**数据库系统原理课程设计**

**指 导 书**

**编者：吕泽华**

**2025**

**华中科技大学 软件学院**

**目 录**

[一、课程简介及基本要求 3](#_Toc181714721)

[二、实验课程目的与要求 3](#_Toc181714722)

[三、主要仪器设备 4](#_Toc181714723)

[四、实验方式与基本要求 4](#_Toc181714724)

[五、考核与报告 4](#_Toc181714725)

[六、实验项目设置与内容 4](#_Toc181714726)

[实验1 熟悉MySQL的安装、使用环境及其基本工具的使用 6](#_Toc181714727)

[实验2 数据库的创建和管理 20](#_Toc181714728)

[实验3 表的创建和管理 28](#_Toc181714729)

[实验4 数据库的完整性设计 33](#_Toc181714730)

[实验5 表数据的简单查询 35](#_Toc181714731)

[实验6 表数据的维护 41](#_Toc181714732)

[实验7 视图管理 43](#_Toc181714733)

[实验8 数据类型的使用 48](#_Toc181714734)

[实验9 表数据的高级查询 54](#_Toc181714735)

[实验10 索引管理 60](#_Toc181714736)

[实验11 数据库的安全性设计 62](#_Toc181714737)

[实验12 触发器实验 65](#_Toc181714738)

[实验13 存储过程实验 68](#_Toc181714739)

[实验14 函数与事件 72](#_Toc181714740)

[实验15 数据库的备份与恢复 77](#_Toc181714741)

[实验16 数据库监视与性能优化实验 87](#_Toc181714742)

[实验17 并发控制技术实验 89](#_Toc181714743)

[实验18 数据库设计 90](#_Toc181714744)

[实验19 数据库应用开发实验 94](#_Toc181714745)

[实验20 综合实验 108](#_Toc181714746)

[七、实验报告模板 110](#_Toc181714747)

**课程编号：**SSE0251

**课程名称：**数据库系统原理课程设计

**课程属性：**专业课

**适用专业：**软件工程

**先修课程要求：**计算机导论、程序设计基础、数据结构等

**总学时：**32学时

**学分：**1学分

**应开实验学期：**三年级第一学期

**教材：**自编课程指导书

## 一、课程简介及基本要求

本课程从基本原理和应用实例两方面介绍关系型数据库系统及使用方法，使学生能够在熟悉数据库系统基本操作及应用的同时，具备基本的数据库应用系统开发的技能，为将来从事数据库应用系统的开发及数据库管理系统的研发工作打下良好的基础。

## 二、实验课程目的与要求

本实验课程可以使学生在理论学习之后，对课程内容有一个直观的了解；并通过实际操作掌握数据库应用系统开发的各种技能。

本实验课程要求学生达到如下基本要求：

1、掌握MySQL8.0的基本情况和基本使用方法；

2、掌握MySQL8.0中各级基本数据库对象的管理，包括：

(1) 数据库的创建和各种基本管理操作；

(2) 数据表格的创建和各种基本管理操作；

(3) 表格数据的编辑和维护；

(4) 表格数据的各种查询操作；

(5) 有关视图的操作。

3、熟悉MySQL8.0中数据库完整性的实现；

4、了解MySQL8.0中数据库安全性的实现；

5、熟悉MySQL8.0中数据库的备份和恢复；

6、数据库性能监视与优化；

7、数据库的并发控制；

8、数据库应用系统的开发。

## 三、主要仪器设备

本实验课程要求的主要仪器设备是计算机。

## 四、实验方式与基本要求

本实验课程以学生按照指导书上机操作为主，教师在课堂教学时进行演示和答疑为辅。

## 五、考核与报告

采用考察的考核方式，平时成绩占30%，实验成绩占70%。

## 六、实验项目设置与内容

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **序号** | **实验名称** | **内容摘要** | **实验学时** | **每组人数** | **实验属性** | **实验者类别** | **开出要求** |
| 1 | 熟悉MySQL8的安装、使用环境 | 了解MySQL8的界面环境；熟悉服务管理器、企业管理器和查询分析器的使用方法 | 1 | 1 | 验证 | 本科 | 必做（课内） |
| 2 | 数据库的创建和管理 | 创建一个数据库；并对其进行基本的管理操作 | 1 | 1 | 验证 | 本科 | 必做（课内） |
| 3 | 表的创建和操作 | 创建若干个数据表格；并对它们进行基本的管理操作 | 2 | 1 | 验证 | 本科 | 必做（课内） |
| 4 | 数据库完整性的实现 | 实现关系型数据库的完整性 | 2 | 1 | 验证 | 本科 | 必做（课内） |
| 5 | 表格数据的基本查询操作 | 对表格数据进行简单的查询操作 | 1 | 1 | 设计 | 本科 | 必做（课内） |
| 6 | 表格数据的编辑和维护 | 对表格数据进行插入、删除和修改操作 | 1 | 1 | 验证 | 本科 | 必做（课内） |
| 7 | 视图管理 | 创建视图；对视图进行基本的管理操作 | 1 | 1 | 设计 | 本科 | 必做（课内） |
| 8 | 数据类型的使用 |  | 1 | 1 | 设计 | 本科 | 选做（课内） |
| 9 | 表格数据的高级查询操作 | 多表连接查询；统计查询；嵌套查询等 | 2 | 1 | 设计 | 本科 | 必做（课内） |
| 10 | 索引管理 | 创建索引：对索引进行基本的管理 | 1 | 1 | 设计 | 本科 | 必做（课内） |
| 11 | 数据库安全性的实现 | 实现数据库的安全性 | 1 | 1 | 验证 | 本科 | 必做（课内） |
| 12 | 触发器实验 | 创建触发器并验证功能性 | 2 | 1 | 设计 | 本科 | 必做（课内） |
| 13 | 存储过程实验 | 存储过程实验 | 2 | 1 | 设计 | 本科 | 必做（课内） |
| 自定义函数实验 | 必做（课内） |
| 游标实验 | 必做（课内） |
| 14 | 函数与事件 | 使用函数与事件进行周期性的数据处理 | 1 | 1 | 设计 | 本科 | 选做（课内） |
| 15 | 数据库的备份和恢复 | 实现数据库的备份和恢复 | 1 | 1 | 验证 | 本科 | 必做（课内） |
| 16 | 数据库监视与性能优化 | 利用数据库监视工具进行监视和性能检测 | 2 | 1 | 设计 | 本科 | 选做（课内） |
| 17 | 并发控制技术 | 并发控制实验 | 2 | 1 | 设计 | 本科 | 必做（课内） |
| 18 | 数据库设计 | 网上书店管理系统的设计 | 2 | 1 | 设计 | 本科 | 必做（课内） |
| 19 | 数据库应用开发 | 基于ODBC、JDBC的数据库应用开发实验 | 2 | 1 | 验证 | 本科 | 选内（课内） |
| 20 | 综合实验 | 网上书店管理系统的实现 | 4 | 1 | 设计 | 本科 | 必做（课外） |

### 实验1 熟悉MySQL的安装、使用环境及其基本工具的使用

**实验学时：1学时**

**实验类型：验证**

**实验要求：必做**

**一、实验目的**

熟悉MySQL的安装和用法环境，掌握MySQL管理工具Navicat的使用，了解在Navicat中执行SQL语句的方法，了解数据库及数据库对象，熟悉服务器的管理与使用。

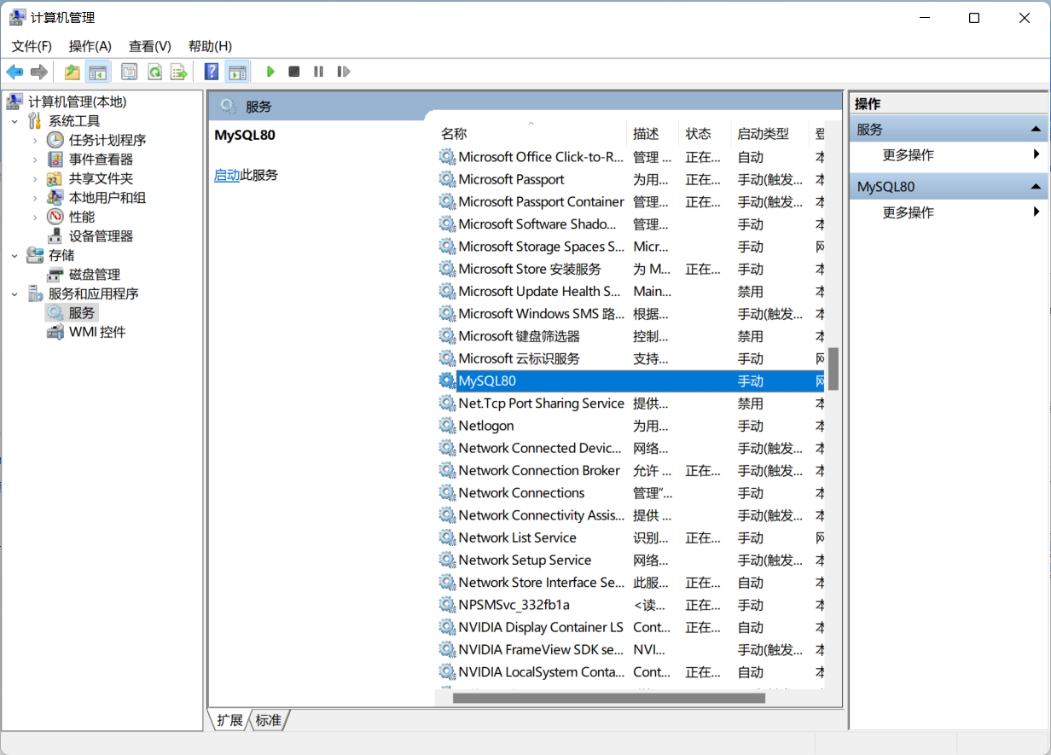
**二、实验内容**

**序1：完全卸载MySQL**

本次实验使用Windows 10 64位操作系统，部分机器可能已经安装了MySQL。由于版本不同，配置未知，需要将已有的MySQL完全卸载，再统一安装。

步骤：

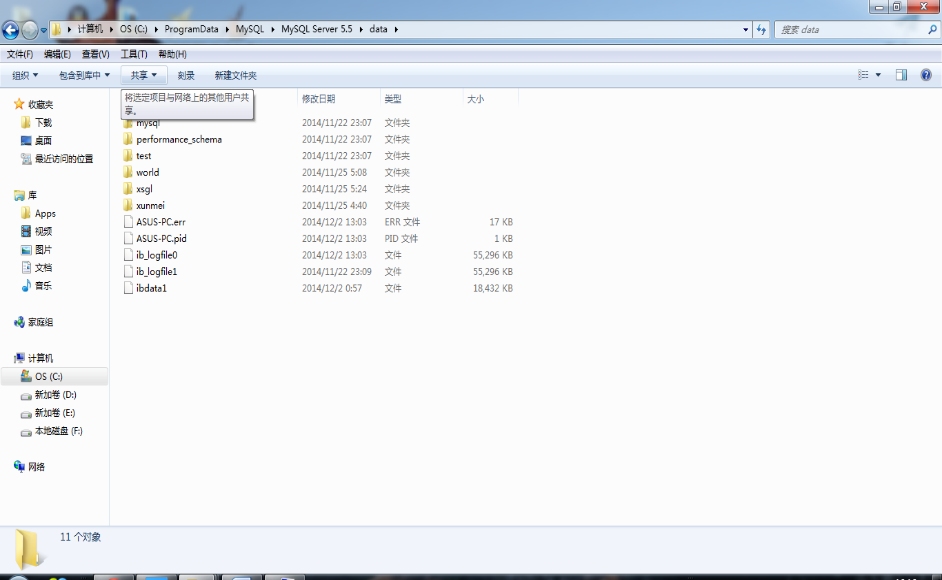
1. 停止MySQL服务 右键点击“计算机—选择“管理”，在弹出的对话框中选择“服务和应用程序”—“服务”，在打开的本地服务中，选择名为“MySQL”的服务，单击右键，选择“停止”如图所示:



1. 卸载MySQL。通过控制面板删除MySQL Server，如图所示：

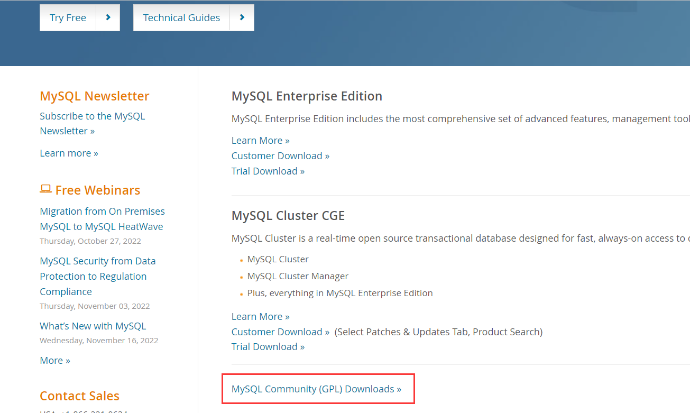


1. 删除残留的安装文件。删除安装路径（默认为“C:\Program Files\MySQL”）下的文件夹MySQL。
2. 删除残留的服务配置文件。删除路径“C:\ProgramData\MySQL\MySQL Server 5.5\data”下的所有文件，也可直接删除MySQL文件夹（ProgramData为隐藏文件夹，需要在“工具”—“文件夹选项”—“查看”中勾选查看隐藏文件。也可以直接通过搜索栏搜索）。如图：

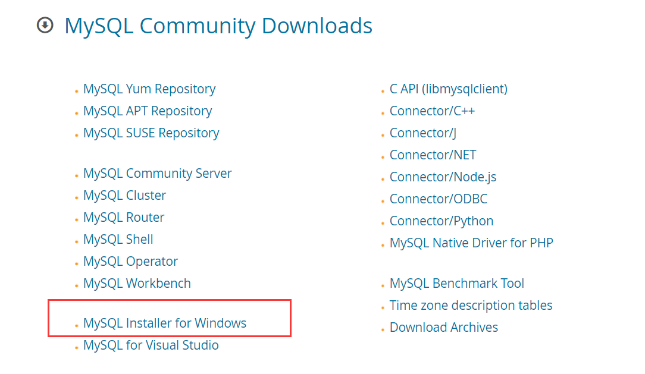


**序2：MySQL安装说明** 步骤：

1. 进入MySQL官方网址，Downloads界面<https://www.mysql.com/downloads/>，在下方选项中选择MySQL Community (GPL)下载

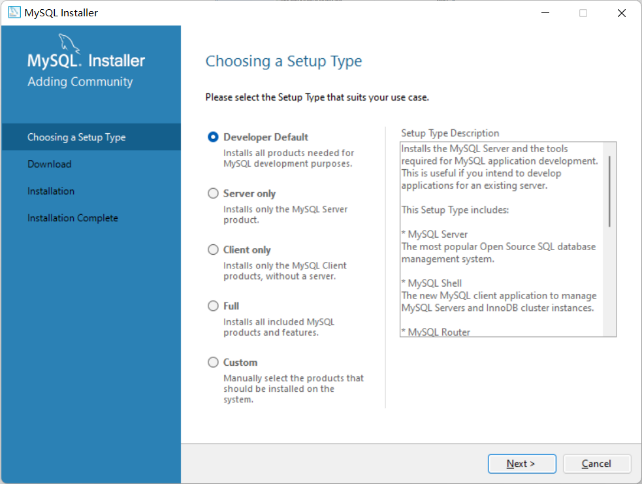


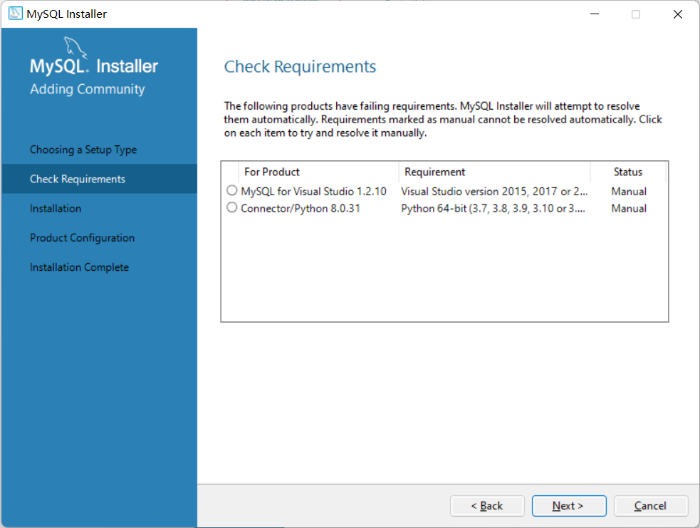
根据自己操作系统选择下载版本，Windows系统同学选择for Windows项下载，随后选择本地安装包，随后选择不登陆/注册，进入下载即可。



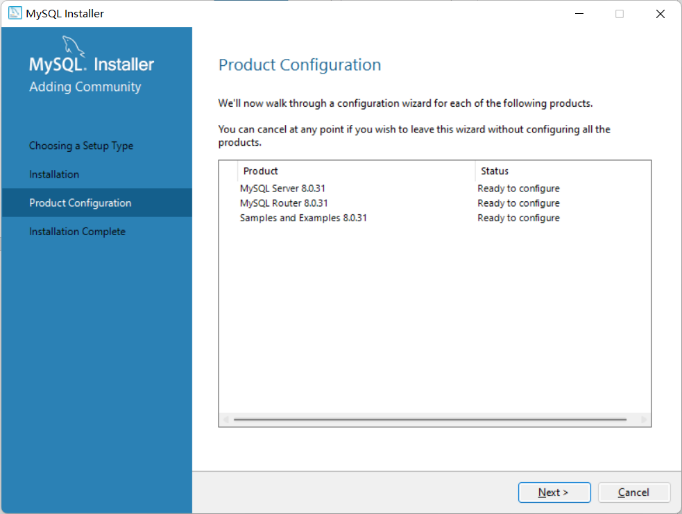


1. 打开MySQL 8.0的安装包mysql-installer-community-8.0.31.0.msi—Next,在如图所示的界面中选择默认模式进行安装。





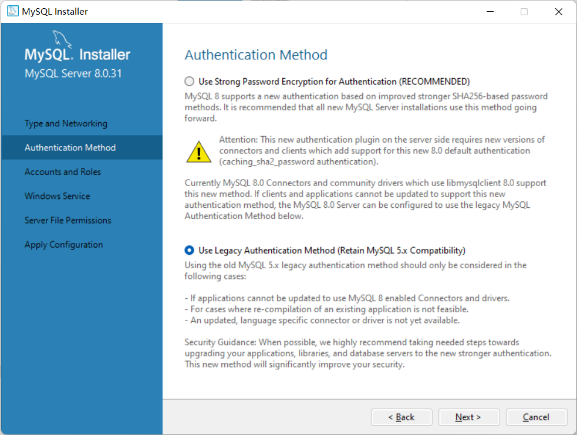
1. 安装完成之后，继续进入如下图所示的配置界面：



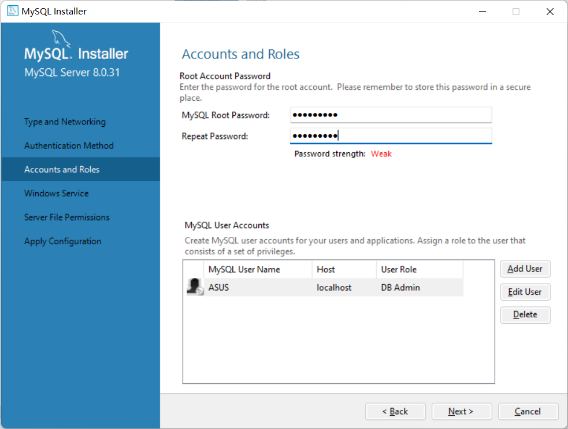
选择next进入配置，“Type and Networking”配置中，config type选项选择服务器类型，“Development Computer（开发测试类，MySQL占用很少资源）”、“Server Computer（服务器类型，MySQL占用较多资源）”、“Dedicated Computer（专门的数据库服务器，MySQL占用所有可用资源）”，一般选“Server Computer”，不会太少，也不会占满，实验所用则选择“Development Computer”。网络协议和端口保持默认即可。



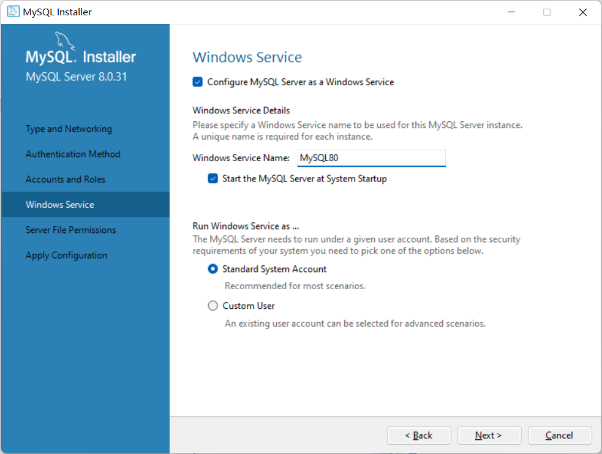
④ 选择密码强度，虽然默认高强度密码安全性最高，但对密码复杂度要求也较高，实验过程中，较容易忘记密码，故而选择下方选项，兼容5.X版本



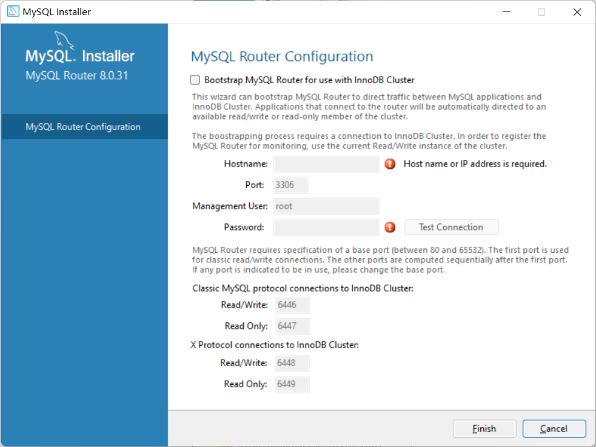
⑤设置密码，创建MySQL账户



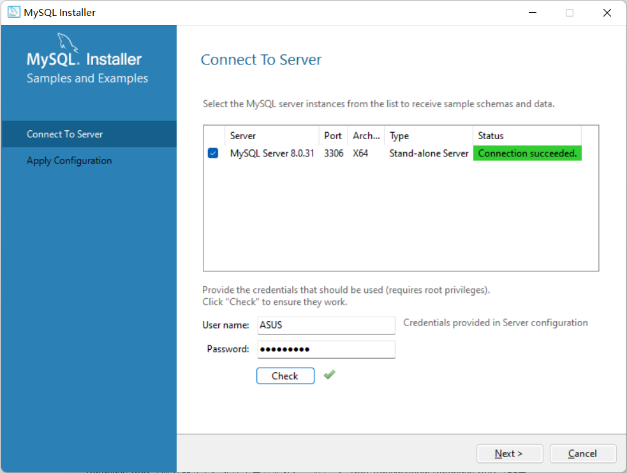
⑥注册Windows服务，保持默认即可



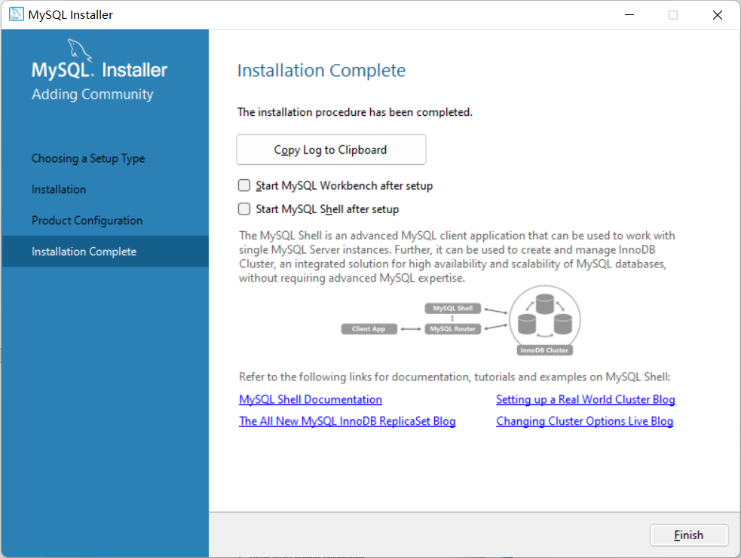
⑦完成MySQL server配置，继续进行下一个模块的配置，MySQL路由配置保持默认即可



⑧进入samples and examples配置中，需要使用之前创建的账户连接MySQL服务



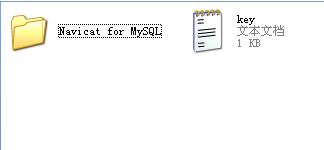
⑨完成配置后配置MySQL workbench和MySQL shell的开机自启选项，按照自己的意愿选择即可



⑩ 完成MySQL安装和配置

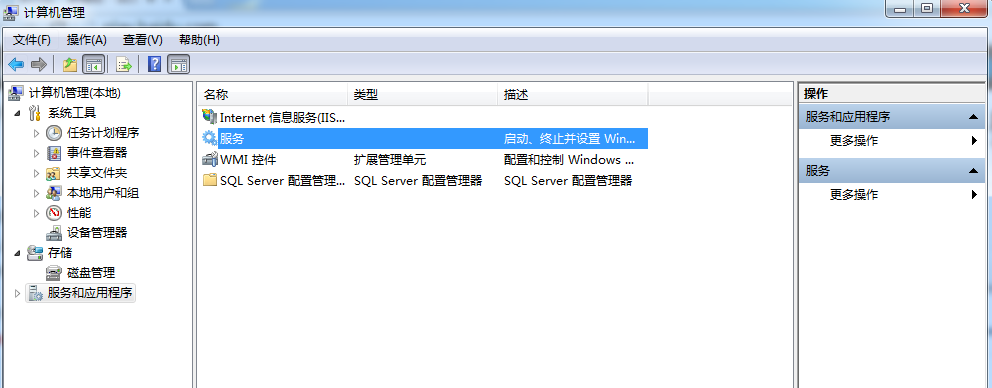
**序3:Navicat配置**

本次实验所使用的Navicat for MySQL为免安装版，解压即可使用。可用“key.txt”中的激活码激活。

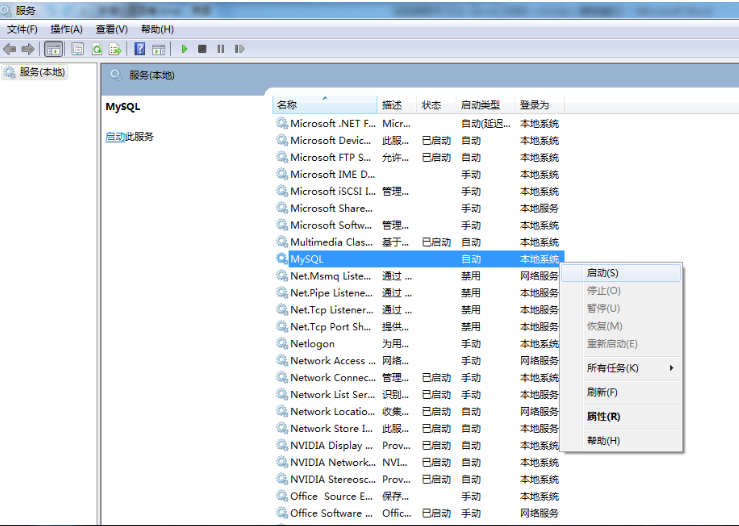


1、打开MySQL服务。

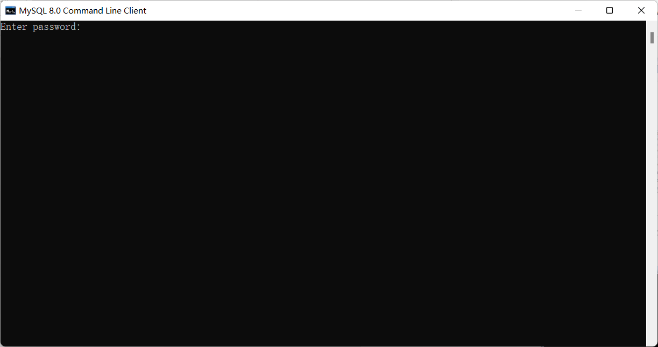
①右键点击“我的电脑—选择“管理”，在弹出的对话框中选择“服务和应用程序”—“服务”，如图所示：



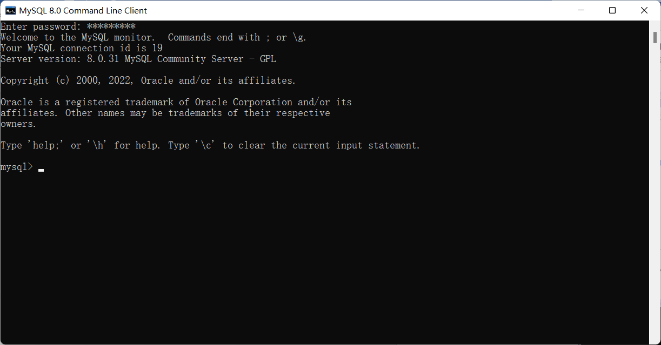
②在打开的本地服务中，确认名为“MySQL80”的服务为开启状态，若不是则单击右键，选择“开启”（也可以在“属性”中设置为随系统的启动而自动启动），如图所示：



③在“开始”菜单—所有程序中找到MySQL文件夹，点击“MySQL 8.0 Command Line Client”,将出现如图所示的窗口：

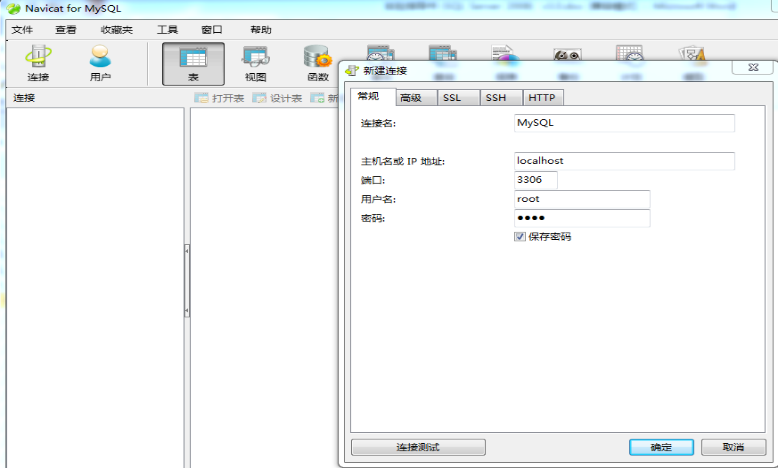


④ 输入自己在安装过程中设置的密码并回车，若出现如图所示结果，说明MysQL服务已正常启动并且可以通过该窗口用相应语句进行操作：

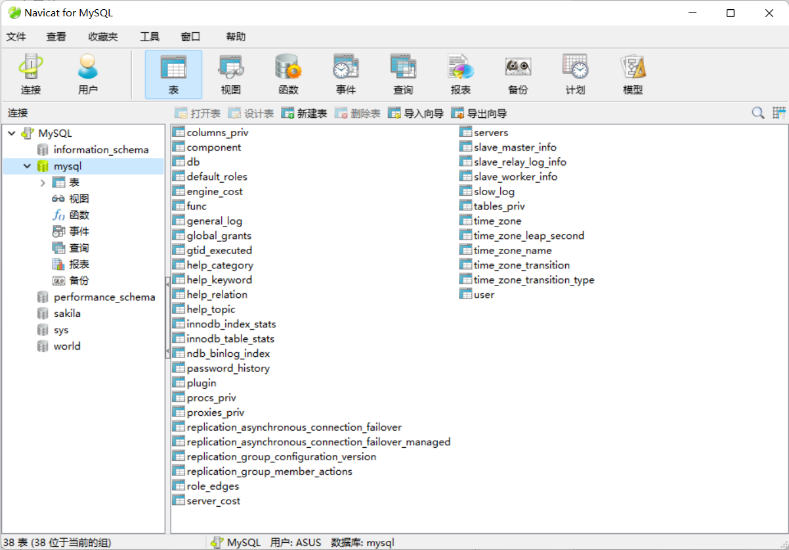


2、学习使用MySQL管理工具Navicat的基本操作，建立连接MySQL并在它左边的树型结构中查看数据库，观察该数据库中的所有数据库对象，如表、视图、存储过程、默认、规则等，学习查询的创建、连接的关闭和删除。

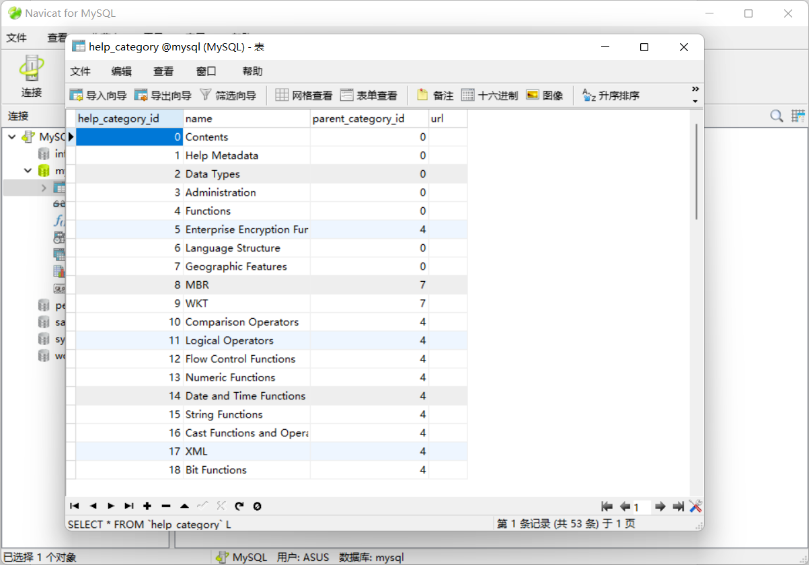
①新建连接MySQL。打开Navicat，进入主界面后点击“连接”，在弹出的对话框中输入连接名“MySQL”，用户名和密码均为“root”，输入之后点击“确定”（可事先点击“连接测试”）,如图所示:



②打开连接MySQL，查看默认生成的数据库、表。右键单击新建的连接MySQL，选择“打开连接”，可以看到默认生成的四个数据库，以数据库“mysql”为例，点击打开，如图所示。

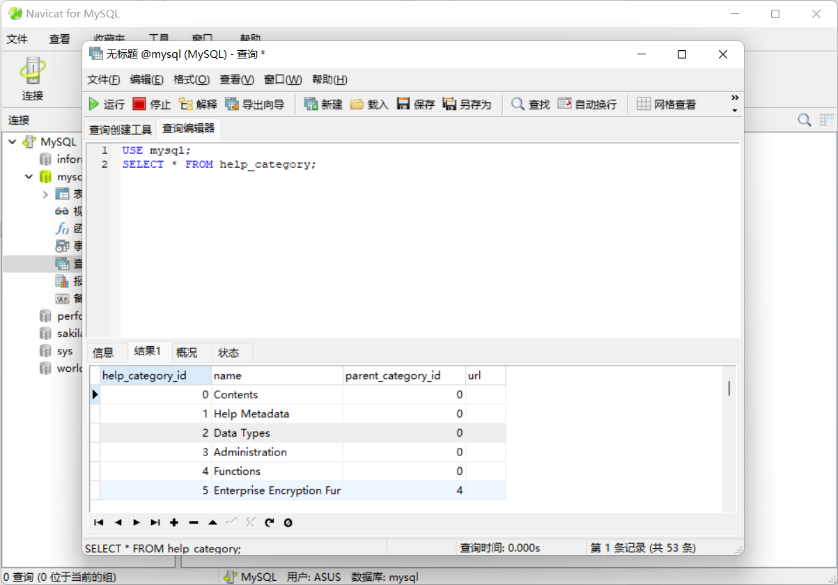


以数据库“mysql”中的表“help\_category”为例，点击打开，如图所示.

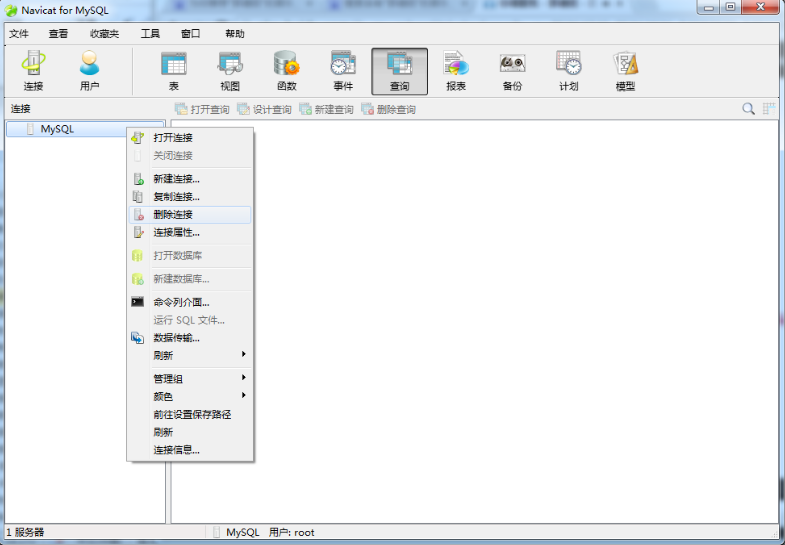


③新建查询。打开数据库“mysql”，点击Navicat主界面上方的“查询”，选择“新建查询”，在弹出的查询编辑器中输入如下语句：

点击上方的“运行”操作及结果如图所示：

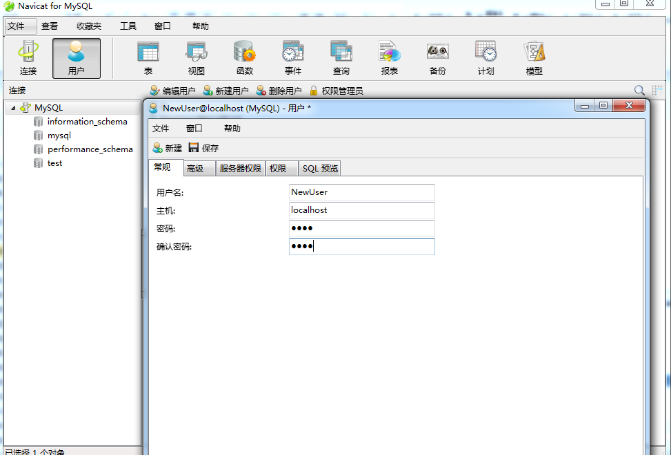


④关闭、删除连接。右键点击连接“MySQL”选择“关闭连接”。同样的操作选择“删除连接”，结果如图所示：

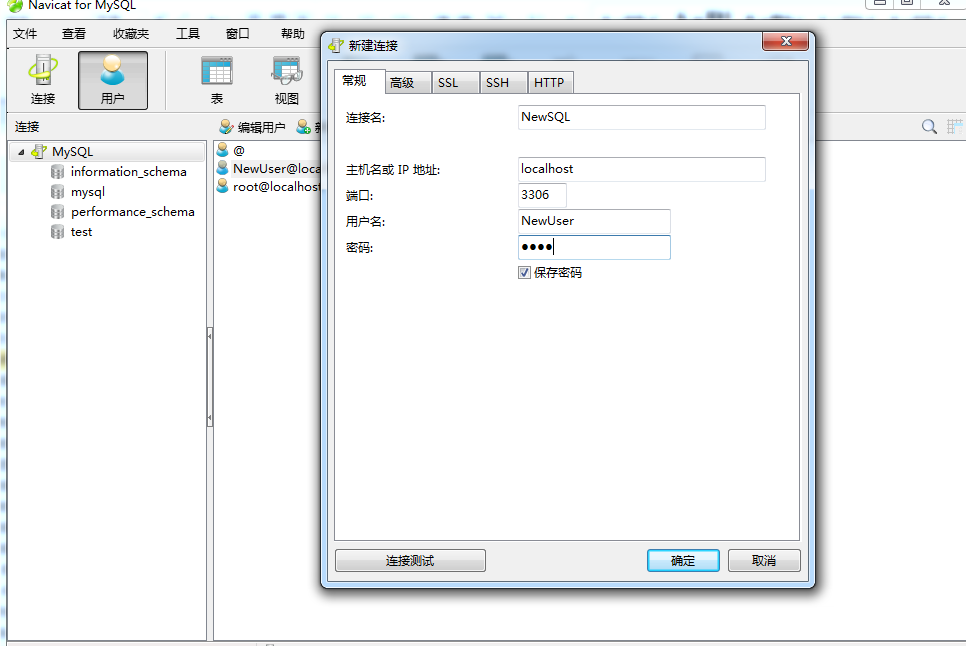


3、学习在Navicat下用户的管理，包括新建、编辑、删除。

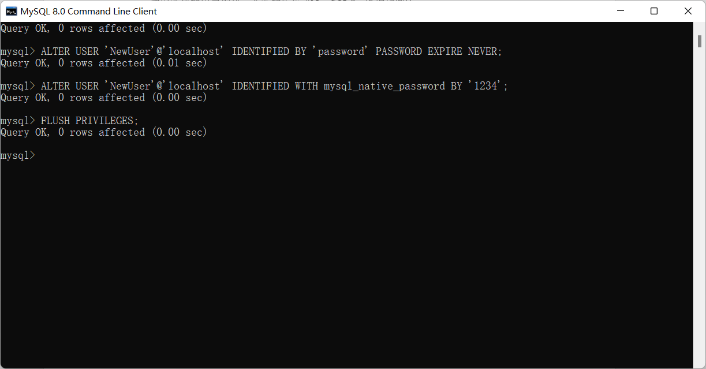
①新建用户。打开连接“MySQL”，点击Navicat主界面上方的“用户”，在弹出的对话框中填写如下内容：用户名“NewUser”、主机“localhost”、密码“1234”并再次确认，将“服务器权限”全部勾选，点击“保存”。如图所示：



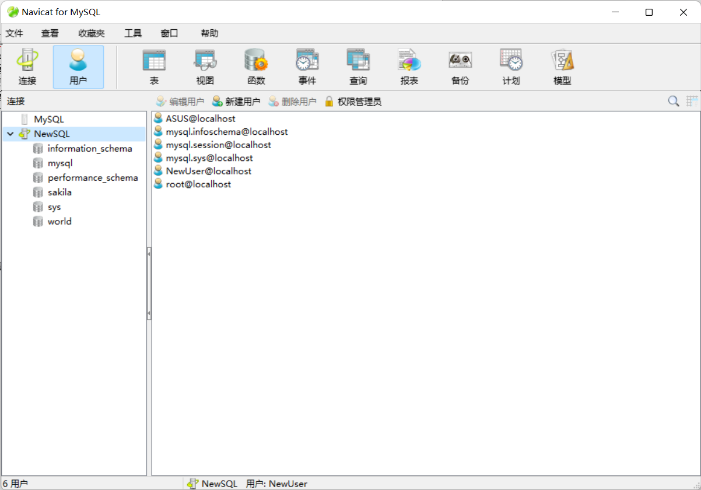
使用新建的用户创建一个新的连接“NewSQL”，操作如图所示：



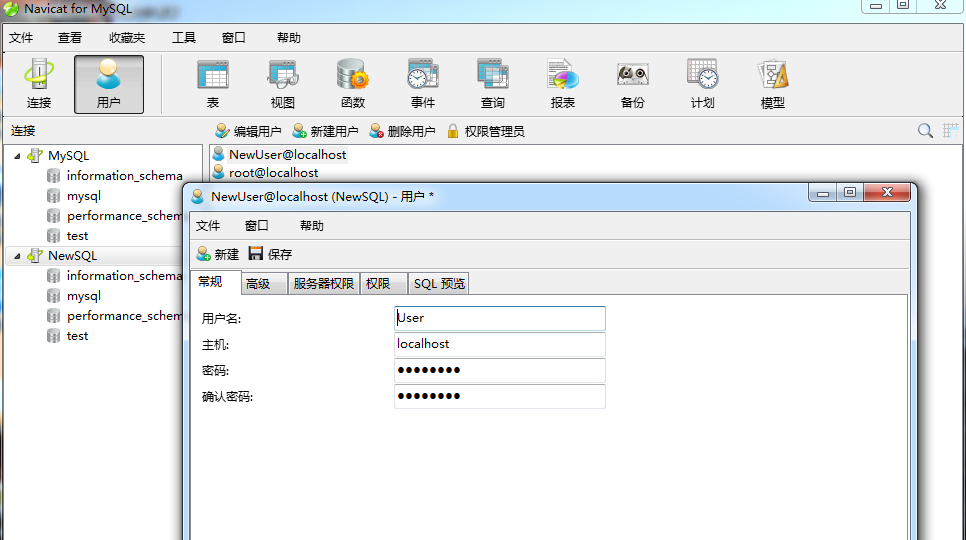
注意，新创建的连接打开时，会报1251号错误，因为MySQL 8之后版本的加密方式改为caching\_sha2\_password，而Navicat使用加密方式仍为MySQL 8之前版本使用的mysql\_native\_password加密。我们此处采用变更MySQL加密方式解决此问题，打开MySQL 8.0 command line client，输入最先创建用户的密码，进入运行后，输入“ALTER USER 'NewUser'@'localhost' IDENTIFIED BY 'password' PASSWORD EXPIRE NEVER;”指令变更NewUser使用的加密方式，变更之后，需要重新设置密码，指令为“ALTER USER 'NewUser'@'localhost' IDENTIFIED WITH mysql\_native\_password BY '1234';”之后刷新数据库“FLUSH PRIVILEGES;”



重新切换到Navicat界面，出现如图所示结果，说明用户创建成功。



②编辑用户。点击用户“NewUser@localhost”，再选择“编辑用户”，修改用户权限后点击保存，同样以新建连接的方法检验。(不要修改密码，此版本Navicat修改密码功能对应的SQL语句格式已过时)如图所示：



③删除用户。选择修改后的用户，点击“删除用户”，选择“确定”，如图所示：



**三、课后练习题**

1、分别通过Navicat主界面和查询编辑器新建查询查看MySQL连接的information\_schema、performance\_schema两个数据库中各一个表的所有数据。

2、新建一个的用户NewUser并用它创建一个新的连接NewSQL，执行练习题1中的操作。

### 实验2 数据库的创建和管理

**实验学时：1学时**

**实验类型：验证**

**实验要求：必做**

**一、实验目的**

熟练掌握使用Navicat和Transact-SQL语言两种方法创建、修改和删除数据库。学习在Navicat中进行数据库的转储和导入。掌握管理数据库的有关系统存储过程。

**二、实验内容**

1、利用Navicat创建满足以下要求的数据库：

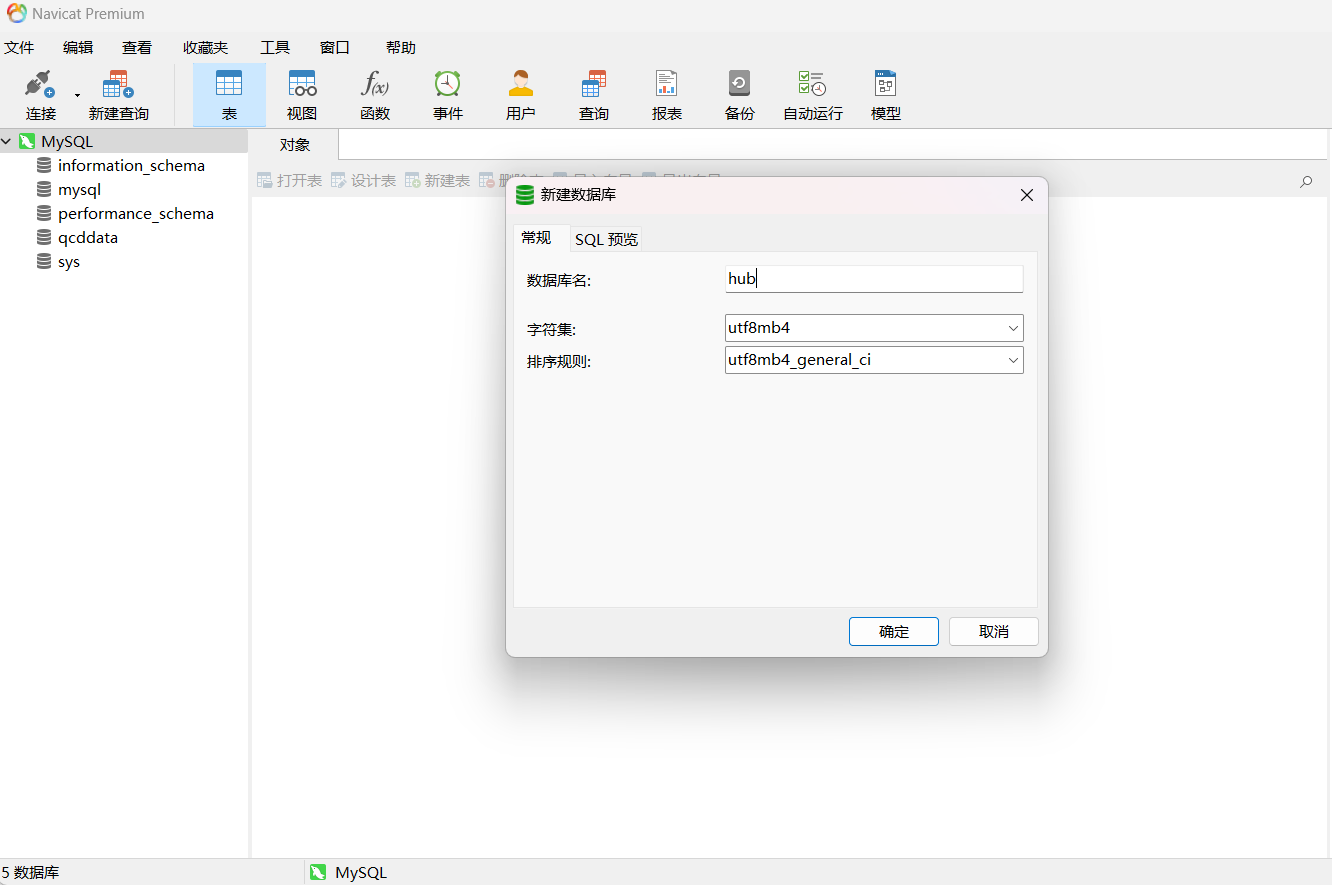
1）数据库存在于连接MySQL中；

2）数据库名称为hub；

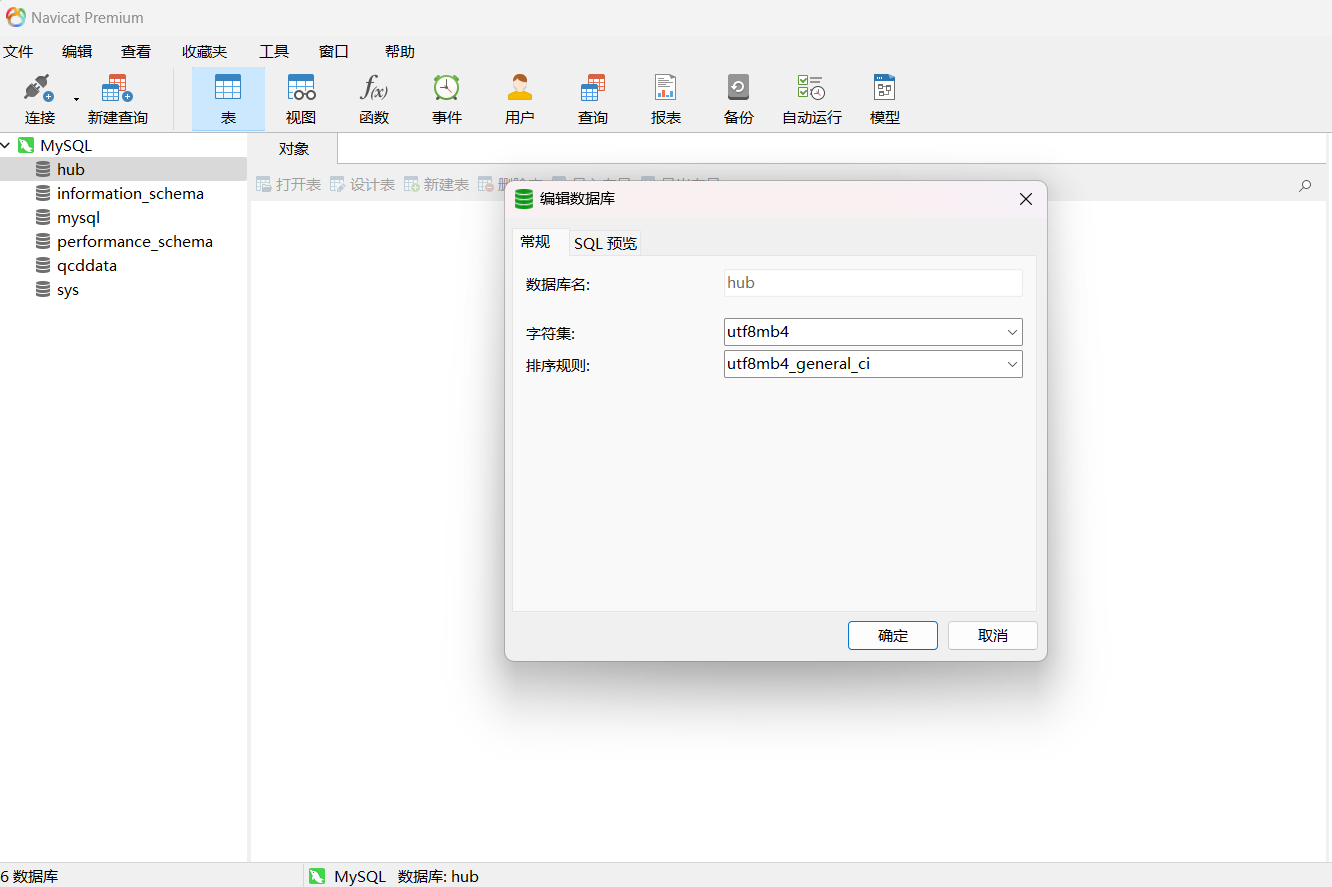
3）字符集选择utf8mb4；

4）排序规则选择utf8mb4\_general\_ci

具体步骤如下：打开“MySQL”连接—单击选择并右键—“新建数据库”—输入数据库名“hub”、字符集“utf8mb4”、排序规则“utf8mb4\_general\_ci”—“确定”。如图所示：



右键—编辑数据库，可以看到如图所示结果：



2、利用Transact-SQL语言创建满足以下要求的数据库：

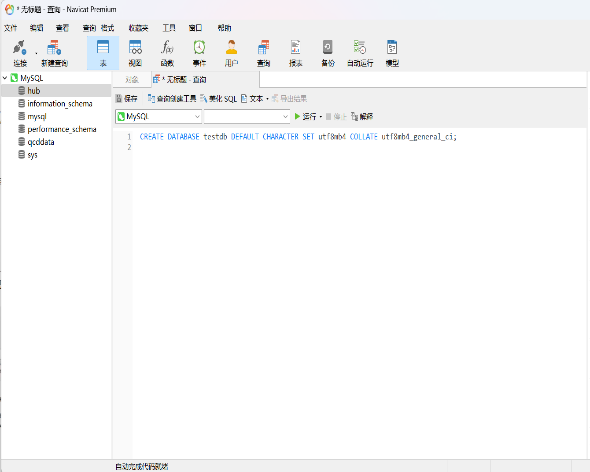
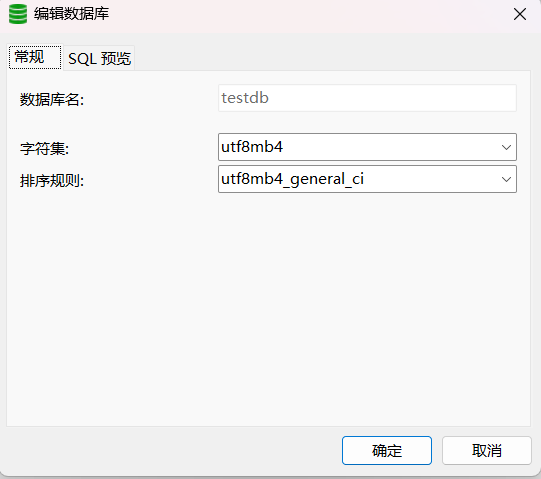
1）数据库存在于连接MySQL中；

2）数据库名称为testdb；

3）字符集选择utf8mb4；

4）排序规则选择utf8mb4\_general\_ci；

具体步骤如下： 点击“新建查询”，在查询编辑器输入以下代码，点击“运行”，完成后，左侧列表中选中连接“MySQL”，点击“刷新”按钮，列表即可见到“testdb”。如图所示：

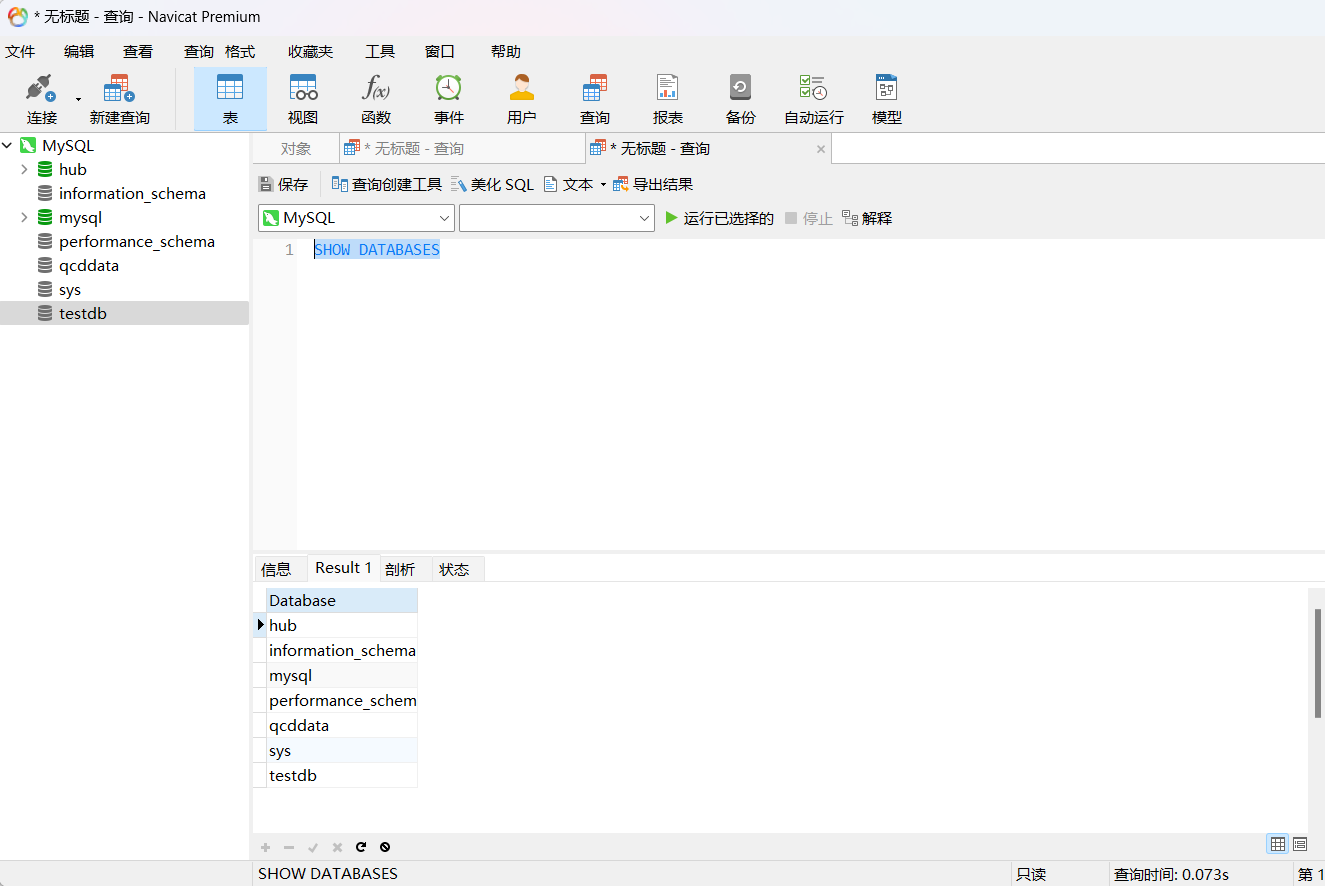
3、利用Transact-SQL语言查看数据库及表的信息。

①查看连接中的所有数据库

步骤：点击“新建查询”，输入

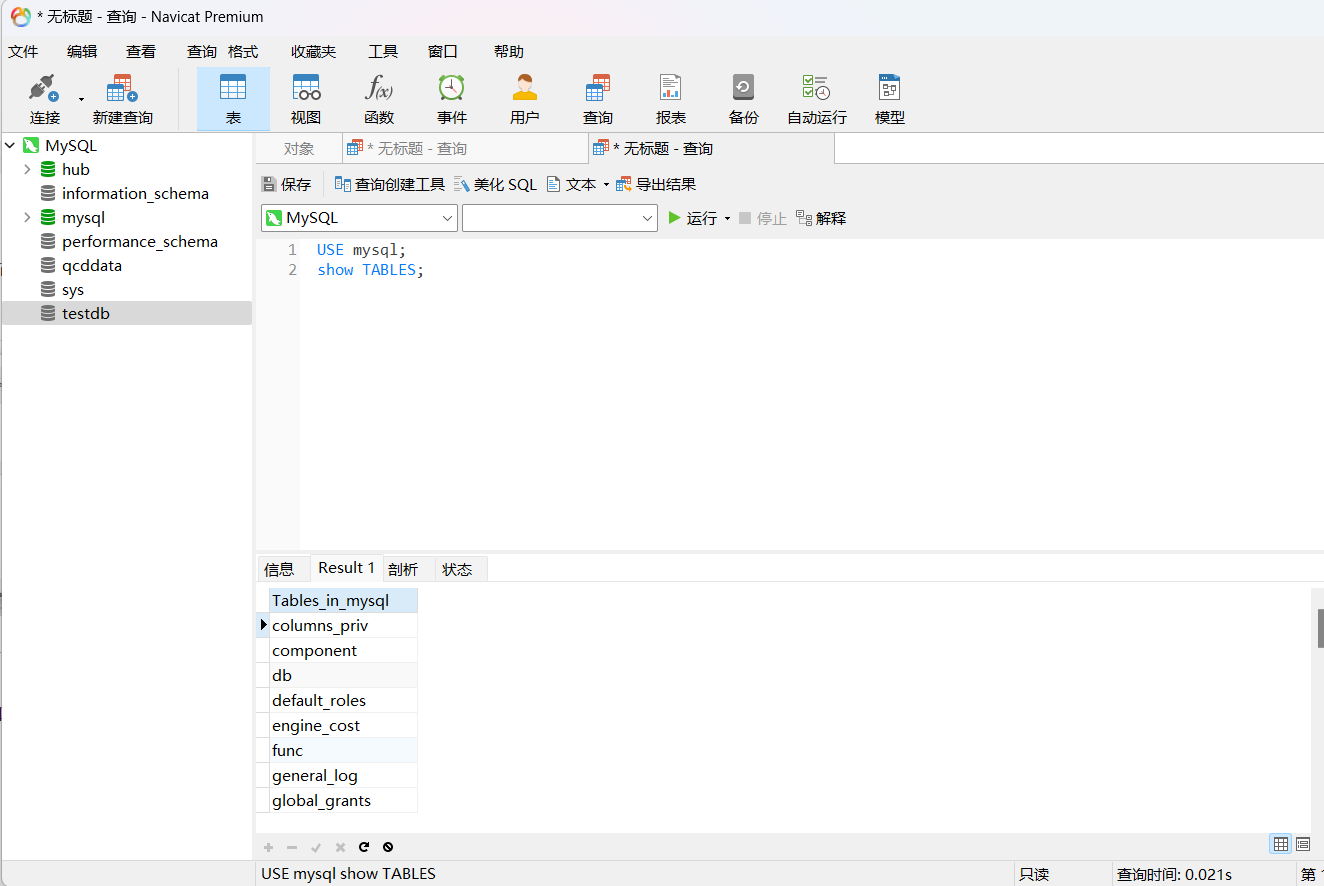
SHOW DATABASES

后运行，结果如图所示：



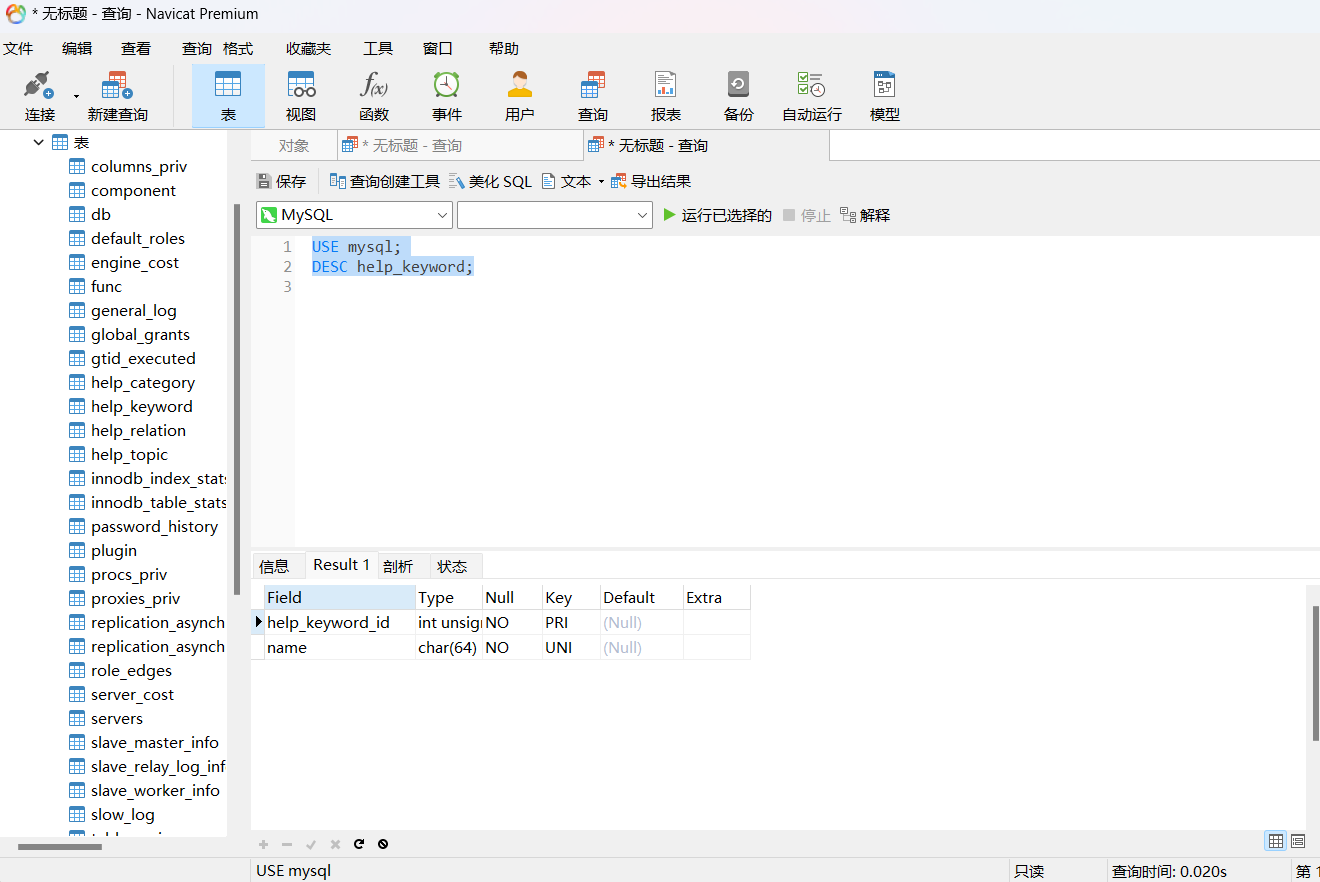
②查看数据库mysql中所有的表

步骤：点击“新建查询”，输入指令后运行，（也可以用语句show tables from mysql;）结果如图所示：



③查看数据库“mysql”中表“help\_keyword”的结构。

步骤：点击“新建查询”，输入后运行，结果如图所示：



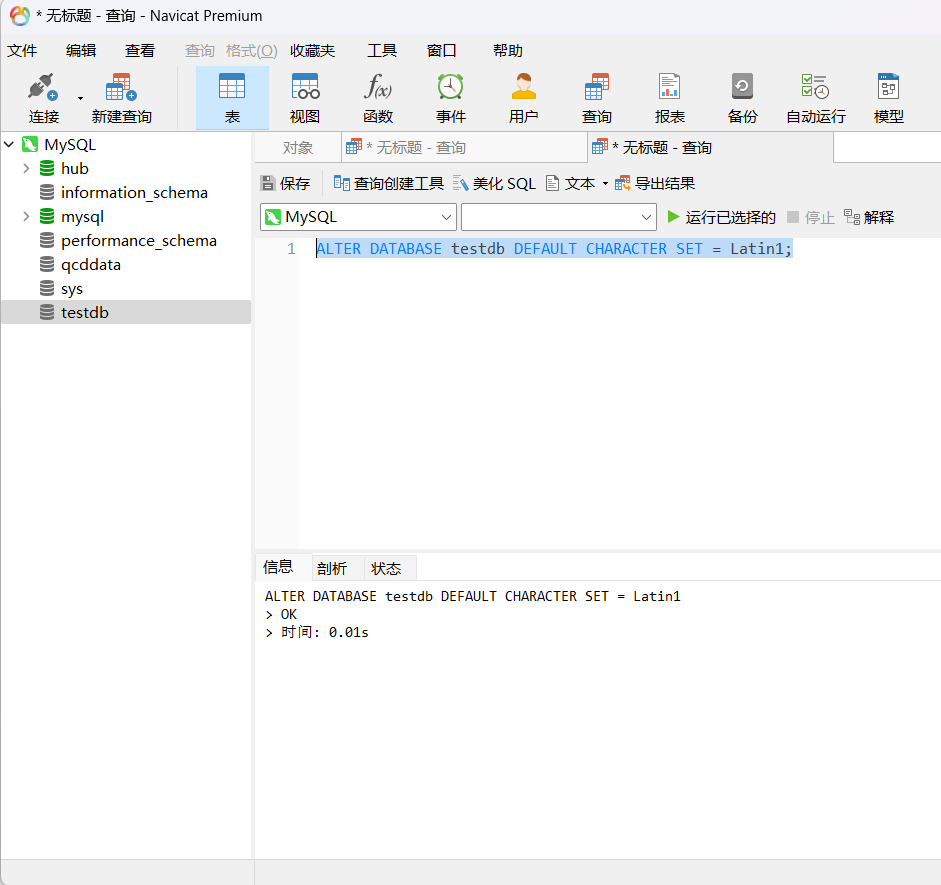
4、利用Navicat修改数据库testdb，修改其排序方式为utf8mb4\_bin。

步骤：右键单击数据库“testdb”—编辑数据库，在排序方式中选择“utf8mb4\_bin”—“确定”。

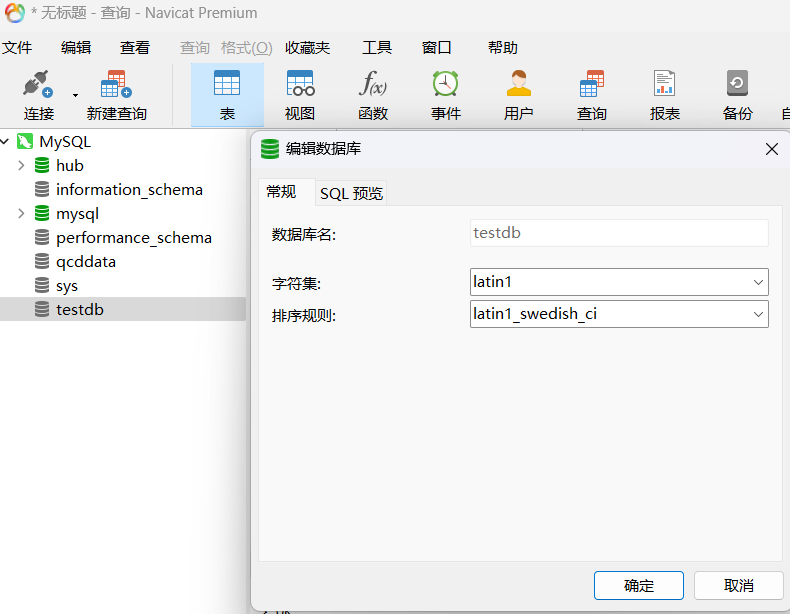
5、利用Transact-SQL语言修改数据库testdb的字符集为Latin1；

步骤：点击“新建查询”，输入以下代码，点击运行

如图所示：



关闭数据库“mydb”后重新打开，查看数据库属性，可以看到如图所示结果：



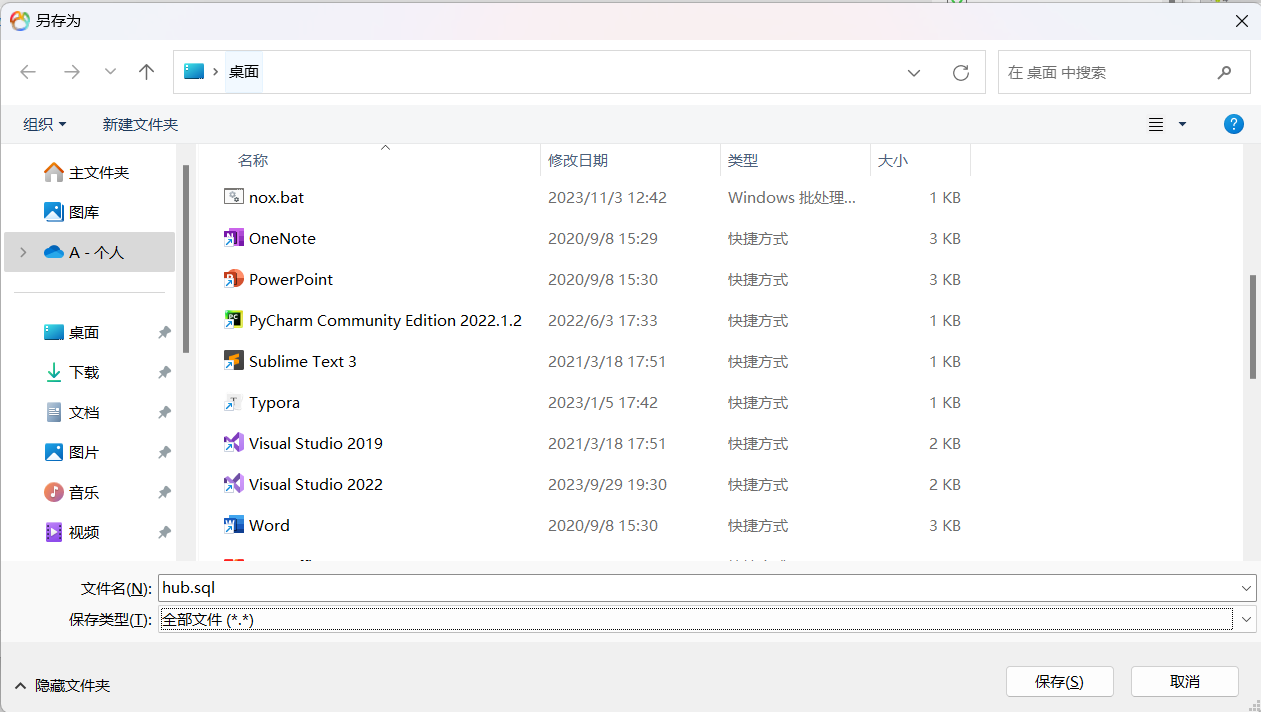
1. 利用Navicat或利用Transact-SQL删除数据库testdb

（1）利用Navicat删除数据库：直接右键点击testdb选择删除即可，注意要关闭当前的SQL语句编写界面。（2）利用Transact-SQL语言删除数据库testdb。

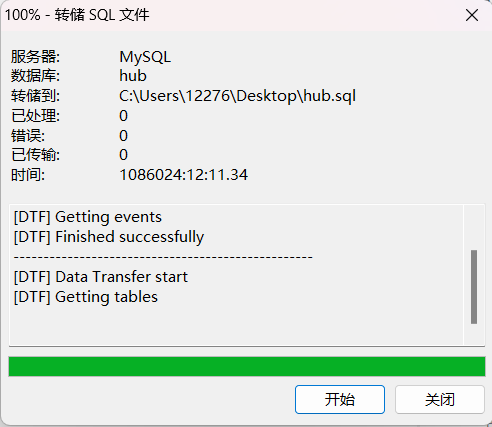
步骤：点击“新建查询”，输入DROP DATABASE testdb 后运行。刷新连接“MySQL”，查看结果。

7、转储MySQL数据库。

步骤：打开连接“MySQL”—右键单击数据库“hub”—选择“转储SQL文件”—将文件名改为“hub.sql”后保存在桌面，如图所示：



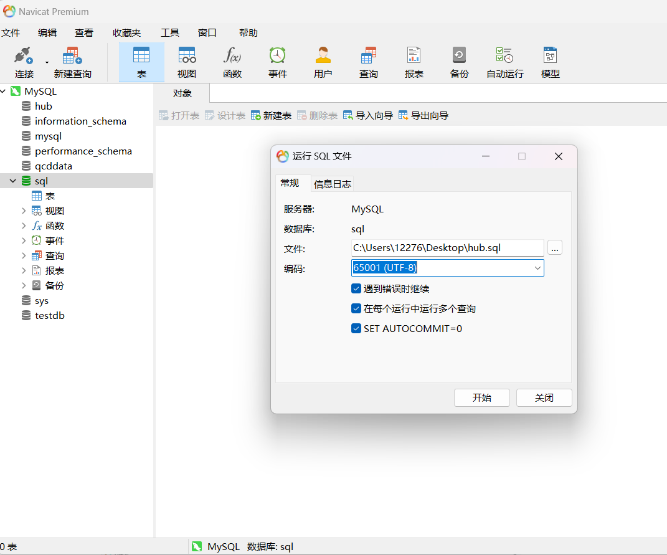
成功后将出现如图所示情况：



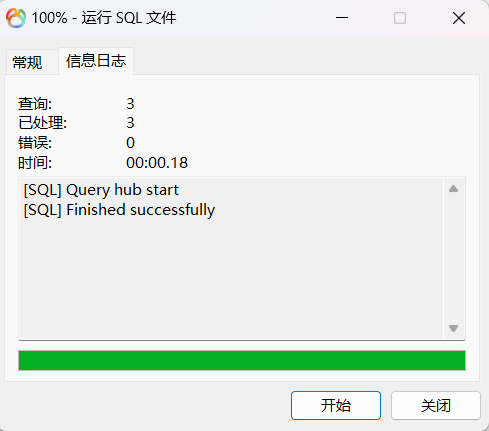
8、导入MySQL数据库。

①在连接“MySQL”中新建数据库“sql”（字符集与排序方式尽量一致）。

②打开数据库“sql”—右键单击—运行SQL文件—选择桌面的“hub.sql”文件—点击“确定”。如图所示：



③执行结果如图所示则表示成功。关闭数据库sql后重新打开，对比数据库“sql”与最初的数据库“hub”，看是否完全相同。



**三、****课后练习题**

1、分别用以下几种语句查看数据库“mysql”中任意表的结构或数据（先输入：“USE mysql；”）：

①SHOW COLUMNS FROM 表名;

②DESCRIBE 表名;

③DESC 表名

④SELECT \* FROM 表名

2、思考以下问题：

1） mysql中utf8编码的utf8mb3\_bin,utf8mb3\_general\_cs,utf8mb3\_general\_ci三者的区别。

2） 不同编码方式的数据库之间进行转储和导入会不会出现错误。

### 实验3 表的创建和管理

**实验学时：1学时**

**实验类型：验证**

**实验要求：必做**

**一、实验目的**

熟练掌握使用Navicat和Transact-SQL语言两种方法创建、修改和删除表。掌握管理表的有关系统存储过程。

**二、实验内容**

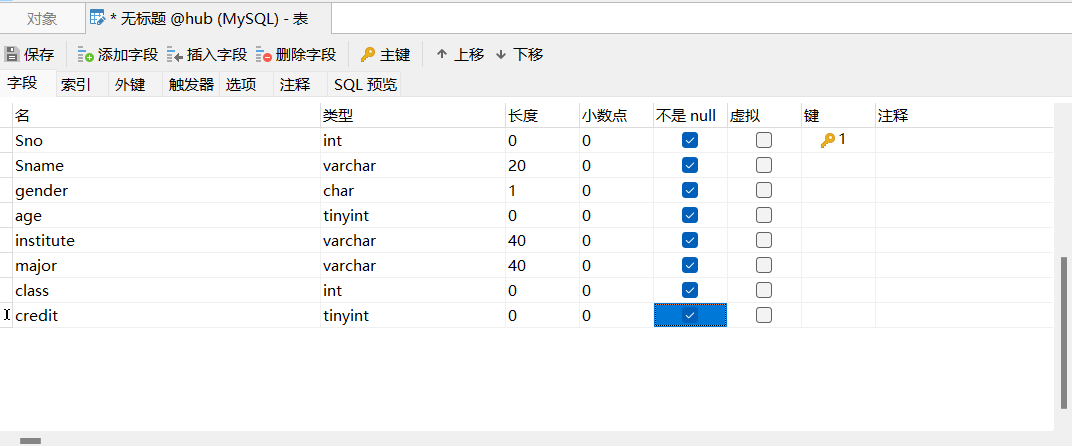
1、在数据库hub中，利用Navicat创建以下表格：

1）表格名为student（学生基本情况表）；

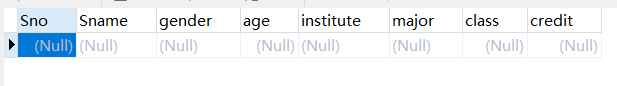
2）表格中各个属性的定义如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 含义 | 数据类型 | 长度 | 能否取空值 | 备注 |
| Sno | 学号 | int |  | no | 主码 |
| Sname | 姓名 | varchar | 20 | no |  |
| gender | 性别 | char | 1 | no |  |
| age | 年龄 | tinyint |  | no |  |
| institute | 院系 | varchar | 40 | no |  |
| major | 专业 | varchar | 40 | no |  |
| class | 班级 | int |  | no |  |
| credit | 总学分 | float | （255，1） | no |  |

具体步骤如下：打开数据库“hub”–右键点击“表”--“新建表”。按照上表内容输入个属性的定义，填写“列名”、“数据类型”，并取消或选择“允许NULL值”前的选项，并右键点击“Sno”，选择将其设为主键，如图所示：



点击“保存”，输入表名：students。结果如图所示：



2、按照以下步骤向表格xs中添加如下记录：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Sno  （学号） | Sname  （姓名） | gender  （性别） | age  （年龄） | institute  （院系） | major  （专业） | class  （班级） | credit  （总学分） |
| 2020102000 | 庄小燕 | 女 | 20 | 软件学院 | 软件工程 | 2001 | 40 |
| 2020102001 | 洪波 | 男 | 21 | 软件学院 | 软件工程 | 2002 | 38 |
| 2020102002 | 肖辉 | 男 | 20 | 计算机学院 | 计算机科学与技术 | 2001 | 36 |
| 2020102003 | 柳嫣红 | 女 | 21 | 计算机学院 | 计算机科学与技术 | 2002 | 36 |
| 2020102004 | 张正正 | 男 | 20 | 数统学院 | 应用数学 | 2001 | 42 |
| 2020102005 | 李丽 | 女 | 21 | 数统学院 | 应用数学 | 2002 | 40 |

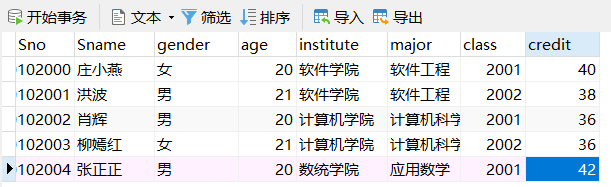
1）打开Navicat；

2）打开连接MySQL找到数据库hub中的表students；

3）在表sdutent上右击鼠标，选择“打开表”；

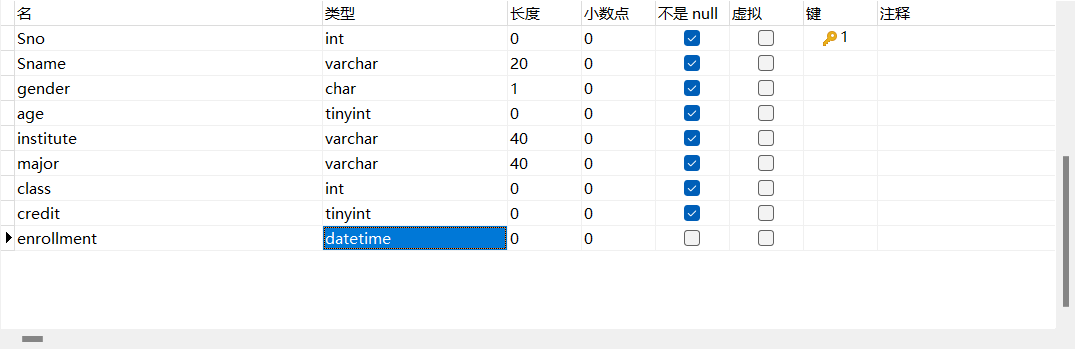
4）在接着出现的记录录入界面上添加记录。（方向键下增加新纪录）

添加完后如图所示：



3、向表students中增加“入学时间”属性列，其列名为enrollment，数据类型为datetime型。

步骤：右键单击students表—设计表—加入列enrollment，设置数据类型为datetime。如图所示：



将“enrollment”属性列删除，步骤右键单击students表—设计表——右键单击该字段——点击删除字段。

4、将表students中age（年龄）列的数据类型改为int型。

步骤：与第4项类似

5、在数据库hub中，利用Transact-SQL语言创建以下表格：

1）表名为course（课程情况表）；

2）表中各个属性的定义如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 含义 | 数据类型 | 长度 | 能否取空值 | 备注 |
| Cno | 课程号 | int | 0 | no | 主码、自增 |
| Cname | 课程名 | varchar | 40 | no |  |
| credit | 学分 | tinyint |  | no |  |

步骤：点击“新建查询”，输入以下代码，点击运行

命令成功完成后，在 “hub”数据库中右键点击“表”，选择“刷新”，结果如图所示：



6、利用Transact-SQL语言修改course表。

①增加“成绩”一列grade，int型，允许为空值，默认为0。

步骤：点击“新建查询”，输入以下代码，点击运行

ALTER TABLE hub ADD COLUMN grade INT DEFAULT 0;

②修改grade列的类型为char。

步骤：点击“新建查询”，输入以下代码，点击运行

ALTER TABLE hub CHANGE COLUMN grade CHAR(4);

③修改cj列的列名为mark。

步骤：点击“新建查询”，输入以下代码，点击运行

ALTER TABLE hub CHANGE grade mark CHAR(4) DEFAULT 0;

④删除mark列。

步骤：点击“新建查询”，输入以下代码，点击运行

ALTER TABLE hub DROP COLUMN grade;

7、利用Transact-SQL将表students重命名为student。

步骤：点击“新建查询”，输入以下代码，点击执行

RENAME TABLE hub.students TO hub.student;

**三、课后练习题**

1、创建一个名为cpxs的数据库，在该数据库中分别通过Navicat和查询分析器创建一个名为cp的表（产品表），表格中各个属性的定义如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 含义 | 数据类型 | 长度 | 能否取空值 | 备注 |
| cpbh | 产品编号 | char | 6 | no | 主码 |
| cpmc | 产品名称 | char | 30 | no |  |
| jg | 价格 | float |  | yes |  |
| kcl | 库存量 | int |  | yes |  |

2、思考以下问题：

1）如果要在一个已经存在的表格上删除一列并增加一列，能不能用一个ALTER TABLE语句来完成？

答：

### 实验4 数据库的完整性设计

**实验学时：1学时**

**实验类型：验证**

**实验要求：必做**

**一、实验目的**

熟练掌握使用Navicat和Transact-SQL语言两种方法创建、修改和删除表。掌握管理表的有关系统存储过程。

**二、实验内容**

1、在数据库hub（已有学生基本情况表student)中，利用Navicat和SQL语句创建以下表格：

1）表名为teacher（教师基本情况表），表中各个属性的定义如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 含义 | 数据类型 | 长度 | 能否取空值 | 备注 |
| Tno | 工号 | int |  | no | 主码 |
| Tname | 姓名 | varchar | 20 | no |  |
| gender | 性别 | char | 1 | no |  |
| age | 年龄 | tinyint |  | no |  |
| institute | 院系 | varchar | 40 | no |  |
| professional\_title | 职称 | varchar | 20 | no |  |

1. 表名为course（课程基本情况表），表中各个属性的定义如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 含义 | 数据类型 | 长度 | 能否取空值 | 备注 |
| Cno | 课程号 | int |  | no | 主码，自增 |
| Cname | 课程名 | varchar | 40 | no |  |
| credit | 学分 | float | （255，1） | no |  |

3）表名为sc（学生选课情况表），表中各个属性的定义如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 含义 | 数据类型 | 长度 | 能否取空值 | 备注 |
| Sno | 学号 | int |  | no | 主码，外键 |
| Cno | 课程号 | Int |  | no | 主码，外键 |
| grade | 分数 | float | （255，2） | yes |  |

4）表名为tc（教师教授课程情况表），表中各属性的定义如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 含义 | 数据类型 | 长度 | 能否取空值 | 备注 |
| Tno | 学号 | int |  | no | 主码，外键 |
| Cno | 课程号 | int |  | no | 主码，外键 |

1. 在teacher表中定义Tno为主键。

2、在sc表中定义Sno和Cno联合构成主键。

3、在tc表中定义Tno和Cno联合构成主键。

4、定义course表中的credit列默认值为0。

5、定义teacher表中的gender列的CHECK约束“男”或“女”。

6、定义course表中的Cno列为自增。

7、在student表与sc表之间定义外键Sno。

8、在course表与sc表之间定义外键Cno。

9、在teacher表与tc表之间定义外键Tno。

10、在course表与tc表之间定义外键Cno。

### 实验5 表数据的简单查询

**实验学时：1学时**

**实验类型：设计**

**实验要求：必做**

**一、实验目的**

掌握SELECT语句的基本用法。掌握计算列的用法。掌握WHERE子句中各类查询条件的写法。

**二、实验内容**

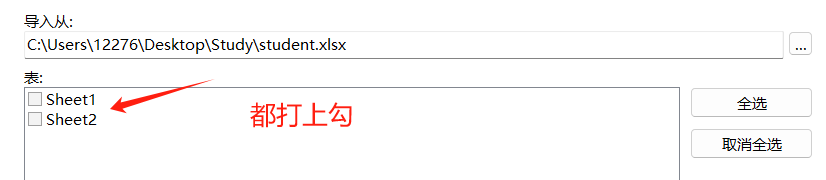
1、准备工作

方式一：在连接“MySQL”中删除原有的数据库“hub”，然后新建数据库“hub”,并将数据库文件hub\_inno.sql导入到该数据库之中。

方式二：将原数据库中各表数据清空后，将excel文件导入到hub数据库的各表中，其中student和sc这两excel文件有两个sheet。

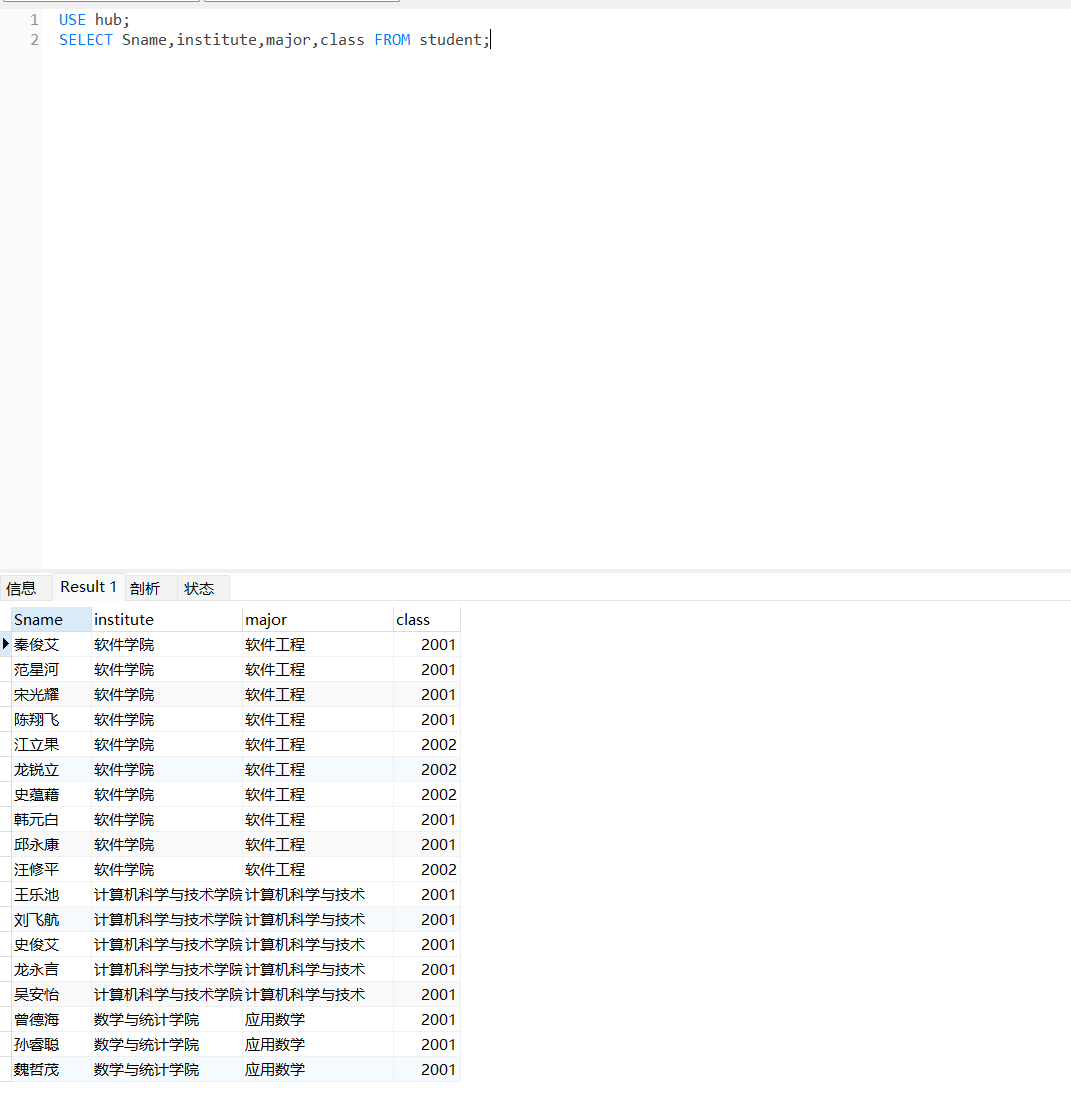






2、查询hub数据库的student表中的学生姓名（Sname）、院系（institute）、专业（major）和班级（class）。

步骤：点击“新建查询”，输入以下代码，点击执行运行结果如图：

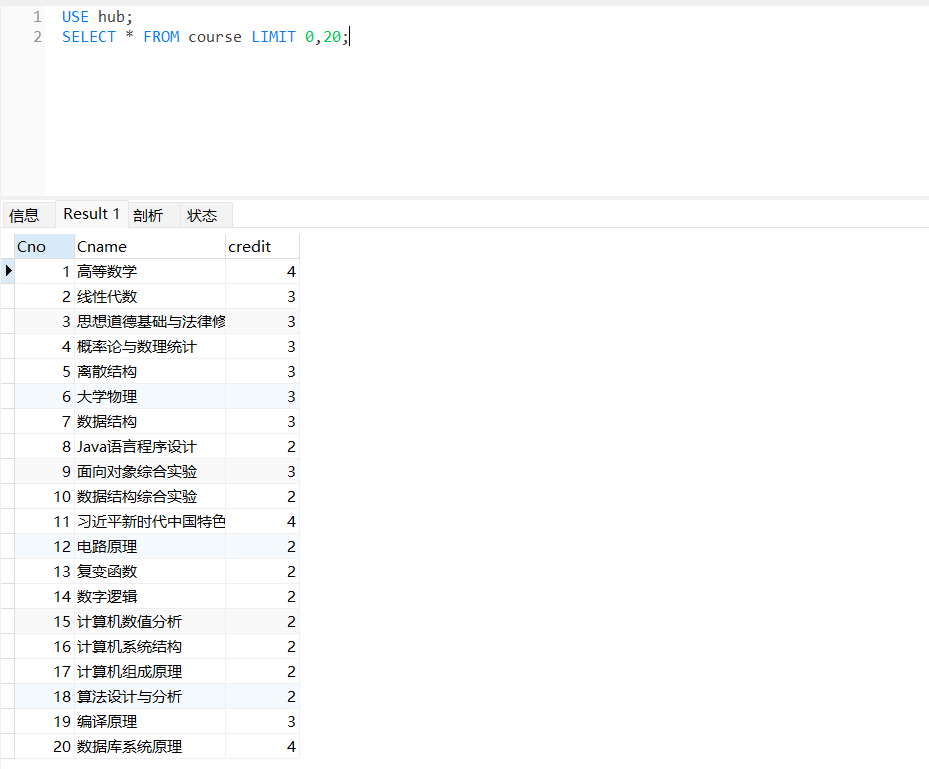


3、从hub数据库的course表中搜索返回前20条的数据。

步骤：点击“新建查询”，输入以下代码，点击执行

其中，“LIMIT M,N”中的M表示从第M条（不包括M）开始

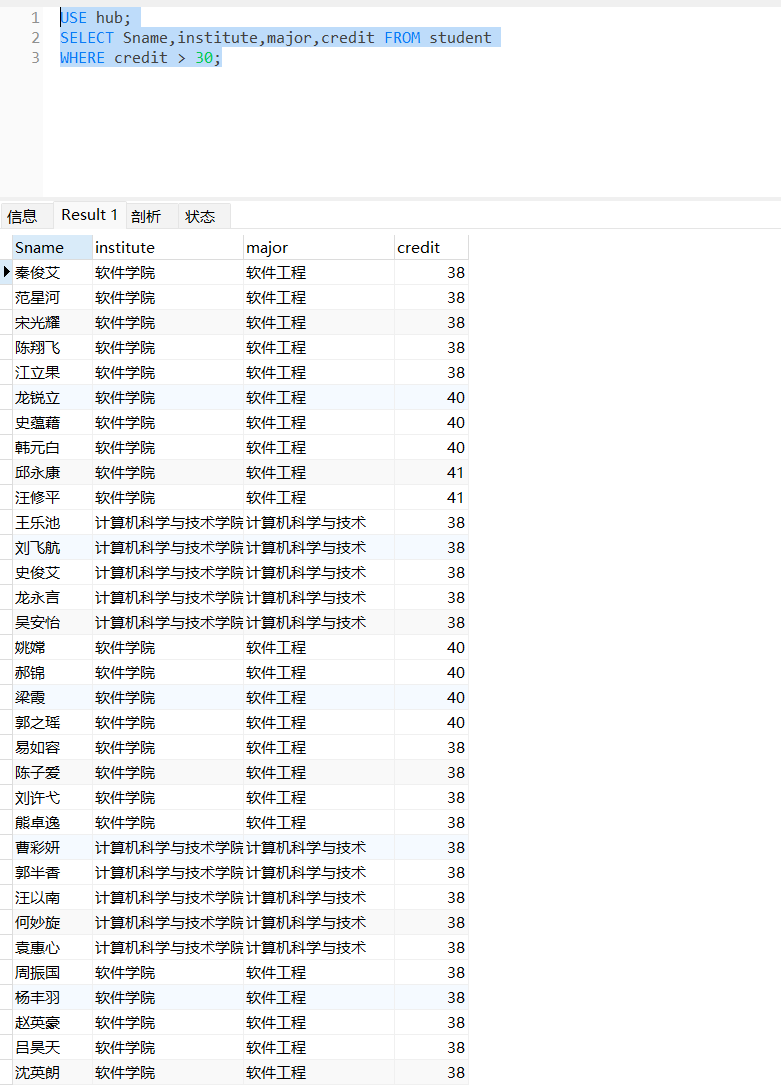
结果如图：



4、使用WHERE 子句从hub 数据库的student表中检索出所有总学分(credit)超过30的学生姓名（Sname）、院系（institute）、专业（major）以及总学分(credit)。

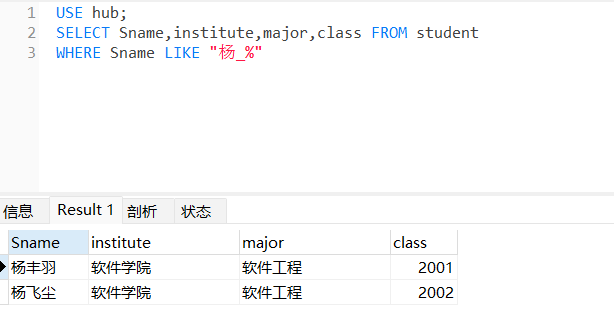
步骤：点击“新建查询”，输入以下代码，点击执行

结果如图：



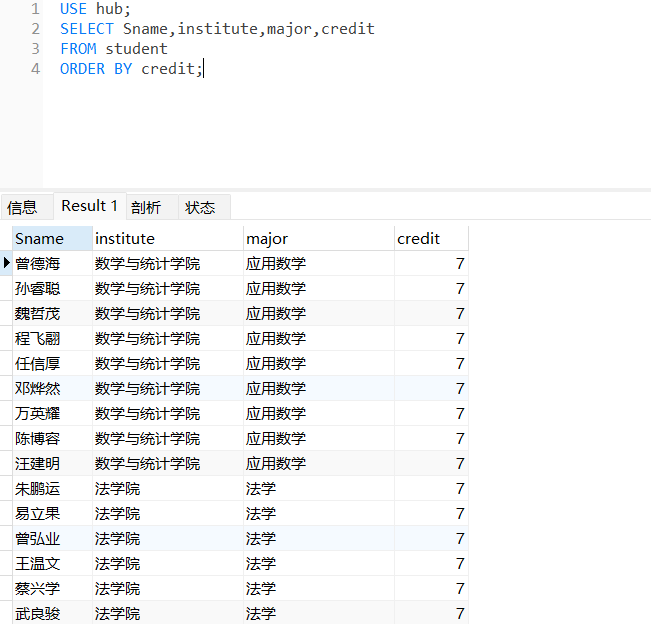
5、查询在hub数据库的student表中Sname以汉字“杨”开头的国家的学生姓名（Sname）、院系（institute）、专业（major）以及班级（class）。

步骤：点击“新建查询”，输入以下代码，点击执行运行结果如图：



6、查询hub数据库student表中所有学生的Sname、institute和major，并按总学分credit从小到大进行排列。

步骤：点击“新建查询”，输入以下代码，点击执行运行结果如图：



练习巩固

7、查询course表中学分（credit）为2分的课程的课程名Cname和学分credit。

8、student表中查询前20个软件学院学生的姓名。

9、student表中查询软件学院所有学生的姓名和总学分，并要求对查询结果按总学分的降序排列。

10、student表中查询总学分介于15和30之间的学生的名字、院系和专业以及总学分。

11、查询名称以“汪”开头的属于软件学院或属于计算机科学与技术学院的学生姓名和院系。

**三、课后练习题**

以下题目在数据库hub中完成。

1、查询teacher表中所有软件学院教师的名字、职称、年龄，要求查询结果按年龄的升序排列。

2、查询student表中属于数学与统计学院或机械科学与工程学院的学生姓名。

3、查询SC表中grade字段大于90分的全部信息。

### 实验6 表数据的维护

**实验学时：1学时**

**实验类型：验证**

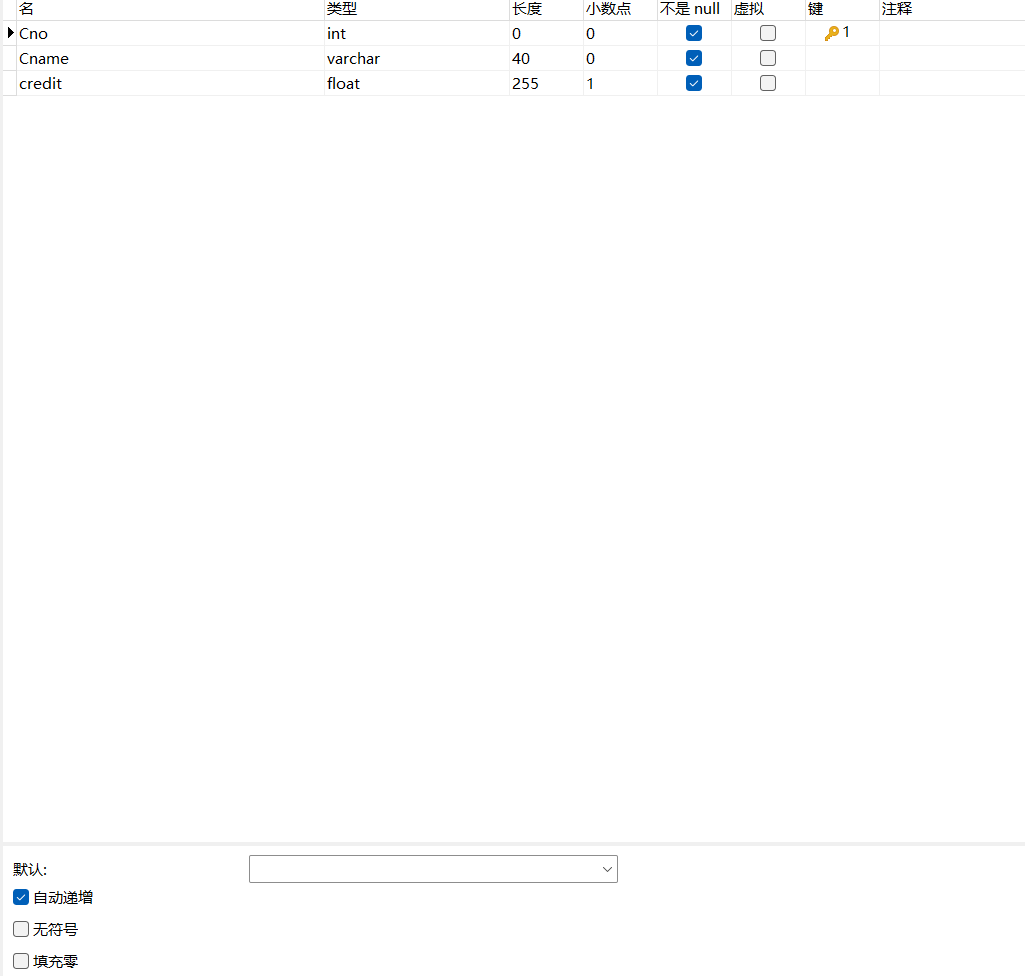
**实验要求：必做**

**一、实验目的**

熟练掌握使用Transact-SQL语言和通过Navicat输入表数据、修改表数据和删除表数据的操作。

**二、实验内容**

1、在数据库hub中建立一个名为newcourse的表，其结构与表course完全一样（注意各字段的默认值、字符集、排序方式等），如图：（其中Cno设置为自增）



2、试将表course中的所有记录插入到表newcourse中去。

3、在表newcourse中插入马克思主义基本原理，其内容为Cname为马克思主义基本原理，credit为4。

4、将表newcourse中credit为“2”的课程的credit改为2.5。

5、将表newcourse中课程的credit均加上0.5。

6、删除表newcourse中课程名为复变函数的课程。

**三、课后练习题**

以下题目在数据库hub中完成。

1、通过Navicat，在newcourse输入10条记录。

2、插入一条课程记录（“null”，“大学英语”，3）到表newcourse。

3、清空表newcourse的所有记录。

4、删除表newcourse。

### 实验7 视图管理

**实验学时：1学时**

**实验类型：设计**

**实验要求：必做**

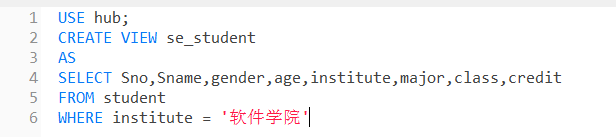
**一、实验目的**

掌握视图的定义与维护操作，加深对视图在关系数据库中作用的理解。

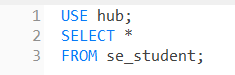
**二、实验内容**

在数据库hub中完成以下操作：

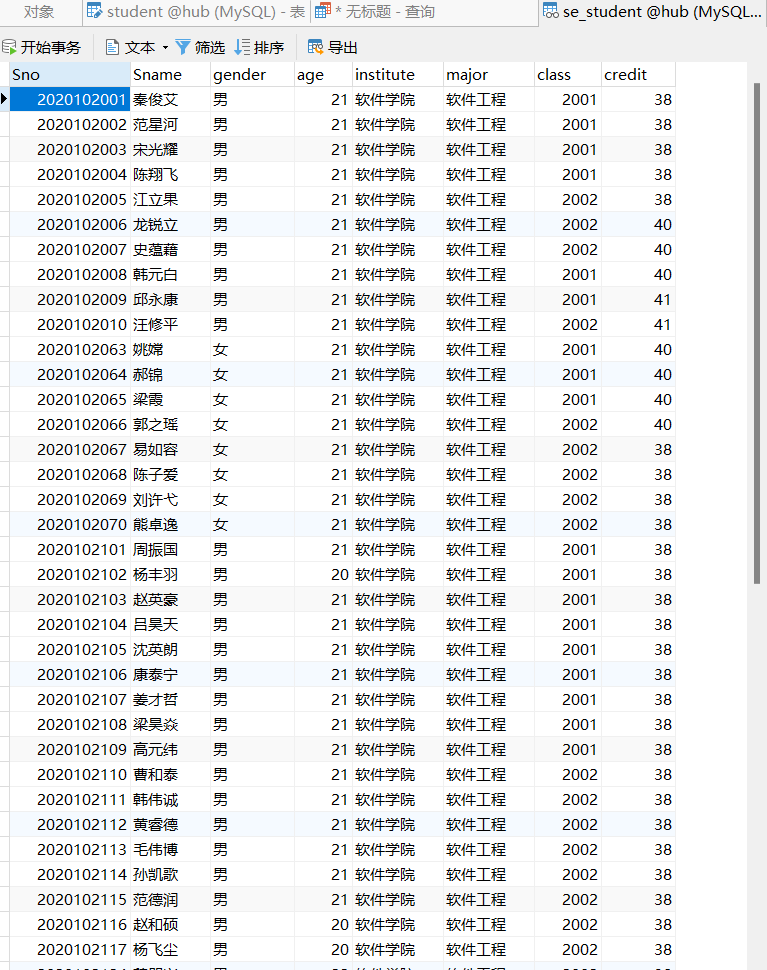
1. 建立软件学院学生的视图se\_student，并要求进行修改和插入操作时仍需保证该视图只有亚洲国家，视图的属性名为Sno，Sname，gender,age,institute，major，class，credit。



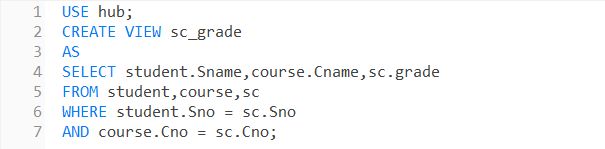
对于视图的使用，将其当做基本表即可，可用下面语句用来显示student表中的所有软件学院的学生及其基本信息：



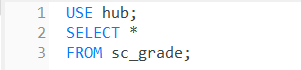
或者直接在左侧目录【hub】下的【视图】下多出的“se\_student”上点击右键，在弹出选项里选择“打开视图”。结果如图所示：



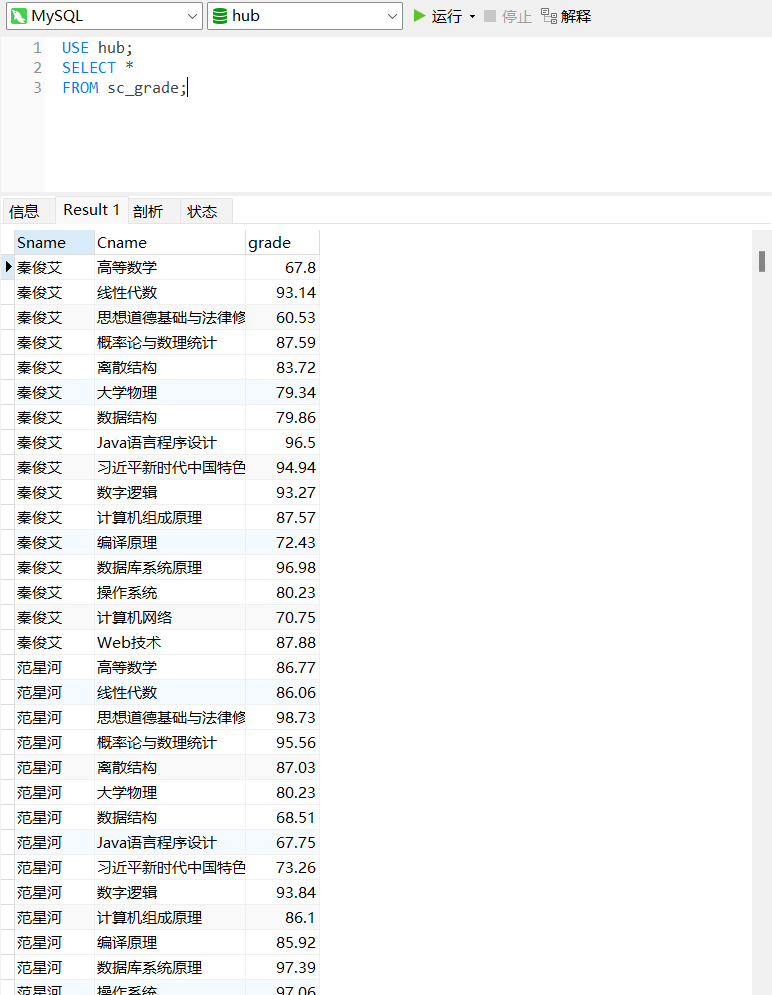
2、建立学生姓名（student.Sname),选修课程名字（course.Cname)和得分情况（grade)的视图。本视图由三个基本表的连接操作导出，其SQL语句如下：



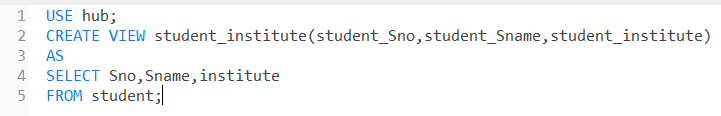
用以下语句查看：



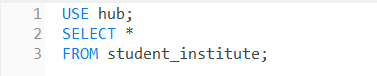
结果如图：



3、定义一个反映学生学号、姓名、院系的视图student\_institute。



用以下语句查看：



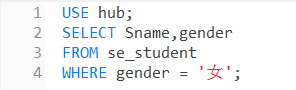
结果如图所示



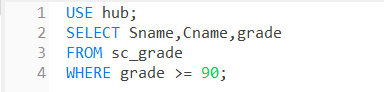
4、删除视图student\_institute。



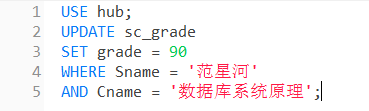
5、在软件学院学生视图se\_student中找出性别(gender)为女的学生名字(Sname)和性别(gerder)。。



6、在选修课程视图sc\_grade中找出成绩大于等于90分的学生名字、课程名和成绩。



7、将学生选修视图sc\_grade中学生姓名为范星河，课程名为数据库系统原理的课程成绩改为90。



8、向软件学院学生视图中插图一条新的学生记录，其中学号为2020102150，姓名为赵阳，性别为男，年龄为21，院系为软件学院，专业为软件工程，班级为2002，总学分为0。



1. **课后练习题**

1、建立选修数据库课程的视图。

2、建立属于计算机科学与技术学院学生的视图，并要求进行修改和插入操作时仍须保证该视图只有数统学院的学生。

3、建立选修数据库系统原理课程且成绩在90分以上的选修视图。

### 实验8 数据类型的使用

**实验学时：1学时**

**实验类型：设计**

**实验要求：选做**

**一、实验目的**

熟练掌握使用Navicat和Transact-SQL语言两种方法创建、修改和删除表。掌握管理表的有关系统存储过程。

**二、实验内容**

1.了解并尝试使用MySQL的各数据类型，不作实验操作内容上的硬性要求。

1.1日期和时间数据类型



1.2整型

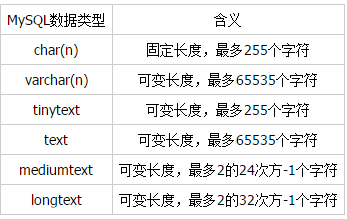


上面定义的都是有符号的，也可以加上unsigned关键字，定义成无符号的类型，那么对应的取值范围就要翻番了，比如：tinyint unsigned的取值范围为0~255。

1.3浮点型



1.4字符串类型



①char（n）和varchar（n）中括号中n代表字符的个数，并不代表字节个数，所以当使用了中文的时候(UTF8)意味着可以插入m个中文，但是实际会占用m\*3个字节。

②同时char和varchar最大的区别就在于char不管实际value都会占用n个字符的空间，而varchar只会占用实际字符应该占用的空间+1，并且实际空间+1<=n。

③超过char和varchar的n设置后，字符串会被截断。

④char在存储的时候会截断尾部的空格，varchar和text不会。

⑤ varchar会使用1-3个字节来存储长度，text不会。

1.5其他类型

① enum(“member1″, “member2″, … “member65535″)：enum数据类型就是定义了一种枚举，最多包含65535个不同的成员。当定义了一个enum的列时，该列的值限制为列定义中声明的值。如果列声明包含NULL属性，则NULL将被认为是一个有效值，并且是默认值。如果声明了NOT NULL，则列表的第一个成员是默认值。

②set(“member”, “member2″, … “member64″)：set数据类型为指定一组预定义值中的零个或多个值提供了一种方法，这组值最多包括64个成员。值的选择限制为列定义中声明的值。

2.了解MySQL数据类型属性，并依次实际操作、验证。

2.1 auto\_increment

auto\_increment能为新插入的行赋一个唯一的整数标识符。为列赋此属性将为每个新插入的行赋值为上一次插入的ID+1。

MySQL要求将auto\_increment属性用于作为主键的列。此外，每个表只允许有一个auto\_increment列。例如：

id smallint not null auto\_increment primary key

2.2 binary

binary属性只用于char和varchar值。当为列指定了该属性时，将以区分大小写的方式排序。与之相反，忽略binary属性时，将使用不区分大小写的方式排序。例如：

hostname char(25) binary not null

2.3 default

default属性确保在没有任何值可用的情况下，赋予某个常量值，这个值必须是常量，因为MySQL不允许插入函数或表达式值。此外，此属性无法用于BLOB或TEXT列。如果已经为此列指定了NULL属性，没有指定默认值时默认值将为NULL，否则默认值将依赖于字段的数据类型。例如：

subscribed enum('0', '1') not null default '0'

2.4 index

如果所有其他因素都相同，要加速数据库查询，使用索引通常是最重要的一个步骤。索引一个列会为该列创建一个有序的键数组，每个键指向其相应的表行。以后针对输入条件可以搜索这个有序的键数组，与搜索整个未索引的表相比，这将在性能方面得到极大的提升。

create table employees

(

id varchar(9) not null,

firstname varchar(15) not null,

lastname varchar(25) not null,

email varchar(45) not null,

phone varchar(10) not null,

index lastname(lastname),

primary key(id)

);

我们也可以利用MySQL的create index命令在创建表之后增加索引：

create index lastname on employees (lastname(7));

这一次只索引了名字的前7个字符，因为可能不需要其它字母来区分不同的名字。因为使用较小的索引时性能更好，所以应当在实践中尽量使用小的索引。

2.5 not null

如果将一个列定义为not null，将不允许向该列插入null值。建议在重要情况下始终使用not null属性，因为它提供了一个基本验证，确保已经向查询传递了所有必要的值。

2.6 null

为列指定null属性时，该列可以保持为空，而不论行中其它列是否已经被填充。记住，null精确的说法是“无”，而不是空字符串或0。

2.7 primary key

primary key属性用于确保指定行的唯一性。指定为主键的列中，值不能重复，也不能为空。为指定为主键的列赋予auto\_increment属性是很常见的，因为此列不必与行数据有任何关系，而只是作为一个唯一标识符。主键又分为以下两种：

①单字段主键: 如果输入到数据库中的每行都已经有不可修改的唯一标识符，一般会使用单字段主键。注意，此主键一旦设置就不能再修改。

②多字段主键: 如果记录中任何一个字段都不可能保证唯一性，就可以使用多字段主键。这时，多个字段联合起来确保唯一性。如果出现这种情况，指定一个auto\_increment整数作为主键是更好的办法。

2.8 unique

被赋予unique属性的列将确保所有值都有不同的值，只是null值可以重复。一般会指定一个列为unique，以确保该列的所有值都不同。例如：

email varchar(45) unique

2.9 zerofill

zerofill属性可用于任何数值类型，用0填充所有剩余字段空间。例如，无符号int的默认宽度是10；因此，当“零填充”的int值为4时，将表示它为0000000004。例如：

orderid int unsigned zerofill not null

3.如果要设计一个表用来存放商品的基本信息，包括商品编号、商品名称、品牌商标、型号、产地、生产厂商、生产日期、保质期、进货价格、销售价格和商品图片等。请设计出这个表的表结构，包括表名、列名、列的数据类型和长度等等。

4.将数据库hub中student表的Sname字段的数据类型改为varchar（18）.

①Navicat—设计表

②查询编辑器，输入以下语句：

USE hub;

alter table student modify column Sname varchar(18);

**三、课后练习题**

1.用Navicat和SQL语言两种方式将数据库hub中teacher表的age字段的数据类型改为int.

2. 在MySQL中建立一个表，有一列为float(5, 3).做以下试验：

①插入12.345，成功则查询该表得到结果；

②插入12.3456，成功则查询该表得到结果；

③插入123.456，成功则查询该表得到结果。

比较三次结果的差异，并分析原因

### 实验9 表数据的高级查询

**实验学时：2学时**

**实验类型：设计**

**实验要求：必做**

**一、实验目的**

本实验的目的是使学生熟练掌握MySQL查询分析器的使用方法，加深SQL语言查询语句的理解。熟练掌握数据查询中的分组、统计、计算和集合的操作方法。

**二、实验内容**

在数据库hub中完成以下操作。

1. 查询所有学生姓名及相应的选修课程名、成绩。
2. 查询系部为软件学院且数据库系统原理成绩在90分以上的学生名字和专业名称以及所属班级。
3. 统计student表中院系为软件学院的学生数。
4. 统计sc表中数据库系统原理课程的平均成绩。
5. 统计course中开设课程总数。
6. 分组统计student表中各院系的学生人数。
7. 分组统计各学院平均的思想道德基础与法律修养课程的成绩。
8. 查询有超过两条教授课程记录的教师的名字和所属院系。
9. 查询不属于软件学院的专业名称和所属院系。
10. 查询选修课程成绩在80分以下的课程名称、学生名字。
11. 查询选修数据库系统原理成绩排名前20的学生名字和成绩。
12. 查询高等数学课程平均成绩低于软件学院的院系名称及其平均成绩，以平均数的降序排列。
13. 查询sc表中多于3条成绩记录且名字以“李”开头的学生的名字和平均成绩，以平均成绩的升序排列。

**三、课后练习题**

1、建立名为SPJ的数据库。它包括S、P、J、SPJ 4个关系模式：

S(SNO,SNAME,STATUS,CITY);

P(PNO,PNAME,COLOR,WEIGHT);

J(JNO,JNAME,CITY);

SPJ(SNO,PNO,JNO,QTY)

供应商表S由供应商代码（SNO）、供应商姓名（SNAME）、供应商状态（STATUS）、供应商所在城市（CITY）组成；

零件表P由零件代码（PNO）、零件名(PNAME)、颜色(COLOR)、重量(WEIGHT)组成；

工程项目表J由工程项目代码(JNO)、工程项目名(JNAME)、工程项目所在城市(CITY)组成；

供应情况表SPJ由供应商代码(SNO)、零件代码(PNO)、工程项目代码(JNO)、供应数量(QTY)组成，标识某供应商 供应某种零件 给某工程项目的数量为QTY。

2、在数据库SPJ中，合理地为库中每一个数据表定义主键和外键。

3、适当为各数据表及其字段添加约束（唯一性、默认值等）

今有若干数据如下,利用navicat进行插入

S表

|  |  |  |  |
| --- | --- | --- | --- |
| SNO | SNAME | STATUS | CITY |
| S1 | 精益 | 20 | 天津 |
| S2 | 盛锡 | 10 | 北京 |
| S3 | 东方红 | 30 | 北京 |
| S4 | 丰泰盛 | 20 | 天津 |
| S5 | 为民 | 30 | 上海 |

P表

|  |  |  |  |
| --- | --- | --- | --- |
| PNO | PNAME | COLOR | WEIGHT |
| P1 | 螺母 | 红 | 12 |
| P2 | 螺栓 | 绿 | 17 |
| P3 | 螺丝刀 | 蓝 | 14 |
| P4 | 螺丝刀 | 红 | 14 |
| P5 | 凸轮 | 蓝 | 40 |
| P6 | 齿轮 | 红 | 30 |

J表

|  |  |  |
| --- | --- | --- |
| JNO | JNAME | CITY |
| J1 | 三建 | 北京 |
| J2 | 一汽 | 长春 |
| J3 | 弹簧厂 | 天津 |
| J4 | 造船厂 | 天津 |
| J5 | 机车厂 | 唐山 |
| J6 | 无线电厂 | 常州 |
| J7 | 半导体厂 | 南京 |

SPJ表

|  |  |  |  |
| --- | --- | --- | --- |
| SNO | PNO | JNO | QTY |
| S1 | P1 | J1 | 200 |
| S1 | P1 | J3 | 100 |
| S1 | P1 | J4 | 700 |
| S1 | P2 | J2 | 100 |
| S2 | P3 | J1 | 400 |
| S2 | P3 | J2 | 200 |
| S2 | P3 | J4 | 500 |
| S2 | P3 | J5 | 400 |
| S2 | P5 | J1 | 400 |
| S2 | P5 | J2 | 100 |
| S3 | P1 | J1 | 200 |
| S3 | P3 | J1 | 200 |
| S4 | P5 | J1 | 100 |
| S4 | P6 | J3 | 300 |
| S4 | P6 | J4 | 200 |
| S5 | P2 | J4 | 100 |
| S5 | P3 | J1 | 200 |
| S5 | P6 | J2 | 200 |
| S5 | P6 | J4 | 500 |

或利用SQL语句插入，执行如下INSERT语句

INSERT INTO S (SNO, SNAME, STATUS, CITY) VALUES

('S1', '精益', 20, '天津'),

('S2', '盛锡', 10, '北京'),

('S3', '东方红', 30, '北京'),

('S4', '丰泰盛', 20, '天津'),

('S5', '为民', 30, '上海');

INSERT INTO P (PNO, PNAME, COLOR, WEIGHT) VALUES

('P1', '螺母', '红', 12),

('P2', '螺栓', '绿', 17),

('P3', '螺丝刀', '蓝', 14),

('P4', '螺丝刀', '红', 14),

('P5', '凸轮', '蓝', 40),

('P6', '齿轮', '红', 30);

INSERT INTO J (JNO, JNAME, CITY) VALUES

('J1', '三建', '北京'),

('J2', '一汽', '长春'),

('J3', '弹簧厂', '天津'),

('J4', '造船厂', '天津'),

('J5', '机车厂', '唐山'),

('J6', '无线电厂', '常州'),

('J7', '半导体厂', '南京');

INSERT INTO SPJ (SNO, PNO, JNO, QTY) VALUES

('S1', 'P1', 'J1', 200),

('S1', 'P1', 'J3', 100),

('S1', 'P1', 'J4', 700),

('S1', 'P2', 'J2', 100),

('S2', 'P3', 'J1', 400),

('S2', 'P3', 'J2', 200),

('S2', 'P3', 'J4', 500),

('S2', 'P3', 'J5', 400),

('S2', 'P5', 'J1', 400),

('S2', 'P5', 'J2', 100),

('S3', 'P1', 'J1', 200),

('S3', 'P3', 'J1', 200),

('S4', 'P5', 'J1', 100),

('S4', 'P6', 'J3', 300),

('S4', 'P6', 'J4', 200),

('S5', 'P2', 'J4', 100),

('S5', 'P3', 'J1', 200),

('S5', 'P6', 'J2', 200),

('S5', 'P6', 'J4', 500);

第4～10题在数据库SPJ中完成。

4、查询重量最轻的零件的零件代码。

5、查询由供应商S1提供零件的工程项目名。

6、查询同时为工程J1和J2提供零件的供应商代码。

7、查询为位于天津的工程提供零件的供应商代码。

8、查询同时为位于天津或北京的工程提供红色零件的供应商代码。

9、查询供应商和工程所在城市相同的供应商能提供的零件代码。

10、查询上海供应商不提供任何零件的工程代码。

### 实验10 索引管理

**实验学时：1学时**

**实验类型：设计**

**实验要求：必做**

**一、实验目的**

本实验的目的是使学生熟练掌握索引设计的原则和技巧，能够创建合适的索引以提高数据库查询、统计分析效率。针对给定的数据库模式和具体应用需求，创建唯一索引、函数索引、复合索引等；修改索引；删除索引。同时设计SQL查询验证索引有效性。

**二、实验内容**

1.创建唯一索引，在student表的学生学号字段上创建唯一索引。



2.创建函数索引（对某个属性的函数创建索引，称为函数索引），在student表的学生名字字段上创建一个学生名字长度的函数索引。



3.创建复合索引（对两个及两个以上的属性创建索引，称为复合索引）在student表的院系名称和专业名称两个字段上创建一个复合索引。



4.创建聚簇索引，在student表的院系名称字段上创建一个聚簇索引。



5.创建Hash索引，在student表的学生名字字段上创建一个Hash索引。



1. 修改索引名称，修改student表的学生名字字段上的Hash索引名。



1. 分析某个SQL查询语句执行时是否使用了索引。



1. 使用SHOW INDEX语句来查看表的索引信息。



**三、课后习题**

1.在一个表的多个字段上创建的复合索引，与相应的每个字段上创建的多个简单索引有何异同？请设计相应的例子加以验证。

### 实验11 数据库的安全性设计

**实验学时：2学时**

**实验类型：验证**

**实验要求：必做**

**一、实验目的**

加深对数据安全性的理解，并掌握Navicat中有关用户，角色及操作权限的管理方法。掌握数据库审计的设置和管理方法，以便监控数据库操作，维护数据库安全。

**二、实验内容**

1、在Navicat中，管理建立用户和分配用户权限。

在这里，我们可以发现root是MySQL最高级别权限的用户，它拥有查看、修改和删除MySQL软件中所有数据库的权限。当需要有多个数据库，并且分配给不同的用户使用，多个用户之间只有查看自己对应数据库的权限，不相互干扰, 需要建立多个数据库和用户，给用户设置管理指定数据库的权限。

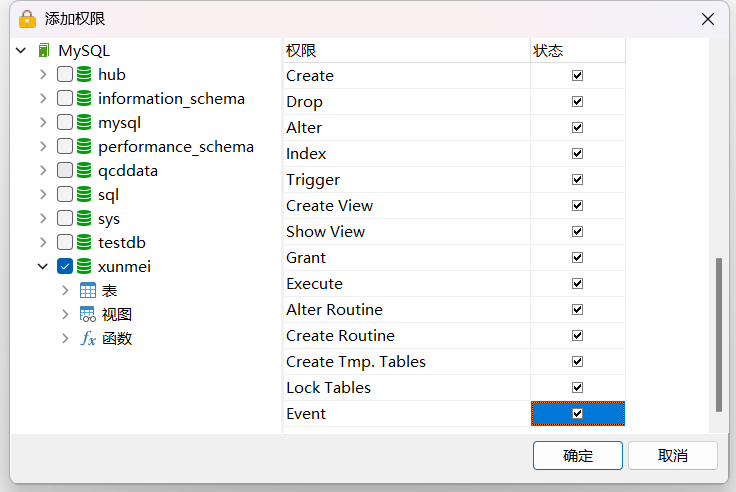
操作方法如下：

①右键点击“MySQL” 连接，选择“连接属性”，将“保存密码”取消。

②鼠标右键点击“MySQL” 连接，选择“创建数据库”，创建名为xunmei的数据库，字符集选择utf8mb4排序规则选择utf8mb4\_general\_ci。

③打开数据库xunmei后，点击界面上方的“用户”—“新建用户”—输入用户名“xmuser”、主机“localhost”、密码“123456”—保存（服务器权限自行定义）。

④点选新建的用户xmuser—编辑用户—权限—添加权限--选择数据库xunmei，添加相应权限（可不全选），如图所示：



⑤设置到此完成。为了验证我们的设置是否正确，我们可以用刚才建立的用户xmuser和密码123456来连接localhost主机地址（方法：关闭连接MySQl，该连接属于最高权限用户root。我们用新用户新建一个到localhost的连接来模拟不同用户使用数据库的过程）。连上后，可使用用户xmuser对数据库xunmei或其他数据库进行操作，看是否与自己设置的权限相吻合。（可只添加添加少量权限，然后尝试在xunmei里进行操作，看是否能成功）。

2、打开数据库审计开关，并查看存储审计信息的存储文件所在位置。

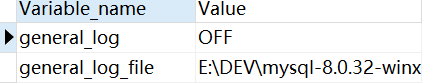
①显示当前审计开关状态。

其中log\_timestamps表示记录审计日志的时间是从那里获取，general\_log表示审计功能是否开启，general\_log\_file表示审计信息的存储文件。

show global variables like 'log\_timestamps';

show global variables like '%general%';

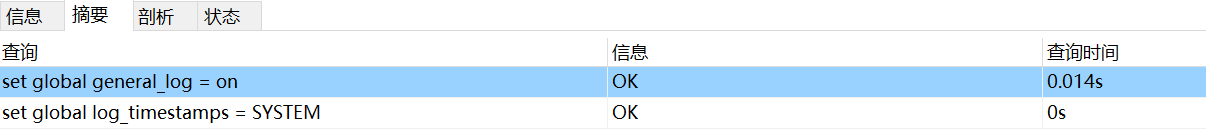




②打开审计开关

set global general\_log = on;

set global log\_timestamps = SYSTEM;



**三、课后练习题**

1、使用两个不同的用户通过Navicat和查询分析器查看hub数据库中student、course两个表的所有数据。

2、删除用户xmuser，尝试新建另一个用户来管理数据库、表等：

添加新用户对数据库hub的权限，比较在有无相关权限的情况之下进行操作的区别：

1. Create。尝试新建表
2. Update。尝试修改表数据或结构
3. ……..（自行设计，总共尝试5组，进行对比）

**四、思考题**

1、解释10个数据库操作权限的含义和影响。

2、试着设计一个例子，分析数据库审计对数据库性能的影响情况。

### 实验12 触发器实验

**实验学时：2学时**

**实验类型：设计**

**实验要求：必做**

**一、实验目的**

本实验的目的是使学生熟练掌握数据库触发器的设计和使用方法。能够理解不同类型触发器的作用和执行原理，验证触发器的有效性。

**二、实验内容**

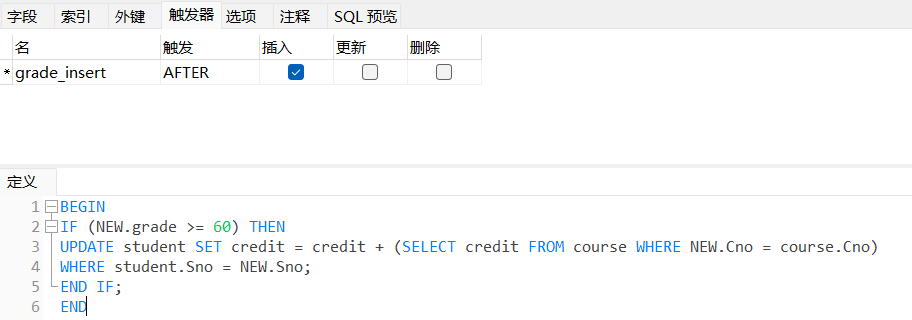
1.创建AFTER触发器并验证触发器的有效性。

①在sc表上定义一个INSERT触发器，插入学生选修课程明细时，当插入的成绩大于等于60（及格）时，自动修改student表中该学生的总学分credit，以保持数据的一致性。

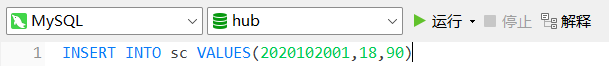
鼠标右键点击sc表，点击设计表，点击触发器

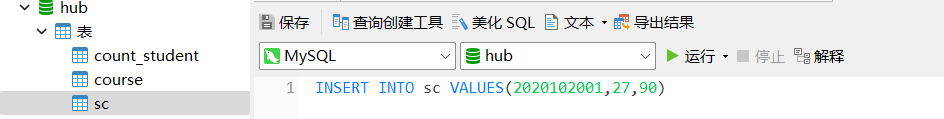


输入出发器的名字，选择触发时机为AFTER，选择插入时触发（在插入的方框中打勾），然后在定义中输入触发器的内容。



新建查询，输入以下SQL语句验证触发器的效果。





打开学生表student验证效果：



总学分credit从48分变成了52分。

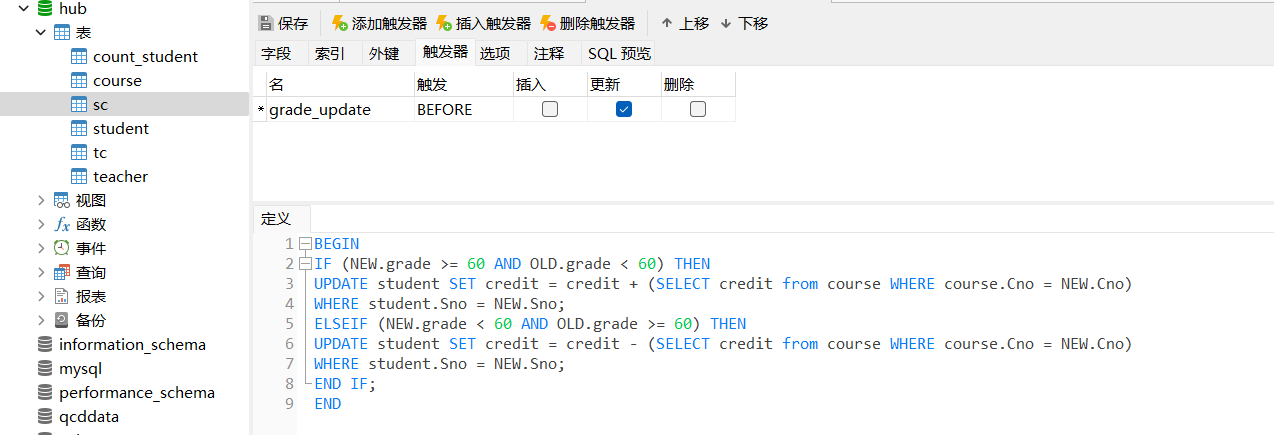
②在sc表上定义一个DELETE触发器，删除学生选修课程明细时，自动修改student表中该学生的总学分credit，以保持数据的一致性。

定义相应的触发器并验证触发器的有效性。

2.创建BEFORE触发器并验证触发器的有效性。

②在sc表上定义一个UPDATE触发器，修改学生选修课程明细时，当修改的成绩大于等于60（及格）且原成绩小于60分（不及格）时或修改的成绩小于60（不及格）且原成绩大于等于60分（及格）时，自动修改student表中该学生的总学分credit，以保持数据的一致性。

定义相应的触发器并验证触发器的有效性。



插入如下数据，同时可以验证在1中①设计的触发器在成绩为50时是否会更新该学生的总学分。



更新该条数据，将成绩50改为90。



在student表中验证总学分credit的更改情况：



从52变成了54。

**三、课后习题**

1.思考BEFORE和AFTER触发器有什么区别，在触发器定义中NEW和OLD又有什么含义？

### 实验13 存储过程实验

**实验学时：2学时**

**实验类型：设计**

**实验要求： 必做**

**一、实验目的**

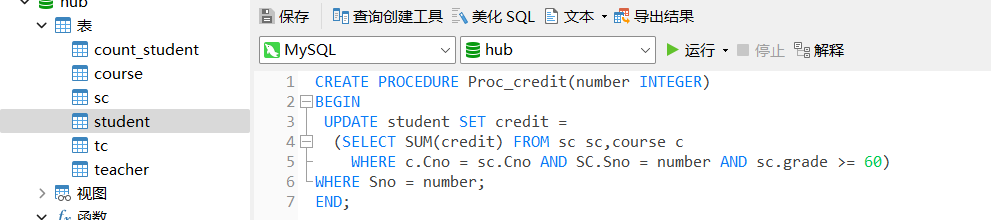
本实验的目的是使学生熟练掌握数据库PL/SQL编程语言，以及数据库存储过程的设计和使用方法；掌握数据库PL/SQL编程语言以及数据库自定义函数的设计和使用方法；掌握PL/SQL游标的设计、定义和使用方法，理解PL/SQL游标按行操作和SQL按结果集操作的区别和联系。

**二、实验内容**

1.存储过程实验：存储过程定义，存储过程运行，存储过程更名，存储过程删除，存储过程的参数传递。掌握PL/SQL编程语言和编程规范，规范设计存储过程。

（1）有参数的存储过程

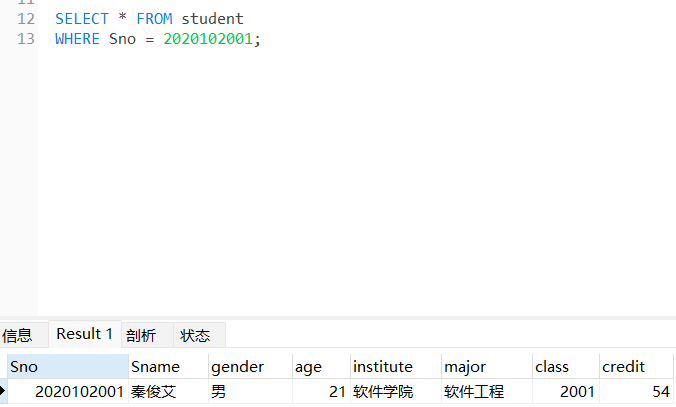
①定义一个存储过程，更新指定学号学生的总学分。



②将student表中学号为2020102001的学生的总学分credit改为0，然后执行上述定义的存储过程Proc\_credit(2020102001)。



1. 查看存储过程执行结果。



（3）修改存储过程

在navicat中点击hub->函数->Proc\_credit->设计函数

进入存储过程定义页面进行存储过程的修改，并把名字改为proc\_Credit。

（2）删除存储过程

删除存储过程proc\_Credit。



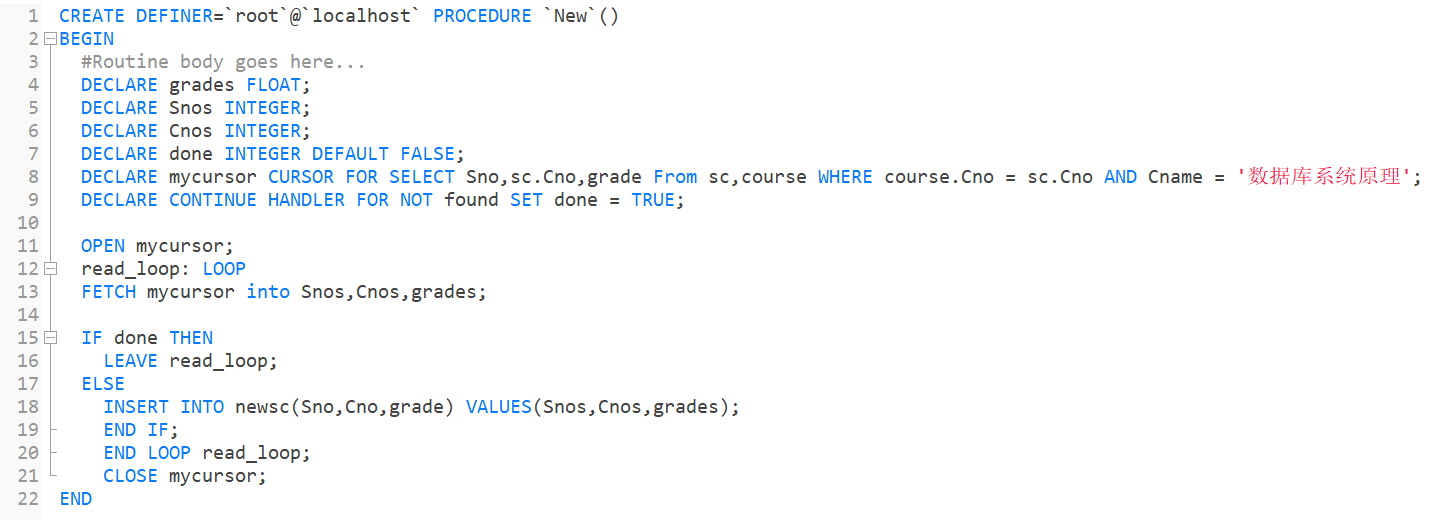
2、自定义函数实验：自定义函数定义，自定义函数运行，自定义函数更名，自定义函数删除，自定义函数的参数传递。掌握PL/SQL和编程规范，规范设计自定义函数。

3、游标定义、游标使用。掌握各种类型游标的特点、区别与联系。

（1）普通游标

①定义一个新的表newsc，其结构与sc表完全一致，用于验证游标实验。

②定义一个存储过程，利用游标将sc表中的数据库系统原理课程的选课记录及相应成绩复制到newsc表中。



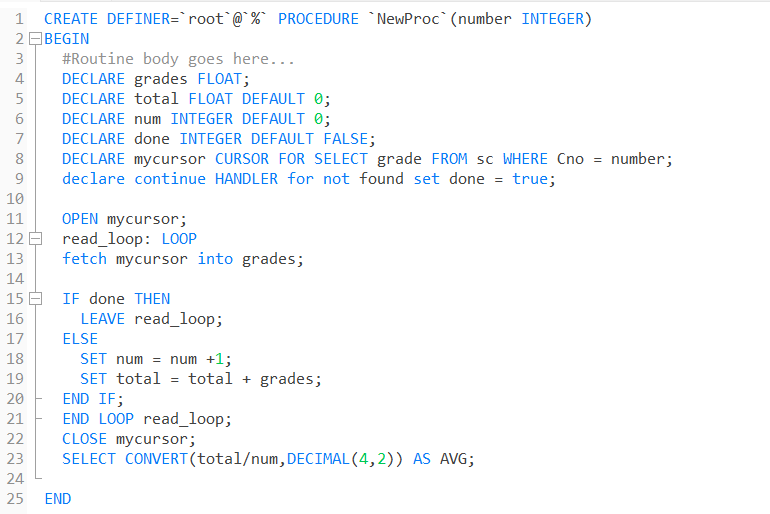
③将该过程重命名为Proc\_Cursor，执行存储过程Proc\_Cursor();



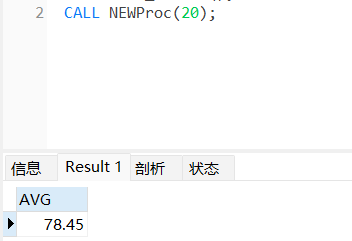
④验证存储过程的正确性，利用SQL语句查询或打开newsc表查看相关数据。

1. 带参数的游标

①定义一个存储过程，用游标计算数据库系统原理课程的平均成绩。



②执行存储过程NewProc()并验证结果。



**三、思考题**

1.试总结几种调试存储过程的方法。

2.存储过程中的SELECT语句与普通的SELECT语句格式有何不同？执行方法有何不同？

**四、课后习题**

1、尝试定义无参数的存储过程、有局部变量的存储过程、有输出参数的存储过程，执行相应存储过程并查看存储过程执行结果。

2、了解REFCURSOR类型游标，阐述与普通游标有什么不同的地方。

### 实验14 函数与事件

**实验学时：2学时**

**实验类型：设计**

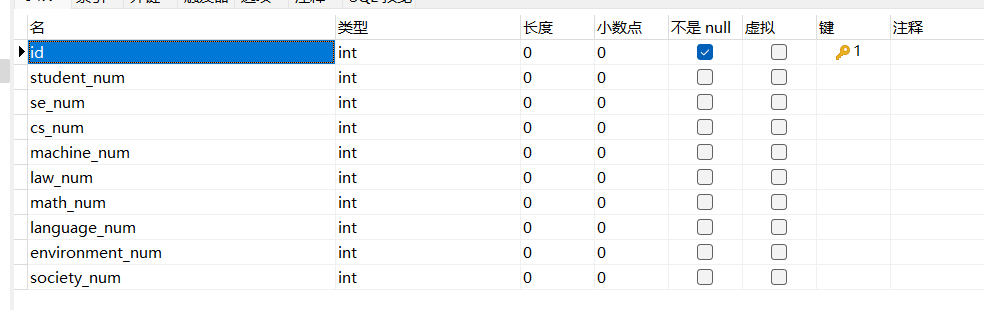
**实验要求：选做**

**一、实验目的**

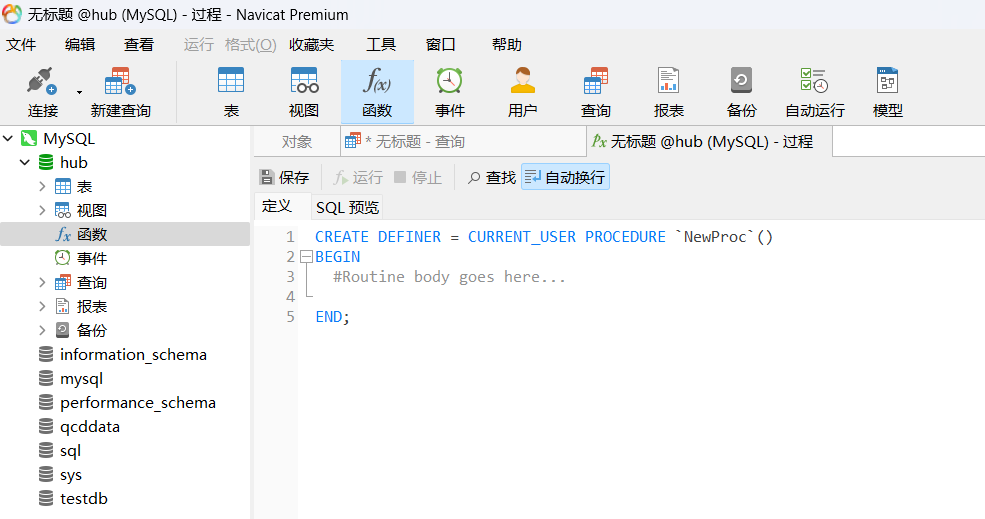
本实验的目的是使学生熟练掌握MySQL函数和事件的使用方法，加深SQL语言查询语句的理解。熟练掌握通过函数和事件使MySQl周期性完成特定事件的方法。

**二、实验内容**

1、新建统计student中institute字段的表count\_student。其中id为自增主键。

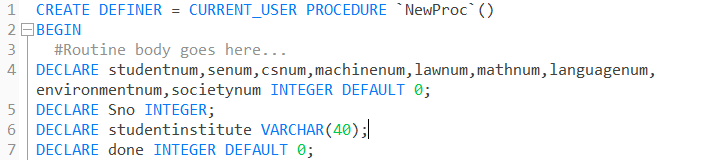


2.新建函数。在数据库hub中，点击“函数-新建函数”，类型为“过程”，点击完成。如图所示，我们将在“BEGIN”和“END”之间添加自己的函数体。



3.编写统计student表中institute的函数

①声明变量



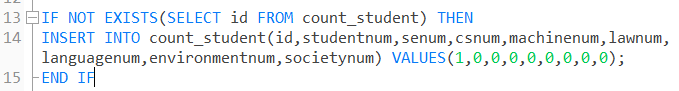
②声明对于表student中institute字段的游标cu。游标用于遍历数据库表中的数据。



③声明异常处理



④初始化count\_student表。（在该表没有统计数据时，插入一条初始数据）

id

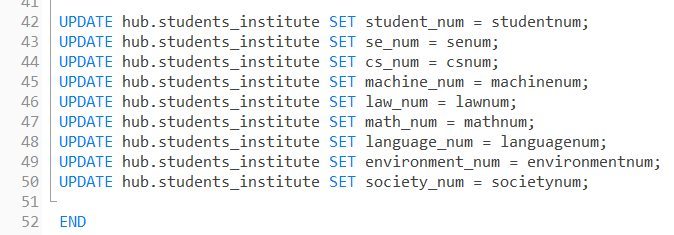
⑤开启游标，使其开始遍历institute，并将获得的值付给studentinstitute（FETCH语句）。



⑥while循环，遍历到表中最后一条数据。在循环体中将遍历得到的数据进行处理，赋值给相应的变量。



⑦更新count\_student表



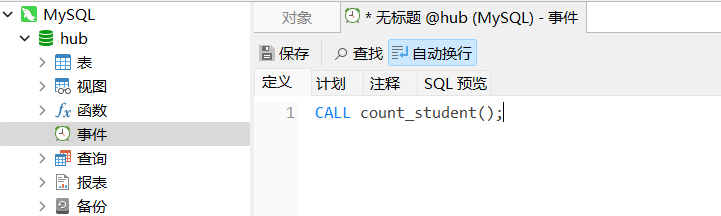
⑧保存函数为count\_student

⑨在数据库hub中选中函数count\_student，点击“运行函数”，并查看count\_student表中的数据是否随之更新。



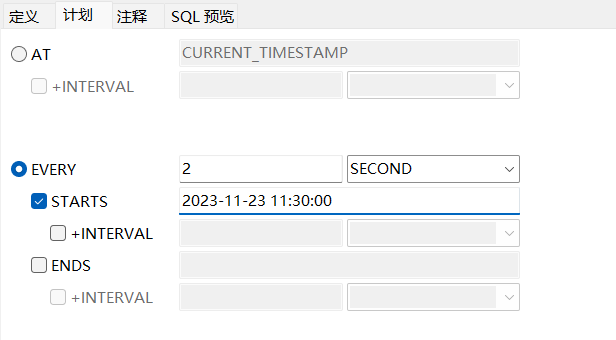
4.新建事件，事件可用于周期性的调用函数。

①在数据库quickcast中新建事件count\_user，如图所示：（注意设置其状态）





②制定事件的计划。可设定其执行间隔，起始和结束时间，持续时间等。如图所示。



③保存事件为count\_student，检查其状态是否为enable。然后修改student表中institute的数据，并按自己设定的周期检查统计表count\_student是否按照预期定时更新。

**三、课后习题**

1.新建函数和事件，每20秒统计student表中gender字段的总数和各性别所占的百分比。（如学生的总数为100，男生的百分比为68%，女生32%等等）。

### 实验15 数据库的备份与恢复

**实验学时：1学时**

**实验类型：验证**

**实验要求：必做**

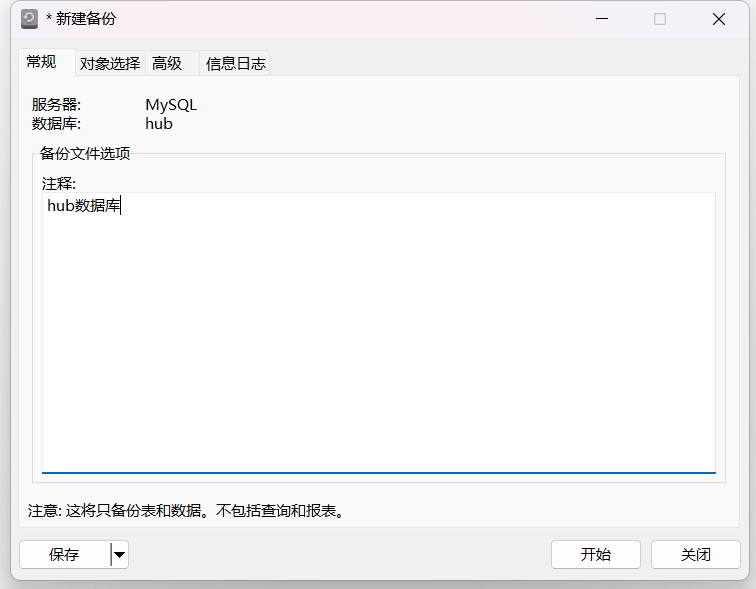
**一、实验目的**

了解Navicat的数据备份和恢复机制，掌握Navicat中数据库备份和恢复的方法。

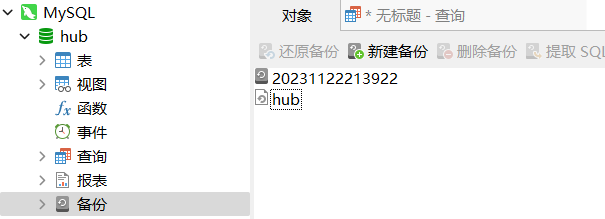
**二、实验内容**

1、备份

步骤：使用用户root打开连接MySQL—打开数据库hub—点击界面上方的“备份”—新建备份，如图所示：



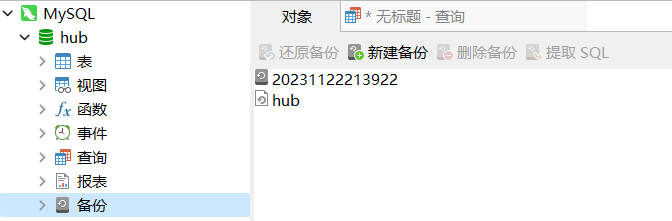
点击“开始”—成功完成后点击保存—保存名称为hub—确定，结果如图所示：



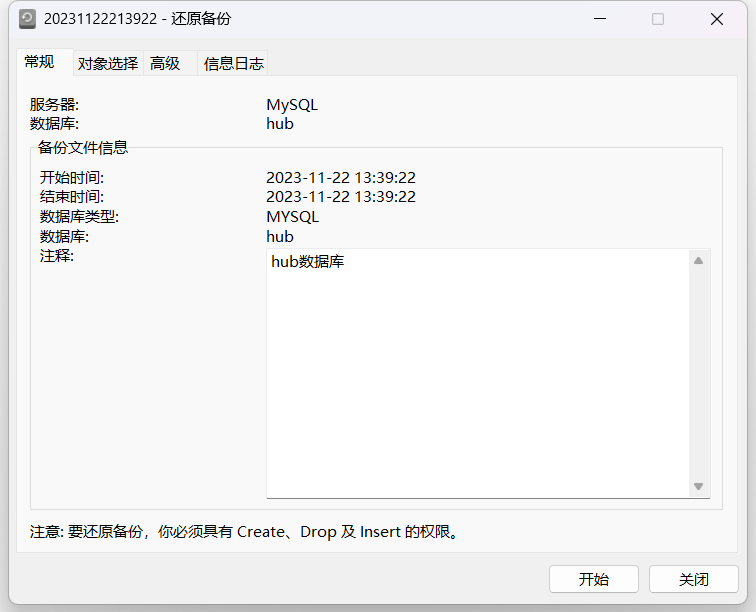
备份完成。

2、恢复还原

步骤：接第1部分操作—删除数据库hub—在连接MySQL中新建一个空的数据库hub（名称需相同）--打开—点击“备份”可以看到如图所示界面：



选择标有时间的备份文件—点击“还原备份”，会弹出如图所示对话框：



点击开始—确定。

过程成功之后会发现数据库hub已成功恢复（可能需要先关闭再打开）。

3. 用Navicat 计划任务备份hub数据库。

备份连接MySQL中的所有数据库（也可以单独选择某一数据库进行备份）：

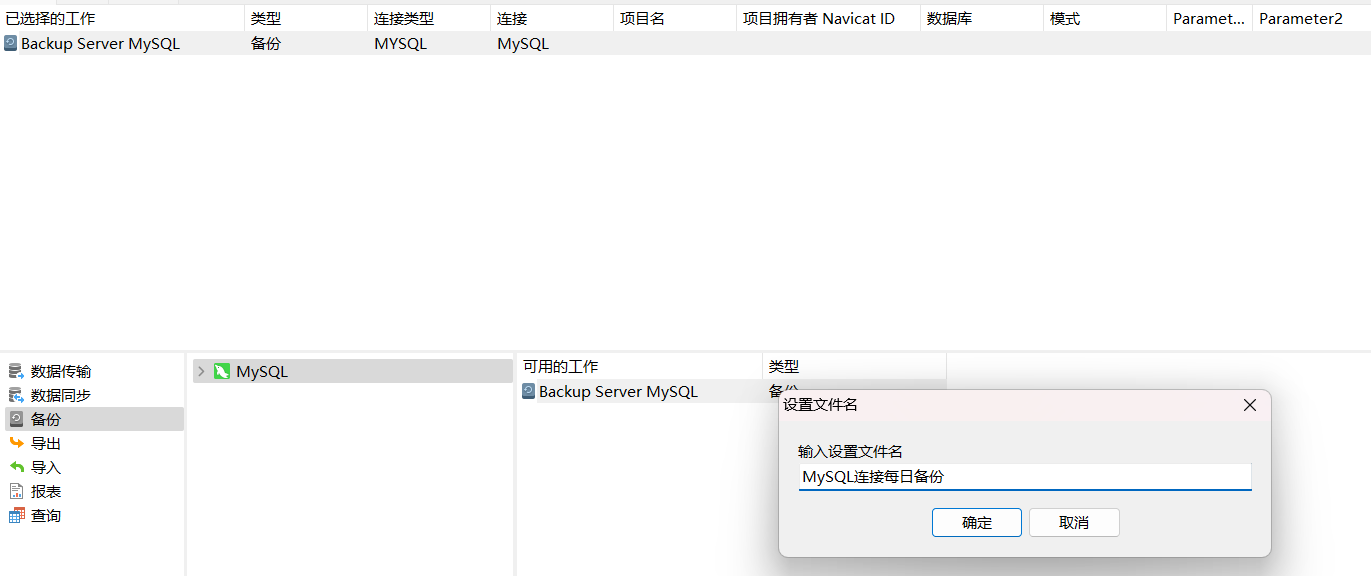
1. 点击“自动运行” ->”新建批处理作业"
2. 默认就有“Backup Server MySQL”这个备份动作。如图所示：



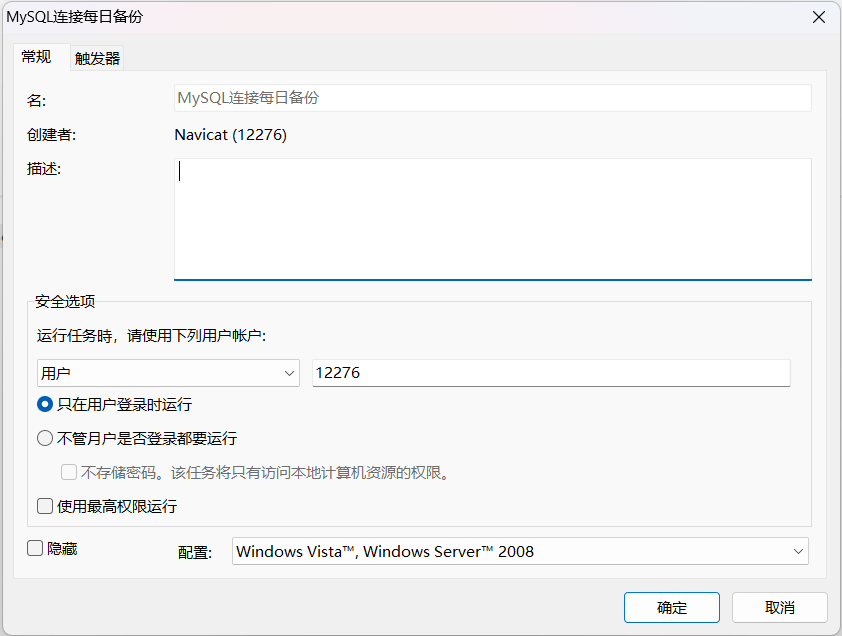
1. 双击它，它就被加入到了如图所示的计划队列中：



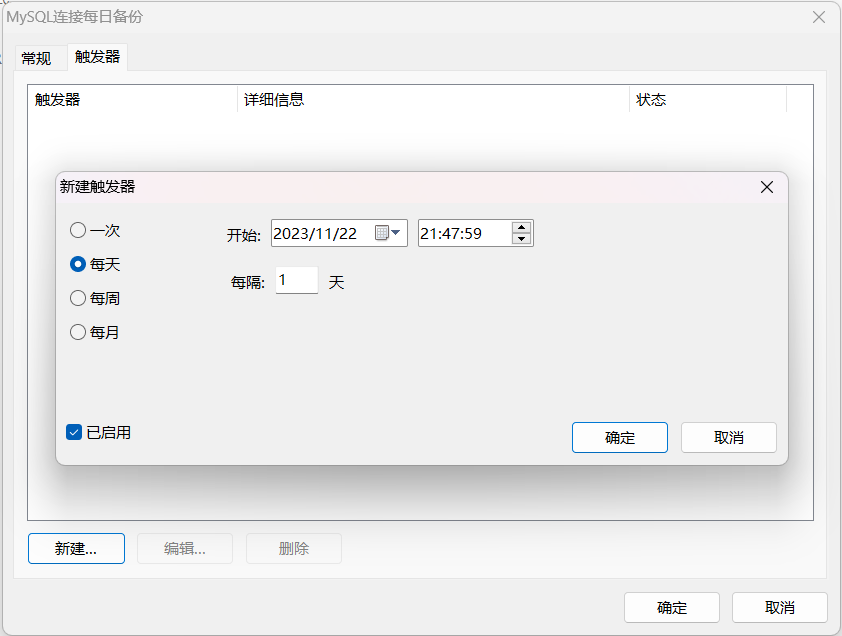
1. 点击“保存”，弹出设置文件名的对话框，输入这个计划的名称即可，如图所示：



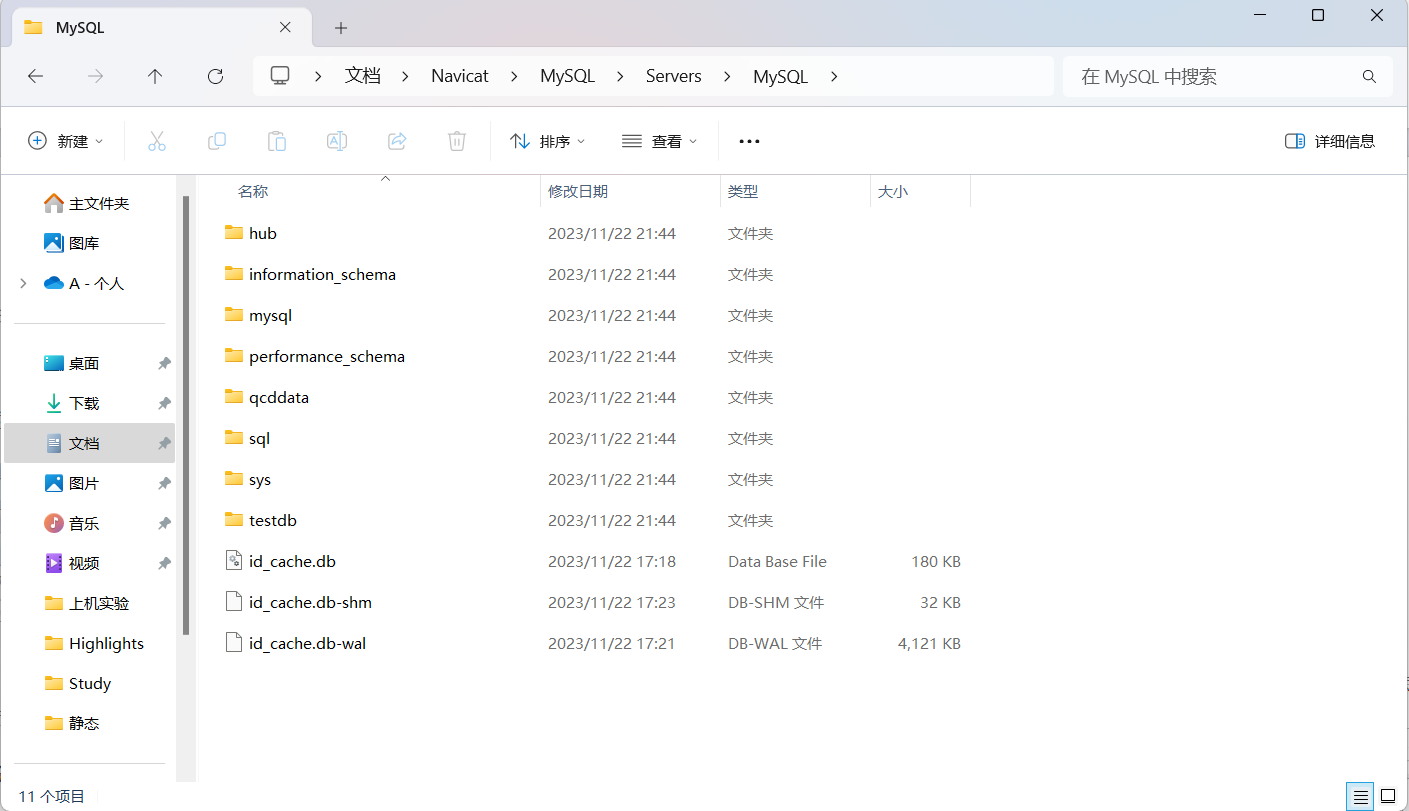
1. 保存之后，任务栏上的“设置计划任务”变成可用状态。点击“设置计划任务”（可能会报错：“指定的错误：0x80041315，任务计划程序服务没有运行。”或者“指定错误：0x80070005：拒绝访问”解决方法见本实验的步骤4），弹出设置的对话框。在“任务”标签下设置密码，就是数据库所在电脑的登录密码，没有的话可以不设。如图3.4所示：



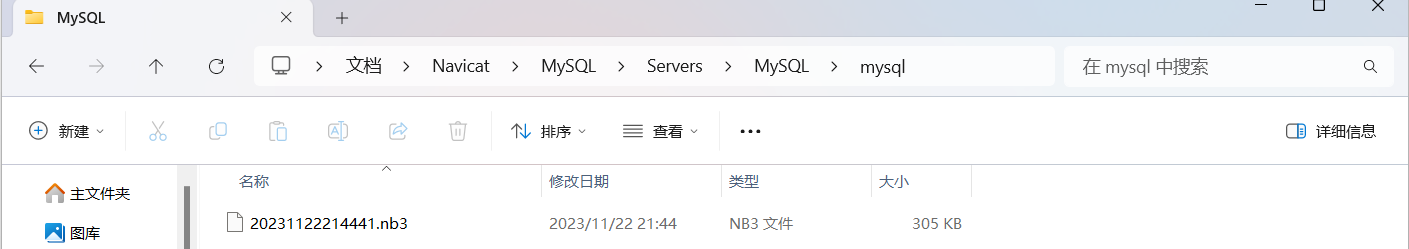
⑥然后设置这个计划的运行时间，在“触发器”标签下，点击“新建”，然后设置具体的运行频率与运行时间点。如图所示：



⑦设置完之后，点击确定，等这个计划运行过之后（可以控制设置的时间以尽快看到结果），看一下备份的效果，备份的文件的位置为：“我的文档” -> “Navicat” -> "MySQL" -> "servers" -> "MySQL"，里面就有各个数据库对应的文件夹，如图所示：(若并未找到对应文件，大概率为Windows本地安全策略屏蔽了任务，解决方案见3.5节)

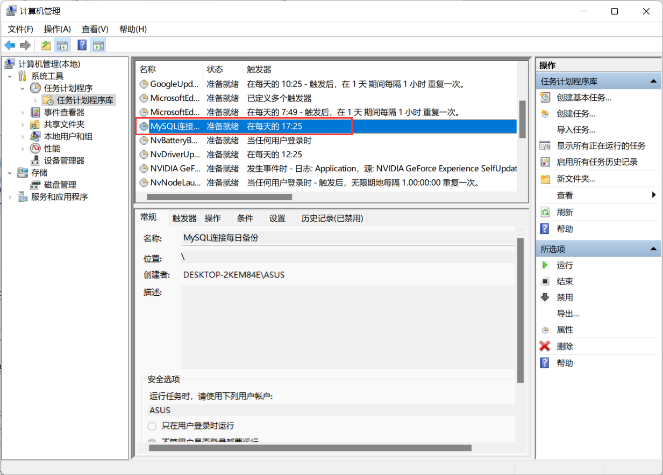


⑧各文件中会生成.psc 文件，包含了这个库的数据，以后可以用来做还原。如图所示：

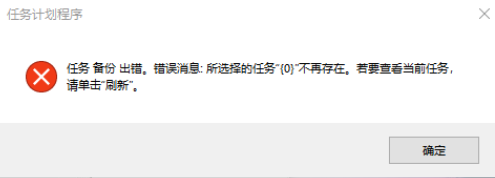


3.5、解决任务设置完成后，任务到时间不运行问题

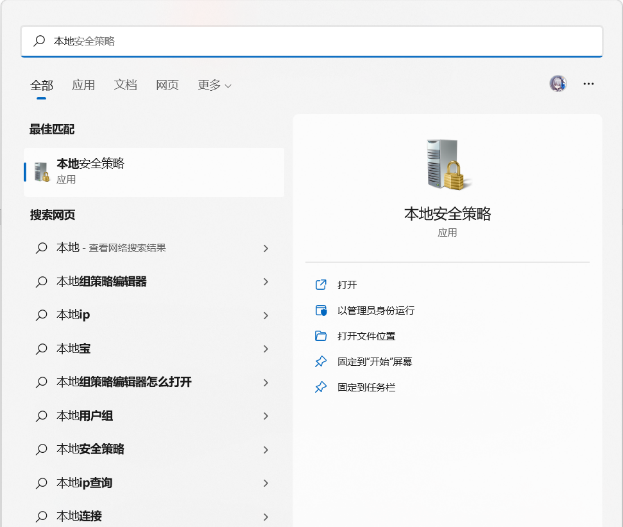
右键“此电脑”>“管理”>左侧导航栏“任务计划程序”>“任务计划程序库”，右侧选择自己创建的任务



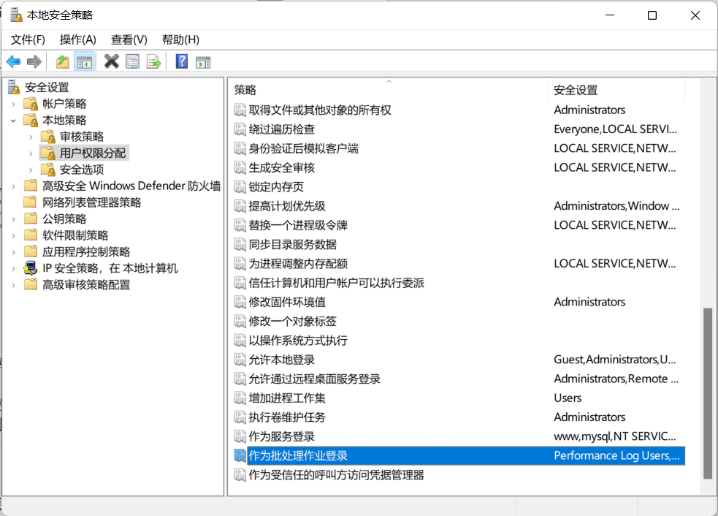
右键此任务，点击运行，若提示任务不存在如下图



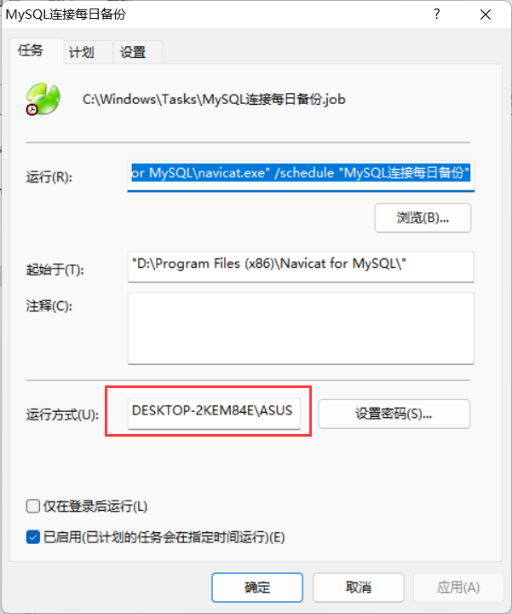
则需要改变计算机的本地安全策略，在桌面开始菜单搜索“本地安全策略”



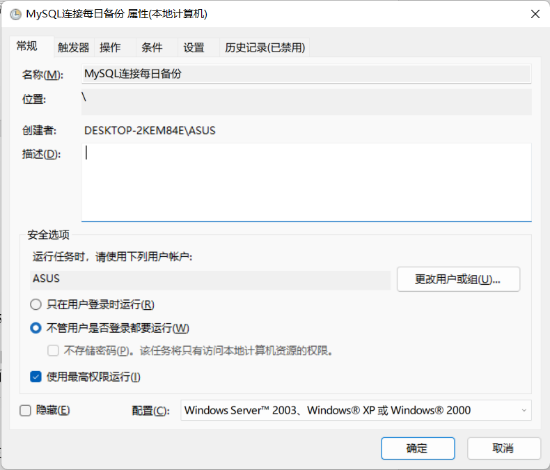
打开此软件后，选择本地策略>用户权限分配>作为批处理作业登录



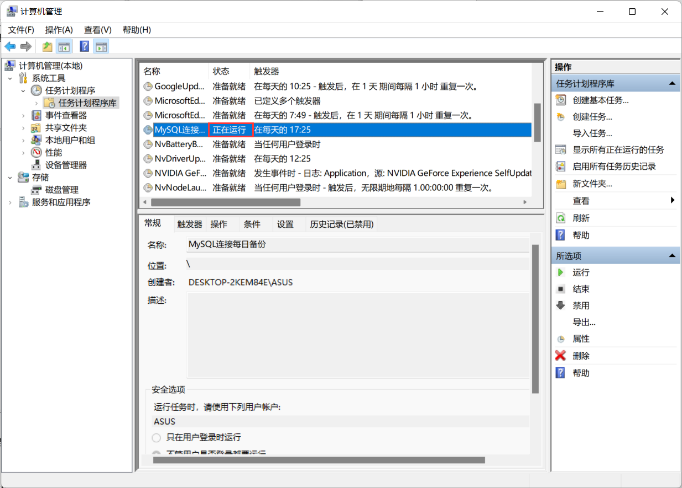
双击此项后，在弹出的界面中选择“添加用户或组”将Navicat中的用户添加到这里。



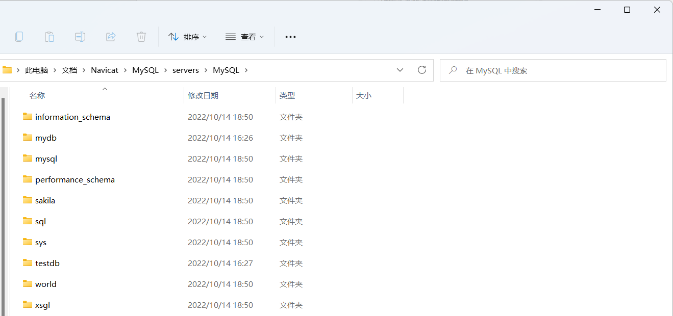
确认修改，再次通过右键“此电脑”>“管理”>左侧导航栏“任务计划程序”>“任务计划程序库”，找到自己的备份任务，右键选择属性，勾选“使用最高权限运行”



确认后，右击此任务，点击运行，任务便正常执行了



文档目录下已经可以找到备份文件，接下来正常继续实验即可



4、指定的错误：0x80041315和0x80070005的解决方法.

①指定错误：0X80041315：任务计划程序服务没有运行。主要原因是你的任务计划服务被禁用，计算机管理里面找到服务，服务里面找到Task Scheduler，看是否已被禁用，开启就行了。具体操作是打开控制面板->管理工具->服务->找到Task Scheduler选项，启动类型选：自动、服务状态选：启动。

② 指定错误：0x80070005：拒绝访问，主要原因有如下几种：

1) 登入Windows时使用空白密码的，应将组策略：“帐户：使用空白密码的本地帐户只允许进行控制台登录”改为“停用”就可以了。具体操作是打开开始运行->输入gpedit.msc打开组策略->计算机配置->windows设置->安全设置->本地策略->安全选项看右边框内：“帐户：使用空白密码的本地帐户只允许进行控制台登录”改为“停用”，就可以了。

2) 可能权限不够或被禁止，具体操作是打开开始运行->输入gpedit.msc打开组策略。(先尝试方案从c，一般可直接解决)

a. 看看组策略的用户权利指派里，禁止用户访问的几个项目有没有对应的名字！

b. 选择计算机配置->windows设置->安全设置->本地策略->用户权利指派 双击右边的 从网络访问此计算机,把需要的用户名添加到列表。

c. 选择计算机配置->Windows设置->安全设定->本地策略->安全选项 双击右边的 域控制器：允许服务器操作员计划任务，打开启用。

**三、课后练习题**

1、创建一个备份设备用于备份数据库master，并尝试还原数据库master。

2、实验二中所用到的导出和导入方法也可以达到备份的效果，尝试用该方法备份和还原数据库hub。

3、思考题：备份策略和备份规划指的分别是什么？

4、给数据库hub设计一个备份规划（频率和时间自定，要可看到结果）。

### 实验16 数据库监视与性能优化实验

**实验学时：1学时**

**实验类型：验证**

**实验要求： 选做**

**一、实验目的**

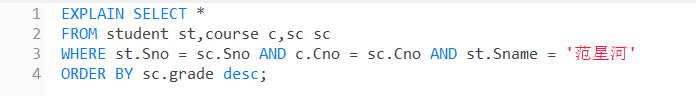
本实验的目的是使学生理解和掌握数据库性能调优的基本原理和方法，了解所使用的DBMS提供的数据库性能监视功能，学习数据库查询性能监视的基本原理和方法，了解数据库系统级参数和连接级参数的配置和调优的基本原理和方法。了解用户可以通过修改这些参数设置来调整系统运行的配置，以优化系统性能。

**二、实验内容**

1.数据库查询性能调优实验：学会使用EXPLAIN命令分析查询执行计划、利用索引优化查询性能、优化SQL语句，以及理解和掌握数据库模式规范化设计对查询性能的影响。能针对给定的数据库模式，设计不同的示例验证查询性能优化效果。

（1）使用EXPLAIN命令查看查询执行计划

查看student、course、sc三个表连接查询的查询执行计划。



该SQL语句时要查询学生名字为’范星河’的学生相关信息、选修课程相关信息和课程成绩，并按照课程成绩（降序），排序输出结果。

1. 利用索引优化查询性能

利用索引实验创建的索引进行SQL查询，优化SQL查询性能。比较在student表的Sname上有索引和无索引时，两种执行计划有何异同，并实际执行该查询，验证有索引和无索引时此查询语句的执行性能。

（3）优化SQL语句

①IN与EXISTS查询：一般地使用EXISTS查询效率要高于IN查询。分别利用IN和EXISTS进行SQL查询，比较两种执行计划，并实际测试执行性能哪种情况好。

②尽可能使用不相关子查询，避免使用相关子查询。不相关子查询一般比相关子查询执行效率高，在可能的情况下，改写相关子查询为不相关子查询。比较两种执行计划，并实际测试执行性能哪种情况好。

（4）数据库模式规范化对查询性能的影响

分析该数据库模式中是否存在不规范的设计。该设计在海量数据的情况下查询效率咋样？如何在设计上进一步提高海量数据的查询效率？

第三范式在一定程度上减少了不必要的冗余，提高了数据库的查询效率，但是如果数据量大且需要大量联合查询的时候，第三范式设计又可能会影响查询效率。

2.（了解）数据库性能监视实验：使用MySQL的数据库性能监视工具，通过标准统计视图和统计访问函数查看数据库系统收集到的性能统计信息、ANALYZE更新数据库统计信息，通过专门工具监视系统性能。希望能够熟悉数据库系统有关性能统计信息的标准视图和统计访问函数，了解如何通过系统收集到的性能数据监视系统性能。

3.（了解）数据库系统配置参数调优实验：熟悉和了解数据库各级参数的作用以及配置，包括系统级参数配置和调优、数据库级参数配置和调优、会话（连接）级参数配置和调优。

**三、课后习题**

1.对表数据的高级查询实验中的查询，使用不同的SQL语句来表达，比较它们的查询效率，体会并总结查询优化的技巧和方法。

### 实验17 并发控制技术实验

**实验学时：2学时**

**实验类型：验证**

**实验要求：选做**

**一、实验目的**

本实验的目的是使学生掌握数据库并发控制的基本原理及其应用方法。

**二、实验内容**

1、验证并发操作带来的数据不一致性问题，包括丢失修改、不可重复读和读“脏”数据等情况。要求通过取消查询分析器的自动提交功能，创建两个不同的用户，分别登录查询分析器，同时打开两个客户端；通过SQL语言设计具体例子展示不同的封锁级别的应用场景，验证各种封锁级别的并发控制效果，以进一步理解封锁技术是如何解决事务并发导致的问题。

DBMS通常提供四级隔离级别：读未提交（read uncommitted）、读已提交（read committed）、可重复读（repeatable read）、可串行化（serializable）。

四级隔离级别与三级封锁协议大致对应关系如下：

①读未提交隔离级别提供最大的事务并发度，但仅能避免丢失修改，对应一级封锁协议。

②读已提交隔离级别提供的事务并发度减弱，能够避免丢失修改和脏读，对应二级封锁协议。

③可重复读隔离级别提供的事务并发度进一步减弱，能够避免丢失修改和脏读和不可重复读，对应增强的二级封锁协议。

④可串行化隔离级别提供最小的事务并发度，能够避免所有的事务并发控制问题，对应三级封锁协议。

**三、思考题**

1.试分析在“读已提交”隔离级别下，设计一个应用程序以避免“不可重复读”和“幻影”。

### 实验18 数据库设计

**实验学时：4学时**

**实验类型：设计**

**实验要求：必做**

**一、实验目的**

1、熟悉数据库应用系统需求分析；

2、熟悉数据库设计的基本方法；

3、掌握数据库和数据表的创建方法；

4、掌握视图的创建方法；

5、了解各种数据完整性约束；

6、对存储过程和触发器有一个完整的认识。

**二、实验要求**

给出网上书店管理系统的需求分析报告（见四），要求根据功能分析和数据分析设计概念模型，实现完整的数据库（包括：各种表，视图，必要的存储过程，触发器等）。

**三、实验内容和步骤**

1、表结构设计：设计文档要求说明每一张表以及每一个字段的含义，说明每个表的主键、外键、约束等；

2、建立数据库；

3、建立表及必要的约束；

4、建立必要的视图；

5、建立必要的存储过程和触发器。

**四、相关资料**

**网上书店管理系统的需求分析报告**

设计一个小型的数据库应用系统，服务于一个网上书店的日常处理业务及网上购书业务，包括供书目录、库存管理、采购管理、客户管理，顾客订单管理、发货管理，供应商管理及网上浏览等。

总体设计要求：书店内部日常处理业务通过C/S模式来实现，而顾客可通过Internet进行网上订书及查询浏览，为改善性能，应尽量将业务规则放在中间层或数据库服务器一端。

该系统的主要功能如下：

1、供书目录及库存管理

供书目录管理及库存管理为书店的主要日常业务，应包括建立和更新供书目录，新书入库功能等。需要记录供书的信息有：书号，书名，作者（最多可包括四个作者，有序），出版社，价格，关键字（最多10个），本书的目录（可选），封皮（可选），存货量，供书商（一本书可有多个供应商，可选）等，并且需考虑丛书的问题（即一个书号包含多本书），库存存放位置（可采用示意图表示，可选）。

2、采购管理

采购管理为书店的主要日常业务之一，包括缺书登记管理及采购单管理。

缺书登记可通过以下几种方法生成：

1）直接进行缺书登记，包括书号，书名，出版社，供书商，数量，登记日期。

2）存书量低于一定限度时，自动生成缺书记录（注：不能有重复）,最低存书量可设定。（可选）

3）顾客在网上进行缺书登记或订货数量超出库存时，生成缺书记录（注：不能有重复），必要时，应记录下顾客的缺书请求，以便答复。（可选）

采购单管理：根据缺书记录单可选择生成采购单，到货时对采购单作相应处理，并增加库存量，删除缺书记录，EMAIL通知缺书登记的顾客（可选）等。

3、客户管理

客户管理包括客户信息管理及信用管理。需要记录客户的信息包括：网上ID，登录密码，名称，地址，帐户余额，信用等级等。

有关客户信息的管理，分别由书店管理和客户网上管理完成，书店管理包括收到客户的款项后，增加客户的帐户余额；调整信用额度。客户其他信息由客户在网上维护，如注册一个新的用户，修改用户信息等等。

信用等级分为五级，一级为10%的折扣，不能透支；二级为15%的折扣，不能透支；三级为15%的折扣，可先发书再付款，透支有额度限制；四级为20%的折扣，可先发书再付款，透支有额度限制；五级为25%的折扣，可先发书再付款，透支无额度限制。信用等级管理有两种形式：一种是书店进行管理；一种是自动进行管理，即在每月月初根据该帐户的余额或累计购书总金额进行调整，如帐户余额超过一定金额，提升其信用额度，或者累计购书总金额超过一定金额提升其信用额度（可选）。

4、顾客订单管理和发货管理

顾客订单包括订单号，订货日期，客户ID，书号，订书数量，金额，发货地址，发货情况等。

订单由顾客在网上申请生成。一个订单可订多本书（可选）。并仅能对书库中已有的书目进行订购，库存量不足时可以先订货；当客户查询的书目在书库中没有找到时，将会询问客户是否希望做进一步的询价及报价。

发货管理为书店日常业务管理之一，必须根据订单情况及客户的信用等级判断其付款额是否到帐，并相应扣减其帐户余额后，才能发货。对一个订单可分次发货（可选）

5、供应商管理

供应商相关信息，包括：供应商基本信息，供应商供货信息，并且不同的供应商发布自己现有书目信息。

6、网上浏览查询

可查询以下二类信息：

1）客户相关信息，包括：客户基本信息，客户历史订单信息，相关订单的发货信息等。

2）书目信息：可按照书号，书名，出版社，关键字（可指定匹配程度（可选））、作者（按第一，第二……作者查询（可选）），进行查询，并可进行模糊查询（可选）。

要求：必须根据所有必选的功能和数据需求设计实现完整的数据库（包括：各种表，视图，必要的存储过程，触发器等）。

**五、课后习题**

1.试选择一个应用，练习数据库设计。

### 实验19 数据库应用开发实验

**实验学时：2学时**

**实验类型：设计**

**实验要求：选做**

**一、实验目的**

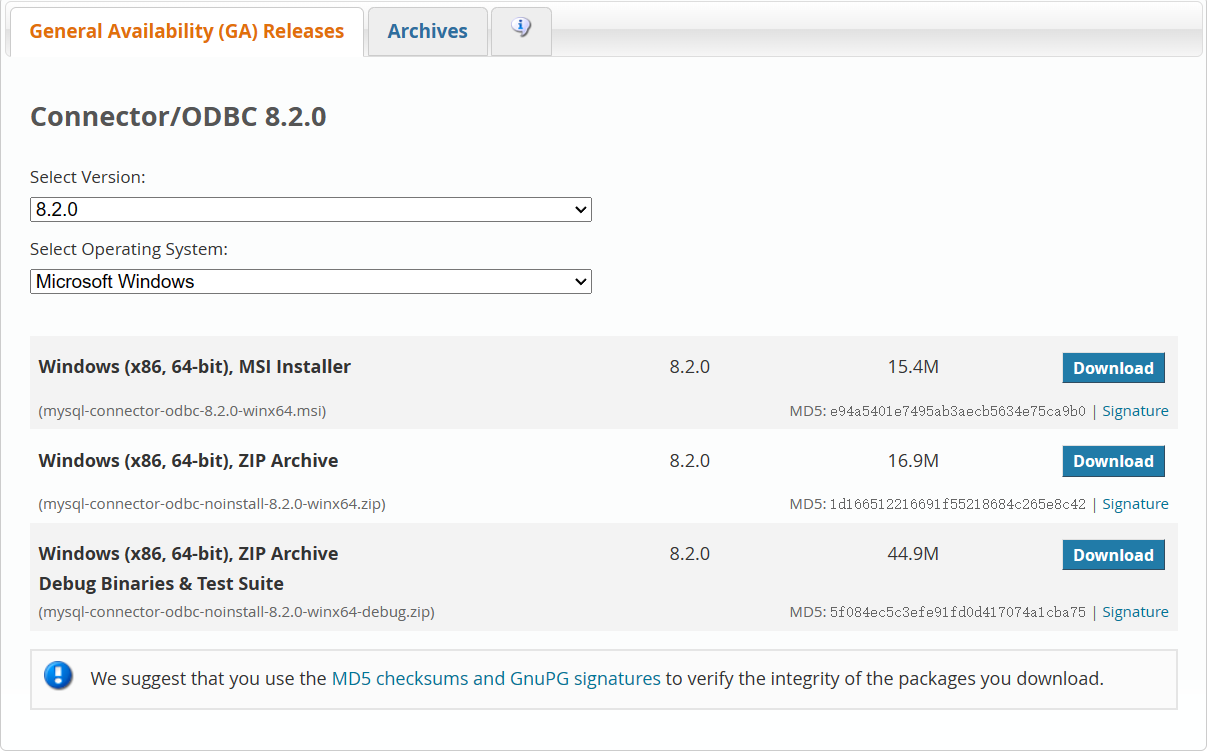
本实验的目的是使学生掌握基于ODBC驱动的数据库应用开发方法和基于JDBC驱动的数据库应用开发方法。

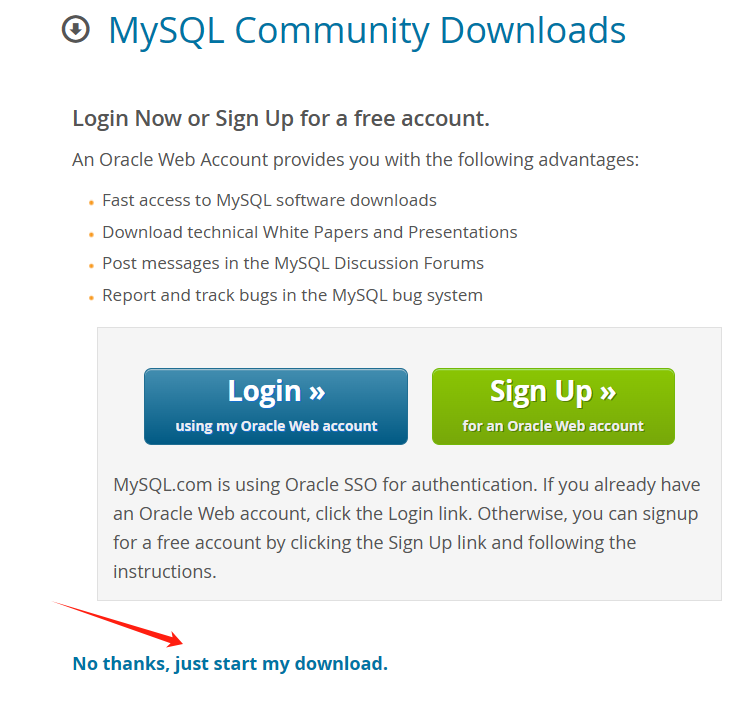
**二、实验内容**

1、基于ODBC的数据库应用开发实验：设置ODBC驱动数据源，基于ODBC驱动的数据库连接方法，实现数据库数据操纵等应用开发常见功能。

（1）下载MySQL ODBC驱动：

下载地址：[MySQL :: Download Connector/ODBC](https://dev.mysql.com/downloads/connector/odbc/)

打开的页面如下图所示，可根据实际使用情况选择下载相应的MSI安装程序，使用Visual C++ 6.0环境编程请下载32位MSI安装程序。**点击下载后出现如下页面，直接点击不用了，请开始下载。**

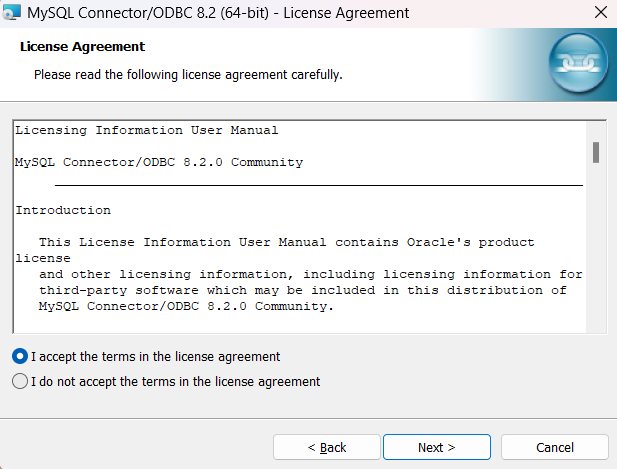


1. 安装MySQL ODBC驱动

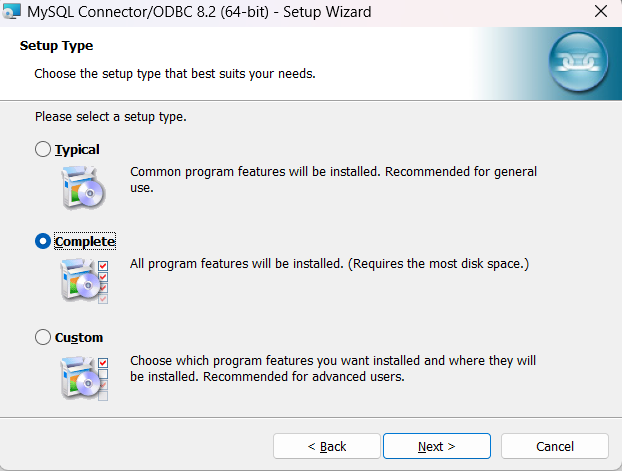
运行MSI安装程序，出现如下页面：



点击Next；



**选择 I accept the terms in the license agreement，点击Next；**

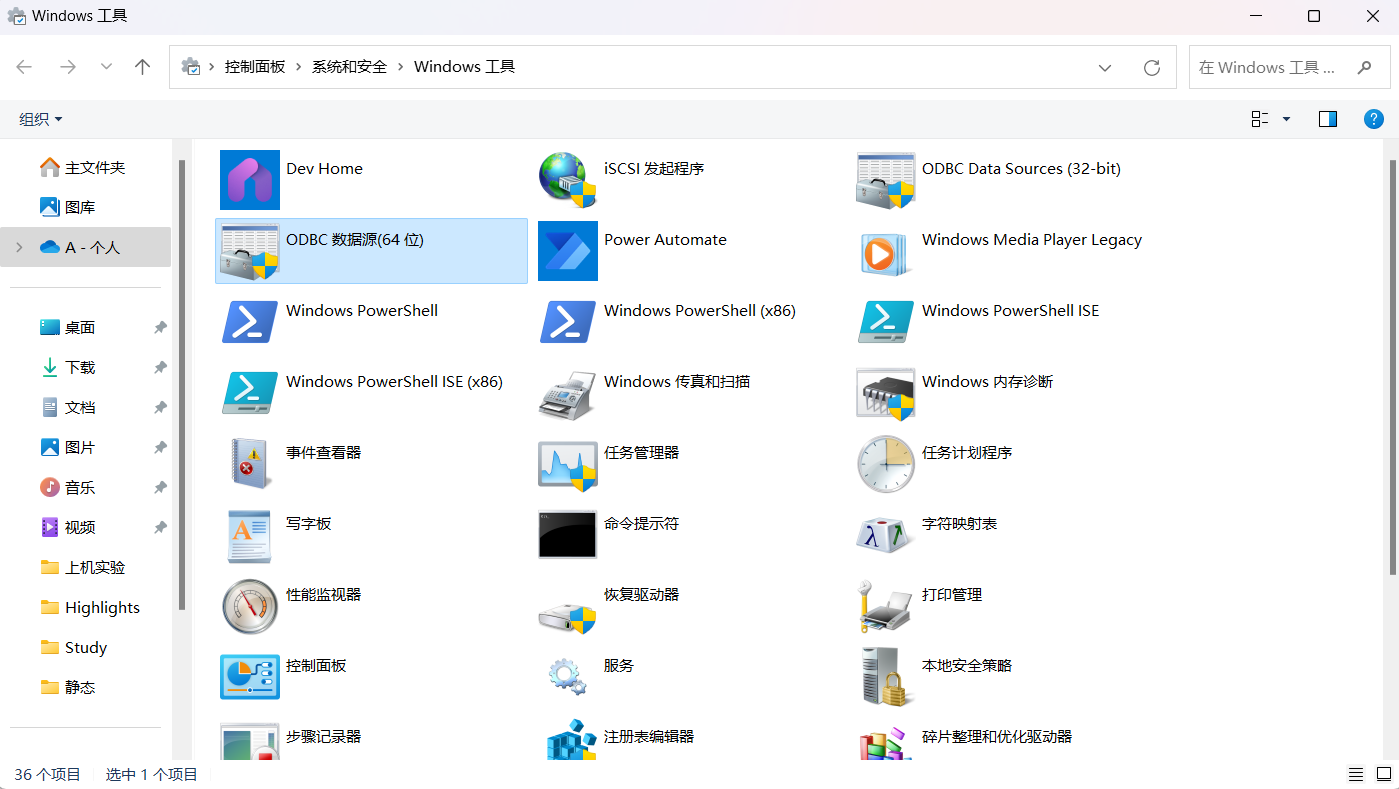


选择Complete；



点击Install，这样就安装完成了。

1. 配置MySQL ODBC数据源

①依次选择控制面板->系统和安全->Windows工具->ODBC数据源（64位）；

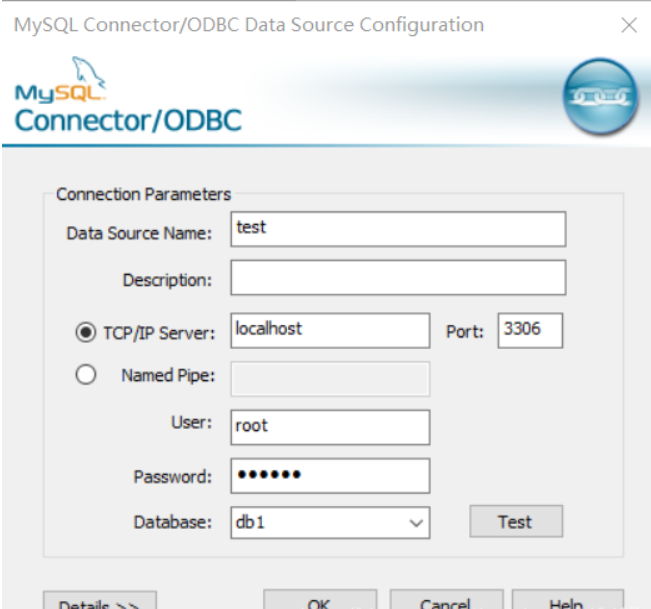
②选择系统DNS中的添加；



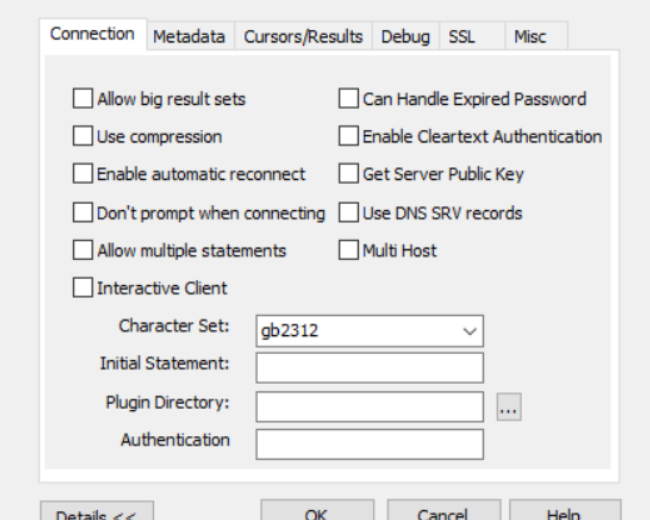
③选择MySQL ODBC 8.2 Unicode Driver；



④Data Source Name填写需要生成的ODBC数据源的名称。Description选填。如果使用远程数据库服务器，TCP/IP Server填写数据库服务器的IP地址，Port填写数据库服务器的相应端口；如果使用本机数据库，TCP/IP Server填写为localhost，端口填写MySQL数据库端口(默认为3306);User为数据库用户名，Password为相应用户名的密码，Database为MySQL下建立的数据库名；



如果使用过程中如果出现中文乱码情况，可选择上图中的Details选项，在弹出的页面中将Character Set改为gb2312。



④参数设置结束后，点击Test，出现如图所示的显示框表示配置成功；



⑤**选择OK保存。如需要修改，选择相应数据源后点击配置，可再次修改相关参数。**

(4)建立连接时的分配连接句柄

ret = SQLAllocHandle(SQL\_HANDLE\_DBC,mysqlhenv,&mysqlhdbc);//mysqlhenv为环境句柄，mysqlhdbc为连接句柄,ret为结果返回集

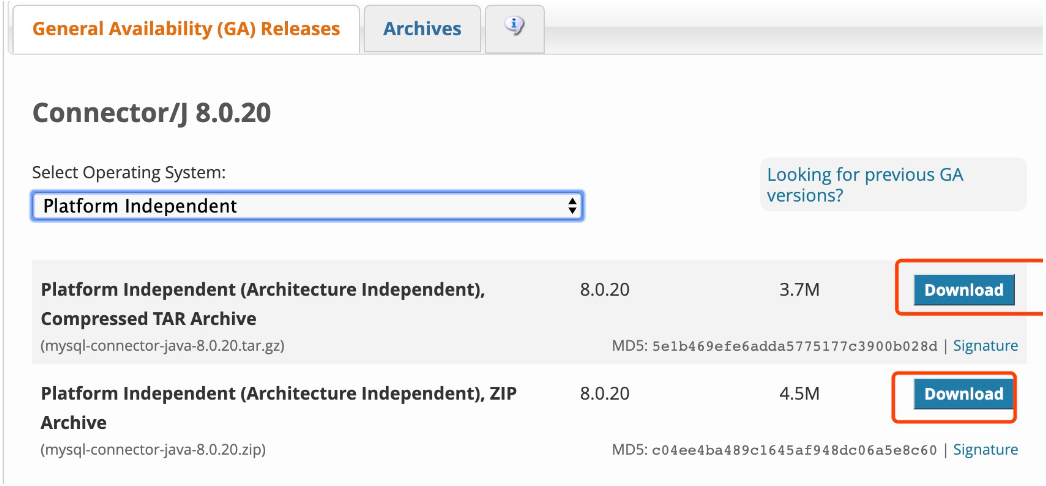
ret = SQLConnect(mysqlhdbc,(SQLCHAR\*)"ODBC数据源名称（也就是第3步（4）中的Data Source Name）",SQL\_NTS,(SQLCHAR\*)"数据库用户名",SQL\_NTS,(SQLCHAR\*)"数据库密码",SQL\_NTS);//mysqlhdbc为连接句柄

if(!SQL\_SUCCEEDED(ret))

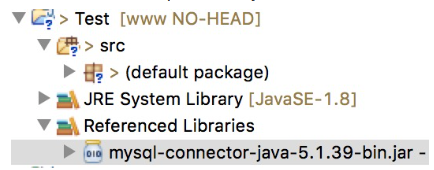
return -1;

2、基于JDBC的数据库应用开发实验：基于JDBC驱动的数据库连接方法，实现数据库数据操纵等应用开发常见功能。

Java 连接 MySQL 需要驱动包，最新版下载地址为：<http://dev.mysql.com/downloads/connector/j/>，解压后得到 jar 库文件，然后在对应的项目中导入该库文件。



本实例使用的是 Eclipse，导入 mysql-connector-java-5.1.39-bin.jar 包：



MySQL 8.0 以上版本的数据库连接有所不同：

1、MySQL 8.0 以上版本驱动包版本 [mysql-connector-java-8.0.16.jar](https://static.runoob.com/download/mysql-connector-java-8.0.16.jar)。

2、com.mysql.jdbc.Driver 更换为 com.mysql.cj.jdbc.Driver。

3、MySQL 8.0 以上版本不需要建立 SSL 连接的，需要显示关闭。

4、allowPublicKeyRetrieval=true 允许客户端从服务器获取公钥。

5、最后还需要设置 CST。

加载驱动与连接数据库方式如下：

Class.forName("com.mysql.cj.jdbc.Driver");

conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/test\_demo?useSSL=false&allowPublicKeyRetrieval=true&serverTimezone=UTC","root","password");

具体步骤：1、导入驱动jar包 2、注册驱动 3、获取数据库的连接对象 4、定义sql语句 5、获取执行sql语句的对象 6、执行sql并接收返回结果 7、处理结果 8、释放资源

实例：

在 MySQL 中创建 RUNOOB 数据库，并创建 websites 数据表，表结构如下：

CREATE TABLE `websites` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`name` char(20) NOT NULL DEFAULT '' COMMENT '站点名称',

`url` varchar(255) NOT NULL DEFAULT '',

`alexa` int(11) NOT NULL DEFAULT '0' COMMENT 'Alexa 排名',

`country` char(10) NOT NULL DEFAULT '' COMMENT '国家',

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=10 DEFAULT CHARSET=utf8;

插入一些数据：

INSERT INTO `websites` VALUES ('1', 'Google', 'https://www.google.cm/', '1', 'USA'), ('2', '淘宝', 'https://www.taobao.com/', '13', 'CN'), ('3', '菜鸟教程', 'http://www.runoob.com', '5892', ''), ('4', '微博', 'http://weibo.com/', '20', 'CN'), ('5', 'Facebook', 'https://www.facebook.com/', '3', 'USA');

以下实例使用了 JDBC 连接 MySQL 数据库，注意一些数据如用户名，密码需要根据你的开发环境来配置：

package com.runoob.test;

import java.sql.\*;

public class MySQLDemo {

// MySQL 8.0 以下版本 - JDBC 驱动名及数据库 URL

static final String JDBC\_DRIVER = "com.mysql.jdbc.Driver";

static final String DB\_URL = "jdbc:mysql://localhost:3306/RUNOOB";

// MySQL 8.0 以上版本 - JDBC 驱动名及数据库 URL

//static final String JDBC\_DRIVER = "com.mysql.cj.jdbc.Driver";

//static final String DB\_URL = "jdbc:mysql://localhost:3306/RUNOOB?useSSL=false&allowPublicKeyRetrieval=true&serverTimezone=UTC";

// 数据库的用户名与密码，需要根据自己的设置

static final String USER = "root";

static final String PASS = "123456";

public static void main(String[] args) {

Connection conn = null;

Statement stmt = null;

try{

// 注册 JDBC 驱动

Class.forName(JDBC\_DRIVER);

// 打开链接

System.out.println("连接数据库...");

conn = DriverManager.getConnection(DB\_URL,USER,PASS);

// 执行查询

System.out.println(" 实例化Statement对象...");

stmt = conn.createStatement();

String sql;

sql = "SELECT id, name, url FROM websites";

ResultSet rs = stmt.executeQuery(sql);

// 展开结果集数据库

while(rs.next()){

// 通过字段检索

int id = rs.getInt("id");

String name = rs.getString("name");

String url = rs.getString("url");

// 输出数据

System.out.print("ID: " + id);

System.out.print(", 站点名称: " + name);

System.out.print(", 站点 URL: " + url);

System.out.print("\n");

}

// 完成后关闭

rs.close();

stmt.close();

conn.close();

}catch(SQLException se){

// 处理 JDBC 错误

se.printStackTrace();

}catch(Exception e){

// 处理 Class.forName 错误

e.printStackTrace();

}finally{

// 关闭资源

try{

if(stmt!=null) stmt.close();

}catch(SQLException se2){

}// 什么都不做

try{

if(conn!=null) conn.close();

}catch(SQLException se){

se.printStackTrace();

}

}

System.out.println("Goodbye!");

}

}

**三、思考题**

1.请调查目前比较流行的软件开发环境在基于ODBC驱动开发数据库应用方面各有什么优缺点？

2.请调查目前比较流行的软件开发环境在基于JDBC驱动开发数据库应用方面各有什么优缺点？

### 实验20 综合实验

**实验学时：4学时**

**实验类型：综合**

**实验要求：必做（课外完成）**

**一、实验目的**

1、熟悉数据库应用系统需求分析；

2、熟悉数据库应用开发的相关技术；

3、了解简单的数据库应用开发流程和步骤；

**二、实验要求**

根据实验十八所设计的数据库，利用实验学习到的数据库应用开发技术，基于ODBC或基于JDBC的数据库应用开发方法，设计并实现网上书店管理系统。

设计一个小型的数据库应用系统，服务于一个网上书店的日常处理业务及网上购书业务，包括供书目录、库存管理、采购管理、客户管理，顾客订单管理、发货管理，供应商管理及网上浏览等。

总体设计要求：书店内部日常处理业务通过C/S模式来实现，而顾客可通过Internet进行网上订书及查询浏览，为改善性能，应尽量将业务规则放在中间层或数据库服务器一端。

该系统的主要功能如下：（具体功能要求见实验十八数据库设计实验—四的相关资料)

1、供书目录及库存管理

2、采购管理

3、客户管理

4、顾客订单管理和发货管理

5、供应商管理

6、网上浏览管理

要求：必须根据所有必选的功能和数据需求，根据实验实验十八所设计的数据库，设计实现数据库应用系统，基本实现相关功能，界面美观，操作简便。

## 七、实验报告模板

**《数据库系统原理》实验报告**

**实验题目： 数据库的创建和管理 表的创建与管理**

**姓名：肖弘基** **实验日期： 2025 年 11 月 6 日**

**实验内容及完成情况：**

**实验总结：**

**教师评语及成绩**：