第三章课后习题答案

3.1试述SQL语言的特点。

答：1.综合统一

2.高度非过程化

3.面向集合的操作方式

4.以同一种语法结构提供两种使用方式

3.2解释下列术语：

SQL模式：SQL语言所支持的关系数据库的体系结构，包括视图、基本表、存储文件三个层次。

基本表：简称基表。是数据库中实际存在的表，在SQL中一个关系对应于一个表。

视图：SQL用视图概念支持非标准的外模式概念。视图是从一个或几个基表导出的表，虽然它也是关系形式，但它本身不实际存储在数据库中，只存放对视图的定义信息（没有对应的数据）。因此，视图是一个虚表（Virtual Table）或虚关系，而基表是一种实关系（Practical Relation）。

单表查询：只涉及到一个关系的查询。

连接查询：同时涉及两个以上的表的查询。

嵌套查询：将一个查询块嵌套在另一个查询块的WHERE子句或HAVING条件中的查询。

3.3试述SQL语言的特点。

(同3.1)

3.4试述SQL体系结构和关系数据库模式之间的关系。

SQL语言支持关系数据库体系结构，即外模式、模式和内模式，利用SQL语言可以实现对三级模式的定义、修改和数据的操纵功能，在此基础上形成了SQL体系结构；在SQL体系结构中的视图对应了关系数据模式的外模式，基本表对应了关系数据模式的模式，存储文件对应了关系数据模式的内模式。

3.5 SQL是如何实现实体完整性、参照完整性和用户定义完整性的？

实体完整性简单的说，就是将表中的每一行看作一个实体。实体完整性要求表的标示符列或主键的完整性。可以通过建立唯一索引、PRIMARY KEY约束、UNIQUE约束，以及列的IDENTITY属性来实施实体完整性。

参照完整性维持被参照表和参照表之间的数据一致性，通过主键（PRIMARY KEY）约束和外键（FOREIGN KEY）约束来实现。引用完整性确保键值在所有表中一致。这样的一致性要求不能引用不存在的值，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。在被参照表中，当其主键值被其他表所参照时，该行不能被删除也不允许改变。在参照表中，不允许参照不存在的主键值。

用户定义完整性使得可以定义不属于其它任何完整性分类的特定业务规则。所有的完整性类型都支持用户定义完整性。

3.6讨论当对一个视图进行更新的时候可能会出现什么样的问题？

1. 若视图的属性来自属性表达式或常数，则不允许对视图执行INSERT和UPDATE操作，但允许执行DELETE操作。
2. 若视图的属性来自库函数，则不允许对此视图更新。
3. 若视图定义中有GROUP BY子句，则不允许对此视图更新。
4. 若视图定义中有DISTINCT任选项，则不允许对此视图更新。
5. 若视图定义中有嵌套查询，并且嵌套查询的FROM子句涉及导出该视图的基本表，则不允许对此视图更新。
6. 若视图由两个以上的基本表导出，则不允许对此视图更新。
7. 如果在一个不允许更新的视图上再定义一个视图，这种二次视图是不允许更新的。

3.7设有两个基本表R(A,B,C) 和 S(A,B,C)试用SQL查询语句表达下列关系代数表达式：

⑴ R∩S ：(select \* from R) INTERSECT (select \* from S)

⑵ R－S ：(select \* from R) EXCEPT (select \* from S)

⑶ R∪S ：(select \* from R) UNION (select \* from S)

⑷ R×S ：select \* from R,S

3.8对于教学数据库的三个基本表：

S(学号，姓名 ，年龄，性别)

SC(学号 ，课程号，成绩)

C(课程号 ，课程名，任课教师姓名）

试用SQL语句表达下列查询：

⑴ 查询姓刘的老师所授课程的课程号和课程名。

SELECT 课程号, 课程名

FROM C

WHERE 任课教师姓名=’刘%’;

⑵ 查询年龄大于23岁的男同学的学号和姓名。

SELECT 学号, 姓名

FROM S

WHERE 性别=’男’ AND 年龄>23;

⑶ 查询学号为S3的学生所学课程的课程号、课程名和任课教师姓名。

SELECT 课程号, 课程名, 任课教师姓名

FROM C

WHERE 课程号 IN

(SELECT 课程号

FROM SC

WHERE 学号=’S3’);

⑷ 查询“张小飞”没有选修的课程号和课程名。

SELECT \*

FROM C

WHERE NOT EXISTS

(SELECT 学习.课程号

FROM SC,S

WHERE SC.课程号=C.课程号 AND S.学号=SC.学号

AND S.姓名='张小飞');

⑸ 查询至少选修了3门课程的学生的学号和姓名。

SELECT 学号, 姓名

FROM S

WHERE 学号 IN

(SELECT 学号

FROM SC

GROUP BY 学号

HAVING COUNT (课程号)>=3);

⑹ 查询全部学生都选修了的课程编号和课程名称。

SELECT 课程号, 课程名

FROM C

WHERE NOT EXISTS

(SELECT \*

FROM S

WHERE NOT EXISTS

(SELECT \*

FROM SC

WHERE SC.课程号=C.课程号AND S.学号=SC.学号));

⑺ 在SC中删除尚无成绩的选课元组。

DELETE

FROM SC

WHERE 成绩 IS NULL;

⑻ 把“高等数学”课的所有不及格成绩都改为60。

UPDATE SC

SET 成绩 = 60

WHERE 成绩< 60 AND 课程号

(SELECT 课程号

FROM C

WHERE 课程名='高等数学');

⑼ 把低于总评均成绩的女同学的成绩提高5%。

UPDATE SC

SET 成绩=成绩\*1.05

WHERE 学号 IN

(SELECT 学号 FROM S WHERE 性别=’女’)

AND 成绩 <

(SELECT AVG(成绩) FROM SC);

⑽ 向C中插入元组（‘C8’，‘VC++’，‘王昆’）

INSERT INTO C VALUES（’C8’,’VC++’,’王昆’）;

3.9 设有下列四个关系模式：

PRODUCT(MAKER,MODEL,TYPE)

PC(MODEL,SPEED,RAM,HD,CD,PRICE)

LAPTOP(MODEL,SPEED,RAM,SCREEN,PRICE)

PRINTER(MODEL,COLOR,TYPE,PRICE)

注：PRODUCT表中TYPE属性列的取值为pc或laptop或printer；PRINTER表中color属性列的取值为ture或false，代表彩色或单色。

试用SQL语句表达下列查询：

⑴ 找出价格高于1万5千元，并且运行速度低于同价位PC的平均运行速度的LAPTOP。

SELECT \*

FROM laptop

WHERE price<3500 And speed>

(select avg(speed)

from pc

where pc.price=laptop.price);

⑵ 找出生产价格最低的彩色打印机的厂家。

SELECT maker

FROM product

WHERE model in

(select model

from printer

where color=true and price<=

(select min(price)

from printer));

⑶ 计算由厂家“HP”生产的PC机和LAPTOP机的平均价格。

(SELECT TYPE, AVG (PRICE) FROM PC, PRODUCT

WHERE PC.MODEL=PRODUCT.MODEL AND

PRODUCT.MAKER=’HP’

GROUP BY TYPE)

UNION

(SELECT TYPE, AVG (PRICE) FROM LAPTOP,PRODUCT

WHERE APTOP.MODEL=PRODUCT.MODEL AND

PRODUCT.MAKER=’ HP’

GROUP BY TYPE;)

⑷ 计算各厂商所生产的LAPTOP机的显示器的平均尺寸。

SELECT MAKER, AVG (SCREEN)

FROM LAPTOP, PRODUCT

WHERE LAPTOP.MODEL=PRODUCT.MODEL

GROUP BY MAKER;

⑸ 找出每一个生产厂商的PC机的最高价格。

SELECT MAKER, MAX (PRICE)

FROM PC, PRODUCT

WHERE PC.MODEL=PRODUCT.MODEL

GROUP BY MAKER;

⑹ 计算生产打印机的各个厂商所生产的PC机的硬盘的平均容量。

SELECT MAKER, AVG (HD)

FROM PC, PRODUCT

WHERE PC.MODEL=PRODUCT.MODEL AND

MAKER IN (SELECT MAKER FROM PRODUCT WHERE TYPE=’printer’)

GROUP BY MAKER;

3.10 试设计如图3-2中显示的数据库模式Library，用来记录书籍、借书人和书籍借出的情况，参照完整性在图中用有向弧来表示。请用SQL语言建立图中的关系模式，并完成下列操作：

CREATE TABLE PUBLISHER

(

NAME CHAR(20) PRIMARY KEY,

ADDRESS CHAR(30),

PHONE CHAR(12)

);

CREATE TABLE BOOK

(

BOOKID CHAR(4) PRIMARY KEY,

TITLE CHAR(20),

PUBLISHERNAME CHAR(20) REFERENCES PUBLISHER(NAME)

);

CREATE TABLE AUTHOR

(

BOOKID CHAR(4) REFERENCES BOOK(BOOKID),

AUTHOR CHAR(8),

PRIMARY KEY(BOOKID)

);

CREATE TABLE BORROWER

(

CARDID CHAR(4) PRIMARY KEY,

NAME CHAR(8),

ADDRESS CHAR(30),

PHONE CHAR(12)

);

CREATE TABLE BORROW

(

BOOKID CHAR(4) REFERENCES BOOK(BOOKID),

CARDID CHAR(4) REFERENCES BORROWER(CARDID),

DATEOUT DATETIME,

DUEOUT DATETIME,

PRIMARY KEY (BOOKID,CARDID);

)

⑴ 查询“高等教育出版社”出版的所有图书名称和编号。

SELECT BOOKID,TITLE

FROM BOOK

WHERE PUBLISHERNAME=’高等教育出版社’;

⑵ 查询所有作者是“郭雨辰”的图书的编号和名称。

SELECT BOOKID,TITLE

FROM BOOK

WHERE BOOKID IN

(SELECT BOOKID

FROM AUTHOR

WHERE AUTHOR=’郭雨辰’);

⑶ 查询“王丽”借过的所有图书的名称。

SELECT TITLE

FROM BOOK

WHERE BOOKID IN

(SELECT BOOKID

FROM BORROW

WHERE CARDID IN

(SELECT CARDID

FROM BORROWER

WHERE NAME=’王丽’);

⑷ 查询“李明”在2018年上半年期间借过的图书名称。

SELECT TITLE

FROM BOOK

WHERE BOOKID IN

(SELECT BOOKID

FROM BORROW

WHERE DATEOUT BETWEEN ‘2018/1/1’AND ‘2018/6/30’ AND CARDID IN

(SELECT CARDID

FROM BORROWER

WHERE NAME=’李明’);

⑸ 建立视图，显示2017年期间没有被人借过的图书编号和名称。

CREATE VIEW UNPOPULARBOOK

AS

SELECT BOOKID,TITLE

FROM BOOK

WHERE BOOKID NOT IN

(SELECT BOOKID

FROM BORROW

WHERE DATEOUT BETWEEN‘2017/1/1’ AND‘2017/12/31’);

⑹ 建立超期未归还书籍的视图，显示图书编号和名称，以及借书人姓名和电话。

CREATE VIEW DELAY

AS

SELECT BOOKID, TITLE, NAME, PHONE

FROM BOOK, BORROW, BORROWER

WHERE BOOK.BOOKID = BORROW.BOOKID AND

BORROW.CARDID = BORROWER.CARDID AND

DUEOUT > DATE();

⑺ 建立热门书籍的视图，显示2017年期间借出次数最多的10本图书名称。

CREATE VIEW POPULARBOOK

AS

SELECT TOP 10 TITLE

FROM BOOK,BORROW

WHERE BOOK.BOOKID=BORROW.BOOKID AND

DATEOUT BETWEEN '2017/1/1' AND '2017/12/31'

GROUP BY TITLE

ORDER BY COUNT(TITLE) desc

⑻ 增加新书《大数据》，书号为“TP319-201”，该书由“广西师范大学出版社”出版，作者为“涂子沛”。

INSERT INTO BOOK

VALUES(‘TP319-201’, ‘大数据’, ‘广西师范大学出版社’);

INSERT INTO AUTHOR

VALUES(‘TP319-201’, ‘涂子沛’);

⑼ 将“高等教育出版社”的电话改为“010－64054588”。

UPDATE PUBLISER

SET PHONE=010-64054588

WHERE NAME=’高等教育出版社’;

⑽ 删除书号为“D001701”的书籍信息。

DELETE

FROM BOOK

WHERE BOOKID=’ D001701’;

3.11 针对3.10建立的表，用SQL语言完成下列操作：

⑴ 把对所有表的INSERT权限授予“张丽”，并允许她将此权限授予其他用户；

GRANT INSERT ON DATABASE Library TO ‘张丽’ WITH GRANT OPTION;

⑵ 把查询和修改BORROWER的权限受给用户“王伟”。

GRANT INSERT, UPDATE ON TABLE BORROWER TO ‘王伟’;