



信息工程系

教

案

课程名称： 数据库系统

教 师： 邱煜佳

总 学 时： 54

理论学时： 18

实训学时： 36

上课班级： 物联网应用技术 241

授课学期： 2025-2026 第二学期

课题名称	第 1 章 数据库入门	计划课时	4 课时
教学引入	数据库技术是一种计算机辅助管理数据的方法，是计算机数据处理与信息管理的核心技术。数据库技术产生于 20 世纪 60 年代末，用于解决数据的组织和存储，以及如何高效地获取和处理数据。本章将围绕数据库的入门知识进行详细讲解。		
教学目标	<ul style="list-style-type: none"> ● 使学生了解数据管理技术的发展，能够说出数据管理技术每个阶段的特点。 ● 使学生了解数据库技术的基本术语，能够说出数据库、数据库管理系统和数据库系统的概念。 ● 使学生了解数据模型，能够说出数据模型的分类和常见术语的含义、E-R 图的表示方式、关系模型的基本概念和完整性约束。 ● 使学生了解常见的数据库产品，能够说出 3 种以上关系型数据库。 ● 使学生熟悉 SQL 的作用，能够根据 SQL 的功能说出 SQL 的 4 个类别。 ● 使学生掌握 MySQL 的安装与配置，能够独立安装 MySQL，并使用账号登录 MySQL。 ● 使学生了解 MySQL 常见的图形化管理工具，能够使用 SQLyog 和 Navicat 连接 MySQL。 		
教学重点	<ul style="list-style-type: none"> ● 安装 MySQL ● 管理 MySQL 服务 ● 登录 MySQL 与密码设置 		
教学难点	<ul style="list-style-type: none"> ● 数据库技术的基本术语 ● 初识数据模型 		
教学方式	课堂教学以 PPT 讲授为主，并结合多媒体进行教学		
课程思政	培养学生的严谨求实的科学精神、团队合作精神；培养学生社会责任感、爱国情怀等。		
教学过程	<p style="text-align: center;">第一课时</p> <p style="text-align: center;">（数据管理技术的发展、数据库技术的基本术语、初识数据模型）</p> <p>一、创设情景，导入新课</p> <p>数据在现实生活中占据着重要的地位，例如，数字、文字、图形、声音等资源都是数据，为了更好的让这些数据更好的被我们使用，一般都需要将数据进行保存。数据的管理技术经历了好几个阶段，现如今，通常使用数据库技术管理数据，为了后续能更好理解和使用数据库技术，本节课将对数据管理技术的发展、数据库技术的基本术语、初识数据模型进行讲解。</p> <p>二、新课讲解</p> <p>知识点 1-数据管理技术的发展</p> <p>(1) 人工管理阶段</p> <p>(2) 文件系统阶段</p> <p>(3) 数据库系统阶段</p> <p>知识点 2-数据库技术的基本术语</p> <p>教师通过 PPT 的方式讲解数据库技术的基本术语。</p> <p>(1) 数据库 (Database, DB)</p>		

(2) 数据库管理系统 (Database Management System, DBMS)

(3) 数据库系统 (Database System, DBS)

- 硬件
- 数据库
- 软件
- 用户

知识点 3-初识数据模型

教师通过 PPT 的方式讲解初识数据模型。

(1) 数据模型的组成要素

- 数据结构
- 数据操作
- 数据约束

(2) 常见的数据模型分类

- 层次模型 (Hierarchical Model)
- 网状模型 (Network Model)
- 关系模型 (Relational Model)
- 面向对象模型 (Object Oriented Model)

(3) 客观对象转换为计算机存储数据

(4) 概念数据模型常用术语

- 实体 (Entity)
- 属性 (Attribute)
- 联系 (Relationship)
- 实体型 (Entity Type)
- 实体集 (Entity Set)

三、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

四、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

第二课时

(E-R 图、关系模型、常见的数据库产品、SQL 简介)

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

E-R 图也称为实体-联系图 (Entity Relationship Diagram)，是一种用图形表示的实体联系模型。E-R 图提供了表示实体、属性和联系的方法，用来描述现实世界的概念模型。接下来针对 E-R 图进行讲解。

三、新课讲解

知识点 1-E-R 图

教师通过 PPT 的方式讲解 E-R 图。

(1) 实体：用矩形表示，将实体名写在矩形框内。

(2) 属性：用椭圆框表示，将属性名写在椭圆框内。实体与属性之间用实线连接。

(3) 联系：用菱形框表示，将联系名写在菱形框内，用连线将相关的实体连接，并在连线旁标注联系的类型，联系的类型分为一对一(1:1)、一对多(1:n)、多对多(m:n)。

知识点 2-关系模型

教师通过 PPT 的方式讲解关系模型。

(1) 关系模型的基本概念

- 关系(Relation)
- 属性(Attribute)
- 元组(Tuple)
- 域(Domain)
- 关系模式(Relation Schema)
- 键(Key)

(2) 关系模型的完整性约束

- 域完整性
- 实体完整性
- 参照完整性
- 用户自定义完整性

知识点 3-常见的数据库产品

教师通过 PPT 的方式讲解常见的数据库产品。

- (1) Oracle
- (2) Microsoft SQL Server
- (3) IBM Db2
- (4) MySQL

知识点 4-SQL 简介

教师通过 PPT 的方式讲解 SQL 简介。

- (1) 数据定义语言 (Data Definition Lanaguage, DDL)
- (2) 数据操纵语言 (Data Manipulation Lanaguage, DML)
- (3) 数据查询语言 (Data Query Language, DQL)
- (4) 数据控制语言 (Data Control Lanaguage, DCL)

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台 (<http://tch.ityxb.com>) 布置本节课作业以及下节课的预习作业。

第三、四课时 (上机练习)

上机一：(考察知识点为获取 MySQL、安装 MySQL、配置 MySQL、管理 MySQL 服务、登录 MySQL 与密码设置、SQLyog、Navicat)

	<p>形式：单独完成</p> <p>题目： 安装和配置 MySQL，要求如下。</p> <ol style="list-style-type: none"> (1) 获取 MySQL 的安装包 mysql-8.0.23-winx64.zip。 (2) 解压 mysql-8.0.23-winx64.zip，安装 MySQL。 (3) 创建 MySQL 配置文件,初始化数据库。 (4) 启动 MySQL 服务。 (5) 登录 MySQL。 (6) 设置 MySQL 的新密码。 (7) 使用设置的新密码登录 MySQL。 (8) 使用 SQLyog 或者 Navicat 登录 MySQL。 <p>上机二：（考察知识点为上机实践：图书管理系统的 E-R 图设计）</p> <p>形式：单独完成</p> <p>题目： 图书管理系统的系统实体信息和系统实体之间的联系设计，设计出图书管理系统的 E-R 图，系统实体信息和系统实体之间的联系具体如下：</p> <p>1. 系统的实体信息</p> <ol style="list-style-type: none"> (1) 图书：编号、名称、价格、上架时间、借阅人编号、借阅时间、状态。 (2) 用户：编号、名称、状态。 (3) 借阅记录：编号、图书编号、借阅人编号、借阅时间、归还时间。 <p>2. 系统实体之间的联系</p> <ol style="list-style-type: none"> (1) 一本尚未被借阅的图书只可以借阅给一个用户，每个用户可以同时借阅多本尚未被借阅的图书。 (2) 一条借阅记录只记录一本图书的借阅信息，一本图书可以有 multiple 借阅记录。 (3) 一个用户可以有多条借阅记录，一条借阅记录只能记录一个用户的借阅情况。
教学后记	

课题名称	第 2 章 数据库和表的基本操作	计划课时	5 课时
教学引入	<p>日常生活中,如果想要将数据存放在 Excel 文档中,就需要先创建一个 Excel 文档,并将数据写入到 Excel 文档的工作表 (Sheet) 中;如果想要将不同类型的数据分别存放,例如,员工的信息和部门的信息想要分开存放,可以创建多个工作表,每个工作表中存放相同类型的数据。在 MySQL 中,数据库就相当于 Excel 文档,数据表就相当于 Excel 中的工作表,MySQL 安装好之后就可以对数据库和数据表进行操作,实现对数据的管理。本章将对数据库和数据表的基本操作进行讲解。</p>		
教学目标	<ul style="list-style-type: none"> ● 使学生掌握数据库的基本操作,能够对数据库进行增删改查操作。 ● 使学生掌握数据表的基本操作,能够对数据表进行增删改查操作。 ● 使学生熟悉数据类型,能够说出 SQL 语句中不同类型数据的表示方式。 ● 使学生掌握表的约束,能够在数据表中设置非空约束、唯一约束、主键约束和默认值约束。 ● 使学生掌握字段自动增长的设置,能够在创建数据表时为字段设置自动增长。 		
教学重点	<ul style="list-style-type: none"> ● 创建数据库 ● 创建数据表 ● 修改数据表 ● 非空约束 ● 唯一约束 ● 主键约束 		
教学难点	<ul style="list-style-type: none"> ● 日期和时间类型 ● 主键约束 		
教学方式	课堂教学以 PPT 讲授为主,并结合多媒体进行教学		
课程思政	培养学生的严谨求实的科学精神、团队合作精神;培养学生社会责任感、爱国情怀等。		
教学过程	<p style="text-align: center;">第一课时</p> <p style="text-align: center;">(创建数据库、查看数据库、选择数据库、修改数据库特征、删除数据库)</p> <p>一、复习巩固</p> <p>教师通过上节课作业的完成情况,对学生吸收不好的知识点进行再次巩固讲解。</p> <p>二、通过直接导入的方式导入新课</p> <p>MySQL 安装好之后就可以进行数据库的相关操作了。数据库类似小区里的车库,可以用来存放汽车,每个车库都有唯一的车库号。在 MySQL 中可以创建多个不同名称的数据库用于存储数据,本节课将针对数据库的基本操作进行详细讲解。</p> <p>三、新课讲解</p> <p>知识点 1-创建数据库</p> <p>教师通过 PPT 结合实际操作的方式讲解创建数据库。</p>		

(1) 创建数据库的基本语法格式如下所示。

```
CREATE {DATABASE | SCHEMA} [IF NOT EXISTS] db_name [create_option];
```

(2) 创建数据库的语法格式的说明具体如下。

- CREATE {DATABASE | SCHEMA}: 创建指定名称的数据库。
- IF NOT EXISTS: 用于判断当前创建的数据库是否存在。
- db_name: 创建数据库的名称。
- create_option: 用于设置所创建的数据库的特征。

(3) 根据语法格式创建一个名称为 itcast, 字符集为 utf8mb4 的数据库。

知识点 2-查看数据库

教师通过 PPT 结合实际操作的方式讲解查看数据库。

(1) 查看语句的语法格式如下所示。

```
SHOW {DATABASES | SCHEMAS} [LIKE 'pattern' | WHERE expr];
```

(2) 查看数据库的语法格式说明如下。

- SHOW {DATABASES | SCHEMAS}: 查看创建好的数据库。
- LIKE 'pattern': 可选项, 表示 LIKE 子句, 可以根据指定匹配模式匹配数据库。
- WHERE expr: 可选项, 表示 WHERE 子句, 用于匹配指定条件的数据库。

(3) 编写 SQL 演示查看数据库系统中已经创建的数据库。

知识点 3-选择数据库

教师通过 PPT 结合实际操作的方式讲解选择数据库。

(1) 在 MySQL 中, 可以使用 USE 语句选择某个数据库为后续操作的默认数据库。USE 语句具体语法**错误!未定义书签**。格式如下。

```
USE <数据库名>
```

(2) 使用 USE 语句演示选择数据库。

知识点 4-修改数据库特征

教师通过 PPT 结合实际操作的方式讲解修改数据库特征。

(1) 展示修改数据库特征的基本语法格式。

```
ALTER {DATABASE | SCHEMA} [db_name] alter_option;
```

(2) 编写 SQL 语句演示将数据库 itcast 的字符集修改为 gbk。

知识点 5-删除数据库

教师通过 PPT 结合实际操作的方式讲解删除数据库。

(1) 删除数据库的基本语法格式如下所示。

```
DROP {DATABASE | SCHEMA} [IF EXISTS] db_name;
```

(2) 编写 SQL 语句演示删除名称为 itcast 的数据库。

四、归纳总结

教师回顾本节课所讲的内容, 并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课

的预习作业。

第二课时

(数值类型、日期和时间类型、字符串类型)

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

使用 MySQL 数据库存储数据时，不同类型数据的存储格式也不相同。MySQL 数据库提供了多种数据类型，主要包括数值类型、日期和时间类型、字符串类型。本节将针对这些数据类型进行讲解。

三、新课讲解

知识点 1-数字类型

教师通过 PPT 结合实际操作的方式讲解数字类型。

(1) 整数类型

- TINYINT
- SMALLINT
- MEDIUMINT
- INT
- BIGINT

(2) 浮点数类型

- FLOAT
- DOUBLE

(3) 定点数类型

定义定点数类型的方式如下所示。

```
DECIMAL (M, D)
```

(4) BIT 类型

定义 BIT 类型的基本语法格式如下所示。

```
BIT (M)
```

知识点 2-日期和时间类型

教师通过 PPT 结合实际操作的方式讲解日期和时间类型。

(1) YEAR

YEAR 类型用于存储年份数据，在 MySQL 中，可以使用以下 3 种表示方式指定 YEAR 类型的值。

- 使用 4 位字符串或数字表示。
- 使用 1 位或 2 位字符串表示。
- 使用 1 位或 2 位数字表示。

(2) DATE

DATE 类型用于存储日期数据，在 MySQL 中，可以使用以下 4 种表示方式指定 DATE 类型的值。

- 以 'YYYY-MM-DD' 或者 'YYYYMMDD' 字符串方式表示。
- 以 'YY-MM-DD' 或者 'YYMMDD' 字符串方式表示。
- 以 YYMMDD 数字方式表示。

- 使用 CURRENT_DATE 或者 NOW()表示当前系统日期。

(3) TIME

TIME 类型用于存储时间数据，可以使用以下 3 种表示方式指定 TIME 类型的值，具体如下。

- 以'D hh:mm:ss'字符串方式表示。
- 以'hmmss'字符串方式或者 hmmss 数字方式表示。
- 使用 CURRENT_TIME 或 NOW()输入当前系统时间。

(4) DATETIME

DATETIME 类型用于存储日期和时间的数据，在 MySQL 中，可以使用以下 4 种表示方式指定 DATETIME 类型的值。

- 以'YYYY-MM-DD hh:mm:ss'或者'YYYYMMDDhhmmss'字符串方式表示的日期和时间。
- 以 hh:mm:ss'或者'YMMDDhhmmss'字符串方式表示的日期和时间。
- 以 YYYYMMDDhhmmss 或者 YMMDDhhmmss 数字方式表示的日期和时间。
- 使用 CURRENT_TIMESTAMP 或 NOW()输入当前系统时间。

(5) TIMESTAMP

TIMESTAMP 类型用于表示日期和时间，TIMESTAMP 类型与 DATETIME 类型的区别，具体如下。

- TIMESTAMP 类型的取值范围比 DATETIME 类型小。
- TIMESTAMP 类型的值和时区相关。

知识点 3-字符串类型

教师通过 PPT 结合实际操作的方式讲解字符串类型。

(1) CHAR 和 VARCHAR 类型

定义 CHAR 和 VARCHAR 类型的方式如下所示。

CHAR (M)

或

VARCHAR (M)

(2) BINARY 和 VARBINARY 类型

定义 BINARY 和 VARBINARY 类型的方式如下所示。

BINARY (M)

或

VARBINARY (M)

(3) TEXT 类型

- TINYTEXT
- TEXT
- MEDIUMTEXT
- LONGTEXT

(4) BLOB 类型

- TINYBLOB
- BLOB
- MEDIUMBLOB
- LONGBLOB

(5) ENUM 类型

ENUM 类型又称为枚举类型，定义 ENUM 类型的数据格式如下所示。

```
ENUM('value1', 'value2', ...)
```

(6) SET 类型

SET 类型的字段通常用于存储字符串对象，SET 类型数据的定义格式如下所示。

```
SET('value1', 'value2', ...)
```

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台 (<http://tch.ityxb.com>) 布置本节课作业以及下节课的预习作业。

第三课时

(创建数据表、查看数据表、修改数据表、删除数据表)

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

在 MySQL 中，所有的数据都存储在数据表中，因此在学习数据表中的数据操作之前，有必要先了解数据库表的基本操作。本节将介绍数据表的基本操作，数据表的基本操作主要包括创建数据表、查看数据表结构、修改数据表结构和删除数据表。

三、新课讲解

知识点 1-创建数据表

教师通过 PPT 结合实际操作的方式讲解创建数据表。

(1) 在 MySQL 中使用 CREATE TABLE 语句创建数据表，其基本语法规则如下所示。

```
CREATE [TEMPORARY] TABLE [IF NOT EXISTS] tbl_name
    (字段名 1 数据类型 1 [列级约束 1]
    [字段名 2 数据类型 2 [列级约束 2]][, ...]
    [, 表级约束(字段名 3[, 字段名 4][, ...])][, ...])
    [table_options] [partition_options];
```

(2) CREATE TABLE 语句语法规则的说明具体如下。

- TEMPORARY: 表示临时表。
- IF NOT EXISTS: 只有在创建的数据表尚不存在时，才会创建数据表。
- tbl_name: 创建的数据表的名称。
- 字段名: 数据表字段的名称。
- 数据类型: 字段中保存数据的数据类型，如日期类型等。
- 约束: 用于保证数据的完整性和有效性的规则。
- table_options: 表示表选项，用于设置数据表的相关选项。
- partition_options: 表示分区选项，用于设置数据表分区的内容。

(3) 使用 SQL 演示数据库表的创建。

知识点 2-查看数据表

教师通过 PPT 结合实际操作的方式讲解查看数据表。

(1) 使用 SHOW TABLES 语句查看数据表

① 通过 SHOW TABLES 语句查看当前数据库中的数据表，基本语法格式如下。

```
SHOW TABLES [LIKE 'pattern' | WHERE expr];
```

② 演示使用 SHOW TABLES 语句查看当前数据库中所有的数据表。

(2) 使用 SHOW CREATE TABLE 查看数据表创建语句

① SHOW CREATE TABLE 语句的基本语法格式如下所示。

```
SHOW CREATE TABLE tbl_name;
```

② 演示使用 SHOW CREATE TABLE 语句查看数据 tb_dept 的创建语句。

(3) 使用 DESCRIBE 语句查看数据表结构信息

① DESCRIBE 语句的基本语法格式如下所示。

```
DESCRIBE 数据表名;
```

上述语法格式可以简写为如下形式。

```
DESC 数据表名;
```

② 演示使用 DESC 语句查看数据表 tb_dept 的表结构信息。

知识点 3-修改数据表

教师通过 PPT 结合实际操作的方式讲解修改数据表。

(1) 在 MySQL 中，可以通过 ALTER TABLE 语句实现数据表结构的修改。

ALTER TABLE 语句的基本语法格式如下所示。

```
ALTER TABLE tbl_name
    [alter_option [, alter_option] ...]
    [partition_options]
```

(2) 展示 alter_option 选项

```
{
    RENAME [TO | AS] 新数据表名
  | RENAME COLUMN 旧字段名 TO 新字段名
  | MODIFY [COLUMN] 字段名 新数据类型[列级约束] [FIRST | AFTER 字段名]
  | CHANGE [COLUMN] 旧字段名 新字段名 新数据类型[列级约束] [FIRST | AFTER 字段名]
  | ADD [COLUMN] 字段名 新字段名 数据类型[列级约束] [FIRST | AFTER 字段名]
  | ADD [CONSTRAINT] {PRIMARY KEY | UNIQUE} (字段名 1[, ...])
  | DROP [COLUMN] 字段名 | {INDEX | KEY}索引名 | PRIMARY KEY
  | table_options
  | ...
}
```

(3) 修改数据表名

(4) 修改数据表选项

(5) 修改字段名

(6) 修改字段的数据类型

(7) 修改字段的排列位置

(8) 添加字段

(9) 删除字段

知识点 4-删除数据表

教师通过 PPT 结合实际操作的方式讲解删除数据表。

(1) 在 MySQL 中, 可以使用 DROP TABLE 语句删除数据表, 其基本的语法格式如下所示。

```
DROP [TEMPORARY] TABLE [IF EXISTS]
tbl_name [, tbl_name] ...
[RESTRICT | CASCADE];
```

(2) DROP TABLE 语句的语法格式的具体说明如下。

- TEMPORARY: 可选项, 表示临时表。
- IF EXISTS: 可选项, 表示在删除之前判断数据表是否存在。
- tbl_name: 表示数据表名称。
- RESTRICT | CASCADE: 可选项, 用于设置外键的级联级别。

(3) 演示使用 SQL 语句删除数据表。

四、归纳总结

教师回顾本节课所讲的内容, 并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

第四课时

(非空约束、唯一约束、主键约束、默认值约束、自动增长)

一、复习巩固

教师通过上节课作业的完成情况, 对学生吸收不好的知识点进行再次巩固讲解。

二、通过情景导入的方式导入新课

为了防止数据表中插入错误的数据库, MySQL 定义了一些规则维护数据库中数据的完整性和有效性, 这些规则即表的约束。常见的约束有非空约束、唯一约束、主键约束、外键约束和默认值约束, 接下来针对外键约束之外的其他约束进行讲解。**错误!未定义书签。**

三、新课讲解

知识点 1-非空约束

教师通过 PPT 结合实际操作的方式讲解非空约束。

(1) 在 MySQL 中, 非空约束通过 NOT NULL 进行限定。

(2) 设置非空约束

① 创建数据表时设置非空约束

创建数据表时给字段设置非空约束, 只需在字段的数据类型后面追加 NOT NULL 即可。

② 修改数据表时添加非空约束

如果需要在已经存在的数据表中添加非空约束, 可以在 ALTER TABLE 语句中通过 MODIFY 或 CHANGE 重新定义字段的方式添加非空约束。

(3) 删除非空约束

可以通过 ALTER TABLE 语句中的 MODIFY 或 CHANGE 重新定义字段的方式实

现。

知识点 2-唯一约束

教师通过 PPT 结合实际操作的方式讲解唯一约束。

(1) 在 MySQL 中，唯一约束通过关键字 UNIQUE 进行设置。

(2) 设置唯一约束

① 创建数据表时设置唯一约束

- 设置列级约束：列级约束紧跟在字段的数据类型之后，只对该字段起约束作用。

- 设置表级约束：表级约束独立于字段，可以对数据表的单个或多个字段起约束作用

② 修改数据表时添加唯一约束

- 在 ALTER TABLE 语句中通过 MODIFY 或 CHANGE 重新定义字段的方式添加唯一约束。

- 通过 ALTER TABLE 语句中的 ADD 添加唯一约束。

(3) 删除唯一约束

如果想要删除字段中已有的唯一约束，可以通过 ALTER TABLE 语句中的“DROP 索引名”实现。

知识点 3-主键约束

教师通过 PPT 结合实际操作的方式讲解主键约束。

(1) 在 MySQL 中，主键约束是通过 PRIMARY KEY 进行设置。

(2) 设置主键约束

- 创建数据表时设置主键约束

- 修改数据表时添加主键约束

(3) 删除主键约束

对于设置错误或者不再需要的主键约束，可以通过 ALTER TABLE 语句的 DROP 将主键约束删除。

知识点 4-默认值约束

教师通过 PPT 结合实际操作的方式讲解默认值约束。

(1) 设置默认值约束

① 创建数据表时设置默认值约束

创建数据表时，给字段设置默认值约束，只需在定义字段时使用如下格式即可。

```
<字段名> <数据类型> DEFAULT <默认值>;
```

② 修改数据表时添加默认值约束

修改数据表时添加默认值约束和修改数据表时添加非空约束类似，可以在 ALTER TABLE 语句中通过 MODIFY 或 CHANGE 重新定义字段的方式添加默认值约束。

(2) 删除默认值约束

删除默认约束可以通过 ALTER TABLE 语句中 MODIFY 或 CHANGE 重新定义字段的方式实现。

知识点 5-自动增长

教师通过 PPT 结合实际操作的方式讲解字段值的自动增长。

- (1) 使用 AUTO_INCREMENT 为新生行自动生成唯一标识。
- (2) 在字段中设置 AUTO_INCREMENT 的基本语法格式如下所示。

字段名 数据类型 AUTO_INCREMENT;

- (3) 使用 AUTO_INCREMENT 时的注意事项。

- 一个数据表中只能有一个字段设置 AUTO_INCREMENT，设置 AUTO_INCREMENT 字段的数据类型可以是任何整数类型，并且该字段必须定义为键，如 UNIQUE、PRIMARY KEY。
- 如果为自动增长字段插入 NULL，则该字段会自动增长值；如果插入的是一个具体的值，则不会自动增长值。
- 默认情况下，设置 AUTO_INCREMENT 的字段的价值是从 1 开始自增。如果插入了一个大于自动增长的具体值，则下次自动增长的价值为字段中最大值加 1。
- 使用 DELETE 删除记录时，自动增长值不会减少或者填补空缺。

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

第五课时（上机练习）

上机：（考察知识点为上机实践：图书管理系统的数据库及相关数据表的创建）

形式：独立完成

题目：创建图书管理系统的数据库及相关数据表

要求：

1. 创建字符集为 utf8mb4 的数据库 bms。
2. 在数据库 bms 中创建数据表 user、数据表 book 和数据表 record，数据表的结构如下。

user 表的结构

字段名称	数据类型	约束	默认值	是否自增	说明
id	INT	主键约束		是	用户编号
name	VARCHAR(20)	非空约束唯一约束		否	用户名称
state	CHAR(1)	非空约束	'0'	否	用户状态 (0 表示启用， 1 表示禁用)

book 表的结构

字段名称	数据类型	约束	默认	是否自增	说明
------	------	----	----	------	----

				值		
	id	INT	主键约束		是	图书编号
	name	VARCHAR(20)	非空约束 唯一约束		否	图书名称
	price	DECIMAL(6,2)	非空约束		否	图书价格
	upload_time	DATETIME	非空约束		否	上架时间
	borrower_id	INT			否	借阅人编号
	borrow_time	DATETIME			否	借阅时间
	state	CHAR(1)	非空约束	'0'	否	图书状态 (0表示可借阅, 1表示已借阅, 2表示已下架)
record 表的结构						
	字段名称	数据类型	约束	默认值	是否自增	说明
	id	INT	主键约束		是	借阅记录编号
	book_id	INT	非空约束		否	图书编号
	borrower_id	INT	非空约束		否	借阅人编号
	borrow_time	DATETIME	非空约束		否	借阅时间
	remand_time	DATETIME	非空约束		否	归还时间
	3. 查看数据表 book 的表结构信息。					
教学后记						

(4) 向部分字段插入数据。

知识点 2-一次插入多条数据

教师通过 PPT 结合实际操作的方式讲解一次插入多条数据。

(1) 使用一条 INSERT 语句同时添加多条数据，其语法格式如下所示。

```
INSERT [INTO] 数据表名 [(字段名 1, 字段名 2, ...)]  
{VALUES | VALUE} (值 1, 值 2, ...), (值 1, 值 2, ...),  
...  
(值 1, 值 2, ...);
```

(2) 演示使用一条 INSERT 语句插入多条数据。

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

第二课时

(更新数据、删除数据)

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

更新数据是指对数据表中已经存在的数据进行修改，比如某个部门名称变更了，就需要对数据表中部门名称字段的值进行修改。删除数据是指对数据表中已经存在的数据进行删除，例如员工离职后，需要将离职的员工从员工表中删除。本节将对更新数据、删除数据进行讲解。

三、新课讲解

知识点 1-更新数据

教师通过 PPT 结合实际操作的方式讲解更新数据。

(1) MySQL 中使用 UPDATE 语句更新表中的数据，其基本的语法格式如下所示。

```
UPDATE 数据表名 SET 字段名 1 = 值 1[, 字段名 2 = 值 2, ...] [WHERE 条件表达式];
```

(2) 更新部分数据

更新部分数据是指根据指定条件更新数据表中的某条或者某几条数据，需要使用 WHERE 子句来指定更新数据的条件。

(3) 更新全部数据

在 UPDATE 语句中如果没有使用 WHERE 子句，则会将表中所有数据的指定字段都进行更新。

知识点 2-删除数据

教师通过 PPT 结合实际操作的方式讲解删除数据。

(1) MySQL 中使用 DELETE 语句删除数据表中的数据，其语法格式如下所示。

```
DELETE FROM 数据表名 [WHERE 条件表达式]
```

(2) 删除部分数据

删除部分数据是指根据指定条件删除数据表中的某一条或者某几条数据，需要使用 WHERE 子句指定删除数据的条件。

(3) 删除全部数据

在 DELETE 语句中如果没有使用 WHERE 子句，则会将数据表中的所有数据都删除。

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台 (<http://tch.ityxb.com>) 布置本节课作业以及下节课的预习作业。

第三课时（上机练习）

上机：（上机实践：图书表的数据操作）

形式：独立完成

题目：操作图书表中的数据

要求：

- (1) 插入单条图书信息，插入单条图书的信息如下表所示。

图书的信息

name	price	state
Java 基础入门（第 3 版）	59.00	'0'

- (2) 插入单条用户信息，插入单条用户的信息如下表所示。

用户的信息

name	state
张三	'0'
李四	'0'

- (3) 同时插入多条图书信息，插入的多条图书信息如下表所示。

多条图书信息

name	price	state
三国演义	69.00	'2'
MySQL 数据库入门	40.00	'0'
Java Web 程序开发入门	49.00	'0'
西游记	59.00	'2'
红楼梦	33.00	'2'
水浒传	66.66	'2'

(4) 上架图书。图书上架其实就是将图书的状态修改为可借阅即可，需要上架的图书是《西游记》。

- (5) 修改单条图书信息，将《水浒传》的价格修改为 66.00。

	<p>(6) 修改多条图书信息，将所有的图书价格下调 10%。</p> <p>(7) 删除图书，删除图书《红楼梦》的信息。</p> <p>(8) 借阅图书。用户借阅图书时，需要在图书表中修改图书的借阅人信息、借阅时间信息、图书状态设置为已借阅。本次为编号为 1 的用户借阅图书《MySQL 数据库入门》。</p>
--	--

课题名称	第 4 章 单表查询	计划课时	4 课时
教学引入	对数据库中的数据进行操作，除了之前章节中讲解到的插入、更新和删除，还有一个使用频率更高、更重要的操作就是查询操作。查询是指从数据库中获得所需要的数据，使用不同的查询方式可以获取不同的数据。一般将只涉及到一张数据表的查询称为单表查询，本章将对单表查询进行讲解。		
教学目标	<ul style="list-style-type: none"> ● 使学生熟悉 Select 语句的作用，能够说出 Select 语句中各子句的含义。 ● 使学生掌握简单查询，能够使用 SELECT 语句查询所有字段、查询指定字段，以及查询去重数据。 ● 使学生掌握条件查询，能够使用比较运算符和逻辑运算符进行条件查询。 ● 使学生掌握高级查询，能够使用聚合函数、分组查询、排序查询和限量查询进行查询。 ● 使学生熟悉别名的设置，能够为数据表和字段设置别名。 		
教学重点	<ul style="list-style-type: none"> ● 查询指定字段 ● 带比较运算符的查询 ● 带逻辑运算符的查询 ● 聚合函数 ● 分组查询 ● 排序查询 ● 限量查询 		
教学难点	<ul style="list-style-type: none"> ● 聚合函数 ● 分组查询 		
教学方式	课堂教学以 PPT 讲授为主，并结合多媒体进行教学		
课程思政	培养学生的严谨求实的科学精神、团队合作精神；培养学生社会责任感、爱国情怀等。		
教学过程	<p style="text-align: center;">第一课时</p> <p style="text-align: center;">（SELECT 语句、查询所有字段、查询指定字段、查询去重数据、带比较运算符的查询、带逻辑运算符的查询）</p> <p>一、复习巩固 教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。</p> <p>二、通过直接导入的方式导入新课 在 MySQL 中，可以使用 SELECT 语句来查询数据。查询数据是指从数据库中根据需求，使用不同的查询方式来获取不同的数据，是使用频率最高、最重要的操作。接下来对 SELECT 语句进行讲解。</p> <p>三、新课讲解</p> <p>知识点 1-SELECT 语句 教师通过 PPT 结合实际操作的方式讲解 SELECT 语句。</p> <p>(1) 从数据表中查询数据的基本语句是 SELECT 语句，SELECT 语句的基本语法格式如下。</p>		

```
SELECT [DISTINCT] *|{select_expr1, select_expr2, ...}  
FROM 数据表名  
[WHERE 条件表达式 1]  
[GROUP BY 字段名 [HAVING 条件表达式 2]]  
[ORDER BY 字段名 [ASC|DESC]]s  
[LIMIT [OFFSET] 记录数]
```

(2) SELECT 语句中各子句的含义如下。

- SELECT [DISTINCT] *| {select_expr1, select_expr2,...}: 指定查询结果中需要返回的值。
- FROM 数据表名: 表示从指定名称的数据表中查询数据。
- WHERE 条件表达式 1: WHERE 子句, 是可选项, 用于指定查询条件。
- [GROUP BY 字段名 [HAVING 条件表达式 2]]: GROUP BY 子句, 是可选项, 用于将查询结果按照指定字段进行分组; HAVING 子句也是可选项, 用于对分组后的结果进行过滤。
- [ORDER BY 字段名 [ASC|DESC]]: ORDER BY 子句, 是可选项, 用于将查询结果按指定字段进行排序。
- [LIMIT [OFFSET] 记录数]: LIMIT 子句, 是可选项, 用于限制查询结果的数量。

知识点 2-查询所有字段

教师通过 PPT 结合实际操作的方式讲解查询所有字段。

(1) 列出所有字段名称进行查询

① 列出所有字段名称进行查询的基本语法格式如下所示。

```
SELECT 字段名 1, 字段名 2, ... FROM 数据表名;
```

② 通过案例, 演示列出所有字段名称进行查询。

(2) 使用通配符*进行查询

① 使用通配符*匹配数据表中所有字段, 其语法格式如下所示。

```
SELECT * FROM 数据表名;
```

② 通过案例, 演示使用通配符*进行查询。

知识点 3-查询指定字段

教师通过 PPT 结合实际操作的方式讲解查询指定字段。

(1) 查询指定字段可以在 SELECT 语句的字段列表中指定要查询的字段, 其语法格式如下所示。

```
SELECT 字段名 1, 字段名 2, ... FROM 数据表名;
```

(2) 通过案例, 演示使用 SELECT 语句查询指定字段。

知识点 4-查询去重数据

教师通过 PPT 结合实际操作的方式讲解查询去重数据。

(1) MySQL 中提供了 DISTINCT 关键字, 可以在查询时去除重复的值, 基本语法格式如下所示。

```
SELECT DISTINCT 字段名 FROM 数据表名;
```

(2) 通过一个案例, 演示在 SELECT 语句中使用 DISTINCT 去除重复值。

知识点 5-带比较运算符的查询

教师通过 PPT 结合实际操作的方式讲解带比较运算符的查询。

MySQL 常见的比较运算符

- =:比较运算符左右两侧的操作数是否相等。
- <>:比较运算符左右两侧的操作数是否不相等。
- !=:比较运算符左右两侧的操作数是否不相等。
- <:比较运算符左侧操作数是否小于右侧操作数。
- <=:比较运算符左侧操作数是否小于等于右侧操作数。
- >:比较运算符左侧操作数是否大于右侧操作数。
- >=:比较运算符左侧操作数是否大于等于右侧操作数。
- BETWEEN ... AND ...:比较数据是否存在于指定范围内。
- IN:比较数据是否存在于指定集合内。
- IS NULL:比较数据是否为 NULL。
- IS NOT NULL:比较数据是否不为 NULL。
- LIKE:通配符匹配, 获取匹配到的数据。

接下来通过一些例子学习 MySQL 常见比较运算符的使用, 具体如下。

(1) 带=运算符的查询

=运算符用于比较运算符左右两边的操作数, 如果操作数的字段类型为字符串, 需要使用单引号对操作数进行包裹。

(3) 带<>运算符的查询

<>运算符和!=运算符都用于比较操作数是否不相等。

(4) 带<运算符的查询

<运算符用于判断左侧操作数是否小于右侧的操作数。

(5) 带>=运算符的查询

>=运算符用于判断左侧操作数是否大于等于右侧操作数。

(6) 带 IN 运算符的查询

IN 运算符用于判断某个值是否在指定集合中, 如果值存在集合中, 则满足条件。IN 运算符的语法格式如下所示。

```
SELECT *|字段名 1, 字段名 2, ...  
FROM 数据表名  
WHERE 值 [NOT] IN (元素 1, 元素 2, ...);
```

(7) 带 IS NULL 运算符的查询

当操作数为 NULL 时, 不能使用运算符=、<>、!=进行比较, 这是因为 NULL 代表未指定或不可预知的值。如果需要判断数据是否为 NULL 的时候, 可以使用 IS NULL 进行比较。

(8) 带 LIKE 运算符的查询

① 使用 LIKE 关键字的 SELECT 语句的语法格式如下所示。

```
SELECT *| {字段名 1, 字段名 2, ...}  
FROM 数据表名  
WHERE 值 [NOT] LIKE 匹配的字符串;
```

② %通配符

%通配符是模糊查询最常用的通配符, 它可以匹配任意长度的字符串, 包括空字符串。

③ _通配符

_通配符用于匹配单个字符, 如果要匹配多个字符, 需要使用多个_通配符, 每个_通配符代表一个字符。

知识点 6-带逻辑运算符的查询

教师通过 PPT 结合实际操作的方式讲解带逻辑运算符的查询。

(1) MySQL 中常见的逻辑运算符

逻辑运算符	说明
NOT !	逻辑非，返回和操作数相反的结果
AND &&	逻辑与，操作数全部为真，则结果为 1，否则结果为 0
OR 	逻辑或，操作数只要有一个为真，则结果为 1，否则结果为 0

(2) 带 NOT 运算符的查询

运算符 NOT 和!都表示逻辑非，返回和操作数相反的结果。

(3) 带 AND 运算符的查询

在 MySQL 中，可以使用 AND 运算符连接两个或者多个查询条件，只有满足所有条件的记录才会被返回。带 AND 运算符的查询语法格式如下所示。

```
SELECT *|{字段名 1, 字段名 2, ...}  
FROM 表名  
WHERE 条件表达式 1 AND 条件表达式 2 ... AND 条件表达式 n;
```

(4) 带 OR 运算符的查询

在使用 SELECT 语句查询数据时，可以使用 OR 运算符连接多个查询条件。带 OR 运算符的查询语法格式如下所示。

```
SELECT *|{字段名 1, 字段名 2, ...}  
FROM 数据表名  
WHERE 条件表达式 1 OR 条件表达式 2 [... OR 条件表达式 n];
```

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台 (<http://tch.ityxb.com>) 布置本节课作业以及下节课的预习作业。

第二课时

(聚合函数、分组查询、排序查询)

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

实际开发中，经常需要做一些数据统计操作，例如统计某个字段的最大值、最小值、平均值等，像这样对一组值执行计算，并将计算后的值返回的操作称为聚合操作，聚合操作一般通过聚合函数实现。接下来对聚合函数进行讲解。

三、新课讲解

知识点 1-聚合函数

教师通过 PPT 结合实际操作的方式讲解聚合函数。

使用聚合函数实现查询的基本语法格式如下所示。

```
SELECT [字段名 1, 字段名 2, ..., 字段名 n] 聚合函数 FROM 数据表名;
```

接下来，通过一些例子学习聚合函数在数据统计中的使用。

(1) COUNT()函数

COUNT()函数用于检索数据表行中的值的计数，COUNT(*)可以统计数据表中记录的总条数，即数据表中有多少行记录。SUM()函数会对指定字段中的值进行累加，并且在数据累加时忽略字段中的NULL值。

(2) SUM()函数

如果字段中存放的是数值型数据，需要统计该字段中所有值的总数，可以使用SUM()函数。

(3) AVG()函数

如果字段中存放的是数值型数据，需要统计该字段中所有值的平均值，可以使用AVG()函数。AVG()函数会计算指定字段值的平均值，并且计算时会忽略字段中的NULL值。

(4) MAX()函数

MAX()函数用于计算指定字段中的最大值，如果字段的值是数值类型，则比较的是值的大小。

(5) MIN()函数

MIN()函数用于计算指定字段中的最小值，如果字段的值是数值类型，则比较的是值的大小。

知识点 2-分组查询

教师通过PPT结合实际操作的方式讲解分组查询。

(1) 在MySQL中，可以使用GROUP BY根据指定的字段对结果集进行分组，如果某些记录的指定字段具有相同的值，那么分组后被合并为一条数据。

(2) 使用GROUP BY分组查询的语法格式如下所示。

```
SELECT 字段名 1, 字段名 2, [表达式] ...
```

```
FROM 数据表名
```

```
GROUP BY 字段名 1, 字段名 2, ... [HAVING 条件表达式];
```

(3) 单独使用GROUP BY分组

单独使用GROUP BY进行分组，将根据指定的字段合并数据行。

(4) GROUP BY和聚合函数一起使用

如果分组查询时要进行统计汇总，此时需要将GROUP BY和聚合函数一起使用。

(5) GROUP BY和HAVING关键字一起使用

通常情况下GROUP BY和HAVING关键字一起使用，用于对分组后的结果进行条件过滤。

知识点 3-排序查询

教师通过PPT结合实际操作的方式讲解排序查询。

(1) 如果想要对查询结果按指定的方式排序，可以使用ORDER BY对查询结果进行排序。

(2) 查询语句中使用ORDER BY的基本语法格式如下所示。

```
SELECT * | { 字段名 1, 字段名 2, ... }
```

```
FROM 表名
```

```
ORDER BY 字段名 1 [ASC | DESC], 字段名 2 [ASC | DESC]…;
```

(3) 使用 ORDER BY 对查询结果进行排序时，如果不指定排序方式，默认按照 ASC 方式进行排序。通过案例进行演示。

(4) 按照指定字段进行排序时，如果字段的值中包含 NULL，NULL 会被当做最小值进行排序。通过案例进行演示。

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台（<http://tch.ityxb.com>）布置本节课作业以及下节课的预习作业。

第三课时

（限量查询、内置函数、为数据表设置别名、为字段设置别名）

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

查询数据时，SELECT 语句可能会返回很多条记录，而用户需要的记录可能只是其中的一条或者几条，例如，在员工管理系统中，希望每一页默认展示 10 条员工信息，并可以通过下拉框更改每页展示的员工信息数，此时就可以通过限量查询完成，接下来针对限量查询进行详细讲解。

三、新课讲解

知识点 1-限量查询

教师通过 PPT 结合实际操作的方式讲解限量查询。

(1) MySQL 中提供了一个关键字 LIMIT，可以指定查询结果从哪一条记录开始以及一共查询多少条信息。

(2) 在 SELECT 语句中使用 LIMIT 的基本语法格式如下所示。

```
SELECT 字段名 1, 字段名 2, ...  
FROM 数据表名  
LIMIT [OFFSET, ] 记录数;
```

(3) 通过案例演示使用 LIMIT 进行限量查询。

知识点 2-内置函数

教师通过 PPT 结合实际操作的方式讲解内置函数。

(1) MySQL 中的内置函数从功能方面划分，可分为如下几类。

- 数学函数
- 字符串函数
- 日期和时间函数
- 条件判断函数
- 加密函数

(2) CONCAT()函数

执行 CONCAT()函数会返回函数参数连接之后的字符串。

(3) IF()函数

IF()函数有 3 个参数，具体格式为 IF(expr,v1,v2)，如果表达式 expr 成立，返回结果 v1；否则，返回结果 v2。

知识点 3-为数据表设置别名

教师通过 PPT 结合实际操作的方式讲解为数据表设置别名。

(1) MySQL 中为数据表起别名的基本语法格式如下所示。

```
SELECT * FROM 数据表名 [AS] 别名;
```

(2) 通过一个案例，演示在 SELECT 语句中为数据表设置别名。

知识点 4-为字段设置别名

教师通过 PPT 结合实际操作的方式讲解为字段设置别名。

(1) SELECT 语句中为字段起别名的基本语法格式如下所示。

```
SELECT 字段名 [AS] 别名 [, 字段名 [AS] 别名, .....] FROM 数据表名;
```

(2) 通过一个案例，演示在 SELECT 语句中为字段设置别名。

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台 (<http://tch.ityxb.com>) 布置本节课作业以及下节课的预习作业。

第四课时（上机练习）

上机：（考察知识点为上机实践：图书管理系统的单表查询）

形式：独立完成

题目：查询图书管理系统中单张数据表中的数据

要求：

(1) 查询可借阅图书清单。查询只显示图书名称和上架时间的可借阅图书的清单。

(2) 根据图书名称排序查询。根据图书名称升序排序的前 5 条图书的信息，查询出的结果只需显示图书的名称、价格和状态。

(3) 根据图书价格查询。查询价格大于 50 的图书的名称。

(4) 根据图书区间价格查询。查询价格大于等于 30 并且小于等于 50 的图书的名称和价格。

(5) 根据借阅状态查询。查询已借阅图书的图书名称、借阅人编号和借阅时间。

(6) 根据书名包含的关键字查询。查询包含“Java”的所有图书名称。

(7) 根据书名结尾关键字查询。查询以“入门”结尾的所有图书名称。

(8) 根据多条件查询。查询“西游记”或者“红楼梦”的图书信息，只需要显示图书名称和价格。

课题名称	第 5 章 多表操作	计划课时	4 课时
教学引入	<p>之前章节对数据的操作都是基于一张数据表完成的，即单表操作，然而实际应用中业务逻辑较为复杂，表与表之间可能存在业务联系，有时候需要基于两张或者两张以上的数据表进行操作，即多表操作。本章将针对多表操作的相关知识进行讲解。</p>		
教学目标	<ul style="list-style-type: none"> ● 使学生掌握多表查询，能够使用交叉连接、内连接、外连接及复合条件连接进行多表查询。 ● 使学生掌握子查询，能够使用子查询结合 IN、EXISTS、ANY、ALL 及比较运算符进行查询。 ● 使学生掌握外键约束的使用，能够为表添加外键约束和删除外键约束。 ● 使学生掌握关联表的三种关联关系，能够向关联表中添加和删除数据。 		
教学重点	<ul style="list-style-type: none"> ● 内连接查询 ● 外连接查询 ● 复合条件连接查询 ● IN 关键字结合子查询 ● ANY 关键字结合子查询 ● ALL 关键字结合子查询 		
教学难点	<ul style="list-style-type: none"> ● 外连接查询 ● EXISTS 关键字结合子查询 		
教学方式	课堂教学以 PPT 讲授为主，并结合多媒体进行教学		
课程思政	培养学生的严谨求实的科学精神、团队合作精神；培养学生社会责任感、爱国情怀等。		
教学过程	<p style="text-align: center;">第一课时</p> <p style="text-align: center;">（交叉连接查询、内连接查询、外连接查询、复合条件连接查询）</p> <p>一、复习巩固</p> <p>教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。</p> <p>二、通过直接导入的方式导入新课</p> <p>在关系型数据库中，一张数据表通常存储一个实体的信息。当两张或多张数据表中存在相同意义的字段时，如果需要同时显示多张数据表中的数据，便可以通过这些意义相同的字段将不同的数据表进行连接，并对连接后的数据表进行查询，这样的查询通常称为连接查询。在 MySQL 中，连接查询包括交叉连接查询、内连接查询、外连接查询、复合条件连接查询，本节课将对这些连接查询进行讲解。</p> <p>三、新课讲解</p> <p>知识点 1-交叉连接查询</p> <p>教师通过 PPT 结合实际操作的方式讲解交叉连接查询。</p> <p>(1) 交叉连接（CROSS JOIN）查询返回的结果是被连接的两张数据表中所有数据行的笛卡尔积。</p>		

(2) 交叉连接的语法格式如下所示。

```
SELECT <字段名> FROM <数据表名 1> CROSS JOIN <数据表名 2> ;
```

或

```
SELECT <字段名> FROM <数据表名 1>, <数据表名 2> ;
```

(3) 通过案例演示交叉连接查询。

知识点 2-内连接查询

教师通过 PPT 结合实际操作的方式讲解内连接查询。

(1) 内连接 (INNER JOIN) 查询又称简单连接查询或自然连接查询, 是常见的连接查询。内连接根据连接条件, 可以对交叉连接查询的部分结果进行筛选, 仅筛选出两张表中相互匹配的记录。

(2) 内连接查询的语法格式如下所示。

```
SELECT 查询字段 FROM 数据表 1 [INNER] JOIN 数据表 2 ON 匹配条件;
```

(3) 通过案例演示内连接查询。

知识点 3-外连接查询

教师通过 PPT 结合实际操作的方式讲解外连接查询。

(1) 除了要查询出符合条件的数据外, 还需要查询出其中一张数据表中符合条件之外的其他数据, 此时就需要使用外连接查询。

(2) 外连接查询的语法格式如下。

```
SELECT 所查字段 FROM 数据表 1 LEFT|RIGHT [OUTER] JOIN 数据表 2 ON 匹配条件
```

- LEFT JOIN: 返回左表中的所有记录和右表中符合连接条件的记录。
- RIGHT JOIN: 返回右表中的所有记录和左表中符合连接条件的记录。

(3) 左连接查询

左连接查询的结果包括 LEFT JOIN 子句中左表的所有记录, 以及右表中满足连接条件的记录。如果左表的某条记录在右表中不存在, 则右表中对应字段的值显示为 NULL。

通过案例演示左连接连接查询。

(4) 右连接查询

右连接查询的结果包括 RIGHT JOIN 子句中右表的所有记录, 以及左表中满足连接条件的记录。如果右表的某条记录在左表中没有匹配, 则左表中对应字段的值显示为 NULL。

通过案例演示右连接连接查询。

知识点 4-复合条件连接查询

教师通过 PPT 结合实际操作的方式讲解复合条件连接查询。

(1) 复合条件连接查询是指在连接查询的过程中, 通过添加过滤条件限制执行结果, 使执行结果更加精确。

(2) 通过案例演示符合条件连接查询。

四、归纳总结

教师回顾本节课所讲的内容, 并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课

的预习作业。

第二课时

(IN 关键字结合子查询、EXISTS 关键字结合子查询、ANY 关键字结合子查询、ALL 关键字结合子查询、比较运算符结合子查询)

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

子查询是指一个查询语句嵌套在另一个语句内部的查询，当某个语句执行所需要的过滤条件是另外一个 SELECT 语句的结果时，可以使用子查询。子查询通常在 WHERE 子句中结合操作符一起使用，操作符可以是 IN、EXISTS、ANY、ALL、比较运算符，本节将结合这几种操作符的子查询进行讲解。

三、新课讲解

知识点 1-IN 关键字结合子查询

教师通过 PPT 结合实际操作的方式讲解 IN 关键字结合子查询。

- (1) IN 关键字结合子查询使用时，需要内层子查询语句返回的结果是一个数据列，这个数据列中的值供外层语句进行比较操作。
- (2) 通过案例演示查询语句中 IN 关键字结合子查询的使用。

知识点 2-EXISTS 关键字结合子查询

教师通过 PPT 结合实际操作的方式讲解 EXISTS 关键字结合子查询。

- (1) EXISTS 关键字用于判断子查询的结果集是否为空，若子查询的结果集不为空，返回 TRUE，否则返回 FALSE。
- (2) 通过案例演示查询语句中 EXISTS 关键字结合子查询的使用。

知识点 3-ANY 关键字结合子查询

教师通过 PPT 结合实际操作的方式讲解 ANY 关键字结合子查询。

- (1) ANY 关键字结合子查询时，表示子查询的查询结果集中的任一查询结果。
- (2) 通过案例演示查询语句中 ANY 关键字结合子查询的使用。

知识点 4-ALL 关键字结合子查询

教师通过 PPT 结合实际操作的方式讲解 ALL 关键字结合子查询。

- (1) ALL 关键字结合子查询时，表示子查询的所有查询集中是所有结果。
- (2) 通过案例演示查询语句中 ALL 关键字结合子查询的使用。

知识点 5-比较运算符结合子查询

教师通过 PPT 结合实际操作的方式讲解比较运算符结合子查询。

通过一个案例，演示查询语句中比较运算符结合子查询的使用

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台（<http://tch.ityx.com>）布置本节课作业以及下节课的预习作业。

第三课时

（添加外键约束、操作关联表、删除外键约束）

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

实际开发的项目中，一个健壮数据库中的数据一定有很好的参照完整性。例如，员工管理系统中有员工表和部门表，如果员工表的部门编号字段使用了 20 的部门编号，部门表中的 20 编号却被删除了，这样就会产生垃圾数据或者错误数据。为了保证数据的完整性，可以在员工表中添加外键约束。本节将对添加外键约束、操作关联表、删除外键约束进行讲解。

三、新课讲解

知识点 1-添加外键约束

教师通过 PPT 结合实际操作的方式讲解添加外键约束。

(1) 外键是数据表中的一个特殊字段，它引用另一张数据表中的一列或多列，被引用的列应该具有主键约束或唯一性约束。

(2) 在 MySQL 中为从表添加外键约束的语法格式如下。

```
ALTER TABLE 从表名 ADD CONSTRAINT [外键名称] FOREIGN KEY(外键字段名)
REFERENCES 主表名(主键字段名);
```

(3) 通过案例演示在数据表中添加外键约束。

知识点 2-操作关联表

教师通过 PPT 结合实际操作的方式讲解操作关联表。

(1) 数据表的关联关系

- 一对一
- 多对一
- 多对多

(2) 关联表添加数据

如果要为两个数据表添加数据，就需要先为主表添加数据。通过案例演示对添加外键约束的关联表添加数据。

(3) 关联表删除数据

2 个数据表之间通过外键约束进行关联后，主表中已经被引用的值不能直接删除。

知识点 3-删除外键约束

教师通过 PPT 结合实际操作的方式讲解删除外键约束。

(1) 删除外键约束的语法格式如下。

```
ALTER TABLE 表名 DROP FOREIGN KEY 外键名;
```

(2) 通过案例演示删除数据表中的外键约束。

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给

	<p>予指导。</p> <p>五、布置作业</p> <p>教师通过高校教辅平台 (http://tch.ityx.com) 布置本节课作业以及下节课的预习作业。</p> <p style="text-align: center;">第四课时（上机练习）</p> <p>上机一：（考察知识点为上机实践：图书管理系统的多表操作）</p> <p>形式：单独完成</p> <p>题目：操作图书管理系统多张数据表中的数据</p> <p>要求：</p> <p>基于图书管理系统的员工表和图书表完成如下操作。</p> <ol style="list-style-type: none"> (1) 查询张三当前借阅的图书信息，图书信息只需显示借阅人编号、借阅人名称、图书名称和借阅时间。 (2) 查询价格比《西游记》的价格高的图书信息，图书信息只需显示图书名称和图书价格。 (3) 查询价格比所有图书的平均价格还低的图书信息，图书信息只需显示图书名称和图书价格。 (4) 查询图书状态和《三国演义》相同的图书信息，图书信息只需显示图书名称、图书价格和状态。 (5) 查询已借阅图书中，价格比任意未借阅的图书价格还低的图书信息，图书信息只需显示图书名称、图书价格和状态。 (6) 查询价格比任意已借阅的图书价格还高的图书信息，图书信息只需显示图书名称、图书价格和状态。 (7) 对图书表中的借阅者编号添加外键约束，以保证数据的完整性。
<p>教学后记</p>	

课题名称	第 6 章 索引和视图	计划课时	3 课时
教学引入	<p>在 MySQL 中，索引类似于书籍的目录，如果想要快速访问数据表中的特定信息，可以建立索引加快数据查询效率。使用数据库时，不仅需要提高对数据的查询效率，也需要考虑数据的安全问题。在 MySQL 中可以创建一种叫视图的虚拟表，让使用视图的用户只能访问被允许访问的结果集，从而提高数据的安全性。除了安全性，视图还具备简化查询语句和逻辑数据独立性等优点，本章将对数据库中的索引和视图分别进行讲解。</p>		
教学目标	<ul style="list-style-type: none"> ● 使学生了解索引的概念，能够说出索引的作用。 ● 使学生掌握索引的创建，能够在创建和修改数据表的同时创建索引、在已有的数据表上创建索引。 ● 使学生熟悉索引的查看，能够通过 show 语句查看数据表中索引的信息。 ● 使学生掌握索引的删除，能够使用 ALTER TABLE 语句和 DROP INDEX 语句删除索引。 ● 使学生了解视图的概念，能够说出视图的优点。 ● 使学生掌握视图的管理，能够创建、查看、修改和删除视图。 ● 使学生掌握视图数据操作，能够通过视图添加、修改和删除基本表中的数据。 		
教学重点	<ul style="list-style-type: none"> ● 索引的创建 ● 视图管理 ● 视图数据操作 		
教学难点	<ul style="list-style-type: none"> ● 视图管理 		
教学方式	<p>课堂教学以 PPT 讲授为主，并结合多媒体进行教学</p>		
课程思政	<p>培养学生的严谨求实的科学精神、团队合作精神；培养学生社会责任感、爱国情怀等。</p>		

第一课时

(索引概述、索引的创建、索引的查看、索引的删除)

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

索引是数据库中用来提高数据查询效率最常用的数据库对象，好比新华字典的音序表，通过音序表可以快速的查找内容。索引对数据表中一列或多列的值与记录行之间按照一定的顺序建立关系，以提高对数据表中数据的查询速度。本节将针对索引概述、索引的创建、索引的查看、索引的删除进行讲解。

三、新课讲解

知识点 1-索引概述

教师通过 PPT 的方式讲解索引概述。

索引的概述:索引是数据库中用来提高数据查询效率最常用的数据库对象。根据索引实现语法不同，MySQL 中常见的索引大致分为 5 种，具体描述如下。

(1) 普通索引

普通索引是 MySQL 中的基本索引类型，使用 KEY 或 INDEX 定义，不需要添加任何限制条件。

(2) 唯一性索引

创建唯一性索引的字段，允许有 NULL 值，但需要保证索引对应字段中的值是唯一的。

(3) 主键索引

主键索引是一种特殊的唯一性索引，用于根据主键自身的唯一性标识每一条记录。主键索引的字段不允许有 NULL 值。

(4) 全文索引

全文索引主要用于提高在数据量较大的字段中的查询效率。

(5) 空间索引

空间索引只能创建在空间数据类型的字段上，其中空间数据类型存储的空间数据是指含有位置、大小、形状以及自身分布特征等多方面信息的数据。

根据创建索引的字段个数，可以将索引分为单列索引和复合索引。

知识点 2-索引的创建

教师通过 PPT 结合实际操作的方式讲解索引的创建。

(1) 创建数据表的同时创建索引

创建数据表的同时创建索引的基本语法格式如下所示。

```
CREATE TABLE 表名 (字段名 1 数据类型[完整性约束条件],
...
    {INDEX | KEY} [索引名] [索引类型] (字段列表)
    | UNIQUE [INDEX | KEY] [索引名] [索引类型] (字段列表)
    | PRIMARY KEY [索引类型] (字段列表)
    | {FULLTEXT | SPATIAL} [INDEX | KEY] [索引名] (字段列表)
...
);
```

根据 CREATE TABLE 语句的基本语法格式，分别演示单列索引和复合索引的

创建。

① 创建单列索引。

② 创建复合索引。

(2) 在已有的数据表上创建索引

若想在已经存在的数据表上创建索引，可以使用 `CREATE INDEX` 语句，`CREATE INDEX` 语句创建索引的具体语法格式如下所示。

```
CREATE [UNIQUE|FULLTEXT|SPATIAL] INDEX 索引名  
      [索引类型] ON 数据表名 (字段列表);
```

根据 `CREATE INDEX` 语句中字段列表的个数，可将创建的索引分为单列索引和复合索引。

① 创建单列索引。

② 创建复合索引。

(3) 修改数据表的同时创建索引

使用 `ALTER TABLE` 语句在修改数据表的同时创建索引，其基本语法格式如下所示。

```
ALTER TABLE 数据表名  
      ADD {INDEX | KEY} [索引名] [索引类型] (字段列表)  
      | ADD UNIQUE [INDEX | KEY] [索引名] [索引类型] (字段列表)  
      | ADD PRIMARY KEY [索引类型] (字段列表)  
      | ADD {FULLTEXT | SPATIAL} [INDEX | KEY] [索引名] (字段列表)
```

根据 `ALTER TABLE` 语句中索引作用字段列表的个数，可将创建的索引分为单列索引和复合索引。

① 创建单列索引。

② 创建复合索引。

知识点 3-索引的查看

教师通过 PPT 结合实际操作的方式讲解索引的查看。

(1) 可以通过如下语法格式的语句进行查看索引。

```
SHOW {INDEXES|INDEX|KEYS} FROM 数据表名;
```

(2) 索引信息字段的含义，具体如下。

- **Table:** 索引所在的数据表的名称。
- **Non_unique:** 索引是否可以重复，0 表示不可以，1 表示可以。
- **Key_name:** 索引的名字，如果索引是主键索引，则它的名字为 PRIMARY。
- **Seq_in_index:** 建立索引的字段序号值，默认从 1 开始。
- **Column_name:** 建立索引的字段。
- **Collation:** 索引字段是否有排序，A 表示有排序，NULL 表示没有排序。
- **Cardinality:** MySQL 连接时使用索引的可能性（精确度不高），值越大可能性越高。
- **Sub_part:** 前缀索引的长度，如字段值都被索引，则 Sub_part 为 NULL。
- **Packed:** 关键词如何被压缩，如果没有被压缩，则为 NULL。
- **Null:** 索引字段是否含有 NULL 值，YES 表示含有，NO 表示不含有。
- **Index_type:** 索引方式，可选值有 FULLTEXT、HASH、BTREE、RTREE。
- **Comment:** 索引字段的注释信息。
- **Index_comment:** 创建索引时添加的注释信息。

- **Visible:** 索引对查询优化器是否可见, YES 表示可见, NO 表示不可见。

- **Expression:** 使用什么表达式作为建立索引的字段, NULL 表示没有。

(3) 在 MySQL 中除了可以查看数据表中的索引信息, 还可以通过 EXPLAIN 关键字分析 SQL 语句的执行情况, 执行语句的字段具体如下。

- **id:** 查询标识符, 默认从 1 开始, 如果使用了联合查询, 则该值依次递增。

- **select_type:** 查询类型。

- **table:** 输出行所引用的数据表的名称

- **partitions:** 匹配的分区

- **type:** 连接的类型。

- **key_len:** 索引字段的长度。

- **ref:** 表示哪些字段或常量与索引进行了比较。

- **rows:** 预计需要检索的记录数。

- **filtered:** 按条件过滤的百分比。

- **Extra:** 附件信息, 如 Using index 表示使用了索引覆盖。

知识点 4-索引的删除

教师通过 PPT 结合实际操作的方式讲解索引的删除。

(1) 使用 ALTER TABLE 删除索引。

使用 ALTER TABLE 删除索引的基本语法格式如下所示。

```
ALTER TABLE 表名
{DROP {INDEX | KEY} index_name
|DROP PRIMARY KEY};
```

(2) 使用 DROP INDEX 删除索引

使用 DROP INDEX 语句删除索引的基本语法格式如下所示。

```
DROP INDEX 索引名 ON 数据表名;
```

四、归纳总结

教师回顾本节课所讲的内容, 并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台 (<http://tch.ityxb.com>) 布置本节课作业以及下节课的预习作业。

第二课时

(视图概述、视图管理、视图数据操作)

一、复习巩固

教师通过上节课作业的完成情况, 对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

视图是一种虚拟存在的表, 并不在数据库中实际存在, 它的数据依赖于真实存在的数据表。通过视图不仅可以看到其依赖数据表中的数据, 还可以像操作数据表一样, 对数据表中的数据进行添加、修改和删除。本节将对视图概述、视图管理、视图数据操作进行讲解。

三、新课讲解

知识点 1-视图概述

教师通过 PPT 结合实际操作的方式讲解视图概述。

与直接操作数据表相比，视图具有以下优点。

- 简化查询语句
- 安全性
- 逻辑数据独立性

知识点 2-视图管理

教师通过 PPT 结合实际操作的方式讲解视图管理。

(1) 创建视图

① 创建视图的基本语法格式如下所示。

```
CREATE [OR REPLACE] VIEW 视图名 [(字段列表)] AS select_statement
```

② 基于单表创建视图

③ 基于多表创建视图

(2) 查看视图

① 查看视图的字段信息

在 MySQL 中，使用 DESCRIBE 语句可以查看视图的字段名、字段类型等字段信息。DESCRIBE 语句的基本语法格式如下所示。

```
DESCRIBE 视图名;
```

或者简写为：

```
DESC 视图名;
```

② 查看视图的状态信息

在 MySQL 中，可以使用 SHOW TABLE STATUS 语句可以查看视图和数据表的状态信息。SHOW TABLE STATUS 语句的基本语法格式如下所示。

```
SHOW TABLE STATUS LIKE '视图名';
```

③ 查看视图的创建语句

在 MySQL 中，使用 SHOW CREATE VIEW 语句可以查看创建视图时的定义语句。SHOW CREATE VIEW 语句的基本语法格式如下所示。

```
SHOW CREATE VIEW 视图名;
```

(3) 修改视图

① 使用 CREATE OR REPLACE VIEW 语句修改视图

使用 CREATE OR REPLACE VIEW 语句修改视图，其基本语法格式如下所示。

```
CREATE OR REPLACE VIEW 视图名 AS SELECT 语句
```

② 使用 ALTER 语句修改视图

使用 ALTER 语句修改视图的基本语法格式如下所示。

```
ALTER VIEW <视图名> AS <SELECT 语句>
```

(4) 删除视图

删除视图的基本语法格式如下所示。

```
DROP VIEW view_name [,view_name1,...];
```

知识点 3-视图数据操作

教师通过 PPT 结合实际操作的方式讲解视图数据操作。

(1) 添加数据

	<p>通过视图向基本表添加数据可以使用 INSERT 语句。</p> <p>(2) 修改数据 通过视图修改基本表的数据，可以使用 UPDATE 语句。</p> <p>(3) 删除数据 通过视图删除基本表的数据，可以使用 DELETE 语句。</p> <p>四、归纳总结 教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并予以指导。</p> <p>五、布置作业 教师通过高校教辅平台 (http://tch.ityxb.com) 布置本节课作业以及下节课的预习作业。</p> <p style="text-align: center;">第三课时（上机练习）</p> <p>上机：（考察知识点为上机实践：图书管理系统中索引和视图的应用）</p> <p>形式：独立完成</p> <p>题目：在图书管理系统的图书表中应用索引和视图</p> <p>要求：</p> <p>(1) 在图书表 book 的图书名称 name 上创建一个索引 index_bookname。</p> <p>(2) 在图书名称 name 和图书状态 state 上建立一个索引 index_bookname_state。</p> <p>(3) 删除索引 index_bookname_state。</p> <p>(4) 创建一个只包含图书名称、图书上架时间和图书状态的视图 view_book_state。</p> <p>(5) 创建一个只包含图书名称、借阅者名称和借阅时间的视图 view_book_borrower。</p> <p>(6) 删除视图 view_book_borrower。</p>
教学后记	

课题名称	第 7 章 事务	计划课时	3 课时
教学引入	<p>通过前几章的学习，大家对数据库的概念、数据库的基本操作以及 SQL 语句的使用有了一定的了解，在数据库开发过程中，经常会为了完成某一功能而编写一组 SQL 语句。为了确保每一组 SQL 语句操作数据的完整性，MySQL 引入了事务的管理，本章将针对事务进行详细地讲解。</p>		
教学目标	<ul style="list-style-type: none"> ● 使学生了解事务，能够说出事务的概念。 ● 使学生掌握事务的基本操作，能够开启、回滚和提交事务，以及创建事务的保存点。 ● 使学生熟悉事务的隔离级别，能够说出 MySQL 中事务的隔离级别，以及每个隔离级别的特点。 		
教学重点	<ul style="list-style-type: none"> ● 事务的基本操作 ● 事务的保存点 		
教学难点	<ul style="list-style-type: none"> ● READ UNCOMMITTED ● READ COMMITTED ● REPEATABLE READ ● SERIALIZABLE 		
教学方式	<p>课堂教学以 PPT 讲授为主，并结合多媒体进行教学</p>		
课程思政	<p>培养学生的严谨求实的科学精神、团队合作精神； 培养学生社会责任感、爱国情怀等。</p>		
教学过程	<p style="text-align: center;">第一课时 (事务概述、事务的基本操作、事务的保存点)</p> <p>一、复习巩固 教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。</p> <p>二、通过直接导入的方式导入新课 事务处理机制在应用程序开发过程中有着非常重要的作用，它可以保证在同一个事务中的操作具有同步性，从而让整个应用程序更加安全，本节将针对事务概述、事务的基本操作、事务的保存点进行详细地讲解。</p> <p>三、新课讲解</p> <p>知识点 1-事务概述 教师通过 PPT 结合实际操作的方式讲解事务概述。 MySQL 中的事务必须满足 4 个特性，具体如下。</p> <p>(1) 原子性 原子性是指一个事务必须被视为一个不可分割的最小工作单元，只有事务中所有的数据库操作都执行成功，才算整个事务执行成功。</p> <p>(2) 一致性 一致性是指事务将数据库从一个一致状态转变为下一个一致的状态。在事务完成之前和完成之后，都要保证数据库内的数据处于一致的状态。</p> <p>(3) 隔离性</p>		

隔离性是指当一个事务在执行时，不会受到其他事务的影响。隔离性保证了未完成事务的所有操作与数据库系统的隔离，直到事务完成之后，才能看到事务的执行结果。

(4) 持久性

持久性是指事务一旦提交，对数据库中数据的修改就是永久性的。

知识点 2-事务的基本操作

教师通过 PPT 结合实际操作的方式讲解事务的基本操作。

(1) 显式开启事务

显式开启事务的语句如下。

```
START TRANSACTION;
```

(2) 手动提交事务

手动提交事务的语句具体如下。

```
COMMIT;
```

(3) 回滚事务

如果不想提交当前事务，还可以使用下列语句取消事务（即回滚），具体如下。

```
ROLLBACK;
```

(4) 通过具体的案例演示事务的使用。

知识点 3-事务的保存点

教师通过 PPT 结合实际操作的方式讲解事务的保存点。

(1) 创建保存点

事务中创建保存点的语法格式如下所示。

```
SAVEPOINT 保存点名;
```

(2) 删除保存点

如果某个保存点不再使用，可以通过如下语法格式删除指定的保存点。

```
RELEASE SAVEPOINT 保存点名;
```

(3) 通过案例演示事务保存点的使用。

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台（<http://tch.ityxb.com>）布置本节课作业以及下节课的预习作业。

第二课时

(READ UNCOMMITTED、READ COMMITTED、REPEATABLE READ、SERIALIZABLE)

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

MySQL 支持多线程并发访问，用户可以通过不同的线程执行不同的事务。为了保证多个事务之间互不影响，就需要为事务设置适当的隔离级别。MySQL 中，事务有 4 种隔离级别，分别为 READ UNCOMMITTED（读未提交）、READ

COMMITTED（读已提交）、REPEATABLE READ（可重复读）和 SERIALIZABLE（串行化）。本节将针对事务隔离级别的相关知识进行详细讲解。

三、新课讲解

知识点 1-READ UNCOMMITTED

教师通过 PPT 结合实际操作的方式讲解 READ UNCOMMITTED。

(1) READ UNCOMMITTED 是事务隔离级别中最低的级别，在该级别下的事务可以读取到其他事务中未提交的数据，这种读取方式也被称为脏读（Dirty Read）。

(2) 通过案例演示脏读现象和解决脏读现象的方式，步骤如下。

- ① 设置客户端 B 中事务的隔离级别为 READ UNCOMMITTED
- ② 演示脏读
- ③ 设置客户端 B 事务的隔离级别为 READ COMMITTED
- ④ 验证是否出现脏读

知识点 2-READ COMMITTED

教师通过 PPT 结合实际操作的方式讲解 READ COMMITTED。

(1) MySQL 中 READ COMMITTED 级别下，事务只能读取其他事务已经提交的内容，可以避免脏读现象，但是会出现不可重复读和幻读的情况。不可重复读是指在事务内重复读取别的线程已经提交的数据，由于多次查询期间，其他事务做了更新操作，出现多次读取的结果不一致的现象。

(2) 通过案例演示不可重复读的现象和解决不可重复读的方式，步骤如下。

- ① 演示不可重复读
- ② 设置客户端 B 中事务的隔离级别为 REPEATABLE READ
- ③ 验证是否出现不可重复读

知识点 3-REPEATABLE READ

教师通过 PPT 结合实际操作的方式讲解 REPEATABLE READ。

(1) REPEATABLE READ 是 MySQL 默认的事务隔离级别，它可以避免脏读、不可重复读。

(2) REPEATABLE READ 理论上会出现幻读，幻读又被称为虚读，是指在一个事务内两次查询中数据条数不一致，幻读和不可重复读类似，同样是在两次查询过程中，不同的是，幻读是由于其他事务做了插入记录的操作，导致记录数有所增加。不过 MySQL 的存储引擎通过多版本并发控制机制解决了该问题，将事务的隔离级别为 REPEATABLE READ 时可以避免幻读。

(3) 通过案例演示幻读的现象和解决幻读的方式，步骤如下。

- ① 设置客户端 B 的隔离级别为 READ COMMITTED
- ② 演示幻读
- ③ 重新设置客户端 B 的隔离级别为 REPEATABLE READ
- ④ 验证是否出现幻读

知识点 4-SERIALIZABLE

教师通过 PPT 结合实际操作的方式讲解 SERIALIZABLE。

(1) SERIALIZABLE 是事务的最高隔离级别，它会在每个读的数据行上加锁，从而解决脏读、幻读、重复读的问题。这个级别，可能导致大量的超时和锁竞

争的现象，因此也是性能最低的一种隔离级别。

(2) 通过案例演示事务的隔离级别设置 SERIALIZABLE 后导致的现象，步骤如下。

- ① 设置客户端 B 中事务的隔离级别为 SERIALIZABLE
- ② 演示可串行化

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台 (<http://tch.ityxb.com>) 布置本节课作业以及下节课的预习作业。

第三课时（上机练习）

上机：（考察知识点为上机实践：图书管理系统中事务的应用）

形式：独立完成

题目：在图书管理系统的应用事务

要求：

(1) 手动开启事务，首先删除数据表 book 原有的全部数据，然后向数据表 book 插入王先生提供的图书信息。

(2) 删除测试数据并插入新数据后，查看数据表中的数据，如果数据无误，提交本次事务，否则对事务进行回滚。

课题名称	第 8 章 数据库编程	计划课时	6 课时
教学引入	<p>为了提高 SQL 语句的重用性，MySQL 可以将频繁使用的业务逻辑封装成程序进行存储，这类程序主要包括存储过程、函数、触发器等。MySQL 在 SQL 标准的基础上，扩展了一些程序设计语言的元素，如变量、流程控制语句等。这些程序设计语言的元素可以让程序更加健全，以提高数据库系统的性能。本章将针对数据库编程的相关内容进行讲解。</p>		
教学目标	<ul style="list-style-type: none"> ● 使学生掌握存储过程的基本操作，能够创建、查看、调用、修改和删除存储过程。 ● 使学生掌握存储函数的基本操作，能够创建、查看、调用和删除存储函数。 ● 使学生掌握变量的使用，能够查看和修改系统变量，对用户变量和局部变量进行定义和赋值。 ● 使学生掌握流程控制语句的使用，能够在程序中灵活使用判断语句、循环语句和跳转语句控制程序执行流程。 ● 使学生掌握错误触发条件和错误处理，能够正确定义错误触发条件和错误处理程序。 ● 使学生掌握游标的使用，能够使用游标检索数据。 ● 使学生了解触发器，能够说出触发器的概念。 ● 使学生掌握触发器的基本使用，能够创建、查看和删除触发器。 		
教学重点	<ul style="list-style-type: none"> ● 创建存储过程 ● 调用存储过程 ● 创建存储函数 ● 调用存储函数 ● 判断语句 ● 循环语句 ● 跳转语句 ● 自定义错误处理程序 ● 使用游标检索数据 ● 触发器的基本操作 		
教学难点	<ul style="list-style-type: none"> ● 自定义错误名称 ● 自定义错误处理程序 ● 使用游标检索数据 		
教学方式	课堂教学以 PPT 讲授为主，并结合多媒体进行教学		
课程思政	培养学生的严谨求实的科学精神、团队合作精神；培养学生社会责任感、爱国情怀等。		

第一课时

(创建存储过程、查看存储过程、调用存储过程、修改存储过程、删除存储过程)

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

开发过程中，经常会遇到重复使用某一功能的情况，为此，MySQL 引入了存储过程。存储过程是一组可以完成特定功能的 SQL 语句集合，它可以将常用或复杂的操作封装成一个代码块存储在数据库服务器中，以便重复使用，大大减少数据库开发人员的工作量。本节将针对存储过程进行详细讲解。

三、新课讲解

知识点 1-创建存储过程

教师通过 PPT 结合实际操作的方式讲解创建存储过程。

(1) 展示创建存储过程的基本语法格式。

```
CREATE PROCEDURE 存储过程名 ([[IN | OUT | INOUT] 参数名称 参数类型])
[characteristic ...] routine_body
```

(2) 创建存储过程的语句中，参数和选项的具体含义如下所示。

- IN: 表示输入参数，该参数需要在调用存储过程时传入。
- OUT: 表示输出参数，初始值为 NULL，它是将存储过程中的值保存到 OUT 指定的参数中，返回给调用者。
- INOUT: 表示输入输出参数，既可以作为输入参数也可以作为输出参数。
- characteristic: 表示存储过程中的例程可以设置的特征。
- routine_body: 表示存储过程中的过程体。

(3) 通过案例演示存储过程的创建。

知识点 2-查看存储过程

教师通过 PPT 结合实际操作的方式讲解查看存储过程。

(1) 使用 SHOW PROCEDURE STATUS 语句显示存储过程的状态信息

SHOW PROCEDURE STATUS 语句显示存储过程状态信息的基本语法格式如下。

```
SHOW PROCEDURE STATUS [LIKE 'pattern']
```

(2) 使用 SHOW CREATE PROCEDURE 语句显示存储过程的创建信息

使用 SHOW CREATE PROCEDURE 语句可以显示存储过程的创建语句等信息，其基本语法格式如下。

```
SHOW CREATE PROCEDURE 存储过程名;
```

(3) 从 information_schema.Routines 表中查看存储过程的信息

存储过程的信息存储在 information_schema 数据库下的 Routines 表中，可以通过查询该表的记录获取存储过程的信息，查询语句如下。

```
SELECT * FROM information_schema.Routines
WHERE ROUTINE_NAME='pro_emp' AND ROUTINE_TYPE='PROCEDURE'\G
```

知识点 3-调用存储过程

教师通过 PPT 结合实际操作的方式讲解调用存储过程。

(1) 调用存储过程的基本语法格式如下。

```
CALL [数据库名称.]存储过程名称 ([实参列表]);
```

(2) 通过一个案例演示存储过程的调用。

知识点 4-修改存储过程

教师通过 PPT 结合实际操作的方式讲解修改存储过程。

(1) 修改存储过程，其基本语法格式如下。

```
ALTER PROCEDURE 过程名称 [characteristic ...];
```

(2) 通过案例演示修改存储过程的特征值。

知识点 5-删除存储过程

教师通过 PPT 结合实际操作的方式讲解删除存储过程。

(1) 删除存储过程的基本语法格式如下所示。错误!未定义书签。

```
DROP PROCEDURE [IF EXISTS] 存储过程名称;
```

(2) 通过案例演示存储过程的删除。

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台 (<http://tch.itymb.com>) 布置本节课作业以及下节课的预习作业。

第二课时

(创建存储函数、查看存储函数、调用存储函数、删除存储函数)

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

MySQL 支持函数的使用，MySQL 中的函数可以分为两种，一种是内置函数，一种是自定义函数。在 MySQL 中，错误!未定义书签。通常将用户自定义的函数称为存储函数，存储函数和 MySQL 内置函数性质相同，都用于实现某种功能。本节将对存储函数进行讲解。

三、新课讲解

知识点 1-创建存储函数

教师通过 PPT 结合实际操作的方式讲解创建存储函数。

(1) 创建存储函数的基本语法格式如下所示。

```
CREATE FUNCTION func_name ([func_parameter[...]])  
RETURNS type  
[characteristic ...]  
Routine_body
```

(2) 通过一个案例演示存储函数的创建。

知识点 2-查看存储函数

教师通过 PPT 结合实际操作的方式讲解查看存储函数。

(1) 使用 SHOW FUNCTION STATUS 语句显示存储函数的状态信息，基本语法格式如下。

```
SHOW FUNCTION STATUS [LIKE 'pattern'];
```

(2) 使用 SHOW CREATE FUNCTION 语句显示存储函数的创建信息，基本语法格式如下。

```
SHOW CREATE FUNCTION 存储函数名;
```

(3) 在 information_schema.Routines 表中查看存储函数的信息，基本语法格式如下。

```
SELECT * FROM information_schema.Routines  
WHERE ROUTINE_NAME='存储函数名' AND ROUTINE_TYPE='FUNCTION' \G
```

知识点 3-调用存储函数

教师通过 PPT 结合实际操作的方式讲解调用存储函数。

(1) 调用存储函数的基本语法格式如下所示。

```
SELECT [数据库名.]函数名 1 (实参列表) [函数名 2 (实参列表) ...];
```

(2) 通过一个案例演示存储函数的调用。

知识点 4-删除存储函数

教师通过 PPT 结合实际操作的方式讲解删除存储函数。

(1) 删除存储函数的语法格式如下所示。

```
DROP FUNCTION [IF EXISTS] 函数名称;
```

(2) 通过一个案例演示存储函数的删除。

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

第三课时

(系统变量、用户变量、局部变量)

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

变量就是在程序执行过程中，其值可以改变的量。在 MySQL 中，可以利用变量存储程序执行过程中涉及的数据，如输入的值、计算结果等。根据变量的作用范围可以将其划分为系统变量、用户变量和局部变量。本节将针对这 3 种变量进行讲解。

三、新课讲解

知识点 1-系统变量

教师通过 PPT 结合实际操作的方式讲解系统变量。

(1) 系统变量又分为全局 (GLOBAL) 变量和会话 (SESSION) 变量，其中全局变量指的是 MySQL 系统内部定义的变量，对所有 MySQL 客户端都有效。

(2) 查看系统变量

在 MySQL 中可以通过 SHOW 语句显示所有的系统变量，其语法格式如下所示。

```
SHOW [GLOBAL | SESSION] VARIABLES [LIKE '匹配字符串' | WHERE 表达式];
```

(3) 修改系统变量

在 MySQL 中，系统变量可以通过 SET 语句进行修改，修改的语法格式如下所示。

```
SET [GLOBAL | @@GLOBAL. | SESSION | @@SESSION.] 系统变量名 = 新值;
```

知识点 2-用户变量

教师通过 PPT 结合实际操作的方式讲解用户变量。

用户变量指的是用户自己定义的变量，它和连接有关，即用户变量仅对当前用户使用的客户端生效，不能被其他客户端看到和使用。

用户变量由符号@和变量名组成，在使用用户变量之前，需要对用户变量进行定义并赋值。MySQL 中为用户变量赋值有以下 3 种方式。

- (1) 使用 SET 语句完成赋值。
- (2) 在 SELECT 语句中使用赋值符号:=完成赋值。
- (3) 使用 SELECT...INTO 语句完成赋值。

知识点 3-局部变量

教师通过 PPT 结合实际操作的方式讲解局部变量。

(1) 局部变量的作用范围仅在语句块 BEGIN...END 之间，在语句块 BEGIN...END 之外，局部变量不能被获取和修改。

(2) 局部变量使用 DECLARE 语句定义，定义的基本语法格式如下。

```
DECLARE 变量名 1 [, 变量名 2...] 数据类型 [DEFAULT 默认值];
```

(3) 通过案例演示局部变量的使用。

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台 (<http://tch.ityxb.com>) 布置本节课作业以及下节课的预习作业。

第四课时

(判断语句、循环语句、跳转语句)

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

程序在执行时，都会按照程序结构（由业务逻辑决定）对执行流程进行控制，程序的结构主要分为顺序结构，选择结构和循环结构，其中顺序结构会按照代码编写的先后顺序，依次执行；选择结构和循环结构会根据程序的执行情况调整和控制程序的执行顺序。程序执行流程由流程控制语句进行控制，MySQL 中的流程控制语句有 IF 语句、CASE 语句、LOOP 语句、LEAVE 语句、ITERATE 语句、REPEAT 语句和 WHILE 语句等。这些语句大体可以分为 3 类，分别为判断语句、循环语句和跳转语句，本节分别对这些语句进行讲解。

三、新课讲解

知识点 1-判断语句

教师通过 PPT 结合实际操作的方式讲解判断语句。

(1) IF 语句

① IF 语句的语法格式如下所示。

```
IF 条件表达式 1 THEN 语句列表
    [ELSEIF 条件表达式 2 THEN 语句列表]...
    [ELSE 语句列表]
END IF
```

② 通过一个案例演示 IF 语句的使用。

(2) CASE 语句

① CASE 语句的语法格式有 2 种，具体如下所示。

```
# 语法格式 1
CASE 表达式
    WHEN 值 1 THEN 语句列表
    [WHEN 值 2 THEN 语句列表]...
    [ELSE 语句列表]
END CASE
```

```
# 语法格式 2
CASE
    WHEN 条件表达式 1 THEN 语句列表
    [WHEN 条件表达式 2 THEN 语句列表]...
    [ELSE 语句列表]
END CASE
```

② 通过一个案例演示 CASE 语句的使用

知识点 2-循环语句

教师通过 PPT 结合实际操作的方式讲解循环语句。

(1) LOOP 语句

① LOOP 语句通常用于实现一个简单的循环，其基本语法格式如下所示。

```
[标签:]LOOP
    语句列表
END LOOP[标签];
```

② 通过一个案例演示 LOOP 语句的使用。

(2) REPEAT 语句

① REPEAT 语句用于循环执行符合条件的语句列表，REPEAT 语句的基本语法格式如下所示。

```
[标签:] REPEAT
    语句列表
    UNTIL 条件表达式
END REPEAT [标签]
```

② 通过示例演示 REPEAT 语句的使用。

(3) WHILE 语句

① WHILE 语句也用于循环执行符合条件的语句列表，WHILE 语句的基本语法格式如下所示。

[标签:]WHILE 条件表达式 DO

语句列表

END WHILE[标签]

② 通过示例演示 WHILE 语句的使用。

知识点 3-跳转语句

教师通过 PPT 结合实际操作的方式讲解 switch 条件语句。

(1) 跳转语句用于实现执行过程中的流程跳转，跳转语句的基本语法格式如下所示。

```
{ITERATE|LEAVE} 标签名;
```

(2) 通过案例演示跳转语句的使用。

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

第五课时

(自定义错误名称、自定义错误处理程序、游标的操作流程、使用游标检索数据、触发器概述、触发器的基本操作)

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过情景导入的方式导入新课

程序在运行过程中可能会发生错误，发生错误时，默认情况下，MySQL 将自动终止程序的执行。有些时候，如果不希望程序因为错误而停止执行，可以通过 MySQL 中的错误处理机制自定义错误名称和错误处理程序，让程序遇到警告或错误时也能继续执行，从而增强程序处理问题的能力。接下来对错误处理进行讲解。

三、新课讲解

知识点 1-自定义错误名称

教师通过 PPT 结合实际操作的方式讲解自定义错误名称。

(1) 在 MySQL 中可以使用 DECLARE 语句为错误声明一个名称，声明的基本语法格式如下所示。

```
DECLARE 错误名称 CONDITION FOR 错误类型;
```

(2) 通过案例演示自定义错误名称。

知识点 2-自定义错误处理程序

教师通过 PPT 结合实际操作的方式讲解自定义错误处理程序。

(1) 自定义错误处理程序的基本语法格式如下所示。

```
DECLARE 错误处理方式 HANDLER FOR 错误类型[, 错误类型...] 程序语句段
```

(2) 错误类型有 6 种可选值，分别如下。

- sqlstate_value: 匹配 SQLSTATE 错误代码。
- condition_name: 匹配 DECLARE 定义的错误条件名称。
- SQLWARNING: 匹配所有以 01 开头的 SQLSTATE 错误代码。

- NOT FOUND: 匹配所有以 02 开头的 SQLSTATE 错误代码。
 - SQLEXCEPTION: 匹配所有没有被 SQLWARNING 或 NOT FOUND 捕获的 SQLSTATE 错误代码。
 - mysql_error_code: 匹配 mysql_error_code 类型的错误代码。
- (3) 通过案例演示自定义错误处理程序的应用。

知识点 3-游标的操作流程

教师通过 PPT 结合实际操作的方式讲解游标的操作流程。

(1) 定义游标

定义游标的基本语法格式如下所示。

```
DECLARE 游标名称 CURSOR FOR SELECT 语句
```

(2) 打开游标

声明游标之后, 要想从游标中提取数据, 需要先打开游标。在 MySQL 中, 打开游标通过 OPEN 关键字实现, 其语法格式如下。

```
OPEN 游标名称
```

(3) 利用游标检索数据

打开游标之后, 就可以通过游标检索 SELECT 语句返回的结果集中的数据, 游标检索数据的基本语法格式如下所示。

```
FETCH 游标名称 INTO 变量名 1 [, 变量名 2]...
```

(4) 关闭游标

游标检索完数据后, 应该利用 MySQL 提供的语法关闭游标, 释放游标占用的 MySQL 服务器的内存资源, 关闭游标的基本语法格式如下所示。

```
CLOSE 游标名称
```

知识点 4-使用游标检索数据

教师通过 PPT 结合实际操作的方式讲解使用游标检索数据。

通过具体的案例演示使用游标检索数据。

知识点 5-触发器概述

教师通过 PPT 结合实际操作的方式讲解触发器概述。

(1) 触发器可以看作是一种特殊的存储过程, 它与存储过程的区别在于, 存储过程使用 CALL 语句调用时才会执行, 而触发器会在预先定义好的事件 (例如 INSERT、DELETE 等操作) 发生时自动调用。

(2) 触发器具有以下优点。

- 触发器自动执行, 在触发器相关联的数据表中的数据发生修改, 触发器中定义的语句会自动执行。
- 触发器对数据进行安全校验, 保障数据安全。
- 通过和触发器相关联的表, 可以实现表数据的级联更改, 在一定程度上保证了数据的完整性。

知识点 6-触发器的基本操作

教师通过 PPT 结合实际操作的方式讲解触发器的基本操作。

(1) 创建触发器

创建触发器的基本语法格式如下所示。

```
CREATE TRIGGER 触发器名称 触发时机 触发事件 ON 数据表名 FOR EACH ROW  
触发程序
```

(2) 查看触发器

① 利用 **SHOW TRIGGER** 语句查看触发器信息的语法格式如下所示。

```
SHOW TRIGGERS;
```

② 通过 **SELECT** 语句查看该数据表获取触发器信息，通过 **TRIGGERS** 数据表查询触发器的语法格式如下所示。

```
SELECT * FROM information_schema.triggers [WHERE trigger_name= '触发器名称'  
'];
```

(3) 触发触发器

通过案例演示触发器的触发。

(4) 删除触发器

MySQL 中使用 **DROP TRIGGER** 语句删除触发器。

① **DROP TRIGGER** 语句的基本语法格式。

```
DROP TRIGGER [ IF EXISTS ] [数据库名.] 触发器名;
```

② 通过案例演示触发器的删除。

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台 (<http://tch.ityx.com>) 布置本节课作业以及下节课的预习作业。

第六课时（上机练习）

上机：（考察知识点为上机实践：数据库编程实战）

形式：独立完成

题目：基于图书管理系统的图书表 and 用户表进行数据库编程

要求：

(1) 创建一个存储过程 **proc_1**，执行后获取图书的名称、价格和借阅状态，如果是未借阅的状态，则显示未借阅；如果是已借阅的状态，则显示借阅人。创建完存储过程 **proc_1** 后，执行该存储过程查看效果。

(2) 创建一个存储过程 **proc_2**，执行后获取所有可借阅的图书信息，图书信息只需显示图书名称、图书价格和上架时间。创建完存储过程 **proc_2** 后，执行该存储过程查看效果。

(3) 创建一个存储函数 **func_1**，执行时输入用户名称，显示用户当前借阅中的图书名称。创建完存储函数 **func_1** 后，执行该存储函数查看效果。

(4) 创建一个存储函数 **func_2**，执行时输入图书名称，显示图书当前的价格档位；如果价格小于等于 40，显示“平民价格”；如果价格大于 40 并且小于等于 60，则显示“主流价格”；如果价格大于 60，则显示“高价格”。创建完存储函数 **func_2** 后，执行该存储函数查看效果。

(5) 查看当前数据库中存储过程 **proc_1** 和存储函数 **func_1** 的信息

(6) 查看存储过程 **proc_1** 和存储函数 **func_1** 的信息后，觉得不是自己想

要的效果，想要删除这 2 个程序。

(7) 创建一个触发器 `trig_book`，在借阅记录表中插入数据时，自动修改图书表中借阅相关的信息。

(8) 创建好触发器 `trig_book` 后，查看该触发器的信息。