

《数据库原理与应用》（第三版）习题参考答案

第1章 数据库概述

1. 试说明数据、数据库、数据库管理系统和数据库系统的概念。

答：数据是描述事物的符号记录。

数据库是长期存储在计算机中的有组织的、可共享的大量数据的集合。

数据库管理系统是一个专门用于实现对数据进行管理和维护的系统软件。

数据库系统是指在计算机中引入数据库后的系统，一般由数据库、数据库管理系统（及相关的实用工具）、应用程序、数据库管理员组成。

2. 数据管理技术的发展主要经历了哪几个阶段？

答：文件管理和数据库管理。

3. 与文件管理相比，数据库管理有哪些优点？

答：与文件系统管理数据相比，数据库系统管理数据带来了如下好处：将相互关联的数据集成在一起，较少的数据冗余，程序与数据相互独立，保证数据的安全可靠，最大限度地保证数据的正确性，数据可以共享并能保证数据的一致性。

4. 在数据库管理方式中，应用程序是否需要关心数据的存储位置和存储结构？为什么？

答：不需要。因为数据库管理系统提供了逻辑独立性和物理独立性。

5. 在数据库系统中，数据库的作用是什么？

答：数据库是数据的汇集，它以一定的组织形式保存在存储介质上。

6. 在数据库系统中，应用程序可以不通过数据库管理系统而直接访问数据文件吗？

答：不能

7. 数据独立性指的是什么？它能带来哪些好处？

答：数据独立性是指应用程序不会因数据的物理表示方式和访问技术的改变而改变，即应用程序不依赖于任何特定的物理表示方式和访问技术，它包含两个方面：逻辑独立性和物理独立性。

物理独立性是指当数据的存储位置或存储结构发生变化时，不影响应用程序的特性；

逻辑独立性是指当表达现实世界的信息内容发生变化时，不影响应用程序的特性。

8. 数据库系统由哪几部分组成，每一部分在数据库系统中的作用大致是什么？

答：数据库系统一般包括数据库、数据库管理系统（及相应的实用工具）、应用程序和数据库管理员四个部分。数据库是数据的汇集，它以一定的组织形式保存在存储介质上；数据库管理系统是管理数据库的系统软件，它可以实现数据库系统的各种功能；应用程序专指以数据库数据为基础的程序，数据库管理员负责整个数据库系统的正常运行。

第2章 数据模型与数据库结构

1. 解释数据模型的概念，为什么要将数据模型分成两个层次？

答：数据模型是对现实世界数据特征的抽象。数据模型一般要满足三个条件：第一是数据模型要能够比较真实地模拟现实世界；第二是数据模型要容易被人们理解；第三是数据模型要能够很方便地在计算机上实现。由于用一种模型来同时很好地满足这三方面的要求在目前是比较困难的，因此在数据库系统中就可以针对不同的使用对象和应用目的，采用不同的数据模型。根据模型应用的不同目的，将这些模型分为两大类：概念层数据模型和组织层数据模型，以方便对信息的描述。

2. 概念层数据模型和组织层数据模型分别是针对什么进行的抽象？

答：概念层数据模型是对现实世界的抽象，形成信息世界模型，组织层数据模型是对信息世界进行抽象和转换，形成具体的DBMS支持的数据组织模型。

3. 实体之间的联系有哪几种？请为每一种联系举出一个例子。

答：实体之间的联系有一对一、一对多和多对多三种。例如：系和正系主任是一对一联系（假设一个系只有一个正系主任），系和教师是一对多联系（假设一个教师只在一个系工作），教师和课程是多对多联系（假设一个教师可以讲授多门课程，一门课程可由多个教师讲授）。

4. 说明实体-联系模型中的实体、属性和联系的概念。

5. 指明下列实体间联系的种类：

- (1) 教研室和教师（设一个教师只属于一个教研室，一个教研室可有多名教师）。
- (2) 商品和顾客。
- (3) 国家和首都（假设一个国家的首都可以变化）。
- (4) 飞机和乘客。
- (5) 银行和账户。
- (6) 图书和借阅者。（设一个借阅者可同时借阅多本书，可在不同时间对同一本书借阅多次）

6. 数据库系统包含哪三级模式？试分别说明每一级模式的作用？

答：数据库系统包含的三级模式为：内模式、模式和外模式。外模式是对现实系统中用户感兴趣的整体数据结构的局部描述，用于满足不同数据库用户需求的数据视图，是数据库用户能够看见和使用的局部数据的逻辑结构和特征的描述，是对数据库整体数据结构的子集或局部重构。模式是数据库中全体数据的逻辑结构和特征的描述，是所有用户的公共数据视图。内模式是对整个数据库的底层表示，它描述了数据的存储结构。

7. 数据库管理系统提供的两级映像的作用是什么？它带来了哪些功能？

答：数据库系统的两级映象是模式与内描述间的映象和外模式与模式间的映象。模式/内模式的映象带来了物理独立性，即如果数据库的存储结构改变了，可通过调整模式/内模式的映象，使模式能够保持不变。外模式/概念模式间的映象带来了逻辑独立性，当概念模式的结构可发生改变时，也可通过调整外模式/模式间的映象关系，使外模式可以保持不变。

8. 数据库三级模式划分的优点是什么？它能带来哪些数据独立性？

答：数据库的三级模式的划分实际上将用户、逻辑数据库与物理数据库进行了划分，使彼此之间的相互干扰减到最少。这三个模式的划分实际上带来了两个数据独立性：物理独立性和逻辑独立性。这使得底层的修改和变化尽量不影响到上层。

第3章 关系数据库

1. 试述关系模型的三个组成部分。

答：关系数据结构、关系操作集合和关系完整性约束。

2. 解释下列术语的含义：

(1) 笛卡尔积：设 D_1, D_2, \dots, D_n 为任意集合，定义笛卡尔积 D_1, D_2, \dots, D_n 为： $D_1 \times D_2 \times \dots \times D_n = \{(d_1, d_2, \dots, d_n) \mid d_i \in D_i, i=1, 2, \dots, n\}$

(2) 主键：也称主码为或主关键字，是表中的属性或属性组，用于唯一地确定一个元组。

(3) 候选键：如果一个属性或属性集的值能够唯一标识一个关系的元组而又不包含多余的属性，则称该属性或属性集为候选键。

(4) 外键：设 F 是关系 R 的一个或一组属性，如果 F 与关系 S 的主键相对应，则称 F 是关系 R 的外键。

(5) 关系：关系就是简单二维表。

(6) 关系模式：二维表的结构称为关系模式。

(7) 关系数据库：对应于一个关系模型的所有关系的集合称为关系数据库。

3. 关系数据库的三个完整性约束是什么？各是什么含义？

答：实体完整性、参照完整性和用户定义的完整性。

实体完整性是保证关系中的每个元组都是可识别的和唯一的。

参照完整性也称为引用完整性，用于表达现实世界中的实体之间的关联关系。

用户定义的完整性也称为域完整性或语义完整性，用于保证数据库中存储的值与现实世界相符。

4. 连接运算有哪些？等值连接和自然连接的区别是什么？

答：连接运算中最重要也是最常用的连接有两个，一个是等值连接，一个是自然连接。

自然连接与等值连接的差别为：

- 自然连接要求相等的分量必须有共同的属性名，等值连接则不要求；
- 自然连接要求把重复的属性名去掉，等值连接却不这样做。

5. 对参与并、交、差运算的两个关系 R, S 有什么要求？

答：必须结构相同，且相应的属性值取自同一个值域。

6. 对参与除运算的两个关系 $(R \div S)$ 有什么要求？除运算的结果关系中包含哪些属性？

答： R 中必须包含 S 的全部或部分属性，除运算的结果包含的属性是只属于 R 不属于 S 的属性。

7. 对参与自然连接和等值连接操作的两个关系R、S有什么要求？

答：有语义相同的属性。

8. 投影操作的结果关系中是否有可能存在重复的记录？为什么？

答：不可能，因为投影运算会自动去掉投影后重复的记录。

9. 利用表3-10至3-12所示的三个关系，写出实现如下查询要求的代数表达式。

(1) 查询“信息系”学生的选课情况，列出学号、姓名、课程号和成绩。

$\Pi_{Sno, Sname, Cno, Grade} (\sigma_{Sdept='信息系'} (SC \text{ Student}))$

或： $\Pi_{Sno, Sname, Cno, Grade} (SC \sigma_{Sdept='信息系'} (Student))$

(2) 查询“VB”课程的考试情况，列出学生姓名、所在系和考试成绩。

$\Pi_{Sname, Sdept, Grade} (\sigma_{Cname='VB'} (Coure \text{ SC Student}))$

或： $\Pi_{Sname, Sdept, Grade} (\sigma_{Cname='VB'} (Coure) \text{ SC Student})$

(3) 查询考试成绩高于90分的学生的姓名、课程名和成绩。

$\Pi_{Sname, Cname, Grade} (\sigma_{Grade>90} (Coure \text{ SC Student}))$

或： $\Pi_{Sname, Cname, Grade} (Coure \sigma_{Grade>90} (SC) \text{ Student})$

(4) 查询至少选修了0512101号学生所选的全部课程的学生姓名和所在系。

$\Pi_{Sname, Sdept} (Student (SC \div \Pi_{Cno} (\sigma_{sno='0512101'} (SC))))$

(5) 查询至少选了“C01”和“C02”两门课程的学生的姓名、所在系和所选的课程号。

$\Pi_{Sname, Sdept, Cno} (Student (SC \div \Pi_{Sno} (\sigma_{cno='C01' \vee Cno='C02'} (SC))))$

(6) 查询没有选修第1学期开设的全部课程的学生学号、姓名和所选的课程号。

$\Pi_{Sno, Sname, Cno} (Student \text{ SC} (\Pi_{sno} (SC) - \Pi_{sno} (\sigma_{semester=1} (Course) \text{ SC})))$

(7) 查询计算机系和信息系选了VB课程的学生姓名。

$\Pi_{Sname} (\sigma_{sdept}$

第4章 SQL Server 2012基础

1. 安装SQL Server 2012对硬盘及内存的要求分别是什么？

答：SQL Server 2012实际硬盘空间需求取决于系统配置和您决定安装的功能，一般应确保系统驱动器中是否有至少 6.0 GB 的可用磁盘空间。

内存：Express 版本：最少512 MB；其他版本：最少1 GB。

2. SQL Server实例的含义是什么？实例名的作用是什么？

答：一个实例代表一个独立的数据库管理系统。实例名是实例的标识，用户通过“计算机名/实例名”的方式可访问指定的命令实例。

3. SQL Server 2012的核心引擎是什么？

答：SQL Server (MSSQLSERVER)

4. SQL Server 2012提供的设置服务启动方式的工具是哪个？

答：配置管理器

5. 在SQL Server 2012中，每个数据库至少包含几个文件？

答：2个文件，一个数据文件，一个日志文件

6. SQL Server 2012数据库文件分为几类？每个文件有哪些属性？

答：主要数据文件和次要数据文件，属性有：物理文件名及其位置、逻辑文件名、初始大小、增长方式、最大大小。

第5章 数据类型及关系表创建

1. Tinyint数据类型定义的数据的取值范围是多少？

答：0-255

2. SmallDatetime类型精确到哪个时间单位？

答：分钟

3. 定点小数类型numeric中的p和q的含义分别是什么？

答：p代表整数位数+小数位数的和值，q代表小数位数。

4. Char(n)、nchar(n)的区别是什么？它们各能存放多少个字符？

答：Char(n)中的n代表能存放n个字节的字符，如果是字母可以存放n个，如果是汉字可存放n/2个。Nchar(n)中的n代表能存放的字符个数。

5. Char(n)和varchar(n)的区别是什么？

答：Char(n)是定长存储，一定占n个字节的空间。Varchar(n)代表最多占n个字节的空间。

6. 数据完整性约束的作用对象有哪些？

答：表和列

7. CHECK约束的作用是什么？

答：限制列的取值范围。

8. UNIQUE约束的作用是什么？

答：限制列取值不重。

9. DEFAULT约束的作用是什么？

答：提供列的默认值。

上机练习

1. 在第4章创建的Students数据库中, 写出创建如下三张表的SQL语句, 要求在定义表的同时定义数据的完整性约束:

(1) “图书”表结构如下:

书号: 统一字符编码定长类型, 长度为6, 主键;

书名: 统一字符编码可变长类型, 长度为30, 非空;

第一作者: 普通编码定长字符类型, 长度为10, 非空;

出版日期: 小日期时间型;

价格: 定点小数, 小数部分1位, 整数部分3位。

```
create table 图书(  
书号 nchar(6) primary key,  
书名 nvarchar(30) not null,  
第一作者 char(10) not null,  
出版日期 smalldate,  
价格 numeric(4,1)  
)
```

(2) “书店”表结构如下:

书店编号: 统一字符编码定长类型, 长度为6, 主键;

店名: 统一字符编码可变长类型, 长度为30, 非空;

电话: 普通编码定长字符类型, 8位长, 每一位的取值均是0~9的数字;

地址: 普通编码可变长字符类型, 40位长。

邮政编码: 普通编码定长字符类型, 6位长。

```
create table 书店(  
书店编号 nchar(6) primary key,  
店名 nvarchar(30) not null  
电话 char(8) check(电话 like '[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]'),  
地址 varchar(40),  
邮政编码 char(6)  
)
```

(3) “图书销售”表结构如下:

书号: 统一字符编码定长类型, 长度为6, 非空;

书店编号: 统一字符编码定长类型, 长度为6, 非空;

销售日期: 小日期时间型, 非空;

销售数量: 小整型, 大于等于1。

主键为(书号, 书店编号, 销售日期);

其中“书号”为引用“图书表”的“书号”的外键;

“书店编号”为引用“书店表”的“书店编号”的外键。

```
create table 图书销售(  
书号 nchar(6) not null,  
书店编号 nchar(6) not null,  
销售日期 smalldate,  
销售数量 smallint check(销售数量 >= 1),  
primary key(书号, 书店编号, 销售日期),  
foreign key(书号) references 图书(书号),  
foreign key(书店编号) references 书店(书店编号)  
)
```

2. 为图书表添加“印刷数量”列, 类型为整数, 同时添加取值大于等于1000的约束。

```
Alter table 图书 add 印刷数量 int check(印刷数量 >= 1000)
```

3. 删除书店表中的“邮政编码”列。

```
Alter table 书店 drop column 邮政编码
```

4. 将图书销售表中的“销售数量”列的数据类型改为整型。

```
Alter table 图书 alter column 销售数量 int
```

第6章 数据操作语句

1. 简单说明SELECT语句中，FROM、WHERE、GROUP BY、HAVING子句的作用。

答：FROM子句指定数据来自的表，WHERE子句指定行数据的筛选条件，GROUP BY子句指定用于分组的列，HAVING子句用于对分组后的统计值进行筛选。

2. 简单说明COUNT(*)与COUNT(列名)的区别。

答：COUNT()用于统计行的个数，并且不忽略NULL；COUNT(列名)用于统计该列中非空值个数，这个函数会去掉NULL。

3. 在聚合函数中，不忽略空值的函数是哪个？

答：COUNT(*)

4. 哪些数据类型的列可以使用SUM(列名)和AVG(列名)函数？

答：数值类型的列

5. 外连接和内连接的区别是什么？

答：外连接结果会包含连接中一个表的全部数据，内连接结果是只包含两个表中满足连接条件的数据。

6. TOP子句的作用是什么？

答：限定取查询结果中的前若干行数据。

7. DISTINCT子句的作用是去掉表中的重复行数据，这个说法对吗？

答：应该是去掉查询结果中的重复行数据。

8. 简单说明嵌套子查询的执行顺序。

答：嵌套子查询是先执行子查询，然后在根据子查询结果执行外层查询。

上机练习

1. 查询学生选课表中的全部数据。

答：select * from SC

2. 查询计算机系的学生姓名、年龄。

答：select sname, sage from student where sdept = '计算机系'

3. 查询成绩在70~80分之间的学生学号、课程号和成绩。

答：select sno,cno,grade from sc
where grade between 70 and 80

4. 查询计算机系年龄在18~20且性别为“男”的学生姓名、年龄。

答：select sname,sage from student
where sdept = '计算机系'
and sage between 18 and 20
and ssex = '男'

5. 查询“c01”课程最高分。

答：select max(grade) from sc where cno = 'C01'

6. 查询计算机系学生的最大年龄和最小年龄。

答：select max(sage) as max_age, min(sage) as min_age
from student
where sdept = '计算机系'

7. 统计每个系的学生人数。

答：select sdept, count(*) 学生人数 from student
group by sdept

8. 统计每个学生的选课门数和考试总成绩，并按选课门数升序显示结果（不包括没选课的学生）。

答:

```
select sno, count(*) 选课门数, sum(grade) 考试总成绩
from sc
group by sno
order by count(*) asc
```

9. 查询总成绩超过200分的学生，列出学号、总成绩。

答: select sno,sum(grade) 总成绩 from sc

```
group by sno
having sum(grade) > 200
```

10. 查询选了“c02”课程的学生姓名和所在系。

答: select sname,sdept

```
from student s join sc on s.sno = sc.sno
where cno = 'C02'
```

11. 查询成绩80分以上的学生姓名、课程号和成绩，并按成绩降序排列结果。

答: select sname,cno,grade

```
from student s join sc on s.sno = sc.sno
where grade > 80 order by grade desc
```

12. 查询哪些学生没有选课，要求列出学号、姓名和所在系。

答: select s.sno, sname, sdept

```
from student s left join sc on s.sno = sc.sno
Where sc.sno is null
```

13. 统计每门课程的选课人数，列出课程号和选课人数。（包括没人选的课程）

答: select c.cno,count(sc.cno) 选课人数

```
from course c left join sc on c.cno = sc.cno
Group by c.cno
```

14. 查询与VB在同一学期开设的课程的课程名和开课学期。

答: select c2.cname,c2.semester

```
from course c1 join course c2 on c1.semester = c2.semester
Where c1.cname = 'VB' and c1.cname != 'VB'
```

15. 查询与李勇年龄相同的学生的姓名、所在系和年龄。

答: select S2.Sname,S2.Sdept, S2.sage

```
from student s1 join student s2 on s1.sage = s2.sage
Where s1.sname = '李勇' and s2.sname != '李勇'
```

16. 查询计算机系年龄最小的2名学生的姓名和年龄。

答: select top 2 with ties sname, sage

```
from student
Where sdept = '计算机系'
Order by sage asc
```

17. 查询VB成绩最高的前2名学生的姓名、所在系和VB成绩，包括并列的情况。

答: select top 2 with ties sname, sdept, grade

```
from student s join sc on s.sno = sc.sno
Join course c on c.cno = sc.cno
```

```
Where cname = 'VB'
Order by grade desc
```

18. 查询选课门数最多的前2名学生的学号和选课门数, 包括并列的情况。

```
答: select top 2 with ties sno, count(*)选课门数
from sc
Group by sno
Order by count(*) desc
```

19. 查询学生人数最多的系, 列出系名和人数。

```
答: select top 1 with ties sdept, count(*)人数
from student
Group by sdept
Order by count(*)desc
```

20. 用子查询实现如下查询:

(1) 查询选修了“c01”号课程的学生姓名和所在系。

```
答: select sname, sdept from student where sno in(
select sno from sc where cno = ' C01' )
```

(2) 查询数学系成绩80分以上的学生学号、姓名、课程号和成绩。

```
答: select sno,sname,cno,grade
from student s join sc on s.sno = sc.sno
where s.sno in(
select sno from sc where grade > 80)
and sdept = '数学系'
```

(3) 查询计算机系考试成绩最高的学生姓名。

```
答: select sname from student
where sno in(
select top 1 sno from sc join student s on s.sno = sc.sno
where sdept = '计算机系'
Order by grade desc )
```

(4) 查询数据结构考试成绩最高的学生姓名、所在系、性别和成绩。

```
答: select sname,sdept,ssex,grade
from student s join sc on s.sno = sc.sno
where s.sno in(
Select top 1 sno from sc join course c on c.cno = sc.cno
Where cname = '数据结构'
Order by grade desc )
```

21. 查询没选VB课程的学生姓名和所在系。

```
答: select sname,sdept from student
Where sno not in (
Select sno from sc join course c on c.cno = sc.cno
Where cname = 'VB' )
```

22. 查询计算机系没选课的学生的姓名和性别。

```
答: select sname,ssex from student
Where sdept = '计算机系'
And sno not in (
Select sno from sc )
```


23. 查询计算机系考试平均成绩最低的学生的姓名及所选的课程名。

```
答: select sname,cname from student s join sc on s.sno = sc.sno
Join course c on c.cno = sc.cno
where s.sno in (
Select top 1 sno from sc
Where sdept = '计算机系'
Group by sno
Order by avg(grade) asc )
```

24. 查询1~5学期中, 选课人数最少的课程的课程名、开课学期和学分。

```
答: select cname,semester,credit from course
Where semester between 1 and 5
And cno in (
Select top 1 cno from sc
Group by cno
Order by count(*) desc )
```

25. 查询计算机系每个学生的考试情况, 列出姓名、课程名和考试成绩, 并将查询结果保存到一个新表中。新表名为: Computer_Dept

```
答: select sname,cname,grade into Computer_Dept
from student s join sc on s.sno = sc.sno
Join course c on c.cno = sc.cno
Where sdept = '计算机系'
```

26. 创建一个新表, 表名为test_t, 其结构为: (COL1, COL 2, COL 3), 其中:

COL1: 整型, 允许空值。

COL2: 普通编码字符型, 长度为10, 不允许空值。

COL3: 普通编码字符型, 长度为10, 允许空值。

试写出按行插入如下数据的语句(空白处表示空值)。

COL1	COL2	COL3
	B1	
1	B2	C2

答: create table test_t(

Col1 int,

Col2 char(10) not null,

Col3 char(10)

)

Insert into test_t values(null, 'B1' ,null)

Insert into test_t values(1, 'B2' , 'C2')

Insert into test_t values(2, 'B3' ,null)

27. 删除考试成绩低于50分的学生的选课记录。

```
答: delete from sc where grade < 50
```

28. 删除没有人选的课程。

```
答: delete from course where cno not in(
Select cno from sc )
```

29. 删除计算机系VB成绩不及格学生的VB选课记录。

答: delete from sc

From student s join sc on s.sno = sc.cno

Join course c on c.cno = sc.cno

where sdept = '计算机系' and cname = 'VB'

and grade < 60

30. 删除VB考试成绩最低的学生的VB选课记录。

答: delete from sc

From course c join sc on c.cno = sc.cno

Where cname = 'VB'

And grade = (

Select min(grade) from sc

Join course c on c.cno = sc.cno

Where cname = 'VB')

31. 将第2学期开设的所有课程的学分增加2分。

答: update course set credit = credit + 2

Where semester = 2

32. 将VB课程的学分改为3分。

答: update course set credit = 3

Where cname = 'VB'

33. 将计算机系学生的年龄增加1岁。

答: update student set sage = sage + 1

where sdept = '计算机系'

34. 将信息系学生的“计算机文化学”课程的考试成绩加5分。

答: update sc set grade = grade + 5

From student s join sc on s.sno = sc.sno

Join course c on c.cno = sc.cno

Where sdept = '信息系'

And cname = '计算机文化学'

35. 将选课人数最少的课程的学分降低1分。

答: update course set credit

第7章 索引和视图

1. 索引的作用是什么?

答: 索引可以加快数据的查询效率。

2. 索引分为哪几种类型? 分别是什么? 它们的主要区别是什么?

答: 分为聚集索引和非聚集索引两种。聚集索引会对数据进行物理排序, 非聚集索引不对数据进行物理排序。

3. 在一个表上可以创建几个聚集索引? 可以创建多个非聚集索引吗?

答: 1个聚集索引。可以。

4. 聚集索引一定是唯一性索引, 对吗? 反之呢?

答: 不对。反之也不对。

5. 在建立聚集索引时, 数据库管理系统是真正将数据按聚集索引列进行物理排序。对吗?

答: 对。

6. 在建立非聚集索引时, 数据库管理系统并不对数据进行物理排序。对吗?

答: 对。

7. 不管对表进行什么类型的操作, 在表上建立的索引越多越能提高操作效率。对吗?

答: 不对。

8. 经常对表进行哪类操作适合建立索引? 适合在哪些列上建立索引?

答:

- 返回范围值的查询: BETWEEN AND、>、>=、< 和 <= ;
- 不返回大型结果集的查询。
- 经常被用作连接的列, 一般来说, 这些列是外键列。
- ORDER BY或GROUP BY操作。
适合建立索引的情况:
- 包含大量非重复值的列。
- 在WHERE子句中经常用于进行BETWEEN AND、>、>=、< 和 <=等操作的列。
- 经常被用作连接操作的列。
- ORDER BY或GROUP BY子句中涉及的列。

9. 使用第5章建立的Student、Course和SC表, 写出实现下列操作的SQL语句。

(1) 在Student表上为Sname列建立一个非聚集索引, 索引名为: Snameldx。

答: Create index Snameldx on student(Sname)

(2) 在Course表上为Cname列建立一个唯一的非聚集索引, 索引名为: CNIdx

答: Create unique index CNIdx on Course(Cname)

(3) 在SC表上为Sno和Cno建立一个组合的聚集索引, 索引名为: SnoCnoldx。

答: Create clustered index SnoCnoldx on SC(Sno,Cno)

(4) 删除Sname列上建立的Snoldx索引。

答: drop index Snoldx on Student

10. 试说明使用视图的好处。

答: 利用视图可以简化客户端的数据查询语句, 使用户能从多角度看待同一数据, 可以提高数据的安全性, 视图对应数据库三级模式中的外模式, 因此提供了一定程度的逻辑独立性。

11. 使用视图可以加快数据的查询速度, 这句话对吗? 为什么?

答: 不对, 因为通过视图查询数据时, 比直接针对基本表查询数据多了一个转换过程, 即从外模式到模式的转换。

12. 使用第5章建立的Student、Course和SC表, 写出创建满足下述要求的视图的SQL语句。

(1) 查询学生的学号、姓名、所在系、课程号、课程名、课程学分。

答: Create view v1 As

```
Select s.sno,sname,sdept,c.cno,cname,credit
```

```
From student s join sc on s.sno = sc.sno
```

```
Join course c on c.cno = sc.cno
```

(2) 查询学生的学号、姓名、选修的课程名和考试成绩。

答: Create view v2 As

```
Select s.sno,sname,cname,grade
```

```
From student s join sc on s.sno = sc.sno
```

```
Join course c on c.cno = sc.cno
```

(3) 统计每个学生的选课门数, 要求列出学生学号和选课门数。

答: Create view v3 As

```
Select sno,count(*) as total
```

```
From sc group by sno
```

(4) 统计每个学生的选课总学分，要求列出学生学号和总学分（说明：考试成绩大于等于60才可获得此门课程的学分）。

答: Create view v4 As

Select sno,sum(credit) as total_credit

From sno join course c on c.cno = sc.cno

Where grade >= 60

Group by sno

13. 利用第12题建立的视图，完成如下查询：

(1) 查询考试成绩大于等于90分的学生的姓名、课程名和成绩。

答: Select sname,cname,grade From v2 where grade >= 90

(2) 查询选课门数超过3门的学生的学号和选课门数。

答: Select * from v3 where total >= 3

(3) 查询计算机系选课门数超过3门的学生的姓名和选课门数。

答: Select sname,total from v3 join student s on s.sno = v3.sno

Where sdept = '计算机系' and total >= 3

(4) 查询选课总学分超过10分的学生的学号、姓名、所在系和选课总学分。

答: Select v4.sno,sname,sdept,total_credit

From v4 join student s on s.sno = v4.sno

Where total_credit >= 10

(5) 查询年龄大于等于20岁的学生中，选课总学分超过10分的学生的姓名、年龄、所在系和选课总学分。

答: Select sname,sage,sdept,total_credit

From v4 join student s on s.sno = v4.sno

Where sage >= 20 and total_credit >= 10

14. 修改12题(4)定义的视图，使其查询每个学生的学号、平均成绩以及总的选课门数。

答: Alter view v4 As

Select sno,avg(grade) as avg_grade,count(*) as total_cno

Group by sno

第8章 关系数据库理论

1. 关系规范化中的操作异常有哪些？它是由什么引起的？解决的办法是什么？

答: 增、删、改异常。数据冗余引起。解决办法: 模式分解。

2. 第一范式、第二范式和第三范式的定义分别是什么？

答: 第一范式: 每个列都是原子项。第二范式: 第一范式且不含部分函数依赖。第三范式: 第二范式且不含传递函数依赖。

3. 什么是部分函数依赖？什么是传递函数依赖？请举例说明。

答: 如果 $X \rightarrow Y$ ，并且对于 X 的一个任意真子集 X' 都有 $X' \not\rightarrow Y$ ，则称 Y 完全函数依赖于 X ，记作 $X \twoheadrightarrow Y$ ；如果 $X' \rightarrow Y$ 成立，则称 Y 部分函数依赖于 X ，记作 $X \rightharpoonup Y$ 。

示例: 学生选课(学号, 姓名, 性别, 选的课程号, 课程名, 考试成绩)

主键: (学号, 课程号)

因为学号 \rightarrow 姓名, 因此(学号, 课程号) \twoheadrightarrow 姓名

4. 第三范式的关系模式是否一定不包含部分函数依赖？

答: 是

5. 对于主键只由一个属性组成的关系模式，如果它是第一范式的，则它是否一定也是第二范式的？

答: 是。

6. 设有关系模式：学生选课（学号，姓名，所在系，性别，课程号，课程名，学分，成绩）。设一个学生可以选多门课程，一门课程可以被多名学生选。一个学生有唯一的所在系，每门课程有唯一的课程名和学分。请指出此关系模式的候选键，判断此关系模式是第几范式的，若不是第三范式的，请将其规范化为第三范式关系模式，并指出分解后的每个关系模式的主键和外键。

答：

(1) 候选码：（学号，课程号）

(2) 学号→姓名，学号→所在系，学号→性别，课程号→课程名，课程号→学分，
（学号，课程号）→成绩

(3) 属于第二范式，因为存在部分函数依赖：学号→姓名。

(4) 第三范式关系模式：

学生（学号，姓名，所在系，性别）

课程（课程号，课程名，学分）

考试（学号，课程号，成绩），学号为引用学生的外码，课程号为引用课程的外码。

7. 设有关系模式：学生（学号，姓名，所在系，班号，班主任，系主任），其语义为：一个学生只在一个系的一个班学习，一个系只有一个系主任，一个班只有一名班主任，一个系可以有多个班。请指出此关系模式的候选键，判断此关系模式是第几范式的，若不是第三范式的，请将其规范化为第三范式关系模式，并指出分解后的每个关系模式的主键和外键。

答：

(1) 候选码：学号

(2) 学号→姓名，学号→所在系，学号→班号，班号→班主任，所在系→系主任

(3) 第二范式，因为有：学号→班号，班号→班主任，因此存在传递函数依赖：

学号 班主任

(4) 第三范式关系模式：

学生（学号，姓名，所在系，班号），班号为引用班的外码，所在系为引用系的外码。

班（班号，班主任）

系（系名，系主任）

8. 设有关系模式：教师授课（课程号，课程名，学分，授课教师号，教师名，授课时数），其语义为：一门课程（由课程号决定）有确定的课程名和学分，每名教师（由教师号决定）有确定的教师名，每门课程可以由多名教师讲授，每名教师也可以讲授多门课程，每名教师对每门课程有确定的授课时数。指出此关系模式的候选键，判断此关系模式属于第几范式，若不属于第三范式，请将其规范化为第三范式关系模式，并指出分解后的每个关系模式的主键和外键。

答：

(1) 候选码：（课程号，授课教师号）

(2) 课程号→课程名，课程号→学分，授课教师号→教师名，（课程号，授课教师号）→授课时数

(3) 属于第一范式。因为有：课程号→课程名，因此存在部分函数依赖关系：

（课程号，授课教师号）课程名

(4) 第三范式关系模式：

课程（课程号，课程名，学分）

教师（教师号，教师名）

授课（课程号，教师号，授课时数），课程号为引用课程的外码，教师号为引用教师的外码。

第9章 事务与并发控制

1. 试说明事务的概念及四个特征。

答：原子性、隔离性、一致性、持久性。

2. 事务处理模型有哪两种？

答：T-SQL事务处理模型、T-SQL事务处理模型。

3. 在数据库中为什么要有并发控制？

答：为避免多个事务之间的相互干扰。

4. 并发控制的措施是什么？

答：加锁。

5. 设有三个事务：T1、T2和T3，其所包含的动作为：

T1: $A = A + 2$; T2: $A = A * 2$; T3: $A = A - 1$

设A的初值为3，若这三个事务并行执行，则可能的调度策略有几种？A的最终结果分别是什么？

答：

T1->T2->T3: $A = 9$

T1->T3->T2: $A = 8$

T2->T1->T3: $A = 7$

T2->T3->T1: $A = 9$

T3->T1->T2: $A = 8$

T3->T2->T1: $A = 6$

5. 当某个事务对某段数据加了S锁之后，在此事务释放锁之前，其他事务还可以对此段数据添加什么锁？

答：S锁

6. 什么是死锁？预防死锁有哪些方法？

答：相互等待对方释放资源。

7. 如何诊断和解除死锁？

答：诊断死锁方法：一般使用超时法和事务等待图法。

解除死锁：通常采用的方法是选择一个处理死锁代价最小的事务，将其撤销，释放此事务所持有的全部锁，使其他事务可以继续运行下去。

8. 怎样保证多个事务的并发执行是正确的？

答：遵守两阶段锁协议。

9. 一级封锁协议对读和写分别加什么锁？加锁范围分别是什么？能避免哪些干扰？

答：写全程加X锁，读不加锁。能避免丢失修改。

10. 二级封锁协议对读和写分别加什么锁？加锁范围分别是什么？能避免哪些干扰？

答：写全程加X锁，读前加S锁，读完释放S锁。能避免丢失修改和不读脏数据。

11. 三级封锁协议对读和写分别加什么锁？加锁范围分别是什么？能避免哪些干扰？

答：写全程加X锁，读全程加S锁。能避免丢失修改、不读脏数据和不可重复读。

第10章 数据库设计

1. 试说明数据库设计的特点。

答：综合性、结构设计和行为设计相分离。

2. 简述数据库的设计过程。

答：设计分为如下几个阶段。

- 需求分析。
- 结构设计，包括概念结构设计、逻辑结构设计和物理结构设计。
- 行为设计，包括功能设计、事务设计和程序设计。
- 数据库实施，包括加载数据库数据和调试运行应用程序。
- 数据库运行和维护阶段。

3. 数据库结构设计包含哪几个过程？

答：包括概念结构设计、逻辑结构设计和物理结构设计。

4. 需求分析中发现事实的方法有哪些？

答：检查文档、面谈、观察操作中的业务、研究和问卷调查等。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/778131137057006034>