第3章习题参考答案

一．选择题

1．A 2．B 3．C 4．A 5．B 6．C 7．A 8．D 9．C 10．A

二．简答题

1．试述SQL的功能有哪些？它有什么特点？

答：SQL语言有以下功能：

（1）数据定义DDL：提供定义、修改、删除关系模式的命令，也提供定义和删除索引及视图的命令，如CREATE，DROP，ALTER等。

（2）数据操纵DML：查询语言，还包括在数据库中插入、删除、修改元组的命令，如SELECT ，INSERT，UPDATE，DELETE等；

（3）数据控制DCL：包括完整性控制（integrity），使数据库中的数据满足完整性约束条件，破坏完整性约束条件的更新将被禁止；事务控制（transaction control）,提供定义事务控制的命令，如COMMIT，ROLLBACK等；权限管理（authorization）,提供对关系或视图访问权限进行说明的命令，如GRANT，REVOKE等。

2．什么是基本表？什么是视图？两者的区别和联系是什么？

答：基本表是独立存在的表，一个关系对应一个表，在数据库管理系统中，表是一个实际存在的物理文件。

视图是从一个或几个基本表导出的虚表。数据库管理系统只存储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。视图在逻辑上与基本表等同，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。通过视图也可以更新数据，但其更新受到一些限制。

3．请举例说明哪一类视图可以更新，哪一类视图不可以更新。

答：目前各DBMS一般都只允许对行列子集视图进行更新，并有以下限制：

（1）若视图是由两个以上基本表导出的，则此视图不允许更新。

（2）若视图的字段来自表达式或常数，则不允许对此视图执行INSERT和UPDATE操作，但允许执行DELETE操作。

（3）若视图的字段来自集函数，则此视图不允许更新。

（4）若视图定义中含有GROUP BY子句，则此视图不允许更新。

（5）若视图定义中含有DISTINCT任选项，则此视图不允许更新。

（6）若视图定义中有嵌套查询，并且内层查询的FROM子句中的表与导出该视图的表相同，则此视图不允许更新。

（7）一个不允许更新的视图所导出的视图不允许更新。

4．相关子查询与不相关子查询的区别是什么？

答：不相关子查询的查询条件不依赖于父查询，它的处理过程是由里向外逐层处理，即每个子查询在上一级查询处理之前求解，子查询的结果用于建立其父查询的查找条件。

相关子查询的查询条件依赖于父查询，它的处理过程是：

（1）首先取外层查询中表的第一个元组，根据它的值与内层查询相关的属性值处理内层查询，若WHERE子句返回值为真，则取此元组放入结果表；

（2）取外层表的下一个元组，重复（1）的操作，直至外层表全部检查完为止。

5．为什么要引入嵌入式SQL ？

答：嵌入式SQL是将SQL嵌入到高级语言中，如PASCAL、FORTRAN、C、ADA等，这些高级语言称为宿主语言，或主语言。嵌入式SQL可以充分利用高级语言的极强的表达计算功能和SQL的数据处理能力，以便完成用户复杂的事务处理。

6．什么叫做游标？为什么要引入游标？

答：游标是用于嵌入式SQL语言中的一种技术，它是相当于一个指示器，指向SQL查询结果集中的第一个元组。

引入游标的原因是：SQL语言与主语言具有不同数据处理方式。SQL语言是面向集合的，一条SQL语句原则上可以产生或处理多条记录，而主语言是面向记录的，一组主变量一次只能存放一条记录，仅使用主变量并不能完全满足SQL语句向应用程序输出数据的要求。嵌入式SQL引入了游标的概念，它可以解决集合性操作语言与过程性操作语言的不匹配的的问题。

三、综合题

1.答

Create table  Student

(

SNO varchar(5) primary key,

NAME varchar(10) not null,

SEX char (2)  not nul,

BIRTHDAY datetime not null,

CLASS int not null

)

Create table  Teacher

(

TNO varchar(5) primary key,

NAME varchar(10) not null,

SEX char (2)  not nul,

BIRTHDAY datetime not null,

PROF varchar(8) not null,

DEPART varchar(20) not null

)

Create table  Course

(

CNO varchar(5) primary key,

CNAME varchar(10) not null,

TNO int(10) not null

)

Create table  Score

(

SNO varchar(5) ,

CNO varchar(10),

DEGREE int(10),

Primary key (SNO,CNO)

)

2.答

1) SELECT NAME, DATEDIFF(year, BIRTHDAY, getdate()) AS 年龄

FROM Student

2) SELECT \*

FROM Student

WHERE NAME LIKE ‘王%’

3) SELECT NAME, CNAME

FROM Teacher, Course

WHERE Teacher.TNO= Course.TNO AND SEX = ‘男’

4) SELECT SNO, CNO, DEGREE

FROM Score

WHERE DEGREE = (SELECT max(DEGREE) FROM score)

5) SELECT NAME

FROM student

WHERE SEX = (SELECT SEX FROM student WHERE NAME = '李军')

AND CLASS = (SELECT CLASS FROM student WHERE NAME = '李军')

6) SELECT \*

FROM Score, Student, Course

WHERE student.SEX='男' and course.CNAME='计算机导论'

And Score.CNO = course.CNO and student.SNO = Score.SNO

SELECT SNO , CNO,DEGREE

FROM Score

WHERE SNO IN

(SELECT SNO

FROM Student

WHERE sex=’男’)

AND CNO =

(SELECT CNO FROM Course

WHERE CNAME=’计算机导论’);

7) SELECT SNO, NAME

FROM Student

WHERE SNO IN

(

SELECT SNO

FROM Score

GROUP BY SNO

HAVING COUNT(CNO)>=2);

或

SELECT SNO,NAME

FROM Student

WHERE EXISTS

( SELECT \*

FROM Score

WHERE Student.SNO=Score.SNO

GROUP BY SNO

HAVING COUNT( \*)>=2);

8) SELECT CNO, COUNT(\*)

FROM Score

GROUP BY CNO

9) SELECT NAME

FROM Student

WHERE NOT EXISTS

(

SELECT \*

FROM Course

WHERE NOT EXISTS

(SELECT \*

FROM Score

WHERE Student.SNO=Score.SNO AND Course.CNO=Score.CNO

)

10) SELECT SNO

FROM Score X

WHERE NOT EXISTS

( SELECT CNO

FROM Score Y

WHERE Y.SNO='109' AND NOT EXISTS

SELECT CNO

FROM Score Z

WHERE X.SNO= Z.SNO AND Y.CNO=Z.CNO

)

3 .答

1) GRANT INSERT

ON TABLE Student

TO 张勇

WITH GRANT OPTION

2) GRANT SELECT, UPDATE(PROF)

ON TABLE Teacher

TO 李天明

3) REVOKE SELECT ON TABLE Score FROM PUBLIC

4.答

1) CREATE VIEW VIEW\_1

FROM BOOKS

WHERE Pname = P1

WITH CHECK OPTION

2) CREATE VIEW VIEW\_2

AS SELECT Bno, Title, Pname, City

FROM BOOKS, PUBLISHERS

WHERE BOOKS.Pname= PUBLISHERS.Pname

3) CREATE VIEW VIEW\_3

AS SELECT Name, Title, Date

FROM BOOKS, BORROWERS, LOANS

WHERE LOAN.CardNo = BORROWERS.CardNo AND LOAN.Bno = BOOKS.Bno

4) CREATE VIEW VIEW\_4

AS SELECT Name

FROM VIEW\_30

WHERE DATEDIFF(mouth, Date, getdate()) > 1