

# 数据管理基础

ch02\_2\_关系代数  
复习思考题  
(参考答案)

智能软件与工程学院



# 复习思考题 1

## 1. 在关系代数中：（[参考答案](#)）

- ① 关系是如何表示的？关系上的数据操纵是如何表示的？
- ② 在关系模型上进行数据操纵，其结果是什么？

## 2. 关于两个关系之间的并、交、差运算（[参考答案](#)）

- ① 什么样的两个关系，可以在它们之间执行并、交或差运算？
- ② 如果在两个关系之间执行并、交或差运算，那么结果关系的关系模式和结果元组集合分别是什么？
- ③ 为什么说‘交’运算不是关系代数中的基本运算？
- ④ 在什么情况下需要使用‘差’运算？在使用‘差’运算时需要注意什么？
- ⑤ 请比较‘差’运算和‘ $\neq$ ’（不等）比较运算之间的区别。

## 3. 关于两个关系之间的‘笛卡尔积’(product) 运算（[参考答案](#)）

- ① 什么样的两个关系，可以在它们之间执行笛卡尔积运算？
- ② 在两个关系之间执行笛卡尔积运算，结果关系的关系模式和结果元组集合分别是什么？
- ③ 笛卡尔积运算在关系数据库访问中有什么实际作用？

## 复习思考题 2

4. 设有一个公司产品零售数据库，其关系模式如下(带下划线的属性是码)：

关系名	属性集	关系模式
顾客	<u>顾客编号</u> , 姓名, 居住城市, 折扣	customers ( <u>cid</u> , cname, city, discent)
供应商	<u>供应商编号</u> , 名称, 所在城市, 佣金比例	agents ( <u>aid</u> , aname, city, percent)
商品	<u>商品编号</u> , 名称, 库存城市, 库存数量, 单价	products ( <u>pid</u> , pname, city, stqty, price)
订单	<u>订单编号</u> , 订购日期, 顾客编号, 供应商编号, 商品编号, 订购数量, 销售金额	orders( <u>ordno</u> , orddate, cid, aid, pid, qty, dols)

在下述六组关系代数查询表达式中，请分析他们各自查询的目标对象、查询语义和结果集的相互关系。

第1组	$Q_1: \sigma_{pid='p01'}(orders)$	$Q_2: \sigma_{pid \neq 'p01'}(orders)$
第2组	$Q_3: \pi_{cid}(\sigma_{pid='p01'}(orders))$	$Q_4: \pi_{cid}(\sigma_{pid \neq 'p01'}(orders))$
第3组	$Q_5: \pi_{cid}(orders) - \pi_{cid}(\sigma_{pid='p01'}(orders))$	$Q_6: \pi_{cid}(orders) - \pi_{cid}(\sigma_{pid \neq 'p01'}(orders))$
第4组	$Q_7: \pi_{cid}(customers) - \pi_{cid}(\sigma_{pid='p01'}(orders))$	$Q_8: \pi_{cid}(customers) - \pi_{cid}(\sigma_{pid \neq 'p01'}(orders))$
第5组	$Q_9: orders - \sigma_{pid='p01'}(orders)$	$Q_{10}: orders - \sigma_{pid \neq 'p01'}(orders)$
第6组	$Q_{11}: \pi_{cid}(orders - \sigma_{pid='p01'}(orders))$	$Q_{12}: \pi_{cid}(orders - \sigma_{pid \neq 'p01'}(orders))$

# 第1题--参考答案

## 1. 在关系代数中：

① 关系是如何表示的？关系上的数据操纵是如何表示的？

② 在关系模型上进行数据操纵，其结果是什么？

□ **关系的表示**：在关系代数中，关系是元组的集合。

设有一个 $n$ 目关系 $R$ ， $A_1, A_2, \dots, A_n$ 是关系 $R$ 的 $n$ 个属性， $domain(A_i)$ 是属性 $A_i$ 的域 ( $i=1, 2, \dots, n$ )

➤ 用关系名来表示一个关系： $R$

➤ 关系模式的表示： $R(A_1, A_2, \dots, A_n)$

➤ 元组及分量的表示：设 $t$ 是关系 $R$ 中的一个元组，那么 $t$ 就是一个具有如下形式的 $n$ 元组：

$(a_1, a_2, \dots, a_n)$  其中： $a_i \in domain(A_i)$  for  $i = 1, 2, \dots, n$

元组 $t$ 在属性 $A_i$ 上的取值可以被表示为  $t[A_i]$

➤ 关系 $R$ 就是如上所示的 $n$ 元组的集合，可以形式化表示为：

$$R = \{ (a_1, a_2, \dots, a_n) \mid a_i \in domain(A_i) \text{ for } i = 1, 2, \dots, n \}$$

$$\text{或者表示为：} R = \{ t \mid t[A_i] \in domain(A_i) \text{ for } i = 1, 2, \dots, n \}$$

$$\text{由笛卡尔积的定义可知：} R \subseteq domain(A_1) \times domain(A_2) \times \dots \times domain(A_n)$$

□ **数据操纵的表示**：以关系为运算对象，用关系代数运算符可以组成一个关系代数表达式，用关系代数表达式来表示用户在关系模型上的数据操纵请求。

□ **数据操纵结果的表示**：在关系数据模型上进行数据操纵，其结果也构成一个关系。关系代数表达式的运算结果就是用户数据操纵请求的结果关系。

## 第2题--参考答案

### 2. 关于两个关系之间的并、交、差运算

- ① 什么样的两个关系，可以在它们之间执行并、交或差运算？
- ② 如果在两个关系之间执行并、交或差运算，那么结果关系的关系模式和结果元组集合分别是什么？
- ③ 为什么说‘交’运算不是关系代数中的基本运算？
- ④ 在什么情况下需要使用‘差’运算？在使用‘差’运算时需要注意什么？
- ⑤ 请比较‘差’运算和‘ $\neq$ ’（不等）比较运算之间的区别。

① 当两个关系“具有相同的目 $n$ （即两个关系都有 $n$ 个属性）”且“相应的属性取自同一个域”时，才能在关系 $R$ 和 $S$ 之间执行并、交、差运算。

② 当两个关系 $R$ 和 $S$ 执行并、交、差运算时，结果关系的关系模式不变，与关系 $R$ （或关系 $S$ ）的关系模式一样。结果关系中的元组集合定义如下：

$$R \cup S = \{t \mid t \in R \vee t \in S\}$$

$$R \cap S = \{t \mid t \in R \wedge t \in S\}$$

$$R - S = \{t \mid t \in R \wedge t \notin S\}$$

$$S - R = \{t \mid t \in S \wedge t \notin R\}$$

③ 可以用‘差’运算来实现‘交’运算的运算功能：

$$R \cap S = R - (R - S) = S - (S - R)$$



## 第2题--参考答案 (cont.)

### 2. 关于两个关系之间的并、交、差运算

- ① 什么样的两个关系，可以在它们之间执行并、交或差运算？
  - ② 如果在两个关系之间执行并、交或差运算，那么结果关系的关系模式和结果元组集合分别是什么？
  - ③ 为什么说‘交’运算不是关系代数中的基本运算？
  - ④ 在什么情况下需要使用‘差’运算？在使用‘差’运算时需要注意什么？
  - ⑤ 请比较‘差’运算和‘ $\neq$ ’（不等）比较运算之间的区别。
- 
- ④ 当查询条件带有‘否定’语义，或者具有明显‘排它性’的时候，通常需要使用到‘差’运算；  
使用‘差’运算时，在减数和被减数关系中，通常需要包含查询目标对象的码。
  - ⑤ 是两种不同类型的运算符，但在应用中又具有一定的‘相关性’。
    - (1) ‘差’运算是两个关系之间的集合运算，‘ $\neq$ ’是两个分量之间的逻辑比较运算，常用于构造数据查询的查询条件表达式；
    - (2) 在数据库查询中，需谨慎使用‘ $\neq$ ’比较运算；
    - (3) 即使查询条件带有‘否定’语义，通常也不是直接使用‘ $\neq$ ’比较运算来构造查询条件，而是使用两个子查询之间的‘差’运算来实现查询目标。（具体应用场景可参考后面的关系代数应用）

## 第3题--参考答案

### 3. 关于两个关系之间的‘笛卡尔积’ (product) 运算 (参考答案)

- ① 什么样的两个关系，可以在它们之间执行笛卡尔积运算？
- ② 在两个关系之间执行笛卡尔积运算，结果关系的关系模式和结果元组集合分别是什么？
- ③ 笛卡尔积运算在关系数据库访问中有什么实际作用？

① 任意两个关系之间，都可以执行笛卡尔积运算。

② 设有一个 $n$ 目关系 $R$ 和 $m$ 目关系 $S$ ，那么笛卡尔积  $R \times S$  的结果是一个 $(n+m)$ 目的关系，关系 $R$ 和关系 $S$ 的所有属性都会被投影到它们的笛卡尔积的结果关系中。

来自关系 $R$ 的每一个元组 $u$ 和来自关系 $S$ 的每一个元组 $v$ ，都会组合得到 $R \times S$ 结果关系中的一个元组，可形式化表示为： $R \times S = \{ (u, v) \mid u \in R \wedge v \in S \}$

③ 笛卡尔积是关系代数中的一个基本运算符。当数据查询需要使用到不同关系时，通常可以先通过笛卡尔积运算将多个关系合并成一个关系，然后再用单个关系上的选择和投影运算实现用户的数据查询请求，实现跨关系的数据访问。

## 第4题--参考答案 1

第1组	$Q_1: \sigma_{pid = 'p01'}(orders)$	$Q_2: \sigma_{pid \neq 'p01'}(orders)$
-----	-------------------------------------	--

□ 查询结果集的语义是什么？

$Q_1$ : p01号商品的所有购买‘订单’

$Q_2$ : 除了p01号商品外，其他所有商品的购买‘订单’

□ 两个结果集的关系是什么？

$Q_1 \cap Q_2 = \text{空集}$

$Q_1 \cup Q_2 = \text{Orders}$



## 第4题--参考答案 2

第2组	$Q_3: \pi_{cid}(\sigma_{pid = 'p01'}(orders))$	$Q_4: \pi_{cid}(\sigma_{pid \neq 'p01'}(orders))$
-----	--	---

### □ 查询结果集的语义是什么？

$Q_3$ : 购买过p01号商品的‘顾客(编号)’

$Q_4$ : 购买过其他商品(pid不等于p01)的‘顾客(编号)’

### □ 两个结果集的关系是什么？

- 两个结果集之间的关系，取决于当前orders表中的订单情况
- 两个结果集之间可能有交集，也可能没有交集，甚至于一个结果集包含着另一个结果集，一切皆有可能！
- $Q_3 \cup Q_4 = \pi_{cid}(orders)$

## 第4题--参考答案 3

第3组	Q <sub>5</sub>	$\pi_{cid}(\mathbf{orders}) - \pi_{cid}(\sigma_{pid = 'p01'}(\mathbf{orders}))$
	Q <sub>6</sub>	$\pi_{cid}(\mathbf{orders}) - \pi_{cid}(\sigma_{pid \neq 'p01'}(\mathbf{orders}))$

□ 查询结果集的语义是什么？

Q<sub>5</sub>: 购买过商品但没有购买过p01号商品的‘顾客(编号)’

Q<sub>6</sub>: 只购买过p01号商品的‘顾客(编号)’

□ 两个结果集的关系是什么？

$$Q_5 \cap Q_6 = \text{空集}$$

$$Q_5 \subseteq Q_4$$

$$Q_4: \pi_{cid}(\sigma_{pid \neq 'p01'}(\mathbf{orders}))$$

$$Q_5 \cup Q_6 = \pi_{cid}(\mathbf{orders})$$

$$Q_6 \subseteq Q_3$$

$$Q_3: \pi_{cid}(\sigma_{pid = 'p01'}(\mathbf{orders}))$$

## 第4题--参考答案 4

第4组	$Q_7$	$\pi_{cid}(customers) - \pi_{cid}(\sigma_{pid = 'p01'}(orders))$
	$Q_8$	$\pi_{cid}(customers) - \pi_{cid}(\sigma_{pid \neq 'p01'}(orders))$

### □ 查询结果集的语义是什么？

$Q_7$ : 没有购买过p01号商品的顾客 (包括“没有购买过任何商品”的顾客)

$Q_8$ : “只购买过p01号商品”的顾客 和 “没有购买过任何商品”的顾客

### □ 两个结果集的关系是什么？

根据顾客的商品购买情况，将所有顾客划分为以下互不相交的四个子集：

- 没有购买过任何商品的顾客集合  $S_1$
- 只购买过p01这一种商品的顾客集合  $S_2$
- 没有购买过p01但购买过其他商品的顾客集合  $S_3$
- 既购买过p01又购买过其他商品的顾客集合  $S_4$

则:  $Q_7 = S_1 \cup S_3$

$Q_7 \cap Q_8 = S_1$

$Q_8 = S_1 \cup S_2$

$Q_7 \cup Q_8 = S_1 \cup S_2 \cup S_3$

## 第4题--参考答案 5

第5组	$Q_9$	$orders - \sigma_{pid = 'p01'}(orders)$
	$Q_{10}$	$orders - \sigma_{pid \neq 'p01'}(orders)$

### □ 查询结果集的语义是什么？

$Q_9$ : 除了p01号商品之外，其他商品的所有销售订单

$Q_{10}$ : p01号商品的所有销售订单

### □ 两个结果集的关系是什么？

$$Q_9 \cap Q_{10} = \text{空集}$$

$$Q_9 \cup Q_{10} = orders$$

$$Q_9 = Q_2$$

$$Q_{10} = Q_1$$

$Q_1: \sigma_{pid = 'p01'}(orders)$
$Q_2: \sigma_{pid \neq 'p01'}(orders)$

(注: orders表中的pid不可能取空值)

## 第4题--参考答案 6

第6组	$Q_{11}$	$\pi_{cid}(orders - \sigma_{pid = 'p01'}(orders))$
	$Q_{12}$	$\pi_{cid}(orders - \sigma_{pid \neq 'p01'}(orders))$

### □ 查询结果集的语义是什么？

$Q_{11}$ : 购买过其他商品 (pid不等于'p01') 的顾客编号的集合

$Q_{12}$ : 购买过商品'p01'的顾客编号的集合

### □ 两个结果集的关系是什么？

就如同前面的 $Q_3$ 和 $Q_4$ 一样,  $Q_{11}$ 和 $Q_{12}$ 的结果集之间也没有确定的相互关系, 一切取决于当前orders表中的订单情况。

- $Q_{11}$ 是“购买过其他商品”的顾客集合, 其中的顾客有的可能也购买过'p01', 也有的可能没有购买过'p01';
- $Q_{12}$ 是“购买过商品p01”的顾客集合, 其中的顾客可能也购买过其他商品, 也可能没有购买过其他商品。

# 总结 & 思考

(注: '其他商品' 是指商品编号pid不等于'p01'的商品)

	查询表达式	目标对象	目标对象需要满足的条件
Q <sub>1</sub>	$\sigma_{pid='p01'}(orders)$	订单	'p01'号商品的销售订单
Q <sub>2</sub>	$\sigma_{pid \neq 'p01'}(orders)$	订单	其他商品的销售订单
Q <sub>3</sub>	$\pi_{cid}(\sigma_{pid='p01'}(orders))$	顾客	该顾客购买过'p01'号商品
Q <sub>4</sub>	$\pi_{cid}(\sigma_{pid \neq 'p01'}(orders))$	顾客	该顾客购买过其他商品
Q <sub>5</sub>	$\pi_{cid}(orders) - \pi_{cid}(\sigma_{pid='p01'}(orders))$	顾客	该顾客没有购买过'p01', 但购买过其他商品
Q <sub>6</sub>	$\pi_{cid}(orders) - \pi_{cid}(\sigma_{pid \neq 'p01'}(orders))$	顾客	该顾客只购买过'p01', 但没有购买过其他商品
Q <sub>7</sub>	$\pi_{cid}(customers) - \pi_{cid}(\sigma_{pid='p01'}(orders))$	顾客	该顾客没有购买过'p01'号商品 (请注意与Q <sub>5</sub> 的区别)
Q <sub>8</sub>	$\pi_{cid}(customers) - \pi_{cid}(\sigma_{pid \neq 'p01'}(orders))$	顾客	该顾客没有购买过其他商品 (请注意与Q <sub>6</sub> 的区别)
Q <sub>9</sub>	$orders - \sigma_{pid='p01'}(orders)$	订单	其他商品的销售订单
Q <sub>10</sub>	$orders - \sigma_{pid \neq 'p01'}(orders)$	订单	'p01'号商品的销售订单
Q <sub>11</sub>	$\pi_{cid}(orders - \sigma_{pid='p01'}(orders))$	顾客	购买过其他商品的顾客 (请注意与Q <sub>5</sub> 的区别)
Q <sub>12</sub>	$\pi_{cid}(orders - \sigma_{pid \neq 'p01'}(orders))$	顾客	购买过'p01'号商品的顾客 (请注意与Q <sub>6</sub> 的区别)