

第二章 关系数据库

关系演算

本次课学习内容

- 关系代数
 - 关系运算（选择、投影、连接，除运算）
- 关系演算
 - 以数理逻辑中的谓词演算为基础的。可分为：
 - 元组关系演算语言（get，put等6条语句）
 - 域关系演算语言QBE

一、关系代数

- 1、除运算
- 2、除运算示例
- 3、除运算练习

1、除运算÷（ Division ，亦称商 ）

- **用途**：除法运算同时从行和列的角度进行运算，在表达某些查询时有用，适合于包含“全部”之类的短语的查询，例如“查询选修了所有课程的学生名字”。
- **定义**：给定关系 $R(X, Y)$ ， $S(Y, Z)$ ， X, Y, Z 为属性列，关系 R 和关系 S 中的 Y 出自相同域集，则：

$$R \div S = \{t_r[X] \mid t_r \in R \wedge \Pi_Y(S) \subseteq Y_x\}$$

其中， Y_x 为 x 在 R 中的象集， $x = t_r[X]$ 。

象集 Z_x

给定一个 X 和关系 $R(X, Z)$ ， Z 为属性组。

当 $t[X]=x$ 时， x 在 R 中的**象集** (Images Set) 为：

$$Z_x = \{ t[Z] \mid t \in R, t[X]=x \}$$

它表示 R 中属性组 X 上值为 x 的诸元组在 Z 上分量的集合

| R | |
|-------|-------|
| x_1 | Z_1 |
| x_1 | Z_2 |
| x_1 | Z_3 |
| x_2 | Z_2 |
| x_2 | Z_3 |
| x_3 | Z_1 |
| x_3 | Z_3 |

- x_1 在 R 中的象集
 $Z_{x1} = \{Z_1, Z_2, Z_3\}$,
- x_2 在 R 中的象集
 $Z_{x2} = \{Z_2, Z_3\}$,
- x_3 在 R 中的象集
 $Z_{x3} = \{Z_1, Z_3\}$

2、除运算示例

例1：设关系 R 、 S 分别为下图的(a)和(b)， $R \div S$ 的结果为图(c)

| R | | |
|-------|-------|-------|
| A | B | C |
| a_1 | b_1 | c_2 |
| a_2 | b_3 | c_7 |
| a_3 | b_4 | c_6 |
| a_1 | b_2 | c_3 |
| a_4 | b_6 | c_6 |
| a_2 | b_2 | c_3 |
| a_1 | b_2 | c_1 |

(a)

| S | | |
|-------|-------|-------|
| B | C | D |
| b_1 | c_2 | d_1 |
| b_2 | c_1 | d_1 |
| b_2 | c_3 | d_2 |

(b)

| $R \div S$ |
|------------|
| A |
| a_1 |

(c)

分析

- 在关系R中, A可以取四个值 $\{a_1, a_2, a_3, a_4\}$
 a_1 的象集为 $\{(b_1, c_2), (b_2, c_3), (b_2, c_1)\}$
 a_2 的象集为 $\{(b_3, c_7), (b_2, c_3)\}$
 a_3 的象集为 $\{(b_4, c_6)\}$
 a_4 的象集为 $\{(b_6, c_6)\}$
- S 在 (B, C) 上的投影为
 $\{(b_1, c_2), (b_2, c_1), (b_2, c_3)\}$
- 只有 a_1 的象集包含了 S 在 (B, C) 属性组上的投影
所以 $R \div S = \{a_1\}$

写出 $R \div S =$

R

| A | B | C | D |
|----|---|---|---|
| a1 | 2 | 4 | 6 |
| a3 | 8 | 2 | 3 |
| a1 | 2 | 3 | 5 |
| a4 | 7 | 7 | 8 |

S

| C | D |
|---|---|
| 3 | 5 |
| 4 | 6 |

$R \div S$

| A | B |
|----|---|
| a1 | 2 |

例2：有关系A、B1、B2和B3，请分别计算A/B1、 A/B2及A/B3。

分析：在关系中， sno可以取四个值 {s1, s2, s3, s4}

s1的象集为 {p1, p2, p3, p4}

s2的象集为 {p1, p2}

s3的象集为 {p2}

s4的象集为 {p2, p4}

| sno | pno |
|-----|-----|
| s1 | p1 |
| s1 | p2 |
| s1 | p3 |
| s1 | p4 |
| s2 | p1 |
| s2 | p2 |
| s3 | p2 |
| s4 | p2 |
| s4 | p4 |

A

| pno |
|-----|
| p2 |

B1

| sno |
|-----|
| s1 |
| s2 |
| s3 |
| s4 |

A/B1

| pno |
|-----|
| p2 |
| p4 |

B2

| sno |
|-----|
| s1 |
| s4 |

A/B2

| pno |
|-----|
| p1 |
| p2 |
| p4 |

B3

| sno |
|-----|
| s1 |

A/B3

例4：查询至少选修1号课程和3号课程的学生学号。

解题思路：首先建立一个临时关系K，

然后求 $\pi_{Sno, Cno}(SC) \div K$

200215121象集{1, 2, 3}

200215122象集{2, 3}

$K = \{1, 3\}$

于是： $\pi_{Sno, Cno}(SC) \div K = \{200215121\}$

| Cno | |
|-----------|-----|
| 1 | |
| 3 | |
| Sno | Cno |
| 200215121 | 1 |
| 200215121 | 2 |
| 200215121 | 3 |
| 200215122 | 2 |
| 200215122 | 3 |

- **关系除法运算的步骤：**

- 1) 将被除关系属性分为象集属性和结果属性
- 2) 对象集属性进行投影—目标数据集
- 3) 将被除关系分组：**结果属性值一样的元组分为一组**
- 4) 找出结果集

3、除运算练习

1、已知关系r、s、w如图所示，计算 $T = (r \cup s) \div w \bowtie S$

r关系

| A | B | C |
|---|---|---|
| 1 | b | 2 |
| 2 | a | 2 |
| 1 | b | 1 |
| 3 | a | 3 |

s关系

| A | B | C |
|---|---|---|
| 2 | b | 1 |
| 3 | b | 1 |
| 3 | b | 2 |
| 3 | a | 2 |

w关系

| B | C |
|---|---|
| a | 2 |
| b | 1 |

解：T=

| A | B | C |
|---|---|---|
| 2 | b | 1 |
| 3 | b | 1 |
| 3 | b | 2 |
| 3 | a | 2 |

2、检索学习全部课程的学生学号

$$\pi_{\text{Sno}, \text{Cno}} (\mathbf{SC}) \div \pi_{\text{Cno}} (\mathbf{Course})$$

3、检索学习全部课程的学生学号和姓名

$$\begin{array}{cc} \pi_{\text{Sno}, \text{Cno}} (\mathbf{SC}) \div \pi_{\text{Cno}} (\mathbf{Course}) & \pi_{\text{Sno}, \text{Sname}} (\mathbf{Student}) \\ \pi_{\text{Sno}, \text{Sname}} (\mathbf{Student} & \pi_{\text{Sno}, \text{Cno}} (\mathbf{SC}) \div \pi_{\text{Cno}} (\mathbf{Course}) \end{array} \bowtie$$

4、检索所学课程包含学生S3所学课程的学生学号

$$\pi_{\text{Sno}, \text{Cno}} (\mathbf{SC}) \div \pi_{\text{Cno}} (\sigma_{\text{sno}='S3'} (\mathbf{SC}))$$

关系代数小结

- 传统的集合运算
 - 并、差、交、笛卡尔积
- 专门的关系运算
 - 选择、投影、连接、除
- 5种基本运算
 - 并、差、笛卡尔积、投影、选择

二、关系演算

- 关系演算是以数理逻辑中的谓词演算为基础的，通过谓词形式来表示查询表达式。
- 根据谓词变元的不同，可将关系演算分为元组关系演算和域关系演算。前者以元组为变量，简称元组演算；后者以域为变量，简称域演算。

- 元组关系演算

- Tuple Relational Calculus ， 简称TRC
- 元组关系演算语言ALPHA
- 元组关系表达式

- 域关系运算

- Domain Relational Calculus ， 简称DRC
- 域关系演算语言QUE

ALPHA语言

- 基本格式是：〈操作符〉〈工作空间名〉(〈目标表〉)[:〈操作条件〉]
 - 操作符有GET、PUT、HOLD、UPDATE、DELETE，DROP等6条语句。
 - 工作空间是指内存空间，**通常用W表示**，也可以用别的字母表示。工作空间是用户与系统的通信区。
 - 目标表用于指定操作（如查询、更新等）出来的结果，它可以是关系名或属性名，一条语句可以同时多个关系或多个属性进行操作。
 - 只有满足操作条件的元组才能进行，操作条件可以为空。
 - 除此之外，还可以在基本格式上加上排序要求，定额要求等。

1) 简单查询

- 下面以学生课程数据库为例，说明ALPHA语言的使用。

例1：查询所有学生的数据

GET W (Student)

- GET语句的作用是把数据库中的数据读入内存空间W，目标表为学生关系Student，代表查询出来的结果，即所有的学生。
- 冒号后面的操作条件缺省，表示无条件查询。

例2：查询所有被选修的课程号码。

GET W (SC.Cno)

- 目标表为选课关系SC中的属性CNO，代表所有被选修的课程号码，查询结果自动消去重复行。

2) 条件查询

- 由冒号后面的逻辑表达式给出查询条件，在表达式中可以使用如下三类运算符：
 - ① 比较运算符： $>$ ， \geq ， $<$ ， \leq ，=等于， \neq ；
 - ② 逻辑运算符： \wedge （与）， \vee （或）， \neg （非）
 - ③ 表示执行次序的括号：（）
- 其中，比较运算符的优先级高于逻辑运算符，可以使用（）改变它们的优先级。

例1：查询所有男学生的姓名。

GET W (Student.Sname) : Student. Ssex = '男'

例2：查询信息系(IS)中年龄小于20岁的学生的学号和年龄

GET W (Student.Sno , Student.Sage):

Student.Sdept='IS' \wedge Student.Sage < 20

例3：查询信息系(IS)或计算机系(CS)的男学生

GET W (Student):

(Student.Sdept='IS' \vee Student.Sdept='CS') \wedge Student. Ssex = '男'

3) 排序检索

例1：查询S3同学所选课程号及成绩，并按成绩降序排列。

```
GET W (SC.CNO,SC.SCORE):SC.SNO='S3' DOWN SC.SCORE
```

例2：查询计算机科学系(CS)学生，按年龄升序排序

```
GET W (Student):Student.Sdept='CS' up Student.Sage
```

- DOWN表示降序，后面紧跟排序的属性名。
- 升序排列时使用UP。

4) 限额检索

例1：查询一名男同学。

GET W (1) (Student) : Student. Ssex = '男'

例2：查询一名男同学，并使他的年龄最小。

GET W (1) (Student) :

Student. Ssex = '男'up Student.Sage

- 所谓的定额查询就是通过在W后面的括号中加上定额数量，限定查询出元组的个数。
- 这里(1)表示查询结果中男同学的个数，取出学生表中第一个男同学的学号和姓名。

5) 使用聚集函数

- 用户在使用查询语言时，经常要作一些简单的运算。
- 例如要统计某个关系中符合某一条件的元组数，或某些元组在某个属性上分量的和、平均值等等。
- 在关系数据库语言中提供了有关这类运算的标准函数，增强了基本检索能力。
- 常用的函数右表所示：

| 函数名称 | 功能 |
|--------------|-----------|
| AVG | 按列计算平均值 |
| TOTAL | 按列计算值的总和 |
| MAX | 求一列中的最大值 |
| MIN | 求一列中的最小值 |
| COUNT | 按列值计算元组个数 |

例1：求学号为S1学生的平均分。

```
GET W (AVG(SC.SCORE)) :Student.SNO='S1'
```

例2：求学校共有多少个系

```
GET W (COUNT(Student.Sdept))
```

例3：求学生的最大年龄

```
GET W (max(Student.Sage))
```

- COUNT函数自动消去重复行，可计算字段“DEPT”不同值的数目。

数据更新

- 更新操作包括修改、插入和删除。

1) 修改操作

- 修改操作使用UPDATE语句实现，具体操作分为以下三步：
 - ①读数据：使用HOLD语句将要修改的元组从数据库中读到工作空间中；
 - ②修改：利用宿主语言修改工作空间中元组的属性；
 - ③送回：使用UPDATE语句将修改后的元组送回数据库中。
- 这里HOLD语句是带上并发控制的GET语句。

例1：把刘伟同学转到信息(IS)系。

```
HOLD W(Student.Sname ,Student.Sdept):Student.Sname='刘伟'
```

```
MOVE 'IS' TO W.Sdept
```

```
UPDATE W
```

- 在ALPHA语言中，不允许修改关系的主码，例如不能使用UPDATE语句修改学生表中的学生学号。
- 如果要修改主码，应该先使用删除操作删除该元组，再插入一条具有新主码值的元组。

2) 插入操作

- 插入操作使用**PUT语句**实现，具体操作分为以下两步：
 - ①建立新元组：利用宿主语言在工作空间中建立新元组；
 - ②写数据：使用PUT语句将元组写入到指定的关系中。

例1：在SC表中插入一条选课记录（S6，C1，85）。

MOVE 'S6' TO W.SNO

MOVE 'C1' TO W.CNO

MOVE 85 TO W.Grade

PUT W(SC)

- PUT语句的作用是把工作空间W中的数据写到数据库中，此例即把已经在工作空间建立的一条选课记录写入到选课关系SC中。
- 注意：**PUT语句只能对一个关系进行操作**，在插入操作时，拒绝接受主码相同的元组。

例2：学校新开设了一门2学分的课程“计算机组织与结构”，其课程号为8，直接先行课为6号课程。插入该课程元组

```
MOVE '8' TO W.Cno
```

```
MOVE '计算机组织与结构' TO W.Cname
```

```
MOVE '6' TO W.Cpno
```

```
MOVE 2 TO W.Ccredit
```

```
PUT W (Course)
```

3) 删除操作

- ALPHA语言中的删除操作不但可以删除关系中的一些元组，还可以删除一个关系。
- 删除操作使用DELETE语句实现，具体操作分为以下两步：
 - ①读数据：使用HOLD语句将要删除的元组从数据库中读到工作空间中；
 - ②删除：使用DELETE语句删除该元组。

例1：删除学号为S6的学生的信息

```
HOLD W(Student): Student.Sno='S6'
```

```
DELETE W
```

例2：删除全部学生的信息

```
HOLD W(Student)
```

```
DELETE W
```

为保证参照完整性, 删除Student中元组时相应地要删除SC中的元组

```
HOLD W (SC)
```

```
DELETE W
```

例3：将学号200215121改为200215126

HOLD W (Student): Student.Sno='200215121'

DELETE W

MOVE '200215126' TO W.Sno

MOVE '李勇' TO W.Sname

MOVE '男' TO W.Ssex

MOVE '20' TO W.Sage

MOVE 'CS' TO W.Sdept

PUT W (Student)

2、域关系演算语言QBE

- QBE是Query By Example的缩写，也称为**示例查询**，它是一种很有特色的**屏幕编辑语言**。
- 域关系演算是关系演算的另一种形式。
- 域关系演算是以元组变量的分量即域变量作为谓词变元的基本对象。
- 域关系演算语言的典型代表是1975年由IBM公司约克城高级研究试验室的M.M.Zloof提出的QBE语言，该语言于1978年在IBM370上实现。

QBE特点

(1) 以表格形式进行操作

- 每一个操作都由一个或几个表格组成，每一个表格都显示在终端的屏幕上，用户通过终端屏幕编辑程序以填写表格的方式构造查询要求，查询结果也以表格的形式显示出来，所以它具有直观和可对话的特点。

(2) 通过例子进行查询

- 通过使用一些实例，使该语言更易于为用户接受和掌握。

(3) 查询顺序自由

- 当有多个查询条件时，不要求使用者按照固定的思路 and 方式进行查询，使用更加方便。

使用QBE语言的步骤

- (1) 用户根据要求向系统申请一张或几张表格，显示在终端上；
- (2) 用户在空白表格的左上角的一栏内输入关系名；
- (3) 系统根据用户输入的关系名，将在第一行从左至右自动填写各个属性名；
- (4) 用户在关系名或属性名下方的一格内填写相应的操作命令，操作命令包括：**P. (打印或显示)**、**U. (修改)**、**I. (插入)**、**D. (删除)**。如果要打印或显示整个元组时，应将“P”填在关系名的下方，如果只需打印或显示某一属性，应将“P”填在相应属性名的下方。

| | | | | |
|------|--------------|--------------|-----|--------------|
| 关系名 | 属性1 | 属性2 | ... | 属性n |
| 操作命令 | 属性值或 查询条件 | 属性值或 查询条件 | ... | 属性值或 查询条件 |

QBE操作框架表

1) 简单查询

例1：显示全部学生的信息

- 方法一：将P.填在关系名的下方

| Student | Sno | Sname | Ssex | Sage | Sdept |
|---------|-----|-------|------|------|-------|
| P. | | | | | |

- 方法二：将“P.”填在各个属性名的下方

| Student | Sno | Sname | Ssex | Sage | Sdept |
|---------|-----------------|--------------|-------------|--------------|--------------|
| | P. <u>95001</u> | P. <u>赵亦</u> | P. <u>女</u> | P. <u>17</u> | P. <u>IS</u> |

- P. 是操作符，表示打印（`print`），实际上是显示。
- T 是示例元素，即域变量，示例元素是这个域中可能的一个值，它不必是查询结果中的元素。QBE要求示例元素下面加下划线。
- 只有目标属性包括所有的属性时，将P. 填在关系名的下方。

2) 条件查询

例1：查询所有女学生的姓名

| Student | Sno | Sname | Ssex | Sage | Sdept |
|---------|-----|--------------|------|------|-------|
| | | P. <u>赵亦</u> | 女 | | |

- 目标属性只有姓名，所以将P. 填在关系名的下方。
- 查询条件中可以使用比较运算符 $>$ ， \geq ， $<$ ， \leq ， $=$ 和 \neq ，其中 $=$ 可以省略。本例的查询条件是SEX='女'， $=$ 被省略。

例2：查询年龄大于18岁的女学生的姓名

方法1：把两个条件写在同一行上：

| Student | Sno | Sname | Ssex | Sage | Sdept |
|---------|-----|--------------|------|------|-------|
| | | P. <u>赵亦</u> | 女 | >18 | |

方法2：把两个条件写在不同行上，但必须使用相同的示例元素。

| Student | Sno | Sname | Ssex | Sage | Sdept |
|---------|-----|--------------|------|------|-------|
| | | P. <u>赵亦</u> | 女 | | |
| | | P. <u>赵亦</u> | | >18 | |

例3：查询年龄大于18岁或者女学生的姓名。

| Student | Sno | Sname | Ssex | Sage | Sdept |
|---------|-----|-------|------|------|-------|
| | | P.赵亦 | 女 | | |
| | | P.李亦 | | >18 | |

- 在QBE中，表示两个条件的“或”，要把两个条件写在不同行上，且必须使用不同的示例元素。

例4：查询选修C1号课程学生的姓名。

| Student | Sno | Sname | Ssex | Sage | Sdept |
|---------|-----------|-------|------|------|-------|
| | <u>S1</u> | P.赵亦 | | | |

| SC | Sno | Cno | Grade |
|----|-----------|-----|-------|
| | <u>S1</u> | C1 | |

本查询涉及两个关系：Student和SC，这两个关系具有公共的属性Sno，Sno作为连接属性，把具有相同的SNO值的两个关系连接起来，Sno在两个表中的值要相同。

例5：查询未选修C1号课程学生的姓名。

| Student | Sno | Sname | Ssex | Sage | Sdept |
|---------|-----------|-------|------|------|-------|
| | <u>S1</u> | P.赵亦 | | | |

| SC | Sno | Cno | Grade |
|--------|-----------|-----|-------|
| \neg | <u>S1</u> | C1 | |

- 查询条件中的“未选修”需使用逻辑非来表示。QBE中的逻辑非运算符为 \neg ，填写在关系名下方。
- 如果S1同学选修C1号课程的情况为假，则符合查询的条件，显示S1同学的学号，然后再查询其他同学。

3) 排序查询

例1：查询全体女同学的学号和姓名，要求查询结果按年龄降序排列。

| Student | Sno | Sname | Ssex | Sage | Sdept |
|---------|-----------------|--------------|------|------|-------|
| | P. <u>95001</u> | P. <u>赵亦</u> | 女 | DO | |

- 对查询结果按照某个属性值升序排列时，则在相应的属性下方填入“AO”，降序排列时，填入“DO”。
- 如果按照多个属性值同时排序，则用“AO(i)”或“DO(i)”表示，其中I为排序的优先级，I值越小，优先级越高。

- 例2：查全体男生的姓名，要求查询结果按所在系升序排序，对相同系的学生按年龄降序排序。

| Student | Sno | Sname | Ssex | Sage | Sdept |
|---------|-----|--------------|------|----------|----------|
| | | P. <u>李勇</u> | 男 | D0 (2) . | A0 (1) . |

4) 使用聚集函数

- 同ALPHA语言类似，QBE语言也提供了一些有关运算的标准函数，以方便用户。

| 函数名称 | 功能 |
|------|-----------|
| AVG | 按列计算平均值 |
| SUM | 按列计算值的总和 |
| MAX | 求一列中的最大值 |
| MIN | 求一列中的最小值 |
| CNT | 按列值计算元组个数 |

QBE常用的库函数及其功能

- 例1：求学号为S1学生的平均分。

| SC | Sno | Cno | Grade |
|----|-----|-----|-------------|
| | S1 | | P. AVG. ALL |

- 例2：求信息（IS）系学生的平均年龄。

| Student | Sno | Sname | Ssex | Sage | Sdept |
|---------|-----|-------|------|-------------|-------|
| | S1 | | | P. AVG. ALL | |

数据更新

- 修改：修改的命令为 **U.**
- 插入：插入的命令为 **I.**
- 删除：删除的命令为 **D.**

1) 修改操作

- 例1：把刘伟同学转到信息系。

| Student | Sno | Sname | Ssex | Sage | Sdept |
|---------|-----|-------|------|------|-------|
| U. | | 刘伟 | | | IS |

- 例2：把刘伟同学年龄改为18岁。

| Student | Sno | Sname | Ssex | Sage | Sdept |
|---------|-----|-------|------|-------|-------|
| | | 刘伟 | | U. 18 | |

例3：把200215121学生的年龄增加1岁

| Student | Sno | Sname | Ssex | Sage | Sdept |
|---------|-----------|-------|------|--------------|-------|
| U. | 200215121 | | | <u>17</u> | |
| | 200215121 | | | <u>17</u> +1 | |

例4：将所有学生年龄增加1岁

| Student | Sno | Sname | Ssex | Sage | Sdept |
|---------|------------------|-------|------|--------------|-------|
| U. | <u>200215121</u> | | | <u>18</u> | |
| | <u>200215121</u> | | | <u>18</u> +1 | |

2) 插入操作

例1：在SC表中插入一条选课记录（S6，C1）

| SC | Sno | Cno | Grade |
|----|-----|-----|-------|
| I. | S6 | C1 | |

注意：新插入的元组必须具有主码值，其他属性值可以为空，如本例中的Grade为空。

3) 删除操作

- 例1：删除S1同学选修C1课程的信息。

| SC | Sno | Cno | Grade |
|----|-----|-----|-------|
| D. | S1 | C1 | |

- 例2：删除删除学生200215089同学的信息。

| Student | Sno | Sname | Ssex | Sage | Sdept |
|---------|-----------|-------|------|------|-------|
| D. | 200215089 | | | | |

第二章 关系数据库小结

- 关系数据结构

- 关系（域，笛卡尔积，关系）
 - 关系，属性，元组，候选码，主码，主属性，基本关系的性质
- 关系模式， 关系数据库

- 关系操作

- 查询（选择、投影、连接、除、并、交、差）
- 数据更新（插入、删除、修改）

- 关系的完整性约束

- 实体完整性
- 参照完整性（外码）
- 用户定义的完整性

- 关系数据语言

- 关系代数语言
- 关系演算语言
 - 元组关系演算语言ALPHA
 - 域关系演算语言QBE