

高职高专“十五”规划教材系列

SQL Server

数据库应用系统开发技术 实验指导及习题解答

朱如龙 刘 焰 编著

1.138S

高职高专“十五”规划教材系列

SQL Server 数据库应用 系统开发技术实验指导及习题解答

朱如龙 刘 焰 编著



机械工业出版社

本书分为3章,第1章为习题解答,包括配套教材《SQL Server 数据库应用系统开发技术》所附习题的全部答案。第2章为实验指导书,实验从基本的建立数据库及表对象入手,由浅入深,逐步引导学生建立其他数据库对象,实施数据完整性,建立查询、视图、索引、触发器、存储过程、进行数据库的备份与恢复。书中共14个实验,与教材配套使用,每个实验对应一个专题内容,学生可以按顺序做,也可以根据具体情况选做。第3章为课程设计指导书,包括客房管理系统课程设计示例和工资管理系统及图书管理系统两个课题的选题。其中,课程设计示例给出了设计的全部SQL脚本和VB程序代码,该示例的所有代码全部经过我们调试、编译并执行,可供大家在进行课程设计时参考。本书是《SQL Server 数据库应用系统开发技术》的配套教材。

本书图文并茂,叙述深入浅出,操作步骤清楚,十分便于读者阅读、理解。

图书在版编目(CIP)数据

SQL Server 数据库应用系统开发技术实验指导及习题解答/朱如龙,刘焰编著. —北京:机械工业出版社,2004.1

(高职高专“十五”规划教材系列)

ISBN 7-111-13753-1

I. S... II. ①朱...②刘... III. 关系数据库—数据库管理系统, SQL Server —高等学校:技术学校—教学参考资料 IV. TP311.138

中国版本图书馆CIP数据核字(2004)第000340号

机械工业出版社(北京市百万庄大街22号 邮政编码100037)

策 划:胡毓坚

责任编辑:蔡 岩

责任印制:施 红

北京忠信诚胶印厂印刷·新华书店北京发行所发行

2004年1月第1版·第1次印刷

787mm×1092mm 1/16·8.25印张·203千字

0001—5000册

定价:14.00元

凡购本图书,如有缺页、倒页、脱页,由本社发行部调换

本社购书热线电话(010)68993821、88379646

封面无防伪标均为盗版

出版说明

为了贯彻国务院发[2002]16号文件《国务院关于大力推进职业教育改革与发展的决定》的精神,进一步落实《中华人民共和国职业教育法》和《中华人民共和国劳动法》,实施科教兴国战略,大力推进高等职业教育改革与发展,我们组织力量,对实现高等职业教育培养目标和保证基本教学规格的文件基础课程、专业技术基础课程和重点建设专业主干课程的教材进行了规划和编写。

本套教材的内容涵盖了普通大专院校计算机及非计算机专业的文化基础课、专业基础课、专业课以及选修课程。主要分为文化基础、编程语言、硬件技术、网络信息、数据库应用及多媒体技术等几大类。为配合高职教育关于“培养 21 世纪与我国现代化建设要求相适应的一线科技实用型人才”的最新理念,我们特为本系列教材配备了实践指导丛书,以利于老师的教学和学生的学习。

本套教材将理论教学和实践教学紧密结合,图文并茂、内容实用、层次分明、讲解清晰,其中融入了作者长期的教学经验和丰富的实践经验。可用作各类大专院校、职业技术学校的教材,也可作为各类培训班的教材。

前 言

对于计算机专业的学生来说,“Microsoft SQL Server 2000”无疑是一门重要的专业课程。它既有较强的理论性,同时又具有较强的实践性。学习任何一门计算机课程,最好的办法是理论联系实际,一边进行理论课的学习,一边自己上机实践,多看联机帮助。如果可能的话,在老师的指导下,选择一个综合性实例进行课程设计,在学习中实践,在实用中提高,只有这样才能真正把所学的知识弄懂弄透。

本实验指导书从基本的建立数据库及表对象入手,由浅入深,逐步引导学生建立其他数据库对象,实施数据完整性,建立查询、视图、索引、触发器、存储过程、数据库的备份与恢复。

在本书写作过程中,作者注重了以下几个方面:

(1) 实用性

立足于实用性,本实验指导书的实验内容重点放在 SQL Server 2000 数据库管理系统开发中常用的基本概念、基本知识上。

(2) 可操作性

书中所有实例、操作步骤都经过我们的实际操作,这样确保了实验的可操作性和正确性。

(3) 简单性

本实验指导书上所有的实验自始至终都在我们精心设计的数据库及表对象中进行,表中所用数据也是为我们所熟悉的,这样便于大家验证实验结果的正确性。

在本书的写作过程中,我们始终贯穿这样一个理念:那就是设想“假如我是一个数据库管理系统开发人员,我需要掌握什么知识,我需要具备哪些技能”,把读者最需要掌握的和数据库开发实践中最常用到的知识传授给读者。衷心地希望经过我们不懈的努力,呈现在读者面前的这本《SQL Server 数据库应用系统开发技术》教材能够使读者受益,受到读者的青睐。

本书是《SQL Server 数据库应用系统开发技术》的配套教材。书中疏漏之处在所难免,恳请大家批评指正。

编 者

目 录

出版说明

前言

第 1 章	SQL Server 2000 数据库管理系统开发技术	1
1.1	Microsoft SQL Server 2000 简介	1
1.2	安装和配置 SQL Server 2000	2
1.3	Transact-SQL 语言	3
1.4	数据库应用系统设计	5
1.5	数据库的创建与管理	5
1.6	表的创建与管理	6
1.7	数据库安全性	8
1.8	数据检索	9
1.9	视图	10
1.10	创建 SQL Server 索引文件	12
1.11	存储过程	13
1.12	触发器	15
1.13	游标及事务	16
1.14	数据库的备份与恢复	17
第 2 章	SQL Server 2000 数据库管理系统开发技术实验指导书	19
2.1	实验 1 SQL Server 2000 的安装	19
2.2	实验 2 T-SQL 语言使用初步	26
2.3	实验 3 数据库的创建及管理	29
2.4	实验 4 创建与管理数据库对象	33
2.5	实验 5 实现数据完整性	37
2.6	实验 6 实现数据库安全性	45
2.7	实验 7 单表与多表查询	51
2.8	实验 8 视图的创建与使用	55
2.9	实验 9 索引的创建与管理	59
2.10	实验 10 存储过程	63
2.11	实验 11 触发器	67
2.12	实验 12 游标及事务的使用	71
2.13	实验 13 数据库的备份与恢复	74
2.14	实验 14 SQL Server 的实际应用	81
第 3 章	SQL Server 数据库管理系统课程设计指导书	87
3.1	课程设计示例——客房管理系统设计	87
3.1.1	课程设计的目的和意义	87

3.1.2	系统功能设计	87
3.1.3	数据库设计	88
3.1.4	“客房管理系统”应用程序结构设计	97
3.1.5	使用 VB 创建工程并编写代码	102
3.2	课程设计选题	117
3.2.1	课题 1——工资管理系统设计	117
3.2.2	课题 2——图书管理系统设计	119
附录		121
附录 A	JWGL 数据库各数据表的结构	121
附录 B	JWGL 数据库各数据表的数据实例	123

第 1 章 SQL Server 2000 数据库 管理系统开发技术

1.1 Microsoft SQL Server 2000 简介

1. 数据库管理系统的功能是什么？

数据库管理系统也称为 DBMS，它是对数据库中数据进行管理的一个中间层软件，主要具有以下功能：

（1）数据定义功能

DBMS 提供数据定义语言（DDL），通过使用 DDL，用户可以方便地对数据库中的相关对象进行定义。

（2）数据操纵功能

DBMS 提供数据操纵语言（DML），通过使用 DML，可以实现对数据库的基本操作，比如数据的查询、删除、更新等操作。

（3）数据通信功能

数据通信功能是分布式数据处理系统中最重要的一项功能之一，它支持与操作系统的联机处理、分时处理和远程作业传输。

（4）数据库的建立和维护功能

数据库的建立和维护主要包括数据库数据的输入、删除、更新功能；数据库数据的转储、恢复功能；数据库的重组和分析功能等。这些功能是数据库管理系统的基本功能。

（5）数据库的运行和管理功能

数据库的运行和管理主要包括安全性检查、完整性约束条件、并发控制及数据库的维护等。为了保证数据的安全性、完整性、一致性以及多个用户对数据的并发操作，所有的数据库操作都要在控制程序的统一管理下进行。

2. 数据库系统有哪几种数据模型？SQL Server 2000 是基于哪种模型的数据库系统？

数据模型是指对客观存在的实体之间的联系进行抽象后呈现出来的数据结构组织形式。根据它们的关系连接起来的结构图形状的不同，通常把数据模型分为三种：层次模型、网状模型和关系模型。所有的数据库系统均是基于某种数据模型的。SQL Server 2000 是基于关系模型的数据库系统。

3. 关系模型的特点有哪些？

1) 数据结构简单、概念清楚。

2) 行数据或列数据是无次序的，正是由于数据库中行与列是无次序的，在实现数据库操作时，可以忽略数据库的物理实现方式。如要使检索出的数据行按一定的次序显示，唯一的办法是检索时指定显示次序。

- 3) 能够直接反映实体之间一对一、一对多、多对多的关系。
- 4) 通过实体间的公共属性可以建立表与表之间的联系, 从而建立实体之间的联系。
- 5) 实现了数据完整性。依据关系理论, 实体集中的每个实体是不重复的。
- 6) 具有严格的理论基础。

4. 什么是主键? 什么是外键? 试举例说明。

主键 (PRIMARY KEY) 是指表中的某一列或几列的组合。该列或列的组合的值惟一地标识一行记录。主键简称为 PK。如学生的学号 (student_id) 必须惟一, 它用于惟一地标识一个学生, 而学生姓名 (student_name) 不能惟一地标识一个学生, 因为完全有可能存在两个学生同名同姓。主键通常用于实施实体完整性, 一般都要求每个表有而且仅有一个主键, 每一个主键值必须惟一, 且不能为空值 (NULL)。应该尽可能不要改变主键值。

外键 (FOREIGN KEY) 是指一个表中含有与另一个表的主键 PK 相同的列组, 则该列组称为外键, 简称为 FK, 外键并不要求惟一。和主键一样, 外键也是由一个或多个列组成。比如 department_id 是系表中的主键, 而在 teacher 表中是外键, 它代表的是一个教师所属的系号。外键的作用是实现参照完整性。

5. SQL Server 2000 有哪几个主要的服务? 分别有什么功能?

SQL Server 2000 主要有三个服务, 即: SQL Server 服务、SQL Server Agent 服务和 Microsoft Distribute Transaction Coordinator (MS DTC) 服务。

SQL Server 服务的功能是处理所有 Transact-SQL 语句、管理服务器上数据库的所有文件; 在多个并发用户之间分配计算机资源, 防止产生逻辑问题, 如控制并要求更新服务器上相同数据的用户请求的时序; 保证数据的安全性、一致性、完整性。

SQL Server Agent 服务器代理是独立于 SQL Server 的一个服务, 需要和 SQL Server 分开启动。SQL Server Agent 服务的主要功能是创建和管理作业、创建和管理报警、创建和管理操作员。

MS DTC 服务也是独立于 SQL Server 的一个服务, 它允许应用程序在一个事务中包括几个不同的数据源。MS DTC 协调分布式事务的正确完成, 保证服务器上所有数据的更新都是一致的, 或者在发生错误的情况下, 取消所有的数据更新。

1.2 安装和配置 SQL Server 2000

1. SQL Server 2000 有哪两种许可模式? 选择许可模式的依据是什么?

SQL Server 2000 中有两种使用许可模式。即“每服务器 (Per Server)”和“每客户 (Per Seat)”许可模式。

“每服务器”许可模式是指每一个同时与服务器的连接都需要独立的“客户访问许可协议”。该设置与每个服务器相关联。网络中的每个服务器都必须有同时与客户连接数目相对应的客户许可。“每客户”许可模式是指每台访问 SQL Server 的计算机需要独立的“客户访问许可协议”。该设置与每一个客户端相关联。每一个客户许可都可以用来同时访问多个服务器。

如果网络中只有一个 Microsoft SQL Server, 就应该选择“每客户”许可模式, 如果网络拥有多个 Microsoft SQL Server, 就应该选择“每服务器”许可模式。

2. SQL Server 应该使用什么样的服务账号？该账号应该满足什么样的条件？为了使 SQL Server 有较好的工作性能，SQL Server 最好安装在什么样的服务器上？

应该使用具有网络访问权限的服务器 NT 域账号作为 SQL Server 的服务账号。该账号应满足如下条件：

- 1) 必须是 Administrators 本地组成员。
- 2) 必须作为一个服务登录。
- 3) 必须具有在 SQL Server 上登录的所有权限。
- 4) 必须具有 Password Never Expires (口令永不失效)属性。

为了使 SQL Server 有较好的工作性能，SQL Server 最好安装在一个独立的服务器上。

3. SQL Server 安装完成后，有哪几个系统数据库？这些数据库的功能是什么？哪几个是样本数据库？

SQL Server 安装完成后，有 4 个系统数据库和 2 个样本数据库。4 个系统数据库分别是：master、model、tempdb 和 msdb，2 个样本数据库分别是：pubs 和 northwind。

master 数据库的功能是存储包括所有可用数据库及为其分配的空间、使用中的进程、用户账户、活动的锁、系统错误等信息和系统存储过程信息等。

msdb 数据库的功能是用来管理警报和任务，存储 SQL Server 管理数据库的每一次备份和恢复的历史信息。它由 SQL Server Agent 服务使用。

model 数据库的功能是为新的数据库充当模板。它可以对 model 数据库进行修改，包括添加用户定义数据类型、规则、缺省和存储过程，对 model 数据库的任何修改都会自动地反映到新建的数据库中。

tempdb 数据库是被所有 SQL Server 数据库和数据库用户共享的数据库。它用来存储临时信息，比如对一个未建索引的表查询时创建的临时索引信息。

4. 练习配置和注册服务器。

略。

5. 在 Windows NT/2000 Server 中的“服务”设置中练习打开和关闭 SQL Server 服务，比较与 SQL Server 2000 中开关服务是否相同。

从 Windows 2000 Server 的“服务”中启动、关闭和暂停 SQL Server：

用鼠标依次单击“开始”、“设置”、“控制面板”，再依次双击“管理工具”、“服务”。用鼠标右键单击“SQL Server 服务”，在系统弹出的快捷菜单上分别单击“启动”、“停止”或“暂停”就可以分别打开、关闭和暂停 SQL Server 服务了。

从 SQL Server 2000 中启动、关闭和暂停 SQL Server：

用鼠标依次单击“开始”、“程序”、“Microsoft SQL Server 2000”、“服务管理器”，系统将弹出“SQL Server 服务管理器”窗口，在此窗口上“服务”对应的下拉框内选择要操作的服务，分别单击“启动”、“停止”或“暂停”命令按钮就可以分别打开、关闭和暂停 SQL Server 服务了。

1.3 Transact-SQL 语言

1. SQL 语句分为哪 4 类？

SQL 语句通常分成 4 类：数据查询语言（Data Query Language）、数据操纵语言（Data

Manipulation Language)、数据定义语言(Data Define Language)和数据控制语言(Data Control Language)。

2. 如何删除一个已经被使用的用户自定义数据类型?

删除一个用户自定义类型的数据可以使用系统存储过程 `sp_droptype`, 或使用企业管理器。已经定义且未被使用的用户定义数据类型可以直接被删除, 但不能删除正在被表或其他数据库对象使用的用户定义数据类型。用鼠标右键单击要删除的用户自定义的数据类型, 点击“属性”, 系统将弹出“用户定义的数据类型”窗口, 在该窗口上点击“使用位置”, 可查看到该用户自定义数据类型被使用的情况, 根据该窗口的信息, 将此用户自定义的数据类型设为未使用状态, 然后将其删除。

3. 什么是局部变量? 什么是全局变量? 如何标识它们?

局部变量是在一个批处理中被声明、定义、赋值和引用的变量, 批处理结束后, 该变量也就消失了。全局变量是用来记录 SQL Server 服务器活动状态的变量, 它预先被定义, 用户只可以使用, 不可以重新定义和赋值。

局部变量是用户定义的变量, 用 `DECLARE` 语句声明, 在声明时它被初始化为 `NULL`, 用户可用 `SET` 语句为其赋值, 局部变量的使用范围是定义它的批、存储过程和触发器。它必须以 `@` 开始, 遵循 SQL Server 的标识符和对象的命名规范, 而且名字不能使用保留字。

全局变量是 SQL Server 系统所提供并赋值的变量。用户不能建立全局变量, 也不能使用 `SET` 语句去修改全局变量的值。全局变量的名字以 `@@` 开头。大多数全局变量的值是报告本次 SQL Server 启动后发生的系统活动。

4. 什么是批? 使用批有何限制?

批是指一个 SQL 语句集, 这些语句一起提交并作为一个组来执行。批结束的符号是“GO”。由于批中的多个语句是一起提交给 SQL Server 的, 所以可以节省系统开销。

使用批时有很多限制:

1) 并不是所有的 SQL 语句都可以和其他语句在一起组合成批。下述语句就不能组合在同一个批中:

```
CREATE PROCEDURE
CREATE RULE
CREATE DEFAULT
CREATE TRIGGER
CREATE VIEW
```

2) 不能在同一个批中既绑定又使用规则和缺省。

3) 不能在同一个批中既定义又使用 `CHECK` 约束。

4) 在同一个批中不能既删除对象又重建它。

5) 用 `SET` 语句改变的选项在批结束时生效。

6) 在同一个批中不能改变一个表再立即引用其新列。

5. 在默认情况下, SQL 脚本文件的后缀是什么? SQL 脚本执行的结果有哪几种形式?

SQL 脚本文件的默认后缀是 `*.sql`, SQL 脚本执行的结果有 3 种形式: 文本显示形式、表格显示形式及文件保存形式。

6. 重新配置查询分析器。
略。

1.4 数据库应用系统设计

1. 数据库应用系统开发的步骤是什么？

数据库应用系统开发的步骤是：

- 1) 认真收集、分析用户需求。
- 2) 根据研究分析后的用户需求，设计和建立应用系统。
- 3) 对应用系统进行测试。
- 4) 安装、实施、验收和评估应用系统。
- 5) 维护应用系统。

2. 在一个实际的应用系统中，客户需求有哪三种？

在实际的应用系统中，客户有 3 种不同类型的需求：

- 1) 功能需求：指包括应用系统应满足的所有功能。
- 2) 数据需求：指完成系统所有功能需求所需要的所有原始信息。
- 3) 性能需求：指整个应用系统必须满足的性能要求。

3. 什么是 ERA 模型？ERA 模型如何向逻辑数据库转化？

ERA 模型就是实体 (Entity) - 关系 (Relation) - 属性 (Attribute) 模型。其作用是描述组织的概念模型。ERA 模型主要由实体、关系、属性 3 个组件组成。

将 ERA 模型转化为逻辑数据库有两种方法。一种是采用如 ERWin、PowerDesign 工具软件进行转化，另一种是用手工的方法进行转化。使用工具转化和使用手工方法转化的原理是一样的。ERA 模型中的实体和关系都转换为表，实体和关系中的属性转化为表中的列。ERA 模型中的关系是通过主键参照关系体现的。

假设存在实体 a 和 b，现在要将其转化为表 A 和表 B，他们的主键分别记为 Pa 和 Pb，则：

1) 实体 a 和实体 b 是一对一关系(1: 1)：既可以将表 A 的主键 Pa 添加到表 B 中充当外键，也可以将表 B 的主键 Pb 添加到表 A 中充当外键。

2) 实体 a 和实体 b 是一对多关系(1: N)：必须把表 A 的主键 Pa 添加到表 B 中充当外键。如：班级和学生实体是一对多的关系，一个班级可以有多个学生，一个学生只能属于一个班级，所以需要把班级号放到学生表中充当外键。

3) 实体 a 和实体 b 是多对多关系(M: N)：除了生成表 A 和表 B 外，还应该生成一张关系表。这个关系表的列是：由表 A 的主键 Pa+表 B 的主键 Pb+关系自己的属性。

1.5 数据库的创建与管理

1. 创建数据库有哪几种方法？

在 SQL_Server 中，创建数据库的方法有 3 种：

- 1) 使用企业管理器创建。
- 2) 使用数据库创建向导创建。

3) 使用 T-SQL 语句创建。

2. 一个数据库中包含哪几种文件?

数据库具有惟一的主数据文件和一个或多个事务日志文件, 数据库文件可能还有次数据文件。这些文件均有操作系统文件名和 SQL 语句使用的逻辑文件名。主数据文件是所有数据文件的起点, 包含指向其他数据库文件的指针, 它的扩展名为.mdf。次数据文件是辅助主数据文件存储数据的, 它的扩展名为.ndf。事务日志文件保存了恢复数据库的所有日志信息, 它的扩展名为.ldf。虽然 SQL Server 2000 并不强制使用 .mdf、.ndf 和 .ldf 作为数据库文件的扩展名, 但仍建议使用这些扩展名以帮助标识文件的用途。

3. 事务日志文件和数据文件分开存放有什么好处?

事务日志文件和数据文件分开存放有如下好处:

1) 事务日志可以单独地备份。

2) 有可能从服务器失效的事件中将服务器恢复到最近的状态。

3) 事务日志不会抢占数据库的空间。

4) 可以很容易地监测事务日志的空间。

5) 在向数据文件和事务日志文件写入数据时会产生较少的冲突, 这有利于提高 SQL Server 的性能。

4. 数据库的更名必须在什么情况下进行?

一般情况下不要做数据库的改名, 如要更名需将数据库的并发用户数设为 1, 更名后再将数据库的并发用户数改回来。设置数据库并发用户的方法是: 用鼠标先点击要改名的数据库, 再依次点击“工具”、“SQL Server 配置属性”、“连接”, 在“并发用户连接的最大数目”输入框中输入相应的数字即可。

5. 事务日志文件的大小一般为数据文件大小的多少合适?

事务日志文件的大小可以在创建数据库时指定, 如果没有指定, 系统将自动创建一个日志文件, 该文件使用系统生成的名称, 大小为数据库中所有数据文件总大小的 25% 为合适。

6. 分别用企业管理器和 T-SQL 语言创建一个数据库。

略。

1.6 表的创建与管理

1. NULL 代表什么含义? 将其与其他值进行比较会产生什么结果? 如果数值型列中存在 NULL, 会产生什么结果?

在数据库中, NULL 是一个特殊值, 表示数值未知。NULL 不同于空字符或数字 0, 也不同于零长度字符串。比较两个空值或将空值与任何其他数值相比均返回未知, 这是因为每个空值均为未知。空值通常表示未知、不可用或以后添加数据。如果某个列上的空值属性为 NULL, 表示接受空值; 空值属性为 NOT NULL, 表示拒绝空值。如果数值型列中存在 NULL, 则在进行数据统计时就会产生不正确的结果。

2. 数据库表中的行有次序吗?

数据库表中的行或列均没有次序。

3. 使用 IDENTITY 列时应注意什么？

IDENTITY 属性使得某一列的取值是基于上一行的列值和为该列定义的步长自动生成的。IDENTITY 列的值可以惟一地标识表中的一行。

定义一个 IDENTITY 列，必须给出一个种子值（初始值），一个步长值（增量）。在一个 IDENTITY 列定义后，每当向表中插入一行数据时，IDENTITY 列就会自动产生下一个值。在插入数据行的语句中，不应包含 IDENTITY 列的值，该列的值应由系统自动给出。

定义 IDENTITY 列时，应注意如下几点：

1) 每张表只允许有一个 IDENTITY 列。

2) IDENTITY 列不能被更新。

3) IDENTITY 列不允许有 NULL 值。

4) IDENTITY 列只能用于具有下列数据类型之一的列：INT、SMALLINT、TINYINT、NUMERIC（小数部分为 0）、DECIMAL（小数部分为 0）。

5) 通过函数 `ident_seed (table_name)` 可获得 seed 值，通过 `ident_incr (table_name)` 可获得步长值，使用全局变量 `@@identity` 可以返回标识列的数据。

4. 使用 T-SQL 语句向表中插入数据应注意什么？

在使用 T-SQL 语句向表中插入数据时要注意以下几点：

1) 当向表中所有列都插入新数据时，可以省略列表名，但是必须保证 VALUES 后的各数据项位置同表定义时的顺序一致。

2) 要保证表定义时的非空列必须有值，即使这个非空列没有出现在插入语句中，也必须如此。

3) 插入字符型和日期型值时，要加入单引号。

4) 没有列出的数据类型应该具有以下属性之一：IDENTITY 属性、Timestamp 数据类型、具有 null 属性或者有一个缺省值。对于具有 IDENTITY 属性的列，其值由系统给出，用户不必往表中插入数据。

5. 什么是数据完整性？如果数据库不实施数据完整性会产生什么结果？

数据完整性是指数据的正确性、完备性和一致性，是衡量数据库质量好坏的重要标准。如果数据库不实施数据完整性，在用 INSERT、DELETE、UPDATE 语句修改数据库内容时，数据的完整性可能会遭到破坏，就可能会存在下列情况：无效的数据被添加到数据库的表中，如：将学生考试成绩输入成负数；对数据库的修改不一致，如：在一个表中修改了某学生的学号，但该学生的学号在另外一个表中却没有得到修改；将存在的数据修改为无效的数据，如：将某学生的班号修改为并不存在的班级号。

6. 数据完整性有哪几类？如何实施？它们分别在什么级别上实施？

数据完整性分为以下 3 类：

1) 域完整性：是指一个列的输入有效性，是否允许为空值。强制域完整性的方法有：限制类型（通过设定列的数据类型）、格式（通过 CHECK 约束和规则）或可能值的范围（通过 FOREIGN KEY 约束、CHECK 约束、DEFAULT 定义、NOT NULL 定义和规则）。如：学生的考试成绩必须在 0~100 之间，性别只能是“男”或“女”。

2) 实体完整性：是指保证表中所有的行惟一。实体完整性要求表中的所有行都有一个惟一标识符。这个惟一标识符可能是一列，也可能是几列的组合，称之为主键。也就是说，

表中的主键在所有行上必须取惟一值。强制实体完整性的方法有：索引、UNIQUE 约束、PRIMARY KEY 约束或 IDENTITY 属性。如：student 表中 student_id（学号）的取值必须惟一，它惟一标识了相应记录所代表的学生，学号重复是非法的。学生的姓名不能作为主键，因为完全可能存在两个学生同名同姓的情况。

3) 参照完整性：是指保证主关键字（被引用表）和外部关键字（引用表）之间的参照关系。它涉及两个或两个以上表数据的一致性维护。外键值将引用表中包含此外键的记录和被引用表中主键和外键相匹配的记录关联起来。在输入、更改或删除记录时，参照完整性保持表之间已定义的关系，确保键值在所有表中一致。这样的一致性要求确保不会引用不存在的值，如果键值更改了，那么在整个数据库中，对该键值的所有引用要进行一致的更改。参照完整性是基于外键与主键之间的关系。如学生学习课程的课程号必须是有效的课程号，表 student_course 外键 course_id 将参考表 course 中主键 course_id 以实现数据完整性。

域完整性、实体完整性及参照完整性分别在列、行、表上实施。数据完整性任何时候都可以实施，但对已有数据的表实施数据完整性时，系统要先检查表中的数据是否满足所实施的完整性，只有表中的数据满足了所实施的完整性，数据完整性才能实施成功。

7. 什么是主键约束？什么是惟一性约束？两者有什么区别？

主键约束保证某一列或一组列值的组合相对于表中的每一行都是惟一的，这些列就是该表的主键。主键不允许有重复值，也不允许有空值。

惟一约束限制表中指定列上所有的非空值必须惟一，即表中任意两行在指定列上都不允许有相同的值。

惟一约束和主键约束的区别是：

1) 惟一约束与主键约束都为指定的列建立惟一索引，即不允许惟一索引的列上有相同的值。主键约束限制更严格，不但不允许有重复值，而且也不允许有空值。

2) 惟一约束与主键约束产生的索引可以是聚簇索引也可以是非聚簇索引，但在缺省情况下惟一约束产生非聚簇索引，主键约束产生聚簇索引。

1.7 数据库安全性

1. SQL Server 有哪两种身份验证模式？

SQL Server 2000 有两种登录验证机制：Windows NT 验证机制和混合验证机制。

NT 验证模式是指要登录到 SQL Server 系统的用户身份由 NT 系统来进行验证。在这种方式下，用户不必提供登录名或密码让 SQL server 验证。

混合验证模式是指用户登录 SQL Server 系统时，其身份验证由 Windows NT 和 SQL Server 共同进行。

提供 SQL Server 身份验证是为了考虑非 Windows 客户及向后兼容，早期 SQL Server 的应用程序可能要求使用 SQL Server 和密码登录。当 SQL Server 实例在 Windows 98/Windows 2000 professional 上运行时，由于 Windows 98/Windows 2000 professional 不支持 Windows 身份验证模式，所以必须使用混合模式。非 Windows 客户端也必须使用 SQL Server 身份验证。

2. 写出 SQL Server 系统的登录验证过程。

略。见教材图 7-2。

3. 登录账户和数据库用户的关系如何？

登录名是访问 SQL Server 的通行证。登录名本身并不能让用户访问服务器中的数据库资源。要访问特定的数据库，还必须要有数据库用户名。新的登录创建以后，才能创建用户，用户在特定的数据库内创建，必须和一个登录名相关联。数据库用户的定义信息存放在与其相关的数据库的 sysusers 表中。这个表包含了该数据库的所有用户对象以及和它们相对应的登录名的标识。在大多数情况下，登录名和用户名使用相同的名称。

在 SQL Server 中，登录账户和数据库用户是 SQL Server 进行权限管理的两种不同的对象。一个登录账户可以与服务器上的所有数据库进行关联，而数据库用户是一个登录账户在某个数据库中的映射，也就是说一个登录账户可以映射到不同的数据库，产生多个数据库用户，而一个数据库用户只能映射到一个登录账户。允许数据库为每个用户对象分配不同的权限，这一特点为在组内分配权限提供了最大的自由度。为数据库用户授权的过程也就是为登录对象提供对数据库的访问权限的过程。

4. 数据库的角色有哪两种？固定数据库的角色能删除吗？

数据库角色分为固定数据库角色和自定义数据库角色。固定数据库角色不能删除。

5. 许可分为哪几种？分别给予解释。

在 SQL Server 2000 中，许可有三种类型：默认许可、对象许可和语句许可。

数据库中的用户根据它们在数据库中的角色被设定了某些缺省权限，这就是用户获得的默认许可。

对象许可是指用户基于数据库对象层次上的访问和操作权限，如果没有对象的许可，用户将不能访问该对象。对象许可有五种：查询、插入、修改、删除和执行。前 4 个许可是用于表和视图的，执行许可只用于存储过程。

语句许可通常授予需要在数据库中创建对象或修改对象、执行数据库和事务日志备份的用户。如果一个用户获得某个语句的许可，该用户就具有了执行该语句的权力。

6. 哪些用户可以管理许可？

有 4 种用户可以管理许可：

- 1) 系统管理员：有 sa 账户或具有相同权限的用户。
- 2) 数据库的属主：当前数据库的拥有者。
- 3) 对象的属主：当前对象的拥有者。
- 4) 数据库用户：不属于以上用户的其他用户。

1.8 数据检索

1. 在数据检索时 BETWEEN 关键字和 IN 关键字的适用对象是什么？

使用基于 BETWEEN 关键字的 WHERE 子句是为了对表中某一范围内的数据进行查询，系统将逐行检查表中的数据是否在 / 不在 BETWEEN 关键字设定的范围内。如果满足条件，则取出该行，如果不满足条件则不取出该行。BETWEEN 关键字一般应用于数字型数据。

使用基于 IN 关键字的 WHERE 子句对表中数据进行查询，系统将逐行检查表中的数据

是否在 / 不在 IN 关键字设定的列表内。如果满足条件，则取出该行，如果不满足条件则不取出该行。IN 关键字一般应用于字符型数据。

2. LIKE 匹配字符有哪几种？如果要检索的字符中包含匹配字符，那么该如何处理？

在 SQL Server 2000 中，共提供了 4 个通配符：

- 1) %：代表任意多个字符。
- 2) _（下划线）：代表一个任意字符。
- 3) []：代表方括号内的任意一个字符。
- 4) [^]：表示任意一个在方括号内没有的字符。

如果要检索的字符中包含匹配字符，则可以使用转义字符。在 ESCAPE 子句中，定义“b”为转义字符，这样，LIKE 子句中紧跟字符“b”后面的字符将被定义为匹配字符而不再是通配符。

3. 要使查询的结果有序显示，应使用什么语句？

要使查询的结果有序显示，应使用 ORDER BY 子句来改变查询结果的显示顺序。

4. 数据检索时使用 COMPUTE 和 COMPUTE BY 产生的结果有何不同？

使用 COMPUTE 子句和 COMPUTE BY 子句都能既浏览明细数据又看到统计的结果。只使用 COMPUTE 子句时，其查询的结果类似于总计，而使用 COMPUTE BY 子句时，其查询的结果将为带具体内容的分类进行统计。

5. 进行连接查询时应注意什么？

连接查询是指以指定表中的某个列或某些列作为连接条件，从两个或更多的表中查询关联数据的查询。有两种连接查询的形式：用于 FORM 子句的 ANSI 连接语法形式和 SQL Server 语法形式。

- 1) 一般而言，基于主键和外键指定查询条件，连接条件可使用“主键=外键”。
- 2) 如果一个表有复合关键字，在连接表时，必须引用整个关键字。
- 3) 应尽可能限制连接语句中表的数目，连接的表越多，查询处理的时间越长。
- 4) 对于连接表的两个列应有相同或类似的数据类型。
- 5) 不要使用空值作为连接条件，因为空值计算不会和其他任何值相等。
6. 什么是分布式查询？

使用 OLE DB 支持，访问来自多种异类数据源的数据，而这些数据可存储在相同或不同的服务器上。

SQL Server 2000 分布式查询支持用户访问：

- 1) 存储在多个 SQL Server 实例中的分布式数据。
- 2) 存储在各种可使用 OLE DB 提供程序访问的关系和非关系数据源中的异类数据。

1.9 视图

1. 什么是视图？使用视图的优点和缺点是什么？

视图是一个由 SELECT 语句指定，用以检索数据库表中某些行或列数据的语句存储定义。从本质上说，视图其实是一种 SQL 查询。

使用视图的优点：