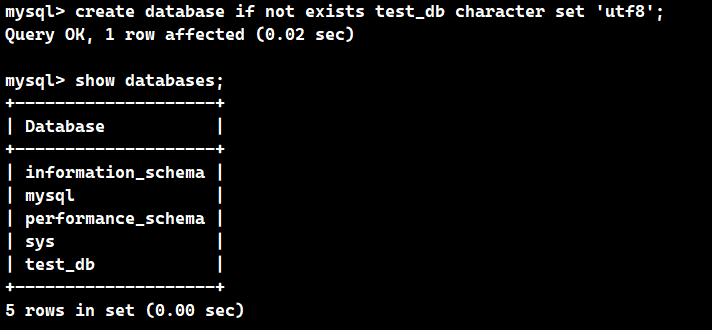


图1.2.1 创建数据库

# 创建学生表

create table if not exists student(

sno varchar(4) primary key,

sname varchar(15) not null,

sage int not null,

ssex varchar(3) not null check(ssex = '男' or ssex = '女'),

scno varchar(4) not null,

constraint fk\_stu\_cls\_no foreign key(scno) references class(cno)

);

# 创建教师表

create table if not exists teacher(

tno varchar(4) primary key,

tname varchar(15) not null,

tage int not null,

tsex varchar(3) not null check(tsex = '男' or tsex = '女'),

tcno varchar(4) not null,

constraint fk\_thr\_cls\_no foreign key(tcno) references class(cno)

);

# 创建班级表

create table if not exists class(

cno varchar(4) primary key,

cname varchar(40) not null,

cnum int default 0

);

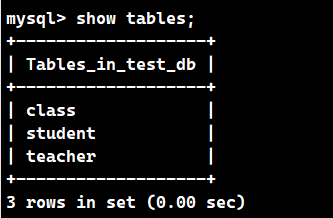


图1.2.2 创建数据表

### 创建触发器

# 增加

DELIMITER //

CREATE TRIGGER increase\_cnum AFTER INSERT ON student

FOR EACH ROW

BEGIN

UPDATE class SET cnum = cnum + 1 WHERE cno = NEW.scno;

END //

DELIMITER ;

# 减少

DELIMITER //

CREATE TRIGGER decrease\_cnum AFTER DELETE ON student

FOR EACH ROW

BEGIN

UPDATE class SET cnum = cnum - 1 WHERE cno = OLD.scno;

END //

DELIMITER ;

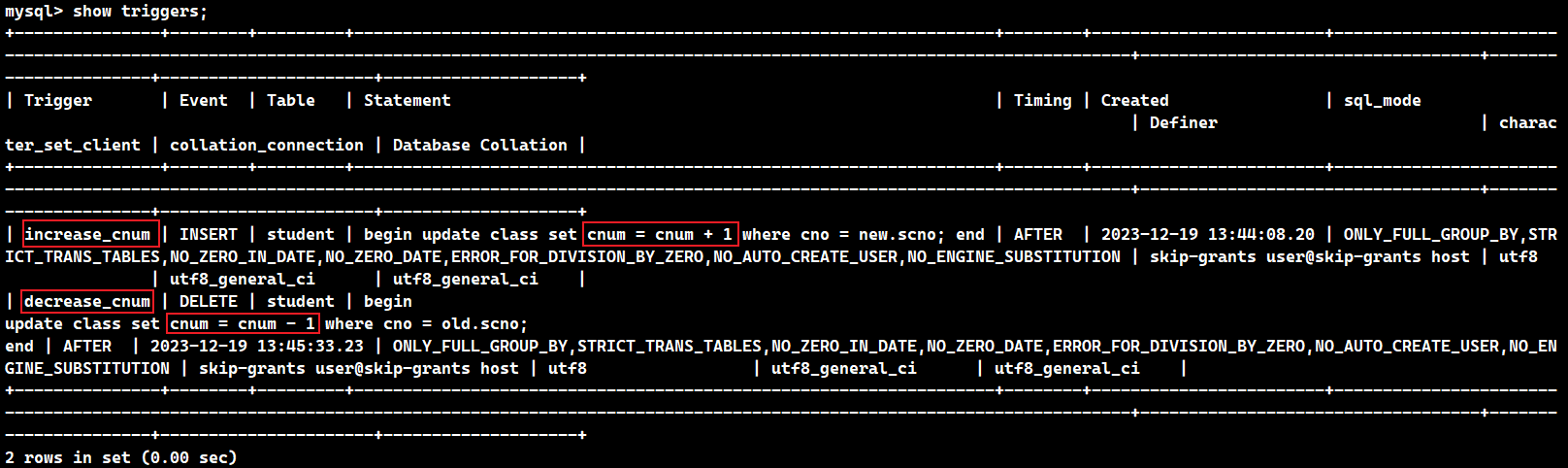


图1.2.3 创建触发器

### 增添预设数据

INSERT INTO class (cno, cname, cnum)

VALUES

('C001', 'Mathematics', 0),

('C002', 'Physics', 0),

('C003', 'Chemistry', 0);

INSERT INTO teacher (tno, tname, tage, tsex, tcno)

VALUES

('T001', 'John', 35, '男', 'C001'),

('T002', 'Mary', 28, '女', 'C002'),

('T003', 'David', 42, '男', 'C003');

INSERT INTO student (sno, sname, sage, ssex, scno)

VALUES

('S001', 'Alice', 20, '女', 'C001'),

('S002', 'Bob', 22, '男', 'C002'),

('S003', 'Charlie', 19, '男', 'C001');

INSERT INTO student (sno, sname, sage, ssex, scno)

VALUES

('S004', 'Emma', 21, '女', 'C002'),

('S005', 'Michael', 23, '男', 'C003'),

('S006', 'Olivia', 20, '女', 'C001');

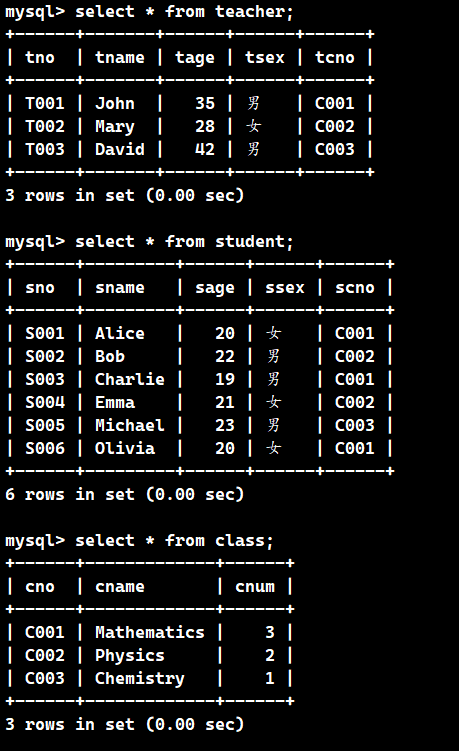


图1.2.4 添加预设数据

### c++连接数据库测试

#include <iostream>

#include <string>

#include <mysql/mysql.h>

using namespace std;

const string host = "127.0.0.1";

const string user = "root";

const string password = "";

const string db = "test\_db";

const int port = 3306;

int main()

{

// 1.初始化

MYSQL\* my = mysql\_init(NULL);

// 2.连接数据库

if(mysql\_real\_connect(my, host.c\_str(), user.c\_str(), password.c\_str(), db.c\_str(), port, NULL, 0) == NULL)

{

cout << "数据库连接失败！" << endl;

return 1;

}

cout << "连接成功" << endl;

// 3.关闭连接

mysql\_close(my);

return 0;

}

编译源程序：

g++ -o mysql test.cpp -L /lib64/mysql -l mysqlclient

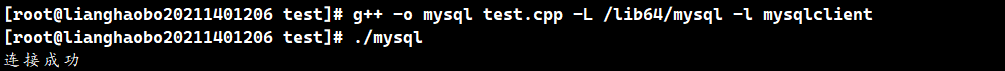


图1.2.5 测试连接数据库

# 代码实现

## mysql.h

#include <iostream>

#include <mysql/mysql.h>

using namespace std;

MYSQL\* get\_connect();

void close\_connect(MYSQL\* my);

## mysql.cpp

#include <iostream>

#include <mysql/mysql.h>

#include "mysql.h"

using namespace std;

const string host = "127.0.0.1";

const string user = "root";

const string password = "";

const string db = "test\_db";

const int port = 3306;

MYSQL\* get\_connect() {

//初始化连接

MYSQL\* my = mysql\_init(NULL);

//2.连接数据库

if (mysql\_real\_connect(my, host.c\_str(), user.c\_str(), password.c\_str(), db.c\_str(), port, NULL, 0) == NULL)

{

cout << "connect failed" << endl;

return my;

}

return my;

cout << "connect success" << endl;

}

void close\_connect(MYSQL\* my) {

//3.关闭连接

mysql\_close(my);

}

## class.h

#include <iostream>

#include <string>

using namespace std;

void class\_select();

void class\_insert();

void class\_update();

void class\_delete();

void \_class(int choice);

## class.cpp

#include <iostream>

#include "class.h"

#include "mysql.h"

using namespace std;

string cno;

string cname;

void class\_select(MYSQL\* my) {

cout << "\*\*\*\*\*\*\*\*\*\*\*\* 查询数据 \*\*\*\*\*\*\*\*\*\*\*\*" << endl;

string sql = "select \* from class;";

if (mysql\_query(my, sql.c\_str()) != 0)

{

cout << sql << " failed" << endl;

return;

}

//获取结果集

MYSQL\_RES\* res = mysql\_store\_result(my);

int row = mysql\_num\_rows(res); //数据行数

int col = mysql\_num\_fields(res); //数据列数

//获取每列的数据，并打印

MYSQL\_FIELD\* fields = mysql\_fetch\_field(res);

for (int i = 0; i < col; ++i)

{

cout << fields[i].name << "\t";

}

cout << endl;

//遍历数据

for (int i = 0; i < row; ++i)

{

//获取每一行

MYSQL\_ROW rows = mysql\_fetch\_row(res);

for (int j = 0; j < col; ++j)

{

cout << rows[j] << "\t";

}

cout << endl;

}

}

void class\_insert(MYSQL\* my) {

cout << "\*\*\*\*\*\*\*\*\*\*\*\* 插入数据 \*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "请输入编辑编号：" << endl;

cin >> cno;

cout << "请输入班级名称" << endl;

cin >> cname;

string sql = "insert into class(cno,cname,cnum) values ('" + cno + "','" + cname + "',0);";

if (mysql\_query(my, sql.c\_str()) != 0)

{

cout << sql << " failed." << endl;

return;

}

cout << sql << " success." << endl;

}

void class\_update(MYSQL\* my) {

cout << "\*\*\*\*\*\*\*\*\*\*\*\* 修改数据 \*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "请输入要修改的班级编号：" << endl;

cin >> cno;

string sql;

cout << "请输入班级名称：" << endl;

cin >> cname;

sql = "update class set cname='" + cname + "' where cno='" + cno + "';";

if (mysql\_query(my, sql.c\_str()) != 0)

{

cout << sql << " failed." << endl;

return;

}

cout << sql << " success." << endl;

}

void class\_delete(MYSQL\* my) {

cout << "\*\*\*\*\*\*\*\*\*\*\*\* 删除数据 \*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "请输入要删除的编辑编号：" << endl;

cin >> cno;

string sql = "delete from class where cno='" + cno + "';";

if (mysql\_query(my, sql.c\_str()) != 0)

{

cout << sql << " failed." << endl;

return;

}

cout << sql << " success." << endl;

}

void \_class(int choice) {

//获取数据库连接

MYSQL\* my = get\_connect();

//3.执行操作

//设置编码格式

mysql\_set\_character\_set(my, "utf8");

switch (choice) {

case 1:

class\_select(my);

break;

case 2:

class\_insert(my);

break;

case 3:

class\_update(my);

break;

case 4:

class\_delete(my);

break;

}

//关闭数据库连接

close\_connect(my);

}

## teacher.h

#include <iostream>

#include <string>

using namespace std;

void teacher\_select();

void teacher\_insert();

void teacher\_update();

void teacher\_delete();

void \_teacher(int choice);

## teacher.cpp

#include <iostream>

#include "teacher.h"

#include "mysql.h"

using namespace std;

string tno;

string tname;

string tsex;

string tage;

string tcno;

void teacher\_select(MYSQL\* my) {

cout << "\*\*\*\*\*\*\*\*\*\*\*\* 查询数据 \*\*\*\*\*\*\*\*\*\*\*\*" << endl;

string sql = "select \* from teacher;";

if (mysql\_query(my, sql.c\_str()) != 0)

{

cout << sql << " failed" << endl;

return;

}

//获取结果集

MYSQL\_RES\* res = mysql\_store\_result(my);

int row = mysql\_num\_rows(res); //数据行数

int col = mysql\_num\_fields(res); //数据列数

//获取每列的数据，并打印

MYSQL\_FIELD\* fields = mysql\_fetch\_field(res);

for (int i = 0; i < col; ++i)

{

cout << fields[i].name << "\t";

}

cout << endl;

//遍历数据

for (int i = 0; i < row; ++i)

{

//获取每一行

MYSQL\_ROW rows = mysql\_fetch\_row(res);

for (int j = 0; j < col; ++j)

{

cout << rows[j] << "\t";

}

cout << endl;

}

}

void teacher\_insert(MYSQL\* my) {

cout << "\*\*\*\*\*\*\*\*\*\*\*\* 插入数据 \*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "请输入教师工号：" << endl;

cin >> tno;

cout << "请输入教师姓名" << endl;

cin >> tname;

cout << "请输入教师年龄：" << endl;

cin >> tage;

cout << "请输入教师性别" << endl;

cin >> tsex;

cout << "请输入教师班级编号" << endl;

cin >> tcno;

string sql = "insert into teacher(tno,tname,tage,tsex,tcno) values ('" + tno + "','" + tname + "'," + tage + ",'" + tsex + "','" + tcno + "');";

if (mysql\_query(my, sql.c\_str()) != 0)

{

cout << sql << " failed." << endl;

return;

}

cout << sql << " success." << endl;

}

void teacher\_update(MYSQL\* my) {

cout << "\*\*\*\*\*\*\*\*\*\*\*\* 修改数据 \*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "请输入要修改的教师工号：" << endl;

cin >> tno;

cout << "\*\*\*\* 请选择要修改的属性： \*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\* 1.姓名 \*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\* 2.年龄 \*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\* 3.性别 \*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\* 4.班级编号 \*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

int choice;

cin >> choice;

string sql;

switch (choice) {

case 1:

cout << "请输入教师姓名" << endl;

cin >> tname;

sql = "update teacher set tname='" + tname + "' where tno='" + tno + "';";

break;

case 2:

cout << "请输入教师年龄：" << endl;

cin >> tage;

sql = "update teacher set tage=" + tage + " where tno='" + tno + "';";

break;

case 3:

cout << "请输入教师性别" << endl;

cin >> tsex;

sql = "update teacher set tsex='" + tsex + "' where tno='" + tno + "';";

break;

case 4:

cout << "请输入教师班级编号" << endl;

cin >> tcno;

sql = "update teacher set tcno='" + tcno + "' where tno='" + tno + "';";

break;

}

if (mysql\_query(my, sql.c\_str()) != 0)

{

cout << sql << " failed." << endl;

return;

}

cout << sql << " success." << endl;

}

void teacher\_delete(MYSQL\* my) {

cout << "\*\*\*\*\*\*\*\*\*\*\*\* 删除数据 \*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "请输入要删除的教师学号：" << endl;

cin >> tno;

string sql = "delete from teacher where sno='" + tno + "';";

if (mysql\_query(my, sql.c\_str()) != 0)

{

cout << sql << " failed." << endl;

return;

}

cout << sql << " success." << endl;

}

void \_teacher(int choice) {

//获取数据库连接

MYSQL\* my = get\_connect();

//3.执行操作

//设置编码格式

mysql\_set\_character\_set(my, "utf8");

switch (choice) {

case 1:

teacher\_select(my);

break;

case 2:

teacher\_insert(my);

break;

case 3:

teacher\_update(my);

break;

case 4:

teacher\_delete(my);

break;

}

//关闭数据库连接

close\_connect(my);

}

## student.h

#include <iostream>

#include <string>

using namespace std;

void student\_select();

void student\_insert();

void student\_update();

void student\_delete();

void \_student(int choice);

## student.cpp

#include <iostream>

#include "student.h"

#include "mysql.h"

using namespace std;

string sno;

string sname;

string ssex;

string sage;

string scno;

void student\_select(MYSQL\* my) {

cout << "\*\*\*\*\*\*\*\*\*\*\*\* 查询数据 \*\*\*\*\*\*\*\*\*\*\*\*" << endl;

string sql = "select \* from student;";

if (mysql\_query(my, sql.c\_str()) != 0)

{

cout << sql << " failed" << endl;

return;

}

//获取结果集

MYSQL\_RES\* res = mysql\_store\_result(my);

int row = mysql\_num\_rows(res); //数据行数

int col = mysql\_num\_fields(res); //数据列数

//获取每列的数据，并打印

MYSQL\_FIELD\* fields = mysql\_fetch\_field(res);

for (int i = 0; i < col; ++i)

{

cout << fields[i].name << "\t";

}

cout << endl;

//遍历数据

for (int i = 0; i < row; ++i)

{

//获取每一行

MYSQL\_ROW rows = mysql\_fetch\_row(res);

for (int j = 0; j < col; ++j)

{

cout << rows[j] << "\t";

}

cout << endl;

}

}

void student\_insert(MYSQL\* my) {

cout << "\*\*\*\*\*\*\*\*\*\*\*\* 插入数据 \*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "请输入学生学号：" << endl;

cin >> sno;

cout << "请输入学生姓名" << endl;

cin >> sname;

cout << "请输入学生年龄：" << endl;

cin >> sage;

cout << "请输入学生性别" << endl;

cin >> ssex;

cout << "请输入学生班级编号" << endl;

cin >> scno;

string sql = "insert into student(sno,sname,sage,ssex,scno) values ('" + sno + "','" + sname + "'," + sage + ",'" + ssex + "','" + scno + "');";

if (mysql\_query(my, sql.c\_str()) != 0)

{

cout << sql << " failed." << endl;

return;

}

cout << sql << " success." << endl;

}

void student\_update(MYSQL\* my) {

cout << "\*\*\*\*\*\*\*\*\*\*\*\* 修改数据 \*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "请输入要修改的学生学号：" << endl;

cin >> sno;

cout << "\*\*\*\* 请选择要修改的属性： \*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\* 1.姓名 \*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\* 2.年龄 \*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\* 3.性别 \*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\* 4.班级编号 \*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

int choice;

cin >> choice;

string sql;

switch (choice) {

case 1:

cout << "请输入学生姓名" << endl;

cin >> sname;

sql = "update student set sname='" + sname + "' where sno='" + sno + "';";

break;

case 2:

cout << "请输入学生年龄：" << endl;

cin >> sage;

sql = "update student set sage=" + sage + " where sno='" + sno + "';";

break;

case 3:

cout << "请输入学生性别" << endl;

cin >> ssex;

sql = "update student set ssex='" + ssex + "' where sno='" + sno + "';";

break;

case 4:

cout << "请输入学生班级编号" << endl;

cin >> scno;

sql = "update student set scno='" + scno + "' where sno='" + sno + "';";

break;

}

if (mysql\_query(my, sql.c\_str()) != 0)

{

cout << sql << " failed." << endl;

return;

}

cout << sql << " success." << endl;

}

void student\_delete(MYSQL\* my) {

cout << "\*\*\*\*\*\*\*\*\*\*\*\* 删除数据 \*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "请输入要删除的学生学号：" << endl;

cin >> sno;

string sql = "delete from student where sno='" + sno + "';";

if (mysql\_query(my, sql.c\_str()) != 0)

{

cout << sql << " failed." << endl;

return;

}

cout << sql << " success." << endl;

}

void \_student(int choice) {

//获取数据库连接

MYSQL\* my = get\_connect();

//3.执行操作

//设置编码格式

mysql\_set\_character\_set(my, "utf8");

switch (choice) {

case 1:

student\_select(my);

break;

case 2:

student\_insert(my);

break;

case 3:

student\_update(my);

break;

case 4:

student\_delete(my);

break;

}

//关闭数据库连接

close\_connect(my);

}

## management.h

#include <iostream>

#include "student.h"

#include "teacher.h"

#include "class.h"

using namespace std;

void menu();

int menu\_select();

void \_select();

void \_insert();

void \_update();

void \_delete();

void \_quit();

## management.cpp

#include <iostream>

#include <stdlib.h>

#include "management.h"

using namespace std;

void menu()

{

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\* 欢迎使用学生管理系统 \*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\* 1.查看数据 \*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\* 2.增加数据 \*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\* 3.修改数据 \*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\* 4.删除数据 \*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\* 0.退出系统 \*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << endl;

}

int menu\_select()

{

int select;

cout << "请选择要操作的数据的选项：" << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\* 请选择数据选项 \*\*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\* 1.学生 \*\*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\* 2.教师 \*\*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\* 3.班级 \*\*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\* 0.退出 \*\*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cin >> select;

return select;

}

void \_select()

{

int sel = menu\_select();

switch (sel) {

case 1:

\_student(1);

break;

case 2:

\_teacher(1);

break;

case 3:

\_class(1);

break;

case 0:

return;

default:

cout << "数据选项输入错误，退回主菜单！！！" << endl;

return;

}

}

void \_insert()

{

int sel = menu\_select();

switch (sel) {

case 1:

\_student(2);

break;

case 2:

\_teacher(2);

break;

case 3:

\_class(2);

break;

case 0:

return;

default:

cout << "数据选项输入错误，退回主菜单！！！" << endl;

return;

}

}

void \_update()

{

int sel = menu\_select();

switch (sel) {

case 1:

\_student(3);

break;

case 2:

\_teacher(3);

break;

case 3:

\_class(3);

break;

case 0:

return;

default:

cout << "数据选项输入错误，退回主菜单！！！" << endl;

return;

}

}

void \_delete()

{

int sel = menu\_select();

switch (sel) {

case 1:

\_student(4);

break;

case 2:

\_teacher(4);

break;

case 3:

\_class(4);

break;

case 0:

return;

default:

cout << "数据选项输入错误，退回主菜单！！！" << endl;

return;

}

}

void \_quit()

{

cout << "正在退出程序......" << endl;

cout << "欢迎下次使用" << endl;

exit(0);

}

## main.cpp

#include <iostream>

#include "management.h"

using namespace std;

int main()

{

int choice;

while (1) {

menu();

cout << "请输入选项：" << endl;

cin >> choice;

switch (choice) {

case 1:

\_select();

break;

case 2:

\_insert();

break;

case 3:

\_update();

break;

case 4:

\_delete();

break;

case 0:

\_quit();

break;

default:

break;

}

}

return 0;

}

## Makefile文件

objects=main.o management.o class.o student.o teacher.o mysql.o

test.exe:$(objects)

g++ $(objects) -o test.exe -L /lib64/mysql -l mysqlclient

class.o student.o teacher.o:mysql.h

management.o:class.h student.h teacher.h

.PHONY:clean

clean:

rm -rf \*.o

# 运行结果分析

## make编译

通过make命令可以执行Makefile文件，对c++源程序进行编译，之后使用make clean命令，来删除在编译过程生成的中间文件（即.o文件）。

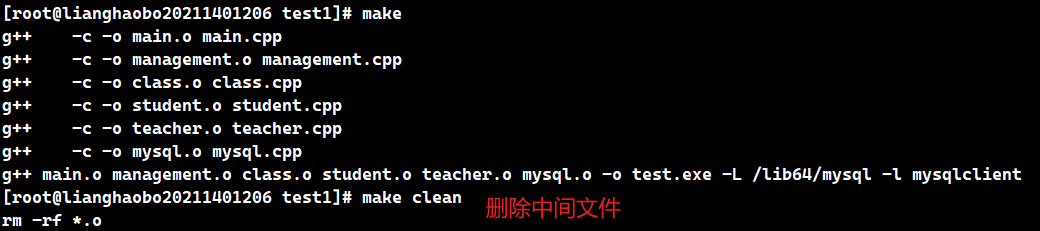


图3.1.1 编译

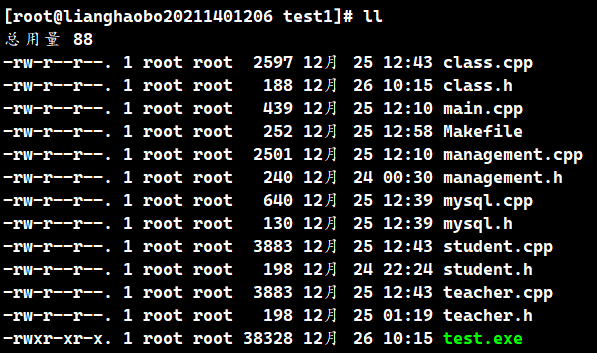


图3.1.2 查看生成文件

## 查看数据

进入系统，并查看预设的学生信息，在数据库中检验正确性



图3.2.1 进入系统

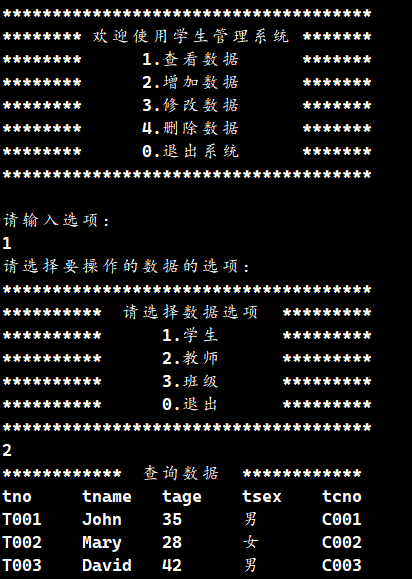


图3.2.2 查询学生数据

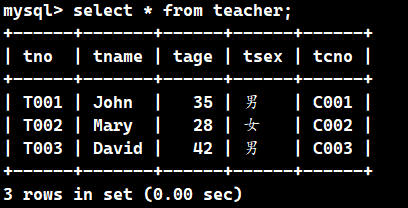


图3.2.3 在数据库中验证

## 添加数据

添加班级jsj21-1，并向这个班中添加学生lhb

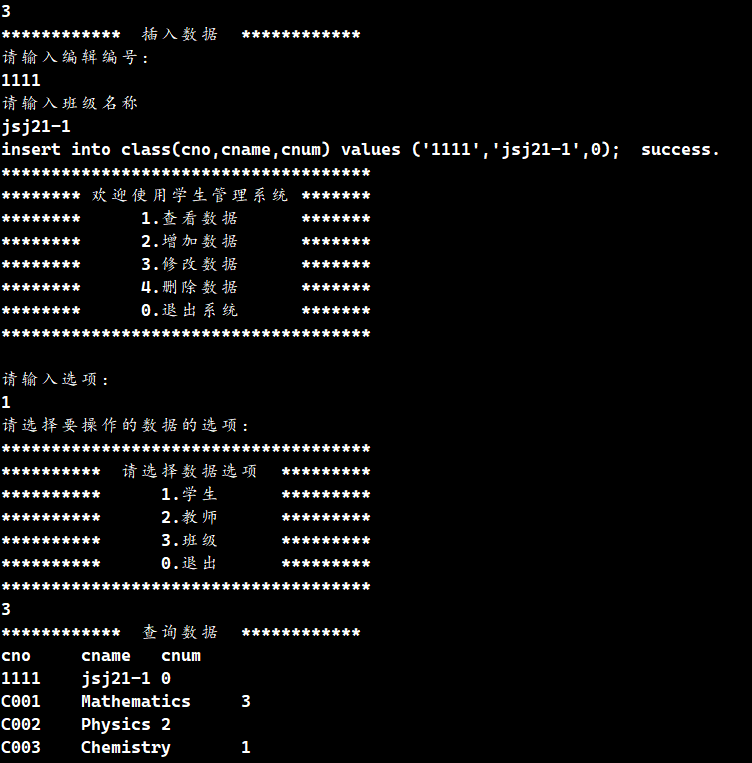


图3.3.1 添加班级jsj21-1并查询

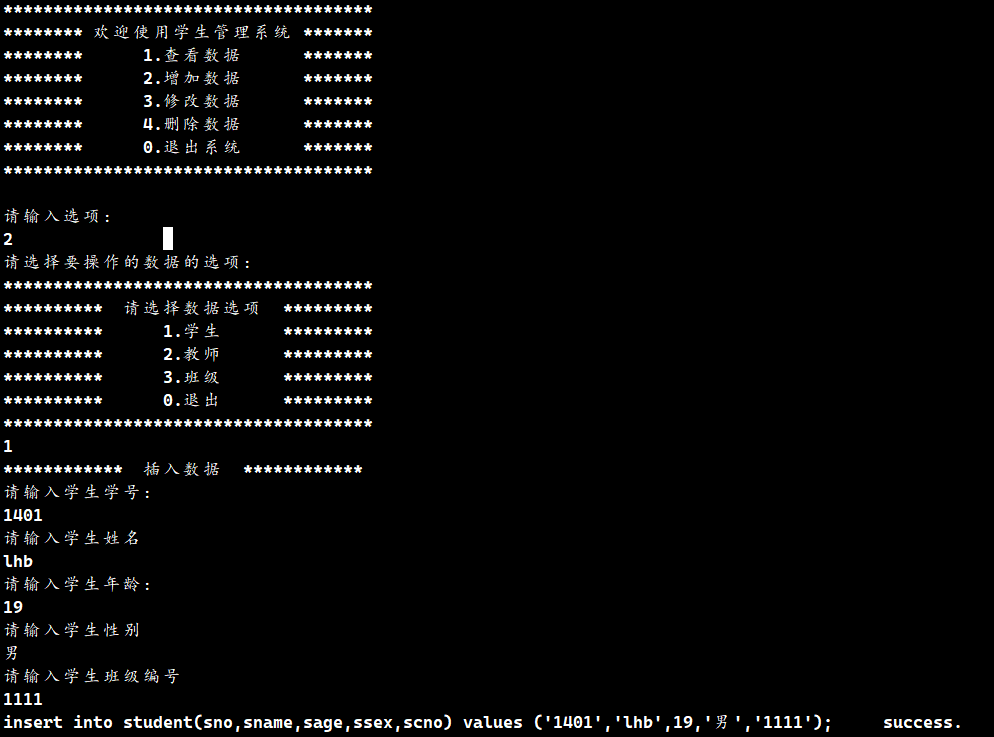


图3.3.2 添加学生lhb

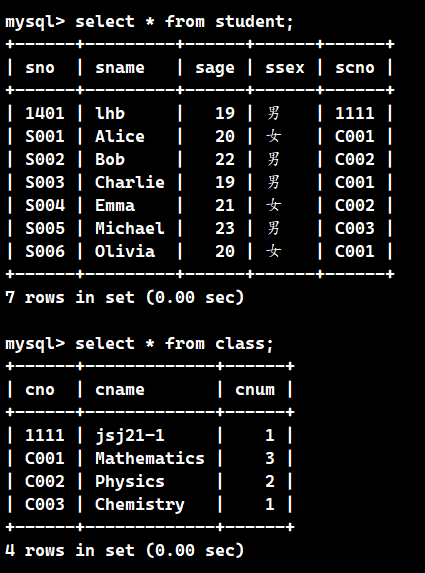


图3.3.3 在数据库中验证是否插入成功

## 修改数据

将学生lhb的年龄从19修改为20

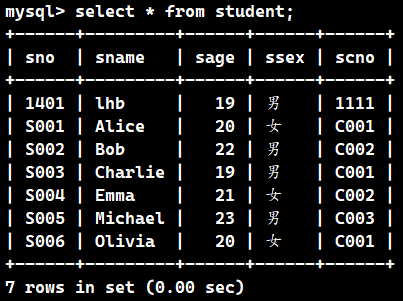


图3.4.1 展示当前学生数据

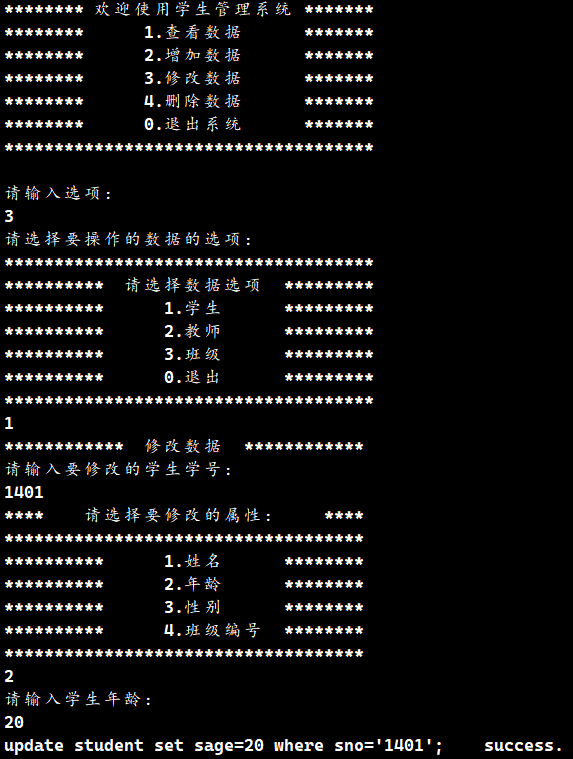


图3.4.2 将lhb的年龄修改为20

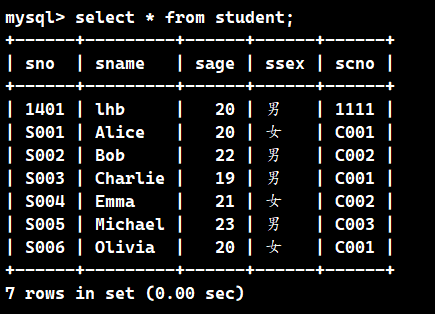


图3.4.3展示修改后的数据

## 删除数据

删除学生lhb

## image-20231226103844300

图3.5.1 展示现有数据

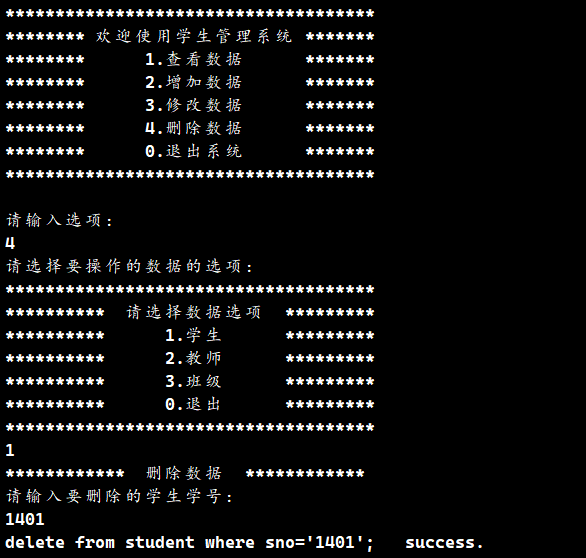


图3.5.2 删除学号为1401的学生

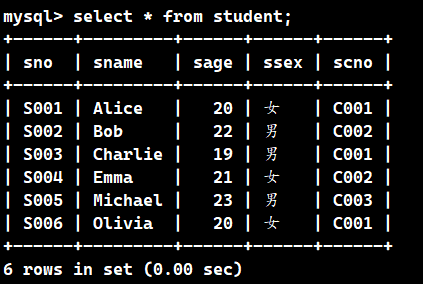


图3.5.3 展示删除后的数据

## 退出系统

## image-20231226104051945

图3.6.1 退出系统

总结

本论文旨在介绍在Linux操作系统中使用C++语言连接MySQL数据库进行班级管理系统开发的过程。通过本论文的研究，深入了解了Linux操作系统、C++编程语言和MySQL数据库的相关知识，并将其应用于实际的软件开发项目中。在开始时，介绍了MySQL数据库的基本概念。之后使用SQL语言在MySQL数据库管理系统中创建了test\_db数据库用于存放数据，又创建了class、teacher、student三张表，分别用于存放班级信息、班主任信息和学生信息，并使用简单的c++程序来验证能否正确地连接数据库。在系统实现中，编写正确的c++代码，展示了班级管理系统各项功能的具体实现方法，之后编写Makefile文件，说明各个文件之间的依赖关系，通过make编译之后生成了可执行文件，通过执行文件演示了使用C++语言编写代码执行SQL查询、插入、更新和删除等数据库操作。

参考文献

1. 梁如军,等. Linux基础及应用教程：基于CentOS7[M]. 2版. 北京：机械工业出版社，2016.5
2. 马忠梅,等. ARM&Linux嵌入式系统教程[M]. 3版. 北京：北京航空航天大学出版社，2014.10

**新疆大学课程论文（设计）、学年论文评分表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 题 目 | 基于Linux和c++的学生管理系统及示例设计 | | | | |
| 作 者 | 梁浩铂 | 专业年级 | 计算机21级 | 指导教师 | 李莉 |
| 指导教师评语及  评分建议 |  |  |  |  |  |
| |  |  |  | | --- | --- | --- | | 评价内容 | 具体要求 | 得分 | | 方案论证  （10分） | 能独立查阅文献和课题调研，能提出较正确、合理、可行的实施方案。 |  | | 论文(设计)内容  （30分） | 论文内容与题目以及任务书内容相吻合。 |  | | 工作量和难度  （20分） | 遵守课程论文（设计）管理制度，按时完成任务书规定的内容，工作量饱满。 |  | | 论文(设计)质量  （20分） | 结构合理、条理清楚、文理通顺、用语符合专业要求；文体格式规范、图表清楚。 |  | | 论文(设计)完整性  （20分） | 论文由学生独立完成，内容完整。 |  | | 总分（100分） | |  |   指导教师：  2023年 月 日 | | | | |
| 院  （部）  或  教  研  室  意  见 | 学院或教研室主任：  年 月 日 | | | | |