



引言

一个完整的数据查询语句的格式

SELECT [ALL DISTINCT] < 目标列表达式1 > [, < 目标列表达式2 > , ...]

FROM <表名或视图名1> [,<表名或视图名2> , ...]

[WHERE <元组选择条件表达式>]

[GROUP BY <属性列名1>[, <属性列名2>,...] [HAVING <组选择条件表达式>]]

[ORDER BY <目标列名1> [ASC|DESC] [, <目标列名2> [ASC|DESC], ...]]



SQL查询语句

SELECT A1, A2, ..., Am FROM R1, R2, ..., Rn WHERE F

实现关系代数中最常见的查询

$$\prod_{A_1,A_2,...,A_m} \sigma_F (R1 \times R2 \times ... \times Rn)$$



辦題內容

- 多表连接查询
- 外连接查询
- 3 自身连接查询



学生选课数据库

学生(学号,姓名,性别,出生时间,所在系)

课程(课程编号,课程名,先修课程号)

选课(学号,课程编号,成绩)

S (SNO, SN, SD, SB, SEX)

C (CNO, CN, PC)

SC ($\frac{SNO}{\sim\sim\sim}$, GRADE)



SELECT [ALL|DISTINCT] < 目标列表达式1 > [, < 目标列表达式2 > , ...] FROM <表名或视图名1 > [, <表名或视图名2 > , ...]

[WHERE <元组选择条件表达式>]

连接条件与元组选择条件进行AND(与)操作

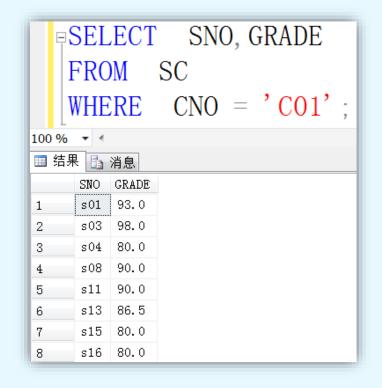
缺少连接条件,在多表的广义笛卡尔积进行查询



▶ 查询选修课程号为"C01"的学生学号和成绩

SELECT SNO, GRADE FROM SC WHERE CNO = 'C01';

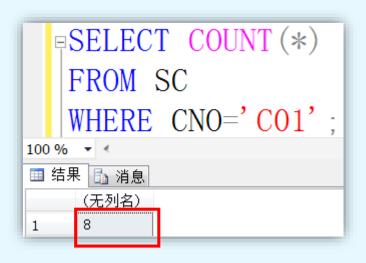
"学号"替换为"姓名"

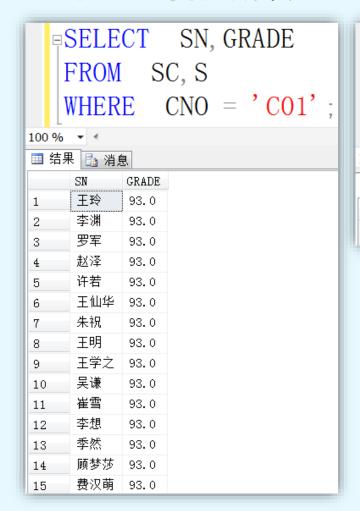


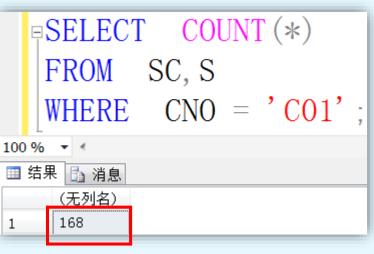


▶ 查询选修课程号为"C01"的学生姓名和成绩

SELECT SN, GRADE FROM SC, S WHERE CNO = 'C01';









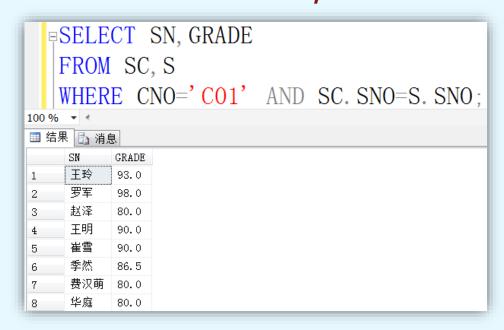
▶ 查询选修课程号为"C01"的学生姓名和成绩

SELECT SN, GRADE

FROM SC, S

SC 与S进行广义 笛卡尔积运算

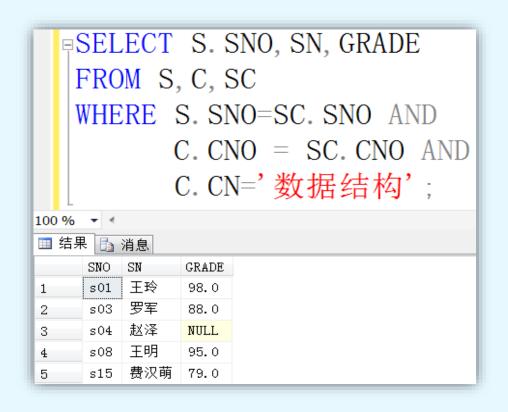
WHERE CNO = 'C01' AND SC.SNO=S.SNO;





▶ 查询选修"数据结构"课程的学生的学号、姓名和成绩。

SELECT S.SNO, SN, GRADE
FROM S, SC, C
WHERE S.SNO=SC.SNO AND
C.CNO = SC.CNO AND
CN='数据结构';





外连接查询

 在SQL语句中,外连接在FROM子句中指定,其语法规则如下: FROM <左关系> LEFT | RIGHT | FULL [OUTER] JOIN <右关系>
 ON <连接条件>

说明:

- 左外连接: LEFT [OUTER] JOIN
- 右外连接: RIGHT [OUTER] JOIN
- 全外连接: FULL [OUTER] JOIN
- 连接条件可包括对参与连接的关系进行元组选择的条件



外连接查询

▶ 查询所有学生姓名及选修课程号为 "C01"的成绩,没有选修 该课程的学生,成绩显示为空。

SELECT SN, GRADE

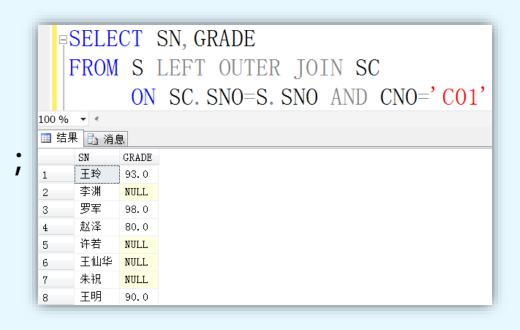
FROM SC, S

WHERE CNO = 'CO1' AND SC.SNO=S.SNO;



查询所有学生姓名及选修课程号为"C01"的成绩,没有选修该课程的学生,成绩显示为空。

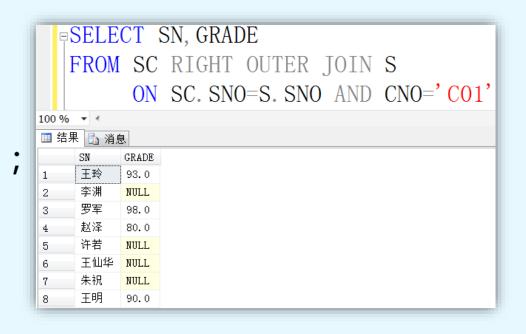
SELECT SN ,GRADE
FROM S LEFT OUTER JOIN SC
ON SC.SNO=S.SNO AND CNO= 'C01';





▶ 查询所有学生姓名及选修课程号为 "C01"的成绩,没有选修 该课程的学生,成绩显示为空。

SELECT SN ,GRADE
FROM SC RIGHT OUTER JOIN S
ON SC.SNO=S.SNO AND CNO= 'C01';





常规连接查询也称为内连接查询,在FROM子句中指定,其语法规则如下:

FROM <左关系> [INNER] JOIN <右关系> ON <连接条件>



▶ 查询选修"数据结构"课程的学生的学号、姓名和成绩。

SELECT S.SNO, SN, GRADE

FROM S, SC, C

WHERE S.SNO=SC.SNO AND C.CNO = SC.CNO AND CN='数据结构';

SELECT S.SNO, SN, GRADE

FROM (S INNER JOIN SC ON S.SNO=SC.SNO)

INNER JOIN C ON C.CNO=SC.CNO

WHERE CN='数据结构';

中国人民解放军陆军工程大学



自身连接查询

- 一个表与其自己进行连接
- 需要对表进行重命名
- 新表名作为表中属性的前缀



自身连接查询

▶ 查询每门课程的间接先修课程号(即先修课程的先修课程号)

课程表C

CNO	CN	PC
C01	高等数学	NULL
C0 2	数据约	c0 1
C0 3	操作系统	C02
C04	数据库	C03
C05	数据分析	C04



自身连接查询

▶ 查询每门课程的间接先修课程号(即先修课程的先修课程号)

课程表C FIRST

CNO	CN	PC
C01	高等数学	NULL
C02	数据结构	C01
C03	操作系统	C02
C04	数据库	C03
C05	数据分析	C04 -

课程表C SECOND

CNO	CN	PC
C01	高等数学	NULL
C02	数据结构	C01
C03	操作系统	C02
C04	数据库	C03
C05	数据分析	C04

SELECT FIRST.CNO, SECOND.PC

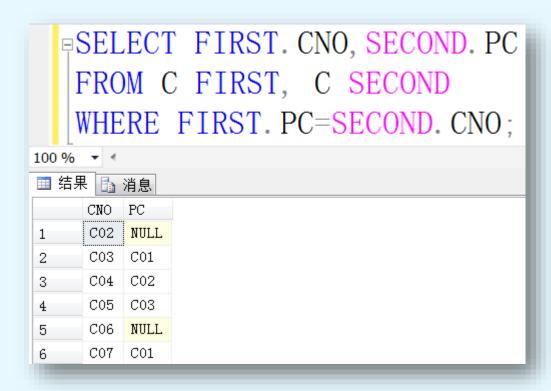
FROM C FIRST, C SECOND

WHERE FIRST.PC=SECOND.CNO;



▶ 查询每门课程的间接先修课程号(即先修课程的先修课程号)

SELECT FIRST.CNO, SECOND.PC
FROM C FIRST, C SECOND
WHERE FIRST.PC=SECOND.CNO;





小结

SQL连接查询的典型格式:

SELECT A1, A2, ..., Am

FROM R1, R2, ..., Rn

WHERE F

外连接:

FROM <左关系> LEFT | RIGHT | FULL [OUTER] JOIN <右关系> ON <连接条件>

自身连接:

FROM S S1,S S2

FROM S AS S1,S AS S2