

## 第三章 关系数据库标准语言 SQL

第三章详细介绍关系数据库语言 SQL。SQL 语言是关系数据库的标准语言,内容十分丰富,是学习关系数据库概念和技术的重要部分。

### 一、基本知识点

关系模型和关系数据库是《概论》的重点,第三章又是重点中的重点,是全书中篇幅最大的一章,因为关系数据库系统的主要功能是通过 SQL 语言来实现的。

需要了解的:SQL 语言发展的过程,从而进一步了解关系数据库技术和 RDBMS 产品的发展过程。

需要牢固掌握的:掌握 SQL 语言的特点、SQL 语言与非关系模型(层次模型、网状模型)数据语言的不同,从而体会 SQL 语言之所以能够为用户和业界所接受,并成为国际标准的原因;体会面向过程的语言和 SQL 语言的区别和优点;体会关系数据库系统为数据库应用系统的开发提供良好环境、减轻用户负担、提高用户生产率的原因。

需要举一反三的:熟练而正确地使用 SQL 语言完成对数据库的查询、插入、删除、更新操作,特别是各种各样的查询,掌握 SQL 语言强大的查询功能。

在完成具体的 SQL 语句时,希望读者能有意识地 and 关系代数、关系演算等语言进行比较,了解它们各自的特点。

难点:本章的难点在于用 SQL 语言正确完成复杂查询。因此在学习的过程中一定要多练习,要在某一个 RDBMS 产品上进行实际运行,检查查询的结果是否正确。只有通过大量练习,才能真正达到举一反三的熟练程度。

### 二、习题解答和解析

1. 试述 SQL 语言的特点。

答

(1) 综合统一。SQL 语言集数据定义语言 DDL、数据操纵语言 DML、数据控

制语言 DCL 的功能于一体。

(2) 高度非过程化。用 SQL 语言进行数据操作,只要提出“做什么”,而无需指明“怎么做”,因此无需了解存取路径,存取路径的选择以及 SQL 语句的操作过程由系统自动完成。

(3) 面向集合的操作方式。SQL 语言采用集合操作方式,不仅操作对象、查找结果可以是元组的集合,而且一次插入、删除、更新操作的对象也可以是元组的集合。

(4) 以同一种语法结构提供两种使用方式。SQL 语言既是自含式语言,又是嵌入式语言。作为自含式语言,它能够独立地用于联机交互的使用方式;作为嵌入式语言,它能够嵌入到高级语言程序中,供程序员设计程序时使用。

(5) 语言简捷,易学易用。

解析

详细的可参考《概论》上 3.1.1。注意不要仅仅背这些特点,关键是要通过具体的练习、使用 SQL 语句来理解这些特点。

2 试述 SQL 的定义功能。

答

SQL 的数据定义功能包括定义表、定义视图和定义索引。

SQL 语言使用 CREATE TABLE 语句建立基本表,ALTER TABLE 语句修改基本表定义,DROP TABLE 语句删除基本表;使用 CREATE INDEX 语句建立索引,DROP INDEX 语句删除索引;使用 CREATE VIEW 语句建立视图,DROP VIEW 语句删除视图。

3 用 SQL 语句建立第二章习题 5 中的 4 个表。

答

对于 S 表: S(SNO, SNAME, STATUS, CITY);

建 S 表

```
CREATE TABLE S
(SNO CHAR(3),
SNAME CHAR(10),
STATUS CHAR(2),
CITY CHAR(10));
```

对于 P 表: P(PNO, PNAME, COLOR, WEIGHT);

建 P 表

```
CREATE TABLE P
(PNO CHAR(3),
PNAME CHAR(10),
```

```
COLOR CHAR(4),  
WEIGHT INT);
```

对于 J 表: J(JNO, JNAME, CITY);

建 J 表

```
CREATE TABLE J  
(JNO CHAR(3),  
JNAME CHAR(10),  
CITY CHAR(10));
```

对于 SPJ 表: SPJ(SNO, PNO, JNO, QTY);

建 SPJ 表

```
CREATE TABLE SPJ  
(SNO CHAR(3),  
PNO CHAR(3),  
JNO CHAR(3),  
QTY INT);
```

4 针对上题中建立的 4 个表试用 SQL 语言完成第二章习题 5 中的查询。

答

读者可以对比 SQL 语言、关系代数、ALPHA 语言、QBE 语言, 体会各种语言的优点。

(1) 求供应工程 J1 零件的供应商号码 SNO;

```
SELECT SNO  
FROM SPJ  
WHERE JNO = J1 ;
```

(2) 求供应工程 J1 零件 P1 的供应商号码 SNO;

```
SELECT SNO  
FROM SPJ  
WHERE JNO = J1  
AND PNO = P1 ;
```

(3) 求供应工程 J1 零件为红色的供应商号码 SNO;

```
SELECT SNO                                     / * 这是嵌套查询 */  
FROM SPJ  
WHERE JNO = J1  
AND PNO IN                                     / * 找出红色零件的零件号码 PNO */  
      (SELECT PNO  
        FROM P                                 / * 从 P 表中找 */  
        WHERE COLOR = 红 );
```

或

```

SELECT SNO
FROM SPJ,P                                / * 这是两表连接查询 */
WHERE JNO = J1                             / * 这是复合条件连接查询 */
      AND SPJ.PNO = P.PNO
      AND COLOR = 红 ;

```

(4) 求没有使用天津供应商生产的红色零件的工程号 JNO;

解析

读者可以对比第二章习题 5 中用 ALPHA 语言来完成该查询的解答。如果大家理解了有关该题的解析说明,那么本题的解答可以看成是把关系演算用 SQL 来表示的过程。

```

GET W (J.JNO):  v SPJX(SPJX .JNO = J.JNO
                v SX ( SX.SNO = SPJX .SNO ( SX .CITY = 天津
                v PX(PX .PNO = SPJX .PNO  PX .COLOR = 红 ))

```

这里的第一种解法是使用多重嵌套查询,第二种方法的子查询是一个多表连接。

注意:从 J 表入手,以包含那些尚未使用任何零件的工程号。

```

SELECT JNO
FROM J
WHERE NOT EXISTS
      (SELECT *
      FROM SPJ
      WHERE SPJ.JNO = J.JNO
            AND SNO IN                                / * 天津供应商的 SNO */
            (SELECT SNO
            FROM S
            WHERE CITY = 天津 )
            AND PNO IN                                / * 红色零件的 PNO */
            (SELECT PNO
            FROM P
            WHERE COLOR = 红 ));

```

或

```

SELECT JNO
FROM J
WHERE NOT EXISTS
      (SELECT *
      FROM SPJ, S, P
      WHERE SPJ.JNO = J.JNO

```

```
AND SPJ.SNO = S.SNO
AND SPJ.PNO = P.PNO
AND S.CITY = 天津
AND P.COLOR = 红 );
```

(5) 求至少用了供应商 S1 所供应的全部零件的工程号 JNO (类似于《概论》P113 例 44)。

```
SELECT DISTINCT JNO
FROM SPJ SPJZ
WHERE NOT EXISTS      / * 这是一个相关子查询 */
    (SELECT *          / * 父查询和子查询均引用了 SPJ 表 */
    FROM SPJ SPJX      / * 用别名将父查询与子查询中的 SPJ 表区分开 */
    WHERE SNO = S1
    AND NOT EXISTS
        (SELECT *
        FROM SPJ SPJY
        WHERE SPJY.PNO = SPJX.PNO
        AND SPJY.JON = SPJZ.JNO));
```

### 解析

本查询的解析可以参考第二章第 5 题,用 ALPHA 语言的逻辑蕴涵来表达。

上述查询可以抽象为:要求这样的工程  $x$ ,使  $(\neg y)p \rightarrow q$  为真。即,对于所有的零件  $y$ ,满足逻辑蕴涵  $p \rightarrow q$ : $P$  表示谓词“供应商 S1 供应了零件  $y$ ”; $q$  表示谓词“工程  $x$  选用了零件  $y$ ”。即,只要“供应商 S1 供应了零件  $y$ ”为真,则“工程  $x$  选用了零件  $y$ ”为真。

逻辑蕴涵可以转换为等价形式:

$$(\neg y)p \rightarrow q \quad (\forall y((p \rightarrow q) \rightarrow (\neg y)p \rightarrow q)) \quad (\forall y((\neg y)p \rightarrow q) \rightarrow (\neg y)p \rightarrow q)$$

它所表达的语义为:不存在这样的零件  $y$ , 供应商 S1 供应了  $y$ , 而工程  $x$  没有选用  $y$ 。

5 针对习题 3 中的 4 个表试用 SQL 语言完成以下各项操作:

答

(1) 找出所有供应商的姓名和所在城市。

```
SELECT SNAME, CITY
FROM S;
```

(2) 找出所有零件的名称、颜色、重量。

```
SELECT PNAME, COLOR, WEIGHT
FROM P;
```

(3) 找出使用供应商 S1 所供应零件的工程号码。

```
SELECT JNO
FROM SPJ
WHERE SNO = S1 ;
```

- (4) 找出工程项目 J2 使用的各种零件的名称及其数量。

```
SELECT P.PNAME, SPJ.QTY
FROM P, SPJ
WHERE P.PNO = SPJ.PNO
      AND SPJ.JNO = J2 ;
```

- (5) 找出上海厂商供应的所有零件号码。

```
SELECT DISTINCT PNO
FROM SPJ
WHERE SNO IN
      (SELECT SNO
       FROM S
       WHERE CITY = 上海 );
```

- (6) 找出使用上海产的零件的工程名称。

```
SELECT JNAME
FROM J, SPJ, S
WHERE J.JNO = SPJ.JNO
      AND SPJ.SNO = S.SNO
      AND S.CITY = 上海 ;
```

或

```
SELECT JNAME
FROM J
WHERE JNO IN
      (SELECT JNO
       FROM SPJ, S
       WHERE SPJ.SNO = S.SNO
             AND S.CITY = 上海 );
```

- (7) 找出没有使用天津产的零件的工程号码。

```
SELECT JNO
FROM J
WHERE NOT EXISTS
      (SELECT *
       FROM SPJ
       WHERE SPJ.JNO = J.JNO
             AND SNO IN
```

```
(SELECT SNO
FROM S
WHERE CITY = 天津 ));
```

或

```
SELECT JNO
FROM J
WHERE NOT EXISTS
  (SELECT *
   FROM SPJ, S
   WHERE SPJ.JNO = J.JNO
        AND SPJ.SNO = S.SNO
        AND S.CITY = 天津 );
```

(8) 把全部红色零件的颜色改成蓝色。

```
UPDATE P
SET COLOR = 蓝
WHERE COLOR = 红 ;
```

(9) 由 S5 供给 J4 的零件 P6 改为由 S3 供应,请做必要的修改。

```
UPDATE SPJ
SET SNO = S3
WHERE SNO = S5
      AND JNO = J4
      AND PNO = P6 ;
```

(10) 从供应商关系中删除 S2 的记录,并从供应情况关系中删除相应的记录。

```
DELETE
FROM SPJ
WHERE SNO = S2 ;
```

```
DELETE
FROM S
WHERE SNO = S2 ;
```

解析

注意删除顺序,应该先从 SPJ 表中删除供应商 S2 所供应零件的记录,然后从 S 表中删除 S2。

(11) 请将 (S2, J6, P4, 200) 插入供应情况关系。

```
INSERT INTO SPJ(SNO, JNO, PNO, QTY) /* INTO 子句中指明列名 */
VALUES (S2, J6, P4, 200);          /* 插入的属性值与指明列要对应 */
```

或

```
INSERT INTO SPJ                / * INTO 子句中没有指明列名 */  
VALUES (S2,P4,J6,200);        / * 插入的记录在每个属性列上有值 */  
                                / * 并且属性列要和表定义中的次序对应 */
```

6 什么是基本表？什么是视图？两者的区别和联系是什么？

答

基本表是本身独立存在的表,在 SQL 中一个关系就对应一个表。

视图是从一个或几个基本表导出的表。视图本身不独立存储在数据库中,是一个虚表。即数据库中只存放视图的定义而不存放视图对应的数据,这些数据仍存放在导出视图的基本表中。视图在概念上与基本表等同,用户可以如同基本表那样使用视图,可以在视图上再定义视图。

7. 试述视图的优点。

答

- (1) 视图能够简化用户的操作;
- (2) 视图使用户能以多种角度看待同一数据;
- (3) 视图对重构数据库提供了一定程度的逻辑独立性;
- (4) 视图能够对机密数据提供安全保护。

详细解释参见《概论》3 5 4。

8 所有的视图是否都可以更新？为什么？

答

不是。视图是不实际存储数据的虚表,因此对视图的更新,最终要转换为对基本表的更新。因为有些视图的更新不能惟一有意义地转换成对相应基本表的更新,所以,并不是所有的视图都是可更新的,如《概论》3 5 1 中的视图 S \_ G (学生的学号及他的平均成绩)

```
CREAT VIEW S _ G(Sno,Gavg)  
AS SELECT Sno, AVG(Grade) / * 设 SC 表中“成绩”列 Grade 为数字型 */  
FROM SC  
GROUP BY Sno;
```

要修改平均成绩,必须修改各科成绩,而我们无法知道哪些课程成绩的变化导致了平均成绩的变化。

9 哪类视图是可以更新的？哪类视图是不可更新的？各举一例说明。

答

基本表的行列子集视图一般是可更新的,如《概论》3 5 3 中的例 1。

若视图的属性来自集函数、表达式,则该视图肯定是不可以更新的,如《概论》3 5 3 中的 S \_ G 视图。



10 试述某个你熟悉的实际系统中对视图更新的规定。

答

(略)

解析

不同的系统对视图更新的规定是不同的,读者必须了解你所用系统对视图更新的规定。

11. 请为三建工程项目建立一个供应情况的视图,包括供应商代码(SNO)、零件代码(PNO)、供应数量(QTY)。针对该视图完成下列查询:

(1) 找出三建工程项目使用的各种零件代码及其数量。

(2) 找出供应商 S1 的供应情况。

答

建视图:

```
CREATE VIEW V _ SPJ AS
SELECT SNO, PNO, QTY
FROM SPJ
WHERE JNO =
      (SELECT JNO
       FROM J
       WHERE JNAME = 三建 );
```

对该视图查询:

(1) 找出三建工程项目使用的各种零件代码及其数量。

```
SELECT PNO, QTY
FROM V _ SPJ;
```

(2) 找出供应商 S1 的供应情况。

```
SELECT PNO, QTY      / * S1 供应三建工程的零件号和对应的数量 */
FROM V _ SPJ
WHERE SNO = S1 ;
```

12 针对习题 3 建立的表,用 SQL 语言完成以下各项操作:

(1) 把对表 S 的 INSERT 权限授予用户张勇,并允许他再将此权限授予其他用户。

答

```
GRANT INSERT
ON TABLE S
TO 张勇
WITH GRANT OPTION;
```

(2) 把查询 SPJ 表和修改 QTY 属性的权限授给用户李天明。

答

```
GRANT SELECT, UPDATE(QTY)
ON TABLE SPJ
TO 李天明;
```

13 在嵌入式 SQL 中是如何区分 SQL 语句和主语言语句的？

答

在 SQL 语句前加上前缀 EXEC SQL, SQL 语句的结束标志则随主语言的不同而不同。

例如,在 PL/1 和 C 中,以分号(;)结束,在 COBOL 中,以 END-EXEC 结束。

14 在嵌入式 SQL 中是如何解决数据库工作单元与源程序工作单元之间通信的？

答

数据库工作单元与源程序工作单元之间的通信主要包括：

(1) SQL 通信区 SQLCA,用来向主语言传递 SQL 语句的执行状态信息,使主语言能够根据此信息控制程序流程。

(2) 主变量(Host Variable):

1) 用来实现主语言向 SQL 语句提供参数;

2) 将 SQL 语句查询数据库的结果交主语言进一步处理。

(3) 游标(Cursor),解决集合性操作语言与过程性操作语言的不匹配,通过游标逐一获取记录,并赋给主变量,交由主语言进一步处理。

详细解释参见《概论》3.7.2。

15 在嵌入式 SQL 中是如何协调 SQL 语言的集合处理方式和主语言的单记录处理方式的？

答

用游标来协调这两种不同的处理方式。游标区是系统为用户开设的一个数据缓冲区,存放 SQL 语句的执行结果,每个游标区都有一个名字。用户可以通过游标逐一获取记录,并赋给主变量,交由主语言进一步处理。

### 三、大 作 业

使用某一个 RDBMS 产品进行 SQL 语言的练习。

你可以使用学校已经安装的 RDBMS 产品,也可以从网上下载一些免费的 RDBMS 软件。最好使用主流的产品,为今后的实际工作积累知识和技术。

练习内容:

1. 了解或学会安装 RDBMS 的步骤;
2. 了解具体产品的功能和特点;
3. 先使用产品自带的样本数据库进行练习;
4. 再用 SQL 语句建立第二章习题 5 中的 4 个表;
5. 使用 SQL 语句尽量多的向 4 个表中插入数据;
6. 完成本章练习题,用实际数据检查你的执行结果是否有错;
7. 使用 SQL 语句建立第三章中的学生 - 课程数据库:  
学生 - 课程数据库中包括 3 个表:  
学生表: Student(Sno, Sname, Ssex, Sage, Sdept)  
课程表: Course(Cno, Cname, Cpro, Ccredit)  
学生选课表: SC(Sno, Cno, Grade)  
使用 SQL 语句尽量多地向 3 个表中插入数据;
8. 完成第三章书上的例题,用实际数据检查你的执行结果是否有错。