Mysql 综合应用设计

学　　院：电子信息（微纳技术）学院

专　　业：微电子科学与工程

年　　级：2019 级

姓　　名：皇甫素素

学　　号：2019205883

2021 年 12 月 20 日

声明

本项目采用 **MIT** 许可协议，代码仓库如下：

https://github.com/suyu610/qt_mysql

This project is licensed under MIT

目录

一、需求分析

1. 基本需求

分别使用 C 语言、QT 和 Web 访问数据库中的成绩。

2. 拓展需求

能对成绩进行简单的增删改查(**未做**)

二、环境安装

1. 更新下载源

> sudo vim /etc/apt/sources.list

deb http://mirrors.aliyun.com/ubuntu/ xenial-security universe

deb http://mirrors.aliyun.com/ubuntu/ xenial-security multiverse

...

> apt-get update

> apt-get upgrade

2. Mysql 的安装与配置

(1)使用 apt-get 安装 mysql

> sudo apt-get install mysql-server

(2)配置 MySQL

> sudo mysql_secure_installation

(3)检查 MySQL 启动情况

> systemctl status mysql.service

```
root1@ubuntu:~$ systemctl status mysql.service
● mysql.service - MySQL Community Server
     Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)
     Active: active (running) since Sun 2021-12-19 09:07:56 PST; 1min 55s ago
   Main PID: 7064 (mysqld)
     Status: "Server is operational"
      Tasks: 38 (limit: 4614)
     Memory: 333.0M
     CGroup: /system.slice/mysql.service
             └─7064 /usr/sbin/mysqld

Dec 19 09:07:55 ubuntu systemd[1]: Starting MySQL Community Server...
Dec 19 09:07:56 ubuntu systemd[1]: Started MySQL Community Server.
```

(4) 测试访问 MySQL

> mysql -uroot -p

```
root1@ubuntu:~$ sudo mysql -uroot -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 14
Server version: 8.0.25-0ubuntu0.20.10.1 (Ubuntu)

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

(5) 设置访问 MySQL 权限

> alter user 'root'@'localhost' identified with mysql_native_password by 'your password'

3. QT5.9 安装

(1) 下载 QT

https://mirrors.tuna.tsinghua.edu.cn/qt/archive/qt/5.9/5.9.0/qt-opensource-linux-x64-5.9.0.run

(2) 安装

```
root1@ubuntu:~$ cd Downloads/
root1@ubuntu:~/Downloads$ ls
qt-opensource-linux-x64-5.9.0.run
root1@ubuntu:~/Downloads$ sudo chmod -R 777 qt-opensource-linux-x64-5.9.0.run
root1@ubuntu:~/Downloads$ sudo ./qt-opensource-linux-x64-5.9.0.run
QStandardPaths: XDG_RUNTIME_DIR not set, defaulting to '/tmp/runtime-root'
```

(3)其他组件的安装

> sudo apt-get install gcc g++

> sudo apt-get install libqt4-dev

> sudo apt-get install build-essential

(4)检查 mysql 驱动是否安装好

> cd /opt/Qt5.9.0/5.9/gcc_64/plugins/sqldrivers

> ldd libqsqlmysql.so

```
root1@ubuntu:/opt/Qt5.9.0/5.9/gcc_64/plugins/sqldrivers$ ldd libqsqlmysql.so
        linux-vdso.so.1 (0x00007ffda7ecd000)
        libQt5Sql.so.5 => /opt/Qt5.9.0/5.9/gcc_64/plugins/sqldrivers/./../../lib/libQt5Sql.so.5
        libQt5Core.so.5 => /opt/Qt5.9.0/5.9/gcc_64/plugins/sqldrivers/./../../lib/libQt5Core.so
        libpthread.so.0 => /lib/x86_64-linux-gnu/libpthread.so.0 (0x00007f245ed10000)
        libmysqlclient.so.18 => not found
        libstdc++.so.6 => /lib/x86_64-linux-gnu/libstdc++.so.6 (0x00007f245eb2e000)
        libm.so.6 => /lib/x86_64-linux-gnu/libm.so.6 (0x00007f245e9df000)
        libgcc_s.so.1 => /lib/x86_64-linux-gnu/libgcc_s.so.1 (0x00007f245e9c2000)
        libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x00007f245e7d8000)
        libicui18n.so.56 => /opt/Qt5.9.0/5.9/gcc_64/plugins/sqldrivers/./../../lib/libicui18n.s
        libicuuc.so.56 => /opt/Qt5.9.0/5.9/gcc_64/plugins/sqldrivers/./../../lib/libicuuc.so.56
        libicudata.so.56 => /opt/Qt5.9.0/5.9/gcc_64/plugins/sqldrivers/./../../lib/libicudata.s
```

如果出现 libmysqlclient.so.18 => Not found

则> sudo wget -O /usr/lib/libmysqlclient.so.18

http://files.directadmin.com/services/es_7.0_64/libmysqlclient.so.18

(5) 启动 QT

> cd opt/Qt5.9.0/Tools/QtCreator/bin

> ./qtcreator

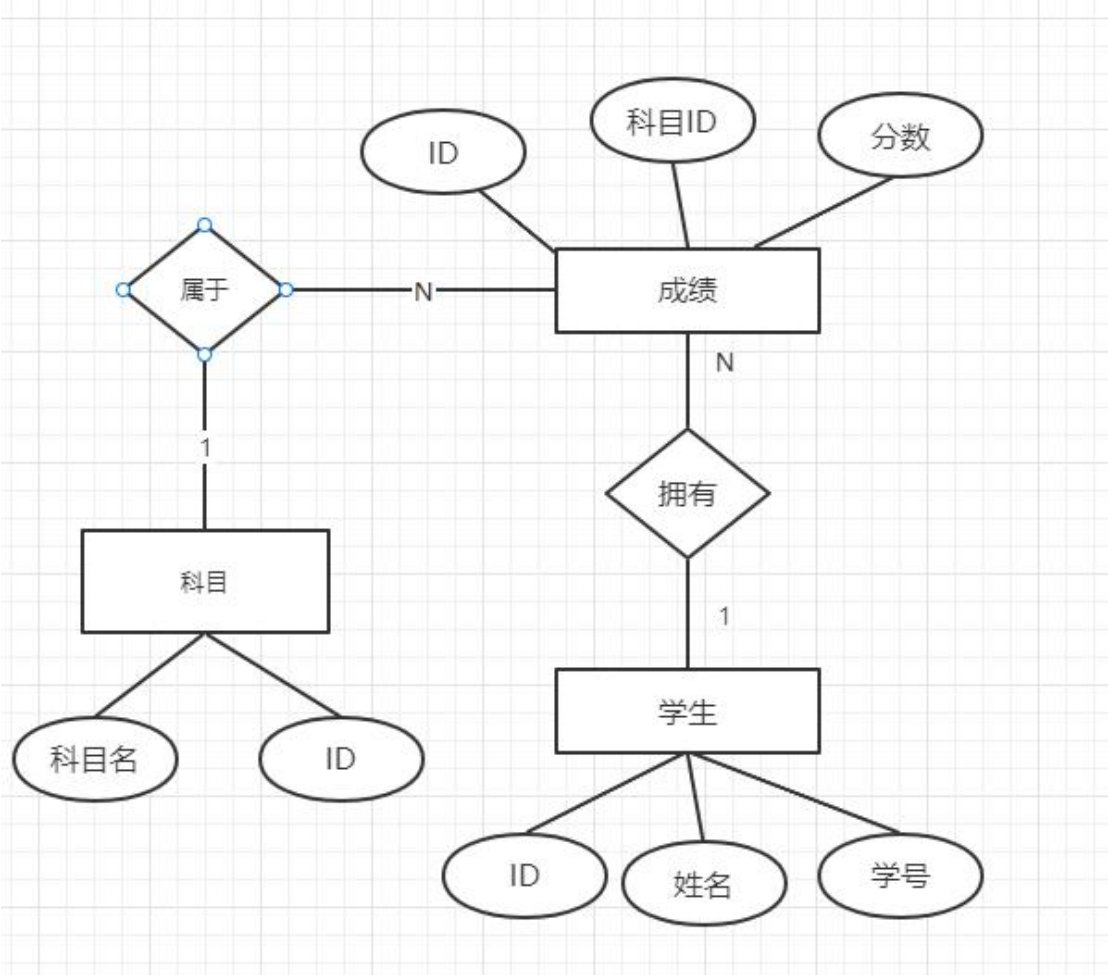
三、业务逻辑

1. 数据库设计

```
mysql> desc grade;
+------------+--------------+------+-----+---------+----------------+
| Field      | Type         | Null | Key | Default | Extra          |
+------------+--------------+------+-----+---------+----------------+
| grade_id   | int          | NO   | PRI | NULL    | auto_increment |
| stu_id     | int          | YES  |     | NULL    |                |
| subject_id | int          | YES  |     | NULL    |                |
| score      | decimal(4,1) | YES  |     | NULL    |                |
+------------+--------------+------+-----+---------+----------------+
4 rows in set (0.01 sec)

mysql> desc stu;
+----------+--------------+------+-----+---------+----------------+
| Field    | Type         | Null | Key | Default | Extra          |
+----------+--------------+------+-----+---------+----------------+
| stu_id   | int          | NO   | PRI | NULL    | auto_increment |
| stu_num  | varchar(30)  | YES  |     | NULL    |                |
| stu_name | varchar(255) | YES  |     | NULL    |                |
+----------+--------------+------+-----+---------+----------------+
3 rows in set (0.00 sec)

mysql> desc subject
    -> ;
+--------------+--------------+------+-----+---------+----------------+
| Field        | Type         | Null | Key | Default | Extra          |
+--------------+--------------+------+-----+---------+----------------+
| subject_id   | int          | NO   | PRI | NULL    | auto_increment |
| subject_name | varchar(255) | YES  |     | NULL    |                |
+--------------+--------------+------+-----+---------+----------------+
2 rows in set (0.00 sec)
```



2. 业务流程图

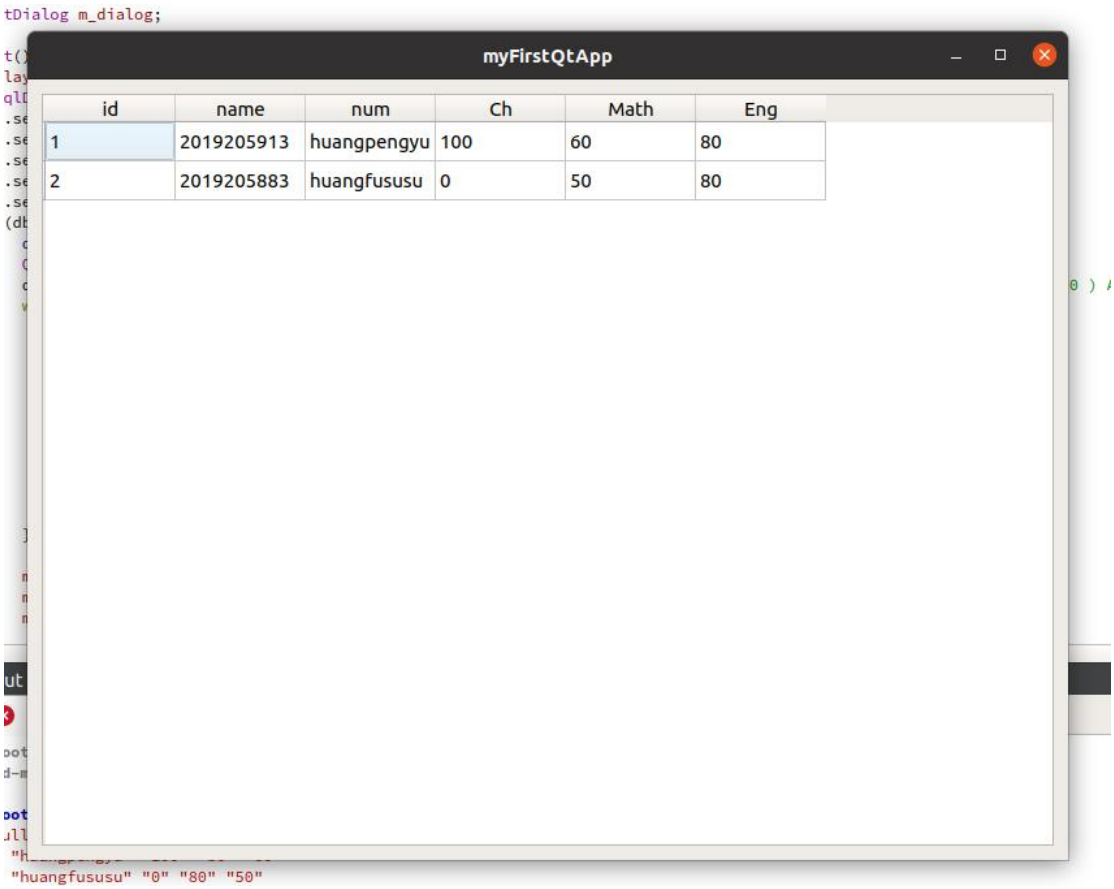
四、效果展示

1. Native-C

> gcc -o sql_c sql_c.c -I/usr/include/mysql -L/usr/lib64/mysql -lmysqlclient

```
[root@VM-0-11-centos ~]# ./sql_c
MySQL client version : 80026
Success connecting
The Result is
(1) 学号: 2019205913 姓名:          hpy        ||      语文: 100.0    数学:  80.0    英语: 60.0
(2) 学号: 2019205883 姓名:   huangfususu       ||      语文:  63.5    数学:  80.0    英语: 50.0
```

2. QT



3. Web

略

五、附录

1. 代码

① Pure_C 连接 Mysql

```c
#include <mysql.h>
#include <stdio.h>
#include <stdlib.h>
int main(){
   MYSQL *conn;
   MYSQL_RES *res;
   MYSQL_ROW row;
   conn = mysql_init(NULL);
   printf("MySQL client version : %d \n",mysql_get_client_version());
   if(!mysql_real_connect(conn,"127.0.0.1","root","HP!!yu610!!++199119","grade_db",0,NULL,0)){
      puts("Error connecting");
      printf("%s\n",mysql_error(conn));
   }
   else{
      puts("Success connecting");
   }
   /* send SQL query */
   if (mysql_query(conn, "SELECT stu_id,stu_num, stu_name,(SELECT score FROM grade WHERE stu_id = stu.stu_id AND subject_id =
      fprintf(stderr, "%s\n", mysql_error(conn));
      exit(0);
   }
   res = mysql_use_result(conn);
   printf("The Result is\n");
   while ((row = mysql_fetch_row(res)) != NULL){
      printf("(%s) 学号: %10s 姓名： %12s \t||\t 语文: %5s\t数学: %5s\t英语:%5s \n", row[0],row[1],row[2],row[3],row[4],row[5]);
   }

   /* close connection */
   mysql_free_result(res);
   mysql_close(conn);
   return 0;
}
```


```c
#include <mysql.h>
#include <stdio.h>
#include <stdlib.h>
int main(){
        MYSQL *conn;
        MYSQL_RES *res;
        MYSQL_ROW row;
        conn = mysql_init(NULL);
        printf("MySQL client version : %d \n",mysql_get_client_version());
        if(!mysql_real_connect(conn,"127.0.0.1","root","密码","grade_db",0,NULL,0)){
                puts("Error connecting");
                printf("%s\n",mysql_error(conn));
         }
          else{
                puts("Success connecting");
            }
        /* send SQL query */
        if (mysql_query(conn, "SELECT stu_id,stu_num, stu_name,(SELECT score FROM grade WHERE stu_id = stu.stu_id AND subject_id = 0 ) AS ch,(SELECT score FROM grade WHERE stu_id = stu.stu_id AND subject_id = 1 ) AS math,(SELECT score FROM grade WHERE stu_id = stu.stu_id AND subject_id = 2 ) AS eng FROM stu;")) {
                fprintf(stderr, "%s\n", mysql_error(conn));
                exit(0);
        }
        res = mysql_use_result(conn);
        printf("The Result is\n");
        while ((row = mysql_fetch_row(res)) != NULL){
                printf("(%s) 学号: %10s 姓名： %12s \t||\t 语文: %5s\t 数学: %5s\t 英语:%5s \n", row[0],row[1],row[2],row[3],row[4],row[5]);
```

```
        }

        /* close connection */
        mysql_free_result(res);
        mysql_close(conn);
        return 0;
}
```

② QT

考虑到本次项目主要目的为学习 QT 与 MySQL 在 unix 下的应用，所以对于业务层并没有做过多的设计与解耦。

Main.cpp

```
#include <QApplication>
#include <QDebug>
#include <QtSql>
#include <QtGui>
#if QT_VERSION_MAJOR > 4
#include <QtWidgets>
#endif

class StuGrade {
    QString m_stuid, m_stuname, m_stunum,m_ch,m_math,m_eng;
public:
    StuGrade(const QString & stuid, const QString & stuname, const QString & stunum,
const QString & ch, const QString & math, const QString & eng) :
        m_stuid{stuid},                                    m_stuname{stuname},
m_stunum{stunum},m_ch{ch},m_math(math),m_eng{eng} {}
    QString stuid() const { return m_stuid; }
    QString stuname() const { return m_stuname; }
    QString stunum() const { return m_stunum; }
    QString ch() const { return m_ch; }
    QString math() const { return m_math; }
    QString eng() const { return m_eng; }
 };

class StuGradeModel : public QAbstractTableModel {
    QList<StuGrade> m_data;
public:
    StuGradeModel(QObject * parent = {}) : QAbstractTableModel{parent} {}
    int rowCount(const QModelIndex &) const override { return m_data.count(); }
    int columnCount(const QModelIndex &) const override { return 6; }
```

```
    QVariant data(const QModelIndex &index, int role) const override {
        if (role != Qt::DisplayRole && role != Qt::EditRole) return {};
        const auto & stu = m_data[index.row()];
        switch (index.column()) {
        case 0: return stu.stuid();
        case 1: return stu.stuname();
        case 2: return stu.stunum();
        case 3: return stu.ch();
        case 4: return stu.eng();
        case 5: return stu.math();
        default: return {};
        };
    }
    QVariant headerData(int section, Qt::Orientation orientation, int role) const override {
        if (orientation != Qt::Horizontal || role != Qt::DisplayRole) return {};
        switch (section) {
        case 0: return "id";
        case 1: return "name";
        case 2: return "num";
        case 3: return "Ch";
        case 4: return "Math";
        case 5: return "Eng";

        default: return {};
        }
    }
    void append(const StuGrade & stu) {
        beginInsertRows({}, m_data.count(), m_data.count());
        m_data.append(stu);
        endInsertRows();
    }
};

class Widget : public QWidget {
    QGridLayout m_layout{this};
    QTableView m_view;
    StuGradeModel m_model;
    QSortFilterProxyModel m_proxy;
    QInputDialog m_dialog;
public:
    Widget() {
        m_layout.addWidget(&m_view, 0, 0, 1, 1);
        QSqlDatabase    db    = QSqlDatabase::addDatabase("QMYSQL");
        db.setHostName("127.0.0.1");
```

```
        db.setDatabaseName("grade_db");
        db.setPort(3306);
        db.setUserName("root");
        db.setPassword("HPyuko12!!");
        if(db.open()){
            qDebug()<<"connect successfully!";
            QSqlQuery query(db);
            query.exec("SELECT stu_id,stu_num, stu_name,(SELECT score FROM grade
WHERE stu_id = stu.stu_id AND subject_id = 0 ) AS ch,(SELECT score FROM grade
WHERE stu_id = stu.stu_id AND subject_id = 1 ) AS math,(SELECT score FROM grade
WHERE stu_id = stu.stu_id AND subject_id = 2 ) AS eng FROM stu;");
            while(query.next()){
                QString id = query.value(0).toString();
                QString stu_num = query.value(1).toString();
                QString stu_name = query.value(2).toString();
                QString ch = query.value(3).toString();
                QString math = query.value(4).toString();
                QString eng = query.value(5).toString();

                m_model.append({id,stu_num,stu_name, ch, math, eng});

                qDebug()<< id << stu_num << stu_name << ch << math << eng;
            }

            m_proxy.setSourceModel(&m_model);
            m_proxy.setFilterKeyColumn(2);
            m_view.setModel(&m_proxy);
        }
        else
            qDebug()<<"failed!!!!";
    }
};

int main(int argc, char *argv[])
{
    QApplication a{argc, argv};
    Widget w;
    w.resize(800,600);
    w.show();
    return a.exec();
}
```

2. Sql 语句

- 初始化

```
CREATE DATABASE grade_db;
DROP TABLE IF EXISTS `grade_db`.`grade`;
DROP TABLE IF EXISTS `grade_db`.`stu`;
DROP TABLE IF EXISTS `grade_db`.`subject`;
CREATE TABLE `grade` (
  `grade_id` int NOT NULL AUTO_INCREMENT,
  `stu_id` int DEFAULT NULL,
  `subject_id` int DEFAULT NULL,
  `score` decimal(4,1) DEFAULT NULL,
  PRIMARY KEY (`grade_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

CREATE TABLE `stu` (
  `stu_id` int NOT NULL AUTO_INCREMENT,
  `stu_num` varchar(30) COLLATE utf8mb4_general_ci DEFAULT NULL,
  `stu_name` varchar(255) COLLATE utf8mb4_general_ci DEFAULT NULL,
  PRIMARY KEY (`stu_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

CREATE TABLE `subject` (
  `subject_id` int NOT NULL AUTO_INCREMENT,
  `subject_name` varchar(255) COLLATE utf8mb4_general_ci DEFAULT NULL,
  PRIMARY KEY (`subject_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

- **Stu.sql**

```
INSERT INTO `stu` (`stu_id`, `stu_num`, `stu_name`) VALUES (1, '2019205913', 'hpy');
INSERT INTO `stu` (`stu_id`, `stu_num`, `stu_name`) VALUES (2, '2019205883', 'huangfususu');
```

- **Grade.sql**

```
INSERT INTO `grade` (`grade_id`, `stu_id`, `subject_id`, `score`) VALUES (0, 1, 0, 100.0);
INSERT INTO `grade` (`grade_id`, `stu_id`, `subject_id`, `score`) VALUES (1, 1, 1, 80.0);
INSERT INTO `grade` (`grade_id`, `stu_id`, `subject_id`, `score`) VALUES (2, 1, 2, 60.0);
INSERT INTO `grade` (`grade_id`, `stu_id`, `subject_id`, `score`) VALUES (3, 2, 0, 63.5);
INSERT INTO `grade` (`grade_id`, `stu_id`, `subject_id`, `score`) VALUES (4, 2, 1, 80.0);
INSERT INTO `grade` (`grade_id`, `stu_id`, `subject_id`, `score`) VALUES (5, 2, 2, 50.0);
```

- **Subject.sql**

```
INSERT INTO `subject` (`subject_id`, `subject_name`) VALUES (0, 'ch');
INSERT INTO `subject` (`subject_id`, `subject_name`) VALUES (1, 'math');
INSERT INTO `subject` (`subject_id`, `subject_name`) VALUES (2, 'eng');
```

3. 参考文献

① Ubuntu18.04 上安装 Qt5.10 步骤
https://blog.csdn.net/weixin_41477306/article/details/95743555