

数据库系统原理

作业 2

软件 42

欧阳鹏程

2141601030

版权声明：BY-NC-SA

2017 年 4 月 9 日

2.3 为什么要对关系模型加以完整性规则的限制？关系模型的完整性约束具体包括哪些内容？

答：完整性约束主要用于保证数据库中数据的正确性和相容性。关系模型的完整性约束包括：域完整性约束、实体完整性约束、参照完整性和用户定义完整性。

2.5 试对笛卡儿积、 θ 连接、等值连接、自然连接等关系运算进行比较。

答：首先连接运算（ θ 连接、等值连接、自然连接）都是建立在笛卡儿积运算之上的，比如 θ 连接是对笛卡儿积做选择，而等值连接是 θ 连接的一种特殊情况；自然连接是对笛卡儿积做投影。

运算	公式
笛卡儿积	$R \times S = \{t^{(m+n)} t^{(m)} \in R \wedge t^{(n)} \in S\}$
θ 连接	$R \bowtie_{A\theta B} S = \{rs r \in R \wedge s \in S \wedge r[A]\theta s[B]\} = \sigma_{A\theta B}(R \times S)$
等值连接	$R \bowtie_{A=B} S = \{rs r \in R \wedge s \in S \wedge r[A] = s[B]\} = \sigma_{A=B}(R \times S)$
自然连接	$R \bowtie S = \Pi_{Attr(R) \cup (Attr(S) - A)}(\sigma_{R.A=S.A}(R \times S))$

2.6 试将 $\cup, -, \bowtie, \sigma, \Pi$ 等运算转换为等价的元组关系演算形式，为什么要对关系演算表达式加以安全性约束？

答：

$$\begin{aligned}
 & - R \cup S = \{t | R(t) \vee S(t)\} \\
 & - R - S = \{t | R(t) \wedge \neg S(t)\} \\
 & - \text{设 } R \text{ 是 } m \text{ 元关系的元组, } S \text{ 是 } n \text{ 元关系的元组, } R \text{ 和 } S \text{ 有 } p \text{ 列} \\
 & \quad \text{表头一致 (} R \text{ 元组的 } m' \sim m' + p \text{ 和 } S \text{ 元组的 } n' \sim n' + p \text{ 的表} \\
 & \quad \text{头一致)} \\
 & R \bowtie S = \{t^{(m+n-p)} | (\exists u^{(m)})(\exists v^{(n)})(R(u) \wedge S(v) \wedge \\
 & \quad \underbrace{u[m'] = v[n'] \wedge u[m'+1] = v[n'+1] \wedge \cdots \wedge u[m'+p] = v[n'+p]}_p) \\
 & \quad \wedge t[1] = u[1] \wedge t[2] = u[2] \wedge \cdots \wedge t[m] = u[m] \wedge \\
 & \quad t[m'+p+1] = v[n'] \wedge \cdots \wedge t[m+n-p] = v[n']\} \\
 & \hspace{15em} (1)
 \end{aligned}$$

- $\sigma_F(R) = \{t | R(t) \wedge F'\}$
其中 F' 是选择条件 F 在元组关系演算中的等价表示。
- $\Pi_{i_1, i_2, \dots, i_k}(R) = \{t^{(k)} | (\exists u)(R(u) \wedge t[1] = u[i_1] \wedge \cdots \wedge t[k] = u[i_k])\}$
- 因为在关系演算中，有可能会出现无限关系，进而出现无穷验证的问题；而无限关系和无穷验证在计算机中都是不允许出现的，所以为了保证运算的安全性需要采取安全性约束。

2.7 设有下列关系：

(1) 求下列表达式的值：

$$\begin{aligned}
 E_1 &= \Pi_{C,D}(\sigma_{A > a'_1 \wedge B < b'_4}(R)) \\
 E_2 &= \Pi_{A,B,E,G}(\sigma_{A < a'_3 \wedge E < e'_3 \wedge G \neq g'_3}(R \bowtie S \bowtie T)) \\
 E_3 &= R \div \Pi_D(\sigma_{F=f'_1}(T)) \\
 E_4 &= \{t | (\exists u)(\exists v)(\exists w)(R(u) \wedge S(v) \wedge T(w) \wedge u[3] > c'_1 \wedge v[2] \neq \\
 & e'_2 \wedge w[3] \neq g'_2 \wedge u[4] = v[1] \wedge v[3] > w[2] \wedge t[1] = u[2] \wedge t[2] = \\
 & u[3] \wedge t[3] = v[1] \wedge t[4] = w[3] \wedge t[5] = w[2])\}
 \end{aligned}$$

(2) 试将 E_4 转化为等价的关系代数表达式。

表 1: R, S, T 关系

R	A	B	C	D
	a_1	b_1	c_1	d_1
	a_1	b_1	c_1	d_2
	a_2	b_2	c_2	d_1
	a_2	b_3	c_2	d_2
	a_2	b_1	c_2	d_3
	a_3	b_2	c_2	d_1
	a_3	b_2	c_3	d_2
	a_4	b_3	c_2	d_1
	a_4	b_3	c_2	d_3
	a_4	b_1	c_2	d_4
	a_4	b_4	c_2	d_2

S	D	E	F
	d_1	e_2	f_1
	d_2	e_1	f_2
	d_2	e_2	f_3
	d_3	e_3	f_1

T	D	F	G
	d_1	f_2	g_1
	d_2	f_2	g_2
	d_3	f_1	g_3

答:

(1) E_1 关系:

C	D
c_2	d_1
c_2	d_2
c_2	d_3
c_2	d_1
c_3	d_2
c_2	d_1
c_2	d_3
c_2	d_2

表 2: R_1

E_2 关系:

A	B	E	G
a_1	b_1	e_2	g_1
a_2	b_2	e_2	g_1
a_3	b_2	e_2	g_1
a_1	b_1	e_1	g_2
a_2	b_3	e_1	g_2
a_3	b_2	e_1	g_2

E_3 关系:

A	B	C
a_4	b_3	c_2

E_4 关系:

B	C	D	G	F
b_3	c_2	d_2	g_1	f_1
b_3	c_2	d_2	g_3	f_1
b_1	c_2	d_2	g_1	f_1
b_1	c_2	d_2	g_3	f_1
b_2	c_3	d_2	g_1	f_1
b_2	c_3	d_2	g_3	f_1
b_4	c_2	d_2	g_1	f_1
b_4	c_2	d_2	g_3	f_1

$$(2) E_4 = \Pi_{B,C,D,G,F}(\sigma_{C>'c'_1}(R) \bowtie \sigma_{E \neq 'e'_2}(S) \bowtie_{S.F>T.F} \sigma_{G \neq 'g'_2}(T))$$

2.8 试用关系代数表达式写出在 Student、Course、SC 关系上进行的下列查询:

- (1) 查询“计算机 07”班同学的学号及姓名;
- (2) 学号为“01055107”的同学所选修的课程名称及成绩;
- (3) 未选修编号为“CS-05”课程的学生学号;
- (4) 选修了“张华”老师所开设课程的学生姓名、课程名称及成绩;
- (5) 选修了全部课程的学生姓名及班级。

答:

- (1) $\Pi_{SNo, SName}(\sigma_{SClass='Computer07'}(Student))$
- (2) $\Pi_{CName, Score}(\sigma_{SNo='01055107'}(SC) \bowtie Course)$
- (3) $\Pi_{SNo}(Student) - \Pi_{SNo}(\sigma_{CNo='CS-05'}(SC))$
- (4) $\Pi_{SName, CName, Score}(\sigma_{TName='ZhangHua'}(Course) \bowtie SC \bowtie Student)$
- (5) $\Pi_{SName, Class}(Student \bowtie (\Pi_{SNo, CNo} \div \Pi_{SNo}(Student)))$