# 数据库系统概论

An Introduction to Database System

第二章 关系数据库

中国人民大学信息学院





## 关系演算

- ❖ 关系演算 以数理逻辑中的谓词演算为基础
- ❖ 按谓词变元不同 进行分类
  - 1.元组关系演算: 以元组变量作为谓词变元的基本对象 元组关系演算语言ALPHA
  - 2.域关系演算: 以域变量作为谓词变元的基本对象 域关系演算语言QBE

#### 2.5.1 元组关系演算语言ALPHA

#### 语句格式

GET 工作空间名 (表达式1) [:条件] [DOWN | UP 表达式2]

表达式1: 指定语句的操作对象

格式: 关系名| 关系名. 属性名| 元组变量. 属性名| 集函数 [, ...]

条件: 结果元组应该满足的条件

格式:逻辑表达式

表达式2: 指定排序方式

格式: 关系名. 属性名| 元组变量. 属性名[, ...]



# (1) 简单检索

GET 工作空间名 (表达式1)

[例1] 查询所有被选修的课程号码。 GET W (SC.Cno)

[例2] 查询所有学生的数据。 GET W (Student)



# (2) 限定的检索

GET 工作空间名 (表达式1):条件

[例3]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' ∧ Student.Sage<20

### (3) 带排序的检索

GET 工作空间名(表达式1)[: 条件]
DOWN | UP 表达式2

[例4]查询计算机科学系(CS)学生的学号、年龄,

结果按年龄降序排序

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

Student.Sdept='CS' DOWN Student.Sage

### (4) 带定额的检索

GET 工作空间名(定额)(表达式1)

[: 条件] [DOWN | UP 表达式2]

[例5]取出一个信息系学生的学号

GET W (1) (Student.Sno): Student.Sdept='IS'

[例6]查询信息系年龄最大的三个学生的学号及其年龄,结果按年龄降序排序。

GET W (3) (Student.Sno, Student.Sage):

Student.Sdept='IS' DOWN Student.Sage

### (5) 用元组变量的检索

#### 元组关系演算:以元组变量作为谓词变元的基本对象

- ❖ 元组变量的含义 表示可以在某一关系范围内变化 也称为范围变量Range Variable
- ❖ 元组变量的用途
  - ① 简化关系名: 设一个较短名字的元组变量来代替较长的关系名。
  - ②条件中使用量词时必须用元组变量。
- ❖ 定义元组变量
  - 格式: RANGE 关系名 变量名
  - ■一个关系可以设多个元组变量

# (6) 用存在量词的检索

[例8] 查询选修2号课程的学生名字。

RANGE SC X

**GET W (Student.Sname):** 

∃X(X.Sno=Student.Sno \ X.Cno='2')

[例9] 查询选修了这样课程的学生学号,其直接先行课是6号课程。

**RANGE Course CX** 

GET W (SC.Sno):

∃CX (CX.Cno=SC.Cno \ CX.Pcno='6')



# 用存在量词的检索(续)

[例10]查询 至少选修一门其先行课为6号课程的学生名字

```
RANGE Course CX
       SC
             SCX
GET W (Student.Sname):
    ∃SCX (SCX.Sno=Student.Sno ∧
         ∃CX (CX.Cno=SCX.Cno \ CX.Pcno='6') )
前束范式形式:
GET W (Student.Sname):
   ∃SCX∃CX (SCX.Sno=Student.Sno∧
            CX.Cno=SCX.Cno \ CX.Pcno='6')
```



#### (7) 带有多个关系的表达式的检索

[例11] 查询成绩为90分以上的学生名字与课程名字。 RANGE SC SCX GET W (Student.Sname, Course.Cname): ∃SCX (SCX.Grade≥90 /\ SCX.Sno=Student.Sno ∧ Course.Cno=SCX.Cno)

### (8) 用量词的检索

[例12] 查询不选1号课程的学生名字

用存在量词表示:

RANGE SC SCX

GET W (Student.Sname): ¬∃SCX

(SCX.Sno=Student.Sno \ SCX.Cno='1')

RANGE SC SCX

GET W (Student.Sname): ∀ SCX

(SCX.Sno≠Student.Sno VSCX.Cno≠'1')

#### (8) 用全称量词的检索

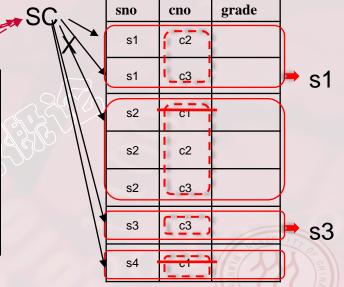
[例12] 查询不选1号课程的学生名字

RANGE SC SCX

GET W (Student.Sname): SCX

(SCX.Sno≠Student.Sno ∨ SCX.Cno≠'c1')

sno	Sname	00000000
s1		
s2		
s3		
s4		



# (9) 用两种量词的检索

[例13] 查询选修了全部课程的学生姓名。

**RANGE Course CX** 

SC SCX

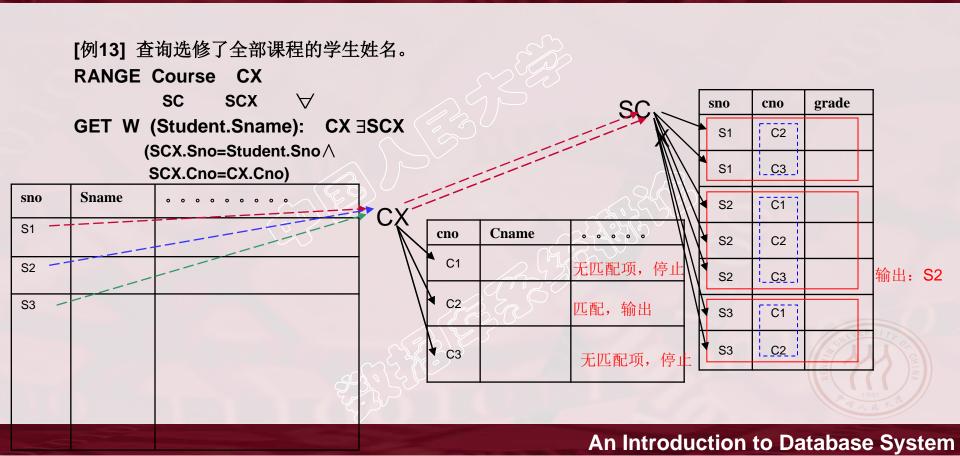
GET W (Student.Sname): ∀CX∃SCX

(SCX.Sno=Student.Sno∧

SCX.Cno=CX.Cno)



### (9) 用两种量词的检索



#### (10) 用蕴函(Implication)的检索

```
[例14] 查询最少选修了S3学生所选课程的学生学号
   RANGE Couse CX
          SC
            SCX
          SC SCY
   (∃SCX(SCX.Sno='S3 ' \ SCX.Cno=CX.Cno) ⇒
  ∃SCY(SCY.Sno=Student.Sno \ SCY.Cno=CX.Cno) )
蕴函:A B A→B 后件包含前件,则蕴函 T;后件不包含前件,则 蕴函 F
            前提为F
  F F T
```

前提为T

F

# (11) 聚集函数

常用聚集函数(Aggregation function)或内部函数(Build-in function)

函数名	功能
COUNT	对元组计数
TOTAL	求总和
MAX	求最大值
MIN	求最小值
AVG	求平均值

表2.5 关系演算中的聚集函数

# 聚集函数(续)

[例15] 查询学生所在系的数目。

GET W (COUNT(Student.Sdept))

COUNT函数在计数时会自动排除重复值。

[例16] 查询信息系学生的平均年龄

**GET W (AVG(Student.Sage):** 

Student.Sdept='IS')



