

第 2 章作业		
姓名	潘腾凯	
学号	37220232203786	
班级		02 班

1. 试述关系模型的三个组成部分

答：关系模型由关系数据结构、关系操作集合，以及完整性约束三部分组成。

2. 简述关系数据语言的特点和分类

答：特点：系数据库的语言，其核心特点是通过数学关系理论（如集合论、谓词逻辑）来处理数据。

可分为三类：关系代数、关系演算，以及结构化查询语言（SQL）

3. 定义并理解下列术语，说明它们之间的联系与区别

(1) 域，笛卡尔积，关系，元组，属性

(2) 候选码，主码，全码，主属性，非主属性，外码

(3) 关系模式，关系，关系数据库

答：（1）域：是一组具有相同数据类型的值的集合。

笛卡尔积：给定一组域 D_1, D_2, \dots, D_n ，这组域的笛卡尔积为

$$D_1 \times D_2 \times \dots \times D_n = \{(d_1, d_2, \dots, d_n) | d_i \in D_i, i=1, 2, \dots, n\}$$

关系：在域 D_1, D_2, \dots, D_n 上笛卡尔积 $D_1 \times D_2 \times \dots \times D_n$ 的子集称为关系，表示为

$$R(D_1, D_2, \dots, D_n)$$

元组：关系中的每个元素成为元组。

属性：关系也是一张二维表，表的每行对应一个元组，表的每列对应一个域。由于域可以相同，为了加以区分，必须给每列起一个名字，称为属性。

(2) 候选码：若关系模式中的某一属性或属性组的值能唯一地标识一个元组，而它的真子集能唯一地标识一个元组，则称该属性或属性组为候选码。

主码：若一个关系由多个候选码，则选定其中一个为主码。

全码：关系中所有属性组合在一起才能唯一标识一个元组，此时关系的候选码包含所有属性，即关系的所有属性都是主属性。

主属性：是包含在任何一个候选码中的属性。

非主属性：是不包含在任何一个候选码中的属性。

外码：设 F 是基本关系 R 的一个或一组属性，但不是关系 R 的码，如果 F 与基本关系的主码 Ks 对应，则称 F 是基本关系 R 的外部码，简称外码。

(3) 关系模式：关系的描述成为关系模式 R (U, D, DOM, F) 其中 R 为关系名，U 为组成该关系的属性集合，D 为 U 中属性所来自的域，DOM 为属性向域的映像集合，F 为属性间数据依赖关系的集合。

关系：关系是关系模式在某一时刻的状态或内容。关系模式是静态的、稳定的，而关系是动态的、随时间不断变化的，这是因为关系操作在不断地更新着数据库中的数据。

关系数据库：关系数据库也有型和值之分。关系数据库的型成为关系数据库模式，是对关系数据库的描述，它包括若干域的定义以及在这些域上定义的若干关系模式。关系数据库是这些关系模式在某一时刻对应的关系的集合，通常称为关系数据库。

4. 举例说明关系模式和关系的区别

答：关系模式是型，关系是值，是关系模型的实例。例如 Student (Sno, Sname, Sage) 是关系模式，而下面的表是关系，即某一时刻关系模式的值：

Sno	Sname	Sage
S1	王大锤	18
S2	张全蛋	18

5. 试述关系模型的完整性约束。在参照完整性中，什么情况下外码属性的值可以为空值？

答：有三类完整性约束：实体完整性、参照完整性和用户定义的完整性（前二者必须满足）。关系模型的完整性约束是对关系的某种约束条件。

参照完整性是两个关系之间的一种约束，当外码属性不是其所在关系的主属性，则外码属性的值可以取空值。

6. 设有一个 SPJ 数据库，包括 4 个关系模式 S、P、J 和 SPJ：

S(SNO,SNAME,STATUS,CITY);
P(PNO,PNAME,COLOR,WEIGHT);
J(JNO,JNAME,CITY);
SPJ(SNO,PNO,JNO,QTY)。

具体数据内容见书此处略。

试用关系代数、元组关系演算语言 ALPHA 和与关系演算语言 QBE 完成如下查询：

- (1) 求供应工程项目 J1 零件的供应商代码 SNO
- (2) 求供应工程项目 J1 零件 P1 的供应商代码 SNO
- (3) 求供应工程项目 J1 红色零件的供应商代码 SNO
- (4) 求没有使用天津供应商生产的红色零件的工程项目代码 JNO。
- (5) 求至少使用了与供应商 S1 所提供的全部零件相同零件代码的工程项目代码 JNO

答：(1) 关系代数： $\pi_{SNO}(\sigma_{JNO='J1'}(SPJ))$

ALPHA 语言：GET W(SPJ.SNO):SPJ.JNO='J1'

QBE 语言：

SPJ	SNO	PNO	JNO	QTY
	P.S1		J1	

(2) 关系代数：

$\pi_{SNO}(\sigma_{JNO='J1' \wedge PNO='P1'}(SPJ))$

ALPHA 语言：GET W(SPJ.SNO):SPJ.JNO='J1'∩SPJ.PNO='P1'

QBE 语言：

SPJ	SNO	PNO	JNO	QTY
	P.S1	P1	J1	

(3) 关系代数:

$$\pi_{SNO}(\pi_{SNO,PNO}(\sigma_{JNO='J1'}(SPJ))) \bowtie \pi_{PNO}(\sigma_{COLOR='红'}(P))$$

ALPHA 语言:

RANGE P PX

GET W (SPJ.SNO): SPJ.JNO='J1' ^ \exists PX (PX.COLOR='红' ^ PX.PNO=SPJ.PNO)

QBE 语言:

SPJ	SNO	PNO	JNO	QTY
	P.S1	P1	J1	

P	PNO	PNAME	COLOR	WEIGHT
	P1		红	

(4) 关系代数:

$$\pi_{JNO}(J) - \pi_{JNO}(\pi_{SNO}(\pi_{SNO}(\sigma_{CITY='天津'}(S))) \bowtie \pi_{SNO,PNO,JNO}(SPJ) \bowtie \pi_{PNO}(\sigma_{COLOR='红'}(P)))$$

ALPHA 语言:

RANGE SPJ SPJX

P PX

S SX

GET W(J.JNO): $\neg \exists$ SPJX (SPJX.JNO = J.JNO ^

\exists SX (SX.SNO = SPJX.SNO ^ SX.CITY = '天津' ^

\exists PX (PX.PNO = SPJX.PNO ^ PX.COLOR = '红'))

QBE 语言:

S	SNO	SNAME	STATUS	CITY
	S1			天津

P	PNO	PNAME	COLOR	WEIGHT
---	-----	-------	-------	--------

	<u>P1</u>		红	
--	-----------	--	---	--

SPJ	SNO	PNO	JNO	QTY
1	P.S1	P1	P. J1	

(5) 关系代数:

$$\pi_{JNO, PNO}(SPJ) \div \pi_{PNO}(\sigma_{SNO='S1'}(SPJ))$$

ALPHA 语言:

RANGE SPJ SPJX

SPJ SPJY

P PX

GET W(J.JNO): $\forall PX(\exists SPJX(SPJX.PNO = PX.PNO \wedge SPJX.SNO = 'S1')$

$\Rightarrow \exists SPJY(SPJY.JNO = J.JNO \wedge SPJY.PNO = PX.PNO))$

QBE 语言:

7. 试述等值连接与自然连接的区别和联系。

答: 自然连接是一种特殊的等值连接, 它要求两个关系中进行比较的分量必须是同名的属性列, 并且要在结果中把重复的属性列去掉。

8. 关系代数的基本运算有哪些? 如何用这些基本运算来表示其他运算?

答: 在 8 种关系代数运算中, 并、差、笛卡尔积、投影和选择 5 种运算为基本运算, 其他三种均可以用这五种基本运算来表达。

交运算: $R \cap S = R - (R - S)$

$$R \bowtie_{A \theta B} S = \sigma_{A \theta B}(R \times S)$$

连接运算:

除运算: $R(X,Y)/S(Y,Z) = \pi_x(R) - \pi_x(\pi_x(R) \times \pi_y(S) - R)$

关系模型

数据库结构 & 形式化定义
 关系: 相同类型的值的集合, {0, 1, 2, 3, 4, 5, ...}
 笛卡尔积: 符号 X , n -tuple (元组), 分量, 基数 (一个成员的不同取值个数)
 关系: (笛卡尔积无意义, 其子集有意义)
 定义: $D_1 \times D_2 \times \dots \times D_n$ 的子集 $R(D_1, D_2, \dots, D_n)$, n 是 degree
 类型: 基本关系, 查询结果, 视图
 (实) (虚) (虚)
 基模式性质: ① 列名不同 ② 不同列可取自相同域 ③ 列次序无所谓 (列名与域名对应, 列名与域名对应)
 关系模式: 关系描述称为关系模式 $R(U, D, F)$, U 是属性名, D 是属性域, F 是属性间的函数依赖, 说明为属性的模型, 扩展
 其他概念: 候选码, 主码 (primary key), 主属性, 非主属性, 码, 码

关系数据库: 支持关系模型的数据库系统

数据库结构: DBMS 设计和实现

关系操作: 查询, select, project, join, divide, union, difference, intersection, 笛卡尔积
 基本操作: 更新, insert, delete, update
 特点: 集合操作方式
 语言: 关系代数 (REL)
 关系演算: 元组关系 (ALPHA, QUEL)
 域关系 (QBE)
 SQL \rightarrow 具有上二者的双至物性

关系完整性: 实体完整性
 参照完整性

(用定义的方式)

关系代数: 传统集合运算
 $R \cup S = \{t \mid t \in R \vee t \in S\}$
 $R - S = \{t \mid t \in R \wedge t \notin S\}$
 $R \cap S = (R - (R - S)) = \{t \mid t \in R \wedge t \in S\}$
 笛卡尔积 $R \times S = \{t \mid t = \langle r, s \rangle \mid r \in R \wedge s \in S\}$
 专门的关系运算: 选择 (限制): $\sigma_F(R) = \{t \mid t \in R \wedge F(t) = \text{true}\}$ $F: X \rightarrow \{0, 1\}$
 投影: $\pi_A(R) = \{t[A] \mid t \in R\}$ 如 $\pi_{\text{smaj}}(\text{Student})$
 连接 (join): $R \bowtie S = \{t \mid t = \langle r, s \rangle \mid r \in R \wedge s \in S \wedge F(r, s)\}$ 外连接 (左/右)
 除 (division): $R \div S = \{t \mid t[X] \wedge t[Y] \in R \wedge \pi_Y(S) \subseteq t[Y]\}$ 在 R 中的记录, $x = t[X]$

关系演算: ALPHA
 检索: ① 简单 GBT WSC (no) ② 用存在量词的 ③ 用全称量词的 ④ 用否定 ⑤ 用连接 ⑥ 用投影 ⑦ 用选择 ⑧ 用并 ⑨ 用差 ⑩ 用交集 ⑪ 用集运算
 更新: ① HOLD ② 修改 ③ 插入 ④ UPDATE
 删除: ① HOLD ② DELETE

