

一、假设存在如下的关系模式

- 顾客 C (cid, cname, city, dis)
- 供应商 A (aid, aname, city, per)
- 商品 P (pid, pname, city, qua, price)
- 订单 O (ordno, orddate, cid, aid, pid, qty, dols)

其中：orddate 是一个日期类型的属性，用于记录订单的创建日期，日期类型的值可以进行大小比较。请用关系代数来表示下述数据查询要求。

1) 只购买过一次商品的顾客的编号；

令 $R := O, S := O$

$O[cid] - ((R \times S) \text{ where } R.cid=S.cid \text{ and } R.ordno <> S.ordno) [R.cid]$

说明：“只购买过一次商品”意思是“这个客户只有一份订单”

2) 在所有有客户的城市中都销售过商品的供应商的编号（请写出两种不同的表示方法：使用除法和不使用除法）；

答案 1（使用除法）

$(C \text{ join } O) [aid, city] \div C[city]$

答案 2（借鉴‘除法’推导公式的思路）

$O[aid] - ((C \times A) [A.aid, C.city] - (O \text{ join } C) [O.aid, C.city]) [aid]$

具体思路如下：

- (1) 所有供应商和所有客户城市的配对关系： $X := (A[aid] \times C[city])$
- (2) 供应商及其客户所在城市的名称： $Y := (O \text{ join } C)[O.aid, C.city]$
- (3) 供应商及其没有向其中的客户销售过商品的城市名称： $Z := X - Y$
- (3) 出现在Z中的供应商即是不满足本题查询要求的供应商

所以最终的查询结果是： $O[aid] - Z[aid]$

3) 查询满足下述条件的顾客 c 的编号：对于每一个“销售过所有商品”的供应商 a，客户 c 都通过 a 购买过商品；

$O[cid, aid] \div (O[aid, pid] \div P[pid])$

4) 查询每一个供应商单笔销售金额最高的订单（即：在一个供应商的所有销售订单中，订单金额 dols 最高的订单），结果返回供应商编号，订单编号，订单金额；

(1) 查询每个供应商订单金额 dols 并非最高的订单：令 $O_1 := O, O_2 := O$

$H := ((O_1 \times O_2) \text{ where } O_1.aid = O_2.aid \text{ and } O_1.dols < O_2.dols) [O_1.aid, O_1.ordno, O_1.dols]$

(2) 查询每个供应商销售金额最高的订单：

$O[aid, ordno, dols] - H$

- 5) 查询每一个客户最后两次订单的订购日期, 结果关系包括三个属性: 客户编号, 该客户的最后两次订单的日期; (**orddate** 越大, 订单越后)

Step 1: 查询每个客户的非最后一次订单 H

令 $O_1 := O, O_2 := O$

$H := ((O_1 \times O_2) \text{ where } O_1.cid=O_2.cid \text{ and } O_1.orddate < O_2.orddate)[O_1.cid, O_1.ordno, O_1.orddate]$

Step 2: 查询每个客户的最后一份订单 R

$R(cid, lastord, lastdate) := O[cid, ordno, orddate] - H$

Step 3: 查询每个客户的倒数第二份订单 T (即在 H 中再查每个客户最后一份订单)

令 $M := H, N := H$

$T(cid, ordno, orddate) :=$

$H - ((M \times N) \text{ where } M.cid=N.cid \text{ and } M.orddate < N.orddate)[M.cid, M.ordno, M.orddate]$

Step 4: 将 R 和 T 进行连接, 得到最终的查询结果

$(R \text{ join } T)[R.cid, R.lastdate, T.orddate]$

二、假设关系模式如下:

- 学生 S(学号 sno, 姓名 sname, 就读院系 dept)
- 教师 T(工号 tno, 姓名 tname, 工作院系 dept)
- 课程 C(课程号 cno, 课程名 cname, 开课院系 dept)
- 选课 L(学号 sno, 课程号 cno, 授课教师工号 tno, 成绩 grade, 选修年份 year)

其中: 同一门课同一个学生只能有一条选课记录。请用关系代数来表示下述数据访问请求。

1. 查询满足下述条件的学生的学号和姓名: 选修过‘计算机’系开设的课程;

$((C \text{ where dept='计算机'})[cno] \text{ join } L) \text{ join } S)[sno, sname]$

注: 关系 C 和 S 都有属性 dept, 在使用自然连接运算前要添加‘投影’运算至少过滤掉其中的一个。

2. 查询满足下述条件的学生的学号和姓名: 所有的课程成绩都及格 (成绩 ≥ 60);

Step 1: 至少有一门课程的成绩低于 60 分的学生

$M := ((S \text{ join } L) \text{ where grade} < 60)[S.sno, S.sname]$

Step 2: 选修过课程的学生

$N := (S \text{ join } L)[S.sno, S.sname]$

Step 3: 所有课程都及格的学生: $N - M$, 将上式代入后得到:

$(S \text{ join } L)[S.sno, S.sname] - ((S \text{ join } L) \text{ where grade} < 60)[S.sno, S.sname]$

【注】另一种表示方法如下: $((L[sno] - (L \text{ where grade} < 60)[sno]) \text{ join } S)[S.sno, S.sname]$

3. 查询满足下述条件的学生的学号和姓名：选修了‘数学’系的所有课程且成绩都合格；

$(L \text{ where grade} \geq 60) [sno, cno] \div (C \text{ where dept='数学'}) [cno]$

【注】这里是指所选修的数学系的课程都及格。如果将本查询的语义理解为“所有的课程成绩都及格”，可使用以下的查询表达式：

$(L [sno, cno] \div (C \text{ where dept='数学'}) [cno]) - (L \text{ where grade} < 60) [sno]$

4. 查询满足下述条件的教师的编号：只讲授过自己工作院系开设的课程；

Step 1: 讲授过其他院系开设课程的教师

$M := ((T \times L \times C) \text{ where } T.tno = L.tno \text{ and } L.cno = C.cno \text{ and } T.dept \neq C.dept) [T.tno]$

Step 2: 讲授过课程的教师 $N := L [tno]$

Step 3: 只讲授过本院系开设课程的教师： $N - M$ ，将上式代入后得到：

$L [tno] - ((T \times L \times C) \text{ where } T.tno = L.tno \text{ and } L.cno = C.cno \text{ and } T.dept \neq C.dept) [T.tno]$

5. 查询每一个教师最后一次的授课年份，结果返回教师的工号、最后一次授课的年份以及所讲授课程的课程号。

Step 1: 查询每个教师的授课历史记录 H（非最后一年的授课纪录）

令 $M := L, N := L$

$H := ((M \times N) \text{ where } M.tno = N.tno \text{ and } M.year < N.year) [M.tno, M.year, M.cno]$

Step 2: 每个教师最后一年的授课记录为 $L[tno, year, cno] - H$ ，将上式代入得：

$L[tno, year, cno] - ((M \times N) \text{ where } M.tno = N.tno \text{ and } M.year < N.year) [M.tno, M.year, M.cno]$