

Examples of SQL

(division)

关系模式

顾客

Customers (cid, cname, city, discnt)

供应商

Agents (aid, aname, city, percent)

产品

Products (pid, pname, city, quantity, price)

订单

Orders (ordno, month, cid, aid, pid, qty, dollars)

关系代数中的除法运算

例1：通过所有供应商购买过商品的顾客的编号(cid)

$$\Pi_{cid,aid} (\mathbf{Orders}) \div \Pi_{aid} (\mathbf{Agents})$$

□ 在SQL中并没有提供除法运算符，该查询的语义如下：

- 如果客户c（或Customers表中的一个元组c）是一个满足结果要求的客户，那么：
- 对Agents表中的任意一个供应商a，客户c都应该通过供应商a购买过商品，即：可以在Orders表中找到一个能够满足以下条件的元组o: $o[cid]=c[cid]$ and $o[aid]=a[aid]$

关系代数中的除法运算

□ 我们也可以用另外一种方式来理解该查询要求：

- 如果客户 **c**（或 **Customers** 表中的一个元组 **c**）是一个满足结果要求的客户，那么在 **Agents** 表中，不存在没有为客户 **c** 购买过商品的供应商。
- 即：在 **Agents** 表中找不到符合以下要求的供应商 **a**
 - 没有为客户 **c** 购买过商品
 - 即：在 **Orders** 表中找不到供应商 **a** 为客户 **c** 购买商品的订单记录，也就是找不到能够满足以下要求的订单元组 **o**： $o[cid]=c[cid]$
 $\text{and } o[aid]=a[aid]$

关系代数中的除法运算

□ 首先，我们可以构造出第一个子查询：

查询供应商a为客户c购买商品的订单记录

```
q1:      select *  
           from orders o  
           where o.cid = c.cid and  
                  o.aid = a.aid
```

关系代数中的除法运算

□其次，利用子查询q1可以构造出第二个子查询：

查询没有“为客户c购买过商品”的供应商

即：查询能够使子查询q1的执行结果为空的供应商

```
q2:      select *  
          from agents a  
          where NOT EXISTS ( q1 )
```

关系代数中的除法运算

□最后，利用子查询q2来构造最终的查询命令：

— 查询不存在“没有为其购买过商品的供应商”
的客户的编号

即：查询能够使子查询q2的执行结果为空的客户

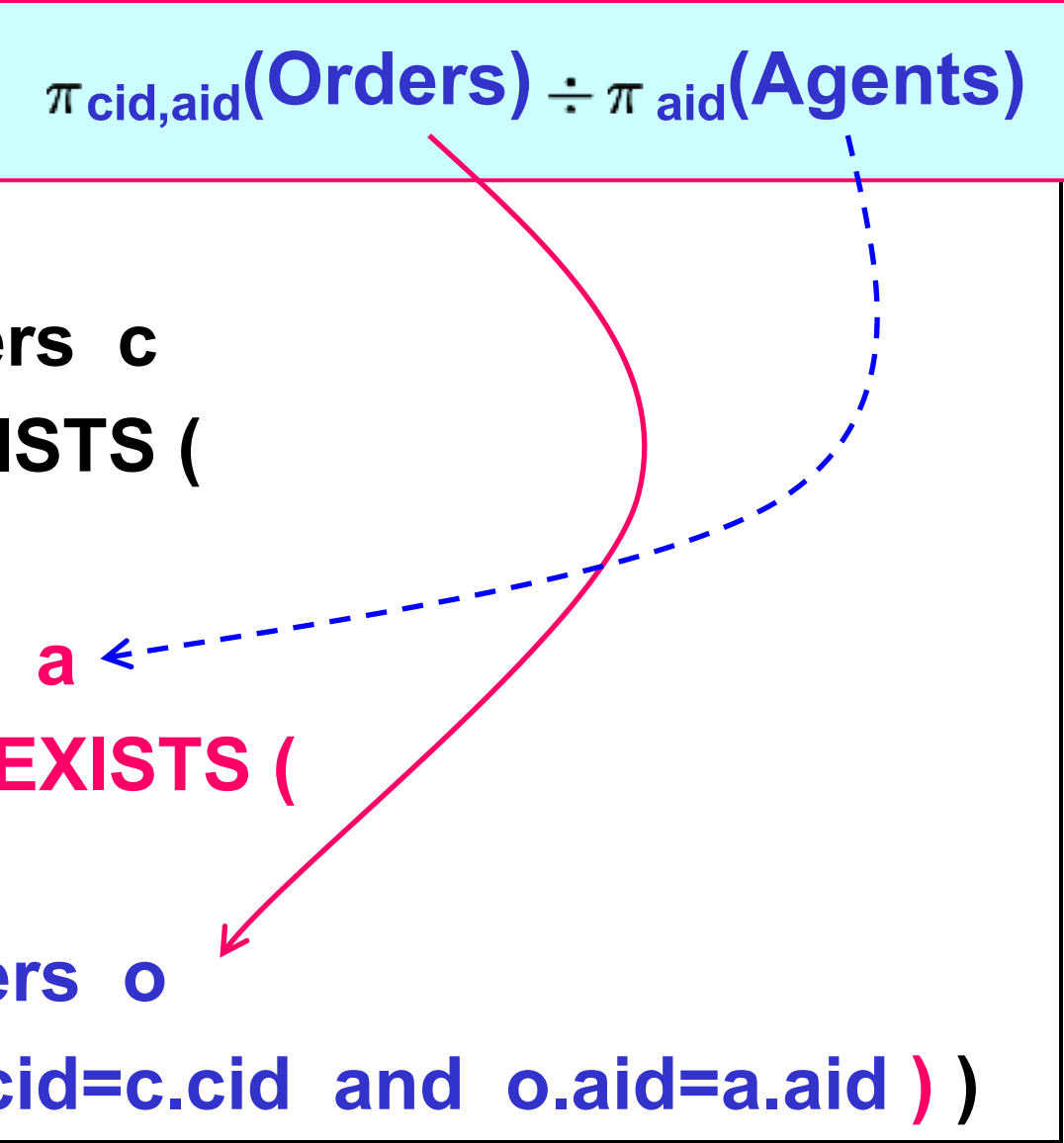
```
select c.cid  
from customers c  
where NOT EXISTS ( q2 )
```

关系代数中的除法运算

□ 将子查询q1和q2作

$\pi_{cid,aid}(\text{Orders}) \div \pi_{aid}(\text{Agents})$

```
SELECT c.cid
FROM customers c
WHERE NOT EXISTS (
  SELECT *
  FROM agents a
  WHERE NOT EXISTS (
    SELECT *
    FROM orders o
    WHERE o.cid=c.cid and o.aid=a.aid ) )
```



例1的另一种解题思路

例1：通过所有供应商购买过商品的顾客的编号(cid)

□（第二种思路）该查询的语义如下：

- 如果客户**c**是一个满足结果要求的客户，那么：
 - 假设**S**是所有为客户**c**购买过商品的供应商的编号的集合，则每一个供应商的编号都属于集合**S**，或者说：
 - 不存在不属于集合**S**的供应商编号

□按照上述思路，最终的查询命令如下(next slide)

关系代数中的除法运算

```
SELECT c.cid
FROM customers c
WHERE NOT EXISTS (
    SELECT *
    FROM agents a
    WHERE a.aid NOT IN (
        SELECT o.aid
        FROM orders o
        WHERE o.cid=c.cid ) )
```

向客户c
销售过商
品的供应
商的集合

关系代数中的除法运算

```
SELECT c.cid  
FROM customers c  
WHERE NOT EXISTS (  
    SELECT *  
    FROM agents a  
    WHERE a.aid NOT IN (  
        SELECT o.aid  
        FROM orders o  
        WHERE o.cid=c.cid ) )
```

没有向客户c
销售过商品的
供应商的集合

关系代数中的除法运算

```
SELECT c.cid
FROM customers c
WHERE NOT EXISTS (
    SELECT *
    FROM agents a
    WHERE a.aid NOT IN (
        SELECT o.aid
        FROM orders o
        WHERE o.cid=c.cid ) )
```

不存在 “没有向
客户c销售过商品
的供应商”

关系代数中的除法运算

例2：通过位于New York的所有供应商都购买过商品的顾客的编号(cid)

$$\Pi_{cid,aid} (\mathbf{Orders}) \div \Pi_{aid} (\delta_{city='New York'} (\mathbf{Agents}))$$

□ 该查询的语义如下：

- 如果客户c（或Customers表中的一个元组c）是一个满足结果要求的客户，那么：
- 对Agents表中的任意一个位于New York的供应商a，客户c都应该通过供应商a购买过商品，即：可以在Orders表中找到一个能够满足以下条件的元组o：o[cid]=c[cid] and o[aid]=a[aid]

关系代数中的除法运算

□ 我们也可以用另外一种方式来理解该查询要求：

- 如果客户 **c**（或 **Customers** 表中的一个元组 **c**）是一个满足结果要求的客户，那么在 **Agents** 表中，不存在 位于 New York 且没有为客户 **c** 购买过商品的供应商。
- 即：在 **Agents** 表中找不到符合以下要求的供应商 **a**：
 - 位于 New York
 - 且没有为客户 **c** 购买过商品

关系代数中的除法运算

□ 首先，我们可以构造出第一个子查询：

查询供应商a为客户c购买商品的订单记录

```
q1:      select *  
           from orders o  
           where o.cid = c.cid and  
                  o.aid = a.aid
```

关系代数中的除法运算

□其次，利用子查询q1可以构造出第二个子查询：

- 查询位于New York且没有“为客户c购买过商品”的供应商

```
q2:      select *  
          from agents a  
          where a.city = 'New York' and  
                NOT EXISTS ( q1 )
```


关系代数中的除法运算


□最后，利用子查询q2来构造最终的查询命令：

```
select c.cid  
from customers c  
where NOT EXISTS ( q2 )
```

□将子查询q1和q2代入后得到如下的查询命令
(next slide)

$\Pi_{cid,aid} (Orders) \div \Pi_{aid} (\sigma_{city='New York'} (Agents))$

```
SELECT c.cid
FROM customers c
WHERE NOT EXISTS (
    SELECT *
    FROM agents a
    WHERE a.city = 'New York' and
        NOT EXISTS (
            SELECT *
            FROM orders o
            WHERE o.cid=c.cid and o.aid=a.aid ) )
```



- ❑ 例3: 查询位于 ‘New York’ 或 ‘Duluth’ 且销售过所有价格大于等于1美圆的商品的供应商的编号

```
SELECT aid
FROM agents a
WHERE (city='New York' or city='Duluth') and
      NOT EXISTS (
        SELECT *
        FROM products p
        WHERE p.price >= 1 and NOT EXISTS (
          SELECT *
          FROM orders o
          WHERE o.aid=a.aid and o.pid=p.pid))
```

- ❑ 例4: 查询购买过 ‘c006’号顾客所购买过的所有商品的顾客的编号

```
SELECT cid
FROM   customers c
WHERE  NOT EXISTS (
    SELECT *
    FROM   orders x
    WHERE  x.cid='c006' and NOT EXISTS (
        SELECT *
        FROM   orders y
        WHERE  y.cid=c.cid and y.pid=x.pid))
```

❑ 例5: 查询居住于 ‘Duluth’ 的顾客都购买过的商品的编号

```
SELECT pid
FROM   products p
WHERE  NOT EXISTS (
    SELECT *
    FROM   customers c
    WHERE  c.city='Duluth' and NOT EXISTS
        (
            SELECT *
            FROM   orders x
            WHERE  x.cid=c.cid and x.pid=p.pid))
```

- ❑ **例6: 查询购买过所有价格为\$0.50的商品的顾客的姓名**

```
SELECT cname
FROM customers c
WHERE NOT EXISTS (
    SELECT *
    FROM products p
    WHERE p.price=0.50 and NOT EXISTS (
        SELECT *
        FROM orders o
        WHERE o.cid=c.cid and o.pid=p.pid ) )
```

例7: 查询购买过所有被购买过的商品的顾客的编号

```
SELECT c.cid  
FROM customers c  
WHERE NOT EXISTS (  
    SELECT *  
    FROM orders o1  
    WHERE NOT EXISTS (  
        SELECT *  
        FROM orders o2  
        WHERE o2.cid=c.cid and o2.pid=o1.pid ))
```

- ❑ 例8: 查询销售过 ‘c004’号顾客购买过的所有商品的供应商的编号 (可以是销售给c004号顾客的, 也可以是销售给其他顾客的)

```
SELECT a.aid
FROM agents a
WHERE NOT EXISTS (
    SELECT *
    FROM orders o1
    WHERE o1.cid='c004' and NOT EXISTS (
        SELECT *
        FROM orders o2
        WHERE o2.aid=a.aid and o2.pid=o1.pid ))
```


Questions

1. 查询为所有折扣(**discnt**)大于8的客户都购买过商品的供应商的编号(**aid**)。
2. 查询满足下述条件的客户的编号:
 - ❑ 如果**c006**号客户通过供应商**y**购买过商品**x**, 那么该客户也通过供应商**y**购买过商品**x**。
3. 查询为所有购买过所有商品的客户都购买过商品的供应商的编号。
4. 查询为所有满足下述条件的客户都购买过商品的供应商的编号:
 - ❑ 该客户购买过所有价格超过1美元的商品

Q1:为所有折扣(discont)大于8的客户都购买过商品的供应商的编号(aid)。

```
SELECT aid  
FROM agents a  
WHERE not exists (  
    SELECT *  
    FROM customers c  
    WHERE c.discont > 8 and not exists (  
        SELECT *  
        FROM orders x  
        WHERE x.cid = c.cid and x.aid = a.aid ) );
```

Q2:查询满足下述条件的客户的编号： 如果c006号客户通过供应商y购买过商品x，那么该客户也通过供应商y购买过商品x。

```
SELECT cid
FROM   customers c
WHERE  NOT EXISTS (
    SELECT *
    FROM   orders x
    WHERE  x.cid='c006' and NOT EXISTS (
        SELECT *
        FROM orders y
        WHERE  y.cid = c.cid and
               y.pid = x.pid and
               y.aid = x.aid ) )
```

Q3:查询为所有购买过所有商品的客户都购买过商品的供应商的编号。

```
SELECT aid FROM agents a
WHERE not exists (
  SELECT * FROM customers c
  WHERE not exists (
    select * from products p
    where not exists (
      select * from orders y
      where y.cid=c.cid and y.pid=p.pid ) )
    and not exists (
      SELECT * FROM orders x
      WHERE x.cid = c.cid and x.aid = a.aid )
  );
```

购买过所有商品的客户

Q4:查询为所有满足下述条件的客户都购买过商品的供应商的编号：该客户购买过所有价格超过1美元的商品

```
SELECT aid FROM agents a
WHERE not exists (
    SELECT * FROM customers c
    WHERE not exists (
        select * from products p
        where p.price > 1 and not exists (
            select * from orders y
            where y.cid=c.cid and y.pid=p.pid ) )
    and not exists (
        SELECT * FROM orders x
        WHERE x.cid = c.cid and x.aid = a.aid )
);
```

【例】 查询至少修读学号为**S4**的学生所修读的所有课程的学生的学号