数据库习题汇总

Organizer: rfhits

Date: 2022-01-05--2022-01-07

Contact: [wysj0117@gmail.com](mailto:wysj0117@gmail.com)

被各种概念、定义整得很无语的一门课，它的难点在于将各种概念用精确的自然语言表示出来并在考试的时候记住。

这门课背PPT很痛苦，但是又必须上，谁叫这是6系开的课。

这种考试就应该开卷，谁闲着没事背概念啊。

要不让学生出一份卷子给老师做一下？🙂

# 概述

此章是开篇之章，告诉我们要从概念模型到数据模型，重点是E-R图和概念模型的了解，然后粗略得讲解了数据库系统。

## 目录

+ 数据管理技术的发展

    + 人工管理阶段

    + 文件系统阶段

    + 数据库系统阶段

+ 概念模型

+ 数据模型

    + 三要素

        + 数据结构

        + 数据操作

        + 完整性约束

            + 完整性的概念

                + 正确性

                + 相容性

            + 约束包括

                + 语义约束

                + 通用性约束

    + 分类

        + 关系模型

        + 层次模型

        + 网状模型

+ 数据库系统的结构

    + 结构特征

        + 三级模式

            + 内模式

            + 模式

            + 外模式

        + 两级映象

            + 内容

                + 外模式/模式

                + 模式/内模式

            + 优点

                + 独立性

                    + 物理独立性

                    + 逻辑独立性

                + 简化接口

                + 安全保密

                + 数据共享

    + 结构

        + 数据库

        + DBA（用户）

        + 硬件

        + DBMS（软件）

            + 功能

                + 数据库定义功能

                + 数据存取功能

                + 数据库运行管理

                    + 并发控制

                    + 存取控制

                    + 完整性约束条件检查和执行

                    + 日志组织和管理

                    + 事务管理和自动恢复

                    + 安全性控制

                + 数据组织、存储和管理

                + 数据库的建立和维护

            + 组成

                + 语言编译处理程序

                + 系统运行控制程序

                + 系统建立和维护程序

                + 数据字典

## 题目

1. 信息与数据含义（2020秋-期末-1.1, 刘瑞PPT）

数据是信息的载体，信息是数据的\_\_\_\_。

答案：内涵。

1. 管理阶段（2020秋-期末-1.4）

数据管理技术的发展经历了 ，文件系统，数据库系统三个阶段。

人工管理

1. 文件系统（2012秋-期末-3.1）

文件系统阶段已经具有了一定的数据独立性。

对

1. 管理阶段特点

数据库和文件系统的根本区别

答：数据结构化是数据库主要特征之一，是数据库与文件系统的根本区别。

1. 数据库系统阶段管理特点有哪些？

面向全组织的复杂的数据结构

数据冗余度小，易扩充

具有较高的数据和程序的独立性

统一的数据控制功能

数据的最小存取单位是数据项

1. 数据库系统阶段数据的最小存取单位是\_\_\_\_。

数据项

1. 三种逻辑数据模型的区别

各种数据模型之间的根本区别在于数据之间联系的表示方式不同：

层次模型：用“树结构”表示数据之间的联系；

网状模型：用“图结构”表示数据之间的联系

关系模型：用“二维表”表示数据之间的联系

1. 数据模型分类（2020秋-期末-1.3）

常见的数据模型（逻辑数据模型）主要包括层次模型， 和关系模型。其中层次模型和关系模型主要区别在于 。

网状模型；不会，应该是二者表示联系的结构上的区别，层次用树，关系用二维表

1. 填空（2011秋-期末-1.1）

数据模型的三要素是\_\_\_\_，\_\_\_\_和\_\_\_\_。

数据结构；数据操作；完整性约束。

1. 数据的完整性（2012秋-期末-1.5）

数据完整性是指:数据的\_\_\_\_和\_\_\_\_。

正确性，相容性。

1. 什么是数据独立性?

物理独立性：数据的存储结构改变，数据的逻辑结构可以不变

逻辑独立性：数据逻辑结构改变，应用程序可以不变。

1. 数据库数据控制功能包括哪些？

数据的安全性控制：保护数据以防止不合法的使用所造成数据的泄密和破坏

数据的完整性控制：指数据的正确性与相容性

并发控制：对多用户的并发操作进行控制、协调，保护数据的完整性

数据库恢复：将数据库从错误状态恢复到某一已知的正确状态

安完并恢

1. 数据库的概念（2012秋-期末-2.1）

\_\_\_\_是长期存储在计算机内的有组织，可共享的数据集合。

A.数据库管理系统B.数据库系统C.数据库D.文件系统

答案：C

1. 数据库的功能（2012秋-期末-1.4）

数据库系统具有统一的数据控制功能，包括数据的\_\_\_\_控制、数据的完整性控制、并发控制等。

安全性控制

1. 区别（2011秋-期末-2.2）

数据库系统与文件系统的本质区别是

Ⅰ. 数据库系统复杂，而文件系统简单。

Ⅱ. 数据库系统实现整体数据的结构化，而文件系统中数据之间是孤立的。Ⅲ.文件系统只能管理程序文件，而数据库系统能够管理各种类型的文件。

Ⅳ.文件系统管理的数据量较少，而数据库系统可以管理庞大的数据量。

1. 只有Ⅲ B.只有Ⅱ C. II,III 和IV D.都是

答案B

1. 模式（2012秋-期末-一.1）

数据库系统的三级模式是指:外模式、\_\_\_\_和内模式。而数据库的逻辑独立性则是通过\_\_\_\_实现的。

模式，外模式/模式映象。

1. 数据库的数据完整性是指:数据的正确性和 。

答案：相容性

1. \_\_\_\_是三级模式的核心。

模式

1. 模式定义了什么？

具体定义数据的逻辑结构（数据记录结构、数据之间的联系）、数据安全性、完整性要求；

1. 2011秋-期末-1.3

数据库的完整性是指\_\_\_\_和\_\_\_\_。

正确性；相容性。

1. 什么是独立性，具体概念是什么？

外模式/模式映象定义某个外模式与模式之间的对应关系。

当模式改变时，外模式/模式映象做相应改变，可以保证外模式不变。——数据的逻辑独立性

模式/内模式映象定义数据逻辑结构与存储结构之间的对应关系。

当内模式改变时，模式/内模式映象做相应修改，使得模式保持不变。——数据的物理独立性

1. \_\_\_\_是数据库系统的核心软件，在\_\_\_\_支持下工作

DBMS；操作系统。

1. DBMS的主要功能有哪些？

数据库定义功能

–提供DDL语言描述外模式、模式、内模式（源模式）。

数据库存取

–提供DML语言（Data manipulation language）对数据库进行检索、插入、修改、删除。

数据库运行和管理

数据组织、存储和管理

数据库建立和维护

1. 2011秋-期末-2.4

数据库管理系统主要的功能包括

I. 数据库定义

II.数据库存取

III.数据库运行和管理

IV .数据组织、存储和管理

V.数据库建立和维护

A.只有1,II,III B,只有I,II,I和IV

C.只有I,II,IV 和V D.都是

D

1. DBMS的组成

语言编译处理程序

系统运行控制程序

–包括系统总控、存取控制、并发控制、完整性控制、保密性控制、数据存取和更新、通信控制等程序。

系统建立和维护程序

–数据装入、数据库系统恢复、性能监督、工作日志等程序。

数据字典

1. 数据库系统组成有哪些？

包括四个主要部分：数据库、用户、软件、硬件

软件——指负责数据库存取、维护和管理的软件系统。包括数据库管理系统DBMS，各种应用系统。DBMS是数据库系统的核心软件，在操作系统支持下工作。

# 关系数据理论和数据库

数据库设计的理论基础是逻辑数据模型里的关系数据模型。从三要素出发，对关系数据库的理论知识进行教学。

## 目录

+ 关系数据模型

    + 数据结构：关系

        + 基本概念

            + 关系，域，元组，……

            + 关系模式

            + 关系数据库

                + 型

                + 值

    + 语义约束

        + 实体完整性

        + 参照完整性

        + 用户定义完整性

    + 数据操作

        + 特点：集合操作

        + 基础：关系运算

            + 关系代数（代数方式）

                + 传统的集合运算：交并差乘

                + 特有的关系运算：选择，投影，连接，自然连接，求商

            + 关系演算（逻辑方式）

                + 元组关系演算

                + 域关系演算

            + 关系运算的安全约束

            + 关系运算的等价性

+ 关系数据语言概述

    + 关系数据库数据语言

        + 组成

            + 数据定义语言

            + 数据操纵语言

            + 数据控制语言

    + 特点

    + 分类

## 题目

1. 什么是关系模式？

关系的描述称作关系模式，它可以形式化地表示为：

**R**（**U**，**D**，**dom**，**F**， **I**）

1. 试说明基本关系的性质。（2010秋-期末-4.1）

1.列是同质的。

2.不同的列可出自同一域,每列必须有不同的属性名。

3.列的顺序无关紧要,即列次序可以互换。

4.任意两个元组不能完全相同(集合内不能有相同的两个元素)。

5.行的顺序无关紧要,即行次序可以互换。

6.每一分量必须是不可再分的数据。

1. 关系的定义是什么？（2021秋-期末-4.X）
2. 什么是关系数据库的型和值？

+ 关系模式的集合构成关系数据库模式—关系数据库的型；

+ 关系的集合则构成具体的关系数据库—关系数据库的值

可以理解为数据库里面表的结构和表里的值。

1. 关系和实例的概念和特点是什么？（2021秋-期末-4.）

不会表达。

1. 关系数据模型的约束条件是什么？

实体完整性（Entity Integrity）：

要有属性或属性组合作为主码，主码值不可为空或部分为空。

参照完整性：

如果关系R的外部码Fk与关系S的主码Pk相对应，则R中的每一个元组的Fk值要么等于S 中某个元组的Pk 值，要么为空值。

用户定义完整性：用户针对具体的应用环境定义的完整性约束条件。

1. 关系模型的完整性约束有两种，即实体完整性约束和参照完整性约束。

错误

还有用户定义的完整性。

1. 什么是数据的安全性和完整性？（2021秋-期末）

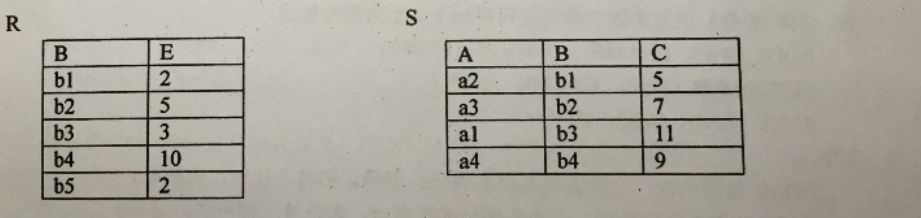
自制的答案：

数据的完整性是为了防止数据库中存在不符合语义的数据，也就是防止数据库中存在不正确的数据。防范对象是不合语义的、不正确的数据。

数据的安全性是保护数据库防止恶意破坏和非法存取。防范对象是非法用户和非法操作。

1. 数据完整性（2012秋-期末-4.2）

设一关系数据库中有如下关系R和S，关系R的主码是属性B，关系S的主码是属性集AB，且关系R中的属性B与关系S中的属性B具有相同的定义。



请回答S中属性B的取值至少要满足哪些约束条件，为什么?

因为关系S的主码是属性集合AB，所以B应该满足实体完整性的要求，即不能取空值，同时AB的取值应当在关系S中是唯一的。

同时，由于B是R的主码，是S的外部码，所以S的取值还应该满足参照完整性的要求。即B取值必须是在R的B中存在的值。

1. 约束性含义（2011秋-期末-4.2）

某企业生产管理系统中有三个表:

产品(产品号，产品名，厂家)，

零件(零件号，零件名，材料)，

组装(产品号，零件号，数量)，

它们的码分别是产品号、零件号、(产品号，零件号)。在向组装关系中添加一个元组时，该元组的属性取值必须满足什么条件?为什么?

产品号和零件号不能为空或者部分为空，要满足实体完整性；

产品号和零件号需要在产品表和零件表中有对应的值，满足参照完整性。

1. 关系操作的特点和分类（自制）

关系数据操作方式的特点是\_\_\_\_，“一次一集合”方式。操作的对象与结果都是\_\_\_\_。关系数据操作的基础是“\_\_\_\_”。关系运算方式有两种：\_\_\_\_，\_\_\_\_。

集合操作；集合；关系运算；代数方式；逻辑方式。

1. 不同角度的代数运算（郎波课件）

从数学角度，基本关系代数运算有5种：\_\_\_\_。

从数据库角度，核心的关系代数运算为：\_\_\_\_。

并、差、乘（笛卡尔积）、选择、投影

选择、投影、连接（或自然连接）

1. 2011秋-2.7

关系代数有五种基本的操作，其他操作可以用这五种操作来表达，这五种基本的操作是\_

A.并、差、交、投影和选择

B..并、差、广义笛卡尔积、投影和选择

C.并、差、连接、投影和选择

D.并、差、交、连接和除法

B

1. 关系代数的简单运用

在学生表中选出年龄大于15岁的学生：\_\_\_\_;

查询学生姓名和系号：\_\_\_\_。

;

1. 给出自然连接的定义，说明选取了什么，删除了什么。

关系R与关系S的自然连接，

是从两个关系的广义笛卡儿积R×S中选取在相同属性列上取值相等的元组，并去掉重复的属性列。

比连接要多做一个删除属性列的操作，得到的列数要少于笛卡尔积。

1. 实战关系代数（郎波课件）

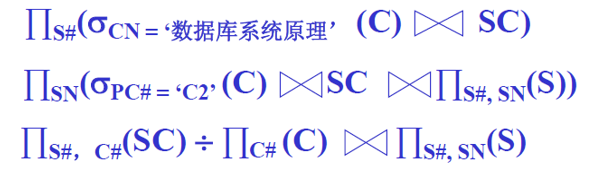
已知有关系模式：S(S#, SN,SA,SD), C(C#, CN, PC#), SC(S#, C#, G)

将下列操作用关系代数运算表示出来：

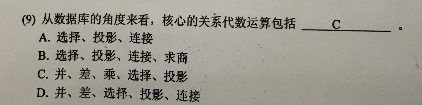
（1）求选修“数据库系统原理”的学生的学号。

（2）求至少选修了这样一门课程的学生姓名，这门课的直接先行课为**C2**。

（3）求选修了全部课程的学生学号和姓名



1. 数据库角度的运算（2012秋-期末-2.9）



这道题和郎波的PPT冲突了。郎波PPT上得选A

1. 元组关系演算的基本概念（自制）

元组关系演算的基本结构是\_\_\_\_。

元组关系表达式的形式定义：\_\_\_\_，它表示了所有使为真的元组的集合。

元组演算表达式；

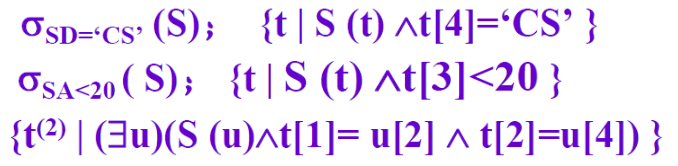
1. 使用元组关系演算完成查询（郎波PPT）

查询计算机系（CS）的全体学生

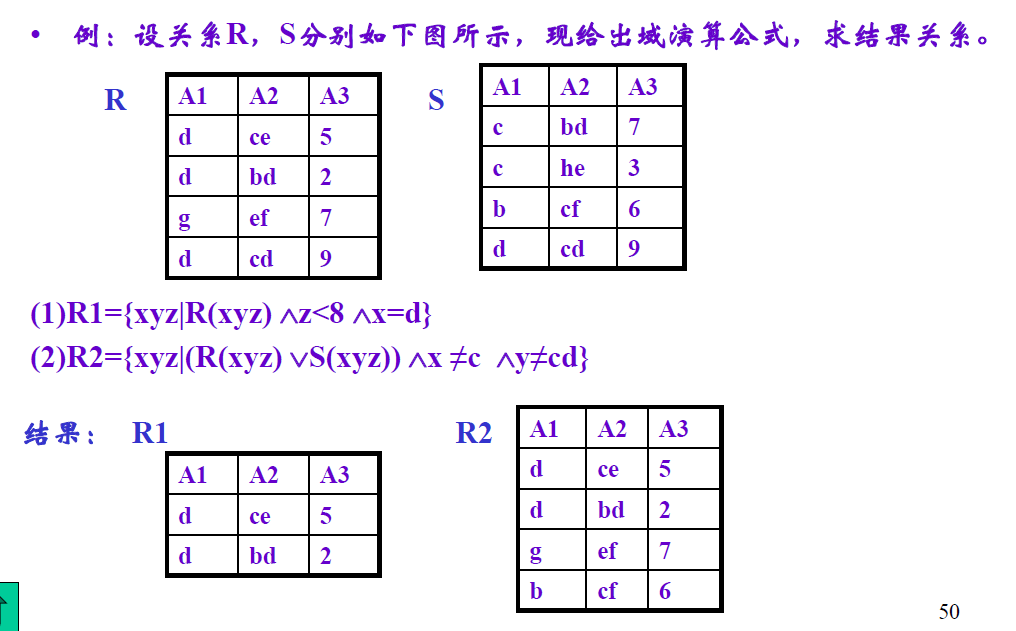
查询年龄小于**20**岁的学生

查询学生的姓名和所在的系

建议先写出关系代数和表达式，在转换成元组关系演算。



1. 给定域关系演算，求结果（郎波PPT）



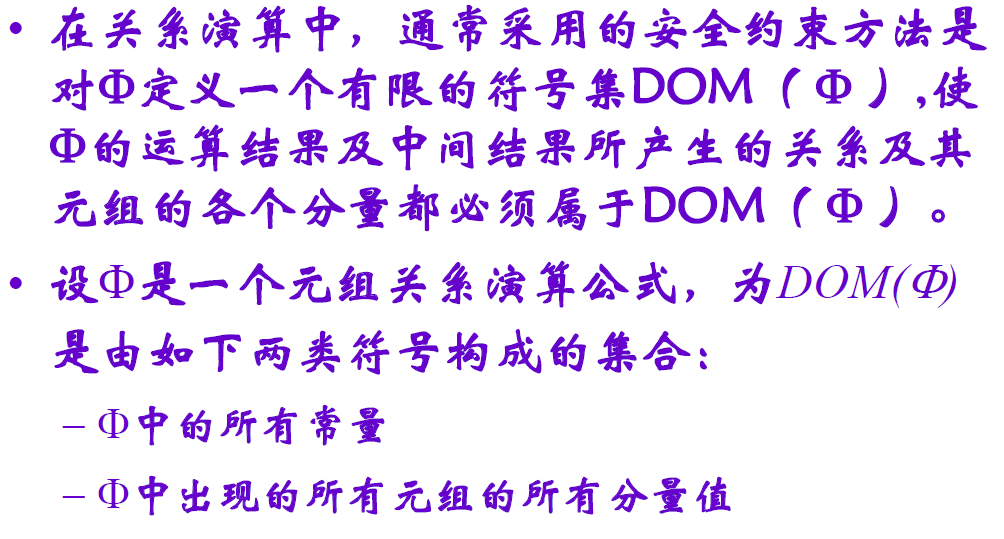
1. 安全运算的概念（自制）

关系运算中把不产生\_\_\_\_和\_\_\_\_的运算称为安全运算；其运算表达式称为\_\_\_\_；对其所采取的限制称为\_\_\_\_。

无限关系；无穷验证; 安全表达式；安全约束

关系代数是安全运算，关系演算则不一定是（毕竟得穷举所有的t才知道结果）所以对关系演算要进行安全约束。

1. 安全约束方法是啥（郎波PPT）



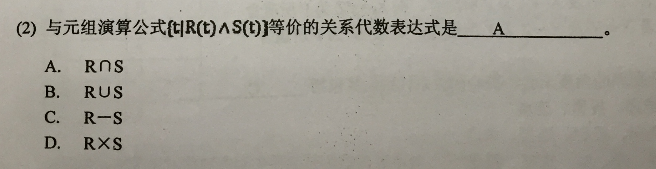
1. 等价性判断（自制）

关系代数、元组关系演算和域关系演算三类关系运算的表达能力是等价的。（）

错误

关系代数是安全运算，关系演算则不一定是。所以对关系演算要进行安全约束。 经过安全约束后的三类关系运算的表达能力是等价的，可以相互转换。

1. 等价转换（2012秋-期末-2.2）



1. 关系数据语言概念（自制）

关系数据语言的组成有哪些？

数据定义（描述）语言（Data definition or description language)，

数据操纵语言（Data Manipulation Language)DML

数据控制语言(Data Control Language)DCL，

1. 各类关系数据语言的功能（自制）

数据定义语言：内模式、模式和外模式的定义

数据操纵语言：数据库有四种基本操作：检索、插入、修改、删除

数据控制语言：完成数据库的安全性控制、完整性控制、并发控制等。

还包括数据查询语言

定义，操作，查询，控制：定操查控

控制：安完并

1. DML功能（自制）

DML有\_\_\_\_和\_\_\_\_。

联机交互方式; 宿主语言方式

联机交互方式下的DML称为自含式语言，可独立使用，适用于终端直接查询；宿主语言方式下的DML称为嵌入式语言，依附于宿主语言，是嵌入高级语言的程序中，以实现数据操作。

1. 关系数据语言的特点有哪些？（自制）

（1）一体化：将数据的定义、查询、更新、控制等功能融为一体，

（2）非过程化：用户只需提出“干什么”，而“怎样干”由DBMS解

决。

（3）面向集合的存取方式：操作对象是一个或多个关系，操作的结果也是一个新关系。

（4）既可独立使用又可与主语言嵌套使用

1. 列出几种关系数据语言，并说明他们的数据操作采用的关系运算的类别。

关系数据语言的核心是查询，所以又称为查询语言。而查询往往表示成一个关系运算表达式，因此关系运算是设计关系数据语言的基础，关系运算的分类也决定了关系语言的分类。



# SQL语言

## 目录

+ 概述

+ 查询功能

+ 投影

+ 选取

+ 排序

+ 连表

+ 子查询嵌套

+ 并差交

+ 库函数

+ 定义功能

+ 视图操作

+ 数据更新

+ 空值处理

+ 数据控制

SQL语言功能丰富，支持

DQL：Data Query Language

DML：Data Manipulation Language，如insert delete update

DCL：Data Control Language，如事务、权限，安全性检查

DDL：Data Definition Language，如定义一个新表

## 题目

1. 什么是基本表与导出表？

基本表：是实际存在的，每个表在存储中可用一个存储文件来表示；

导出表:是从基本表导出的表，有视图（View）和快照(Snapshot)

1. 定义表，并确定约束

CREATE TABLE 课程表 (

课程号varchar(255) NOT NULL PRIMARY KEY ,

课程名varchar(255) UNIQUE ,

先修课varchar(255),

学分int CHECK (学分>0 AND 学分<5)

)

CREATE TABLE 选课表

(

学号varchar(255) NOT NULL REFERENCES 学生表(学号),

课程号varchar(255) NOT NULL FOREIGN KEY REFERENCES 课程表(课程号),

成绩float CHECK (成绩>=0 AND 成绩<=100),

PRIMARY KEY (学号,课程号)

)

1. alter有何作用，举例说明

**Alter Table <**表名**>**

**[Add <**新列名**><**数据类型**>[<**完整性约束**>]]**

**[Drop <**完整性约束名**>]**

**[Modify <**列名**><**数据类型**>]**

修改属性的数据类型

alter table table\_name

alter column column\_nametype

添加新的属性

ALTER TABLE 学生表

ADD 电子邮件varchar(20)

新增约束

ALTER TABLE 课程表

ADD CHECK (学分>0 AND 学分<6)

1. 选择的条件（PPT）

检索选修C1或C2且成绩高于70分的学生学号、课程号和成绩。

SELECT S#，C#，G

FROM SC

WHERE （C#=’C1’ OR C#=’C2’）AND G>=70

1. 创建cluster索引

CREATE CLUSTERED INDEX 学号

Index ON 学生表(学号)

1. 为学生选课表建立unique索引，学号升序，课程号降序（郎波PPT）

Create Unique Index Scno

On SC(S# ASC, C# DESC)

1. 排序检索（PPT）

检索全体学生信息，并按系号升序，同一个系按年龄降序排列。

SELECT \*

FROM S

ORDER BY SD , SA DESC;

在最后一句话加上`order by 列名 序`

1. 连表

检索学生张华所学课程的成绩。

SELECTSN，C#，G

FROM S，SC

WHERE S.S#=SC.S# AND SN=‘张华’;

注意连表条件

1. 表的自连接（PPT）

检索所有比李勇年龄大的学生姓名、年龄。

SELECT X.SN，X.SA

FROM S X，S Y

WHERE X.SA>Y.SA AND Y.SN=’李勇’;

通过定义别名，将一个表看成两个表，进行连接（笛卡尔积）。

1. 外连接（PPT）

检索所有学生的全部信息。

SELECT \*

FROM S，SC

WHERE S.S#=SC.S#(\*);

在连接谓词某一边加（\*或+），则逻辑上为\*所在边的表增加了一个空行。它可以与另一个表中所有不满足连接条件的元组进行连接，使这些元组能够输出。

1. 找出学生S1没有选修的课程

SELECT C#

FROM SC X

WHERE NOT EXISTS(

SELECT \*

FROM SC Y

WHERE Y.S#=’S1’ AND X.C#=Y.C#)

遍历每一门课，不存在(S1,C#)这条记录。

1. 子查询嵌套检索

检索至少选修了学生S2选修的全部课程的学生学号。



不存在（这样的课）

这样的课：S2选了，（该学生没选）

1. 交差并检索
2. 选修了C1或C2课程的学生学号。
3. 检索无人选修的课程号和名称。

SELECTS#FROM SC WHERE C#='C1'

UNION

SELECT S# FROM SC WHERE C#=‘C2’;

SELECT C#，CN

FROM C

WHERE C# IN (

SELECT C# FROM C

MINUS

SELECT DISTINCT C# FROM SC) ;

1. 库函数检索

库（集）函数

–COUNT( ) 按列值计个数，COUNT(\*)对行记数。

–SUM() 对数值列求总和

–AVG() 求数值列的平均值

–MAX() 在列中找出最大值

–MIN() 在列中找出最小值

•GROUP只能在SELECT子句以及HAVING子句中出现

1. 检索学生总数。
2. 检索选修了课程的学生人数
3. 检索选修了C1课程的学生最高分。
4. 求学号为S4的学生的总分和平均分

SELECT COUNT(\*)FROM S ;

SELECT COUNT(DISTINCT S#) FROM SC ;

SELECT MAX(G)

FROM SC

WHERE C#='C1';

SELECT SUM(G)，AVG(G)

FROM SC

WHERE S#='S4’;

1. 2011秋-期末-2.6

ALTER TABLE属于SQL语言的\_\_语句。

A.数据定义

B.数据操纵

C.数据查询

D.数据控制

A

alter可以重命名列

1. 写出增删改元组的BNF形式（自制）

插入：

**Insert Into <**表名**>**

**[(<**属性列**>[{,<**属性列**>}])]**

**Values(<**值**>[{,<**值**>}])**

1. 插入查询结果（郎波PPT）

插入系-平均年龄

**Insert Into Dept\_Age (Sdept, Avgage)**

**Select SD, AVG(SA) From S**

**Group By SD;**

1. 数据更新（郎波PPT）

将学生S1的年龄改为22岁

**Update S**

**Set SA = 22**

**Where S#=’S1’;**

1. 数据删除（郎波PPT）

删除学号为**S19**的学生的记录

删除所有学生的选课记录

删除计算机系所有学生的选课记录

**Delete From S Where S#=„S19‟**

**Delete From SC**

**Delete From SC**

**Where ‘CS’ = (Select SD From S**

**Where S .S# = SC.S#)**

1. 利用空值，查找没有分配系号的学生和已经被分配系号的学生（郎波PPT）

SELECT \*

FROM S

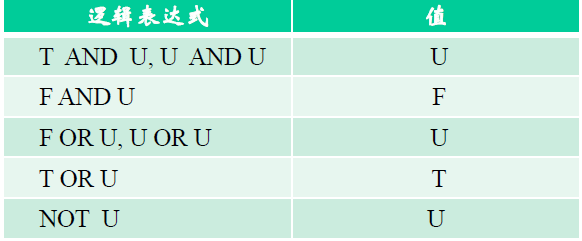
WHERE SD IS NULL;

SELECT \*

FROM S

WHERE SD IS NOT NULL;

1. 写出空值、True和False三者逻辑运算的基本结果（自制）



1. 视图

建立计算机系的学生视图

Create View CS\_Student

As

SELECT S#, SN, SA FROM S

WHERE SD =’CS’ ;

1. 视图消解

DBMS执行对视图的查询时，从\_\_\_\_中取出视图的定义，把定义中的子查询和用户的查询结合起来，转换成等价的对基本表的查询，然后再执行修正的查询。这一转换过程称为视图消解。

数据字典。

1. 视图创建有查询（PPT）

在计算机系学生的视图中找出年龄小于20岁的学生。并写出出视图消解后查询语句。

Create View CS\_Student

As

SELECT S#,SN, SA

FROM S

WHERE SD='CS';

Select S#, SA

From CS\_Student

Where SA <20;

转换后为:

Select S#, SA From S

Where SD=CS’And SA <20;

1. 视图的作用
2. 能够简化用户操作
3. 使用户能够以多种角度看待同一数据
4. 提供了一定程度的逻辑独立性
5. 能够对数据提供安全保护
6. 数据库操作（2012秋-期末-6）

已知某图书销售公司有如下关系数据库模式:

BOOK(BNO,BNAME,AUTHOR,PRESS)

CUSTOMER （CNO,CNAME,CITY)

SALE (BNO,CNO,QTY)

其中,

BOOK是图书表,它的各属性依次表示:书号，书名，作者，出版社;

CUSTOMER是客户表,它的各属性依次表示:客户号，客户名，城市;

SALE是销售表,它的各属性依次表示:书号,客户号,购买数量。

1.用SQL语句实现下列操作

(1)创建表SALE,要求定义表的主码和外码，表中各属性的类型和长度可根据语义自行定义。

CREATE TABLE SALE(

BNO CHAR(10) REFERENCE BOOK(BNO),

CNO CHAR(10) REFERENCE CUSTOMER(CNO),

QTY INT,

PRIMARY KEY(BNO,CNO));

(2)找出购买ISBN1101号图书的客户名和所在城市。(3分)

SELECT CNAME, CITY

FROM CUSTOMER

JOIN SALE ON CUSTOMER.CNO=SALE.CNO

WHERE BNO='ISBN1101'

(3）找出既购买ISBN1101号图书又购买ISBN1102 号图书的客户号和客户名。(3分)

SELECT CUSTOMER.CNO,CNMAE

FROM CUSTOMER

JOIN SALE ON CUSTOMER.CNO=SALE.CNO

WHERE BNO=’ISBN1101’

AND CUSTOEMER.CNO IN (

SELECT CNO,CNMAE

FROM CUSTOMER

JOIN SALE ON CUSTOMER.CNO-SALE.CNO

WHERE BNO='ISBN1102')

(4)检索书名含有“数据库”字样图书的基本信息和购买数量。(3分）

SELECT BOOK.BNO, BNAME,AUTHOR, PRESS, SUM(QTY)

FROM BOOK

JON SALE ON BOOK.BNO=SALE.BNO

WHERE BNAME LIKE‘%数据库%

GROUP BY BOOK.BNO, BNAME, AUTHOR, PRESS

特别注意每一条是购买记录，所以需要group by

1. 定义一个北京客户基本信息的视图BJ\_CUSTOMER。(2分)

CREATE VIEW BJ\_CUSTOMER

AS

SELECT \* FROM CUSTOMER WHERE CTY='北京’

(6)利用第（5）题中定义的视图BJ\_CUSTOMER，检索北京客户都购买的图书的图书号。(3分)

SELECT BNO FROM BOOK

WHERE NOT EXISTS (

SELECT \* FROM BJ\_CUSTOMER

WHERE NOT EXISTS(

SELECT \*FROM SALE

WHERE SALE.BNO = BOOK.BNO

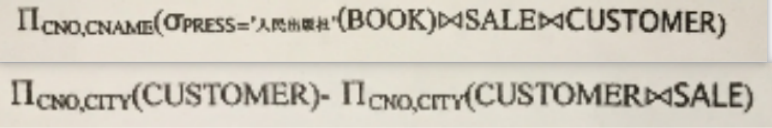
AND SALE.CNO=BJ\_CUSTOMER.CNO))

对每一本书，不存在这样的北京客户，北京客户没有买它。

2.用关系代数表达下列查询:(每题3分，共6分)

(1)找出购买了人民出版社出版图书的客户编号和姓名。

(2)找出没有购买图书的客户名称和所在城市。



3.用元组关系演算表达下列查询:(共2分)

(1)检索被一名顾客一次购买数最达500本(含500本）以上的图书名及购买数量。



1. 查（2011秋-期末-7）

现有以下关系模式:

学生表Student (SNO,SNAME,SEX,AGE, DEPT);

课程表Course （CNO,CNAME);

选课表SC(SNO,CNO, SCORE);

用 SQL语句实现下列操作:(每题2分,共14分)

(1)查找既选修了‘数据库’，也选修了‘操作系统’的同学的学号。

SELECT SNO From SC, COURSE

WHERE SC.CNO=COURSE.CNO AND CNAME=”数据库”

AND SNO IN

(SELECT SNO FROM SC, COUSE

WHERE SC.CNO=COURSE.CNO AND CANME=”操作系统”)

(2)查找被每个同学都选过的课程

SELECT CANME FROM COURSE

WHERE NOT EXIST

(SELECT \* FROM STUDENT

WHERE NOT EXIST

(SELECT \* FROM SC

WHERE SC.CNO = COURSE.CNO AND SC.SNO = STUDENT.SNO))

不存在这样的学生，他没选这门课

(3)累出每门课程的课程号和平均成绩

SELECT CNO, AVG(SCORE) FROM SC

GROUP BY CNO

(4)查找“计算机”系年龄在20岁到22岁之间，姓“王”的学生的学号和姓名

SELECT SNO, SNAME FROM STUDENT

WHERE DEPT=’计算机’ AND AGE BETWEEN (20, 22) AND SNAME LIKE ‘王%’

(5)创建计算机系学生的选课情况视图（包括学号、姓名、所选课程名和成绩四个属性）

CRETE VIEW CS.CS

AS

SELECT SNO, SNAME, CNAME, SCORE

FROM STUDENT, SC, SCORE

WHERE STUDENT.SNO=SC.SNO AND SC.CNO = COURSE.CNO

(6)检索比MARY年龄小3岁的学生的姓名、年龄、性别和所在系。

SELECT SNAME, AGE, SEX, DEPT FROM STUDENT

WHERE AGE+3=

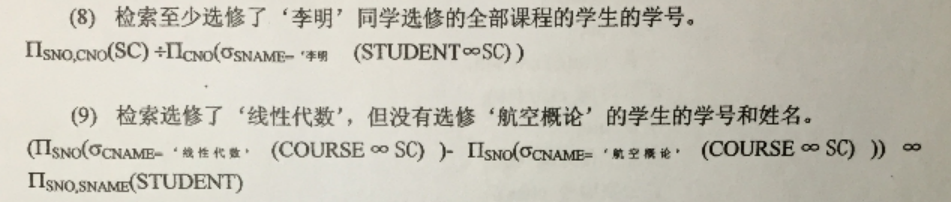
(SELCT AGE FORM STUDENT WEHERE SANME = ‘MARY’)

(7) 在学生表中插入学号为26，姓名为‘李婷’，性别为‘女'，年龄为20，所在系为‘计算机系’的一条记录。

INSERT INTO STUDENT

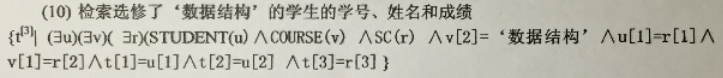
VALUES( ‘26'，‘李婷”,‘女’，20，‘计算机系’);

2.用关系代数表达下列查询:(每题3分,共6分)



3.用元组关系演算表达下列查询:(2分)

(10)检索选修了‘数据结构’的学生的学号、姓名和成绩



1. 数据控制功能包括哪些？

定义完整性约束条件

支持事务操作

提供安全控制功能

–授权

•GRANT 〈权限〉[ON 〈对象类型〉〈对象名〉]

TO 〈用户〉

–收回权限

•REVOKE 〈权限〉[ON 〈对象类型〉〈对象名〉]

FROM 〈用户〉

# 函数依赖与模式分解

## 目录

+ 关系数据理论

+ 函数依赖

+ 分类

+ 完全函数依赖

+ 部分函数依赖

+ 传递函数依赖

+ 公理系统

+ 内容

+ 自反律：自己推出子集

+ 增广律：if x->y, then xz->yz

+ 传递律: if x->y and y->z, then x->z

+ 属性集的闭包

+ 有效性与完备性

+ 最小依赖集

+ 规范化

+ 1NF

+ 2NF

+ 3NF

+ BCNF

+ 多值依赖与第四范式

+ 模式分解的理论--概念

+ 无损连接

+ 保持函数依赖

+ 分解算法

+ 3NF函数依赖

+ 3NF无损分解

+ 候选码求解算法

+ 单属性依赖集的求解算法

+ 多属性依赖集的求解算法

给我们一个需求，如何分析我们要存储哪些数据，要以什么方式存储这些数据。

运用本章的语言来说，也就是如何确定表的结构，表的码等内容。

在进入本章前，要知道一个数据依赖的概念。

## 题目

1. 数据依赖：其中最重要的有两种：\_\_\_\_和\_\_\_\_。

函数依赖( Functional Dependency)，也就是单值依赖

多值依赖(Multivalued Dependency

1. 什么是逻辑蕴含，什么是函数依赖集的闭包？

逻辑蕴含：关系模式R< U, F >中，X、Y是R的属性集合，如果从F中的函数依赖能够推出，则称F逻辑蕴涵。

函数依赖集F的闭包：在关系模式中，为F所逻辑蕴涵的函数依赖的全体称作F的闭包，记作。

1. Armstrong公理的内容（20

若F是属性组U上的一组函数依赖，下列哪一条不属于Armstrong公理系统中的基本推理规则?

A.若Y⊆X⊆U，则X→Y为F所蕴含

B.若X→Y为F所蕴含,且Z⊆U,则XZ→YZ为F所蕴含

C.若X→Y及X→Z为F所蕴含，则X→YZ为F所蕴含

D.若X→Y及Y→Z为F所蕴含，则X一Z为F所蕴含

答案：C

1. 求最小依赖集的步骤

右部单值化，箭头的右边一定是单个属性

删去无用左部属性，箭头的左边属性尽可能

化简有向图，逐个检查，直到不变

注意答案的不唯一

如果给定的都是单对单的函数依赖，可以画出一个有向图，删除的本质是，两个点直接不止一条可达路径。

1. 求最小依赖集（郎波课件）

，求Fm

先把拿掉，看看剩下的函数依赖集能不能再推出B，不行就必须得包含。

逐个拿掉尝试看行不行。

可以去掉

1. 求最小依赖集（郎波课件）

，求Fm 。

因为，所以可以化简为，然后再删除。

1. 2012秋-期末-3.4

二维表的任何两个行不能全同。

答案：对

1. 关系规范化的目的是\_\_\_\_。（2020秋-期末-1.5）

我也不知道答案，但是在郎波课件上有这么一句话：“这些关系常有一些异常或冗余等弊病。规范化的目的就是要消除这些弊病。 ”

或许答案是“消除关系中的异常或冗余等弊病”。

谁知道呢？言之有理就行吧。

1. 规范化的基本思想（自制）

规范化的基本思想是\_\_\_\_，使数据库模式中各关系模式达到某种程度的“分离”，使一个关系只描述\_\_\_\_或者\_\_\_\_。即“一事一地”的设计原则。规范化的实质是\_\_\_\_。

逐步消除数据依赖中不合适的部分; 一个实体; 实体间的一种联系; 概念的单一化。

1. 简述各个范式的内容。

若，且每个非主属性完全依赖于码，则称。

消除非主属性对码的传递依赖。

且每个非主属性都不传递依赖于R的任何码，则。

BCNF：函数依赖的左边一定含有码，

4NF：多值依赖左边一定含有码

1. 什么是多值依赖？（自制）

设**R(U)**是属性集**U**上的一个关系模式，**X**、**Y**、**Z**是**U**的子集，并且**Z = U – X – Y**，关系模式**R(U)**中多值依赖**X** ->-> **Y**成立，当且仅当对**R(U)**的任一关系**r**，给定的一对（**x**，**z**）值有一组**Y**的值，这组值仅仅决定于**x**值而与**z**值无关。

1. 给出多值依赖的形式定义

在**R(U)**的任一关系**r**中，如果存在元组**t**，**s**使得**t[X]=s[X]**，那么就必然存在元组**w**，**v**∈**r**，（**w**，**v**可以与**s**，**t**相同），使得：

**w[X] = s[X] = v[X] = t[X]**

**w[Y] = t[Y]**，**v[Y] = s[Y]**

**w[Z] = s[Z]**， **v[Z] = t[Z]**

则称**Y**多值依赖与**X**，记作**X** ->-> **Y**。

s和t的Y，Z两两组合分给v和w

1. 多值依赖概念（2010秋-期末-3.x）

关系模式R(U)中，X、Y、Z是U的子集，并且Z=U-X-Y，多值依赖X→→Y 成立,当且仅当R(U)的任一关系r，给定的一个X值有一组Y的值。（）

错

1. 多值依赖的概念（2011秋-期末-2.8）

设U为所有属性，X，Y，Z为属性集，Z=U-X-Y。下列关于函数依赖和多值依赖的叙述中,正确的是 \_。

1. 若X→Y，则X→→Y
2. 若X→→Y，则X→Y
3. 若X→Y,则X→Z
4. 若X→→Y，则X→→Z
5. 若X→Y, ，则X→→Y'

A.只有I,II和III

B.只有II,III和V

C.只有I和IV

D.只有IV和V

答案：C

可以参考例子里的C，T，B三个属性

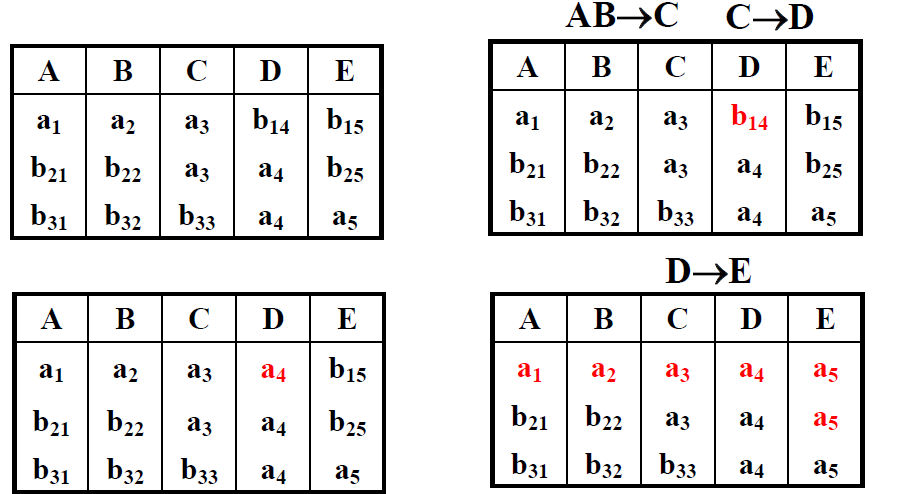
V是不对的，X🡪{Y×Z}不一定X🡪{Y+a,Z-a}。

1. 简述Armstrong公理的有效性和完备性。

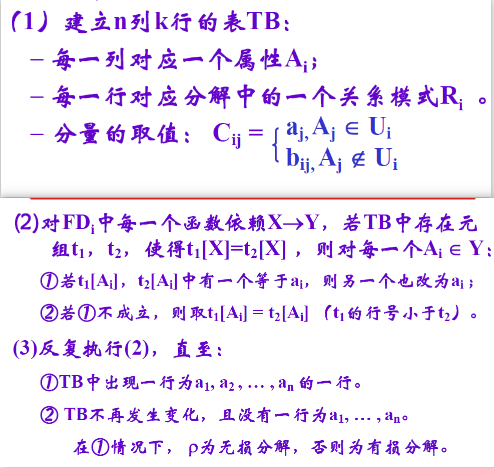
有效性：指由F出发根据Armstrong公理推导出来的每个函数依赖一定在F所蕴含的函数依赖的全体之中。

完备性：F所蕴含的函数依赖的全体中的每一个函数依赖，必定可以由F根据Armstrong公理导出。

1. 验证如下无损是不是无损分解？（郎波课件）



算法：



就是利用函数依赖，看看能不能把原来零散的关系拼起来。

1. 判断无损分解（2012秋-期末-2.4）

设有关系模式R(ABCDE)，其函数依赖集 F={A→BC, CD→E}，则下列分解结果中不属于无损分解的分解是:

A.{R(ABC), R(ADE)}

B.{R(ABCD), R(CDE)}

C.{R(AB), R(ACDE)}

D.{R(AC), R(BCDE)}

答案：D

无损分解的充要条件：

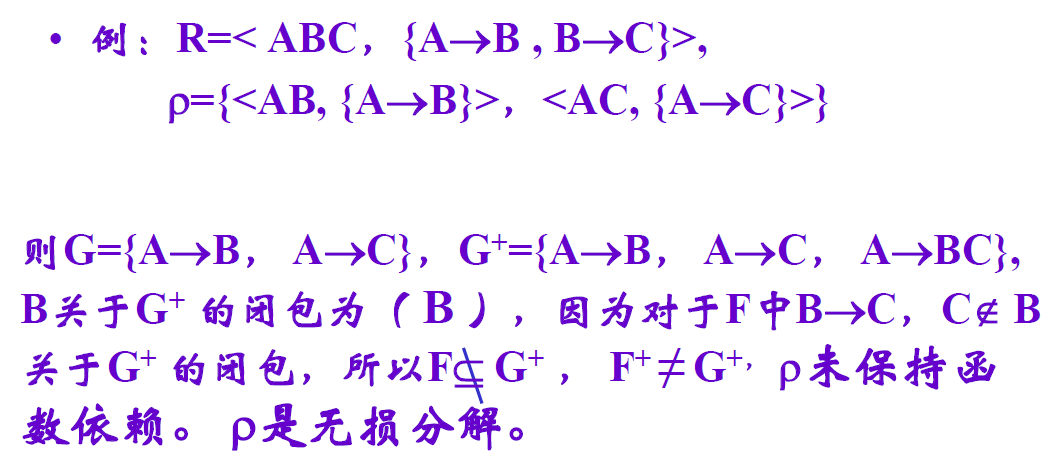
定理：R<U，F>的一个分解具有无损连接性的充分必要条件是 或。

即R1， R2的共同属性至少构成 R1、R2 二者之一的侯选码。

1. 无损分解与保持函数依赖的分解有什么关系？

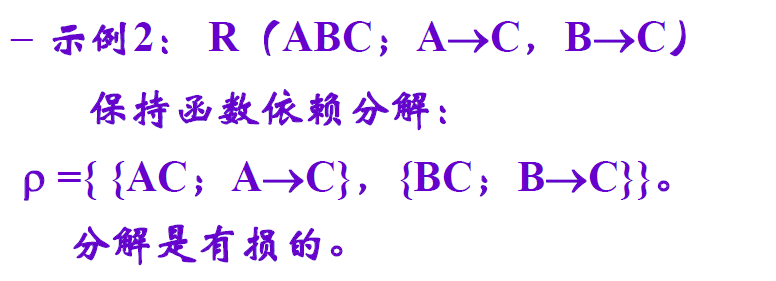
无损分解未必保持函数依赖，保持函数依赖，未必是无损分解。

无损分解可能丢掉有向图的一些信息，导致不是函数依赖。



保持函数依赖可能无法合并成原来的表。

e.g. ，拆开后，就拼不回去了。



1. 模式分解理论上能到达的范式等级？

若仅要求分解保持函数依赖，那么模式分解总可以达到3NF，但不一定能达到BCNF；

若仅要求分解具有无损连接性，那一定可以达到4NF或更高;

若要求分解既保持函数依赖，又具有无损连接性，可以达到3NF，但不一定能达到BCNF。

1. 2011秋-期末-2.9

若关系模式R的码由全部属性构成，则R一定可以达到的范式等级是-

I.2NF

II.3NF

III. BCNF

IV.4NF

A.只有III B.只有IV

C.只有I,II, III D.I, II,III, IV

C

1. 模式分解极限（2010秋-期末-2.10）

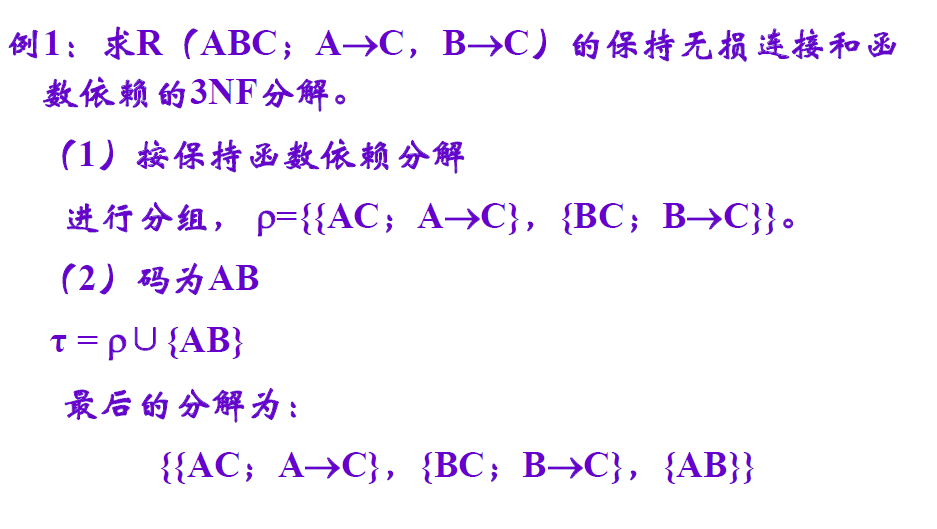
(10)如果关系模式R只包含两个属性,则可以断言R属于的最高范式为:

A.2NF B.3NF

C. BCNF D.4NF

D

1. 找个3NF分解（课件）



1. 分解到BCNF



1. 范式判断与规范（2012秋-期末-5）

有一个房层租赁信息表Rental (CNo，CName，PNo，PAddr，STime，ETime,

Rent，ONo，OName，OPhone)。

表中各属性的含义为:

CNo——客户编号，CName——客户名，

PNo——房屋编号，PAddr——房屋地址，

STime——租赁起始时间，ETime——租赁结束时间，

Rent——租金，

ONo——房东编号，OName——房东名称，OPhone——房东的电话。

这些数据有如下语义:

CNo唯一标识一个客户，PNo唯一标识一个房屋，ONo唯一标识一个房东;

一个客户可以租赁多个房屋，一个房屋可以被多个客户租赁，一个客户租赁一个房屋具有一个开始时间和一个结束时间;

每个房屋对应一个地址,每个地址可以有多个房屋;

每个房屋有一个租金,不同的房屋可以有相同的租金;

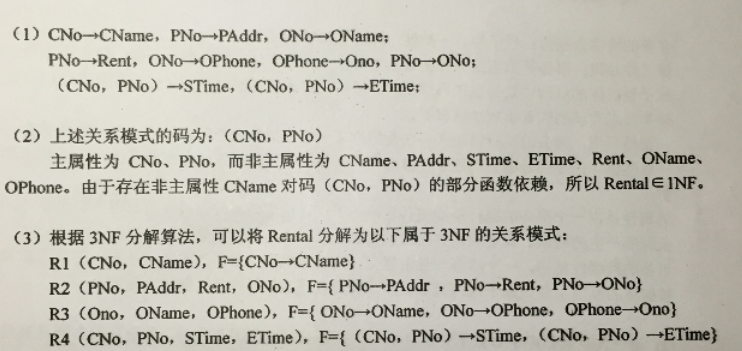
一个房东可以拥有多个房屋,每个房屋只能属于一个房东;

一个房东只能有一个电话,每个电话只能属于一个房东;

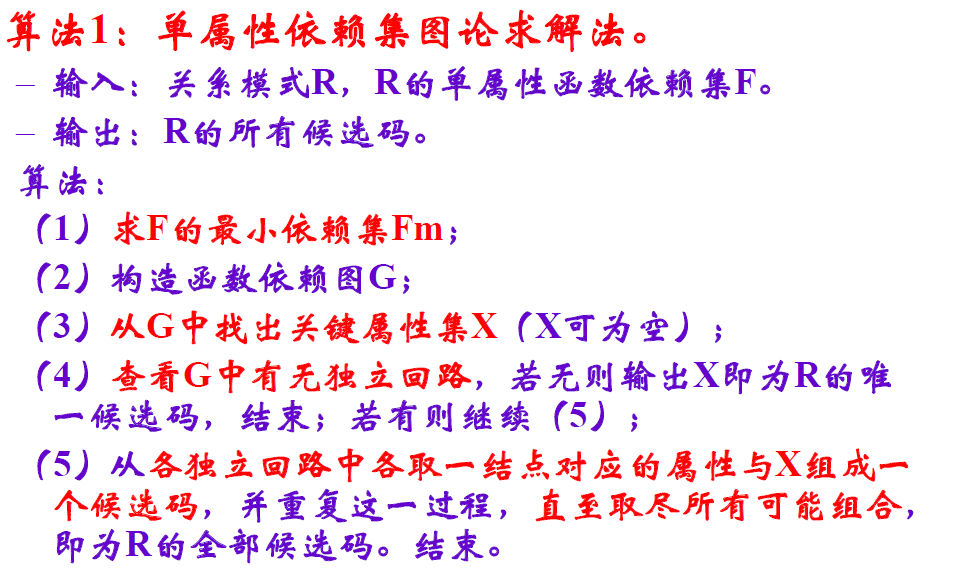
(1)根据上述语义写出关系Rental 的函数依赖集F，并求F的极小函数依赖集Fm。(4分)

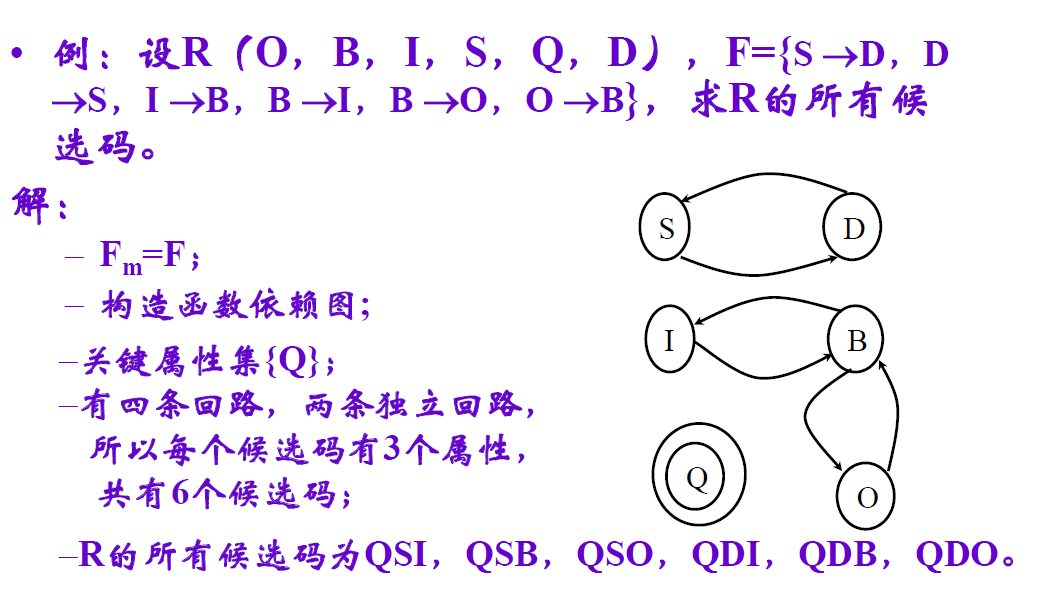
(2)判断关系Rental 所达到的最高范式等级。(4分)

(3)将Rental分解为具有无损连接性和保持函数依赖的3NF。(4分)

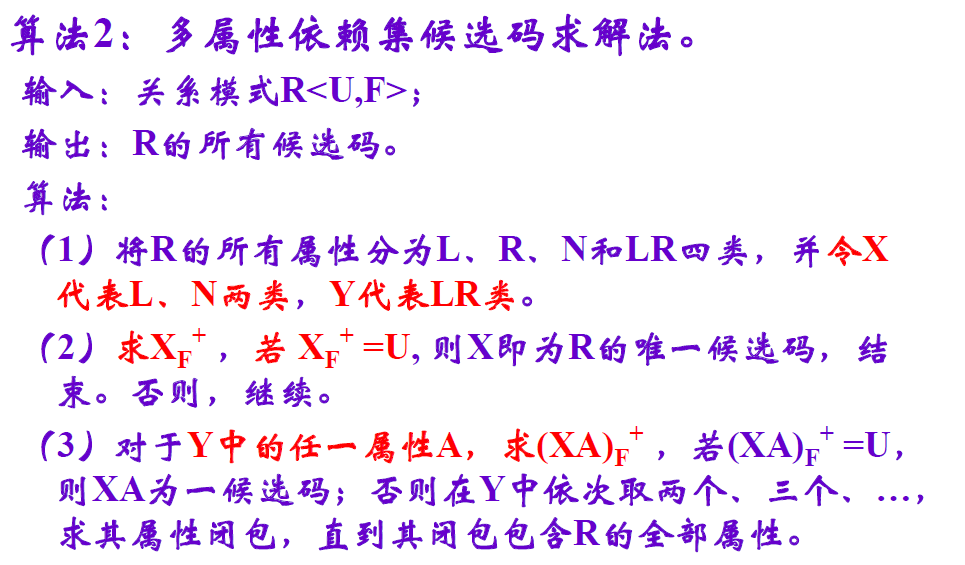


1. 给出单属性依赖集的候选码求解算法



1. 单属性依赖集的候选码（郎波课件）
2. 

从关键属性开始，穷举LR类属性，



1. 规范化题（2011秋-期末-5）

设关系模式R(A，B,C, D,G),R的函数依赖集

，求:

(1)求

(2)求R的所有候选码;(3分)

(3)判断R属于第几范式;(2分)

(4)保持无损连接性和函数依赖,将R分解为3NF。(5分)

1. ABCDG
2. 终结点D、G一定不选，A一定要选，BC之间挑，所以答案是

(AB), (AC)

1. 1NF，因为存在部分依赖
2. R1（ACBG），R2（CD）

# 数据库设计

画数据流图进行需求分析，然后画E-R图建立概念模型，接着把概念模型建立关系模型，也就是数据库中的表。

## 目录

+ 数据库设计流程

    + 需求分析：收集支持系统目标的基础数据及其处理;

        + 调查用户需求

        + 表达需求

            + 数据流图：表达数据和处理之间的关系；

            + 数据字典；描述系统中各类数据

    + 概念结构设计；形成独立于数据库逻辑结构与具体DBMS的概念模型，可以用E-R图表示

        + E-R图

            + 局部E-R图设计

            + 综合局部E-R图形成总E-R图

                + 合并：初步E-R图

                + 修改和重构

    + 逻辑结构设计：将概念结构转换为某个DBMS所支持的逻辑数据模型

        + 形成初始关系数据库模式

        + 关系模式规范化：确定范式等级

        + 关系模式优化

            + 水平分解

            + 垂直分解

        + 子模式定义

            + 优点

                + 符合用户习惯的别名

                + 不同视图保证安全性

                + 降低查询的难度

            + how

    + 物理结构设计；设计数据库在物理设备上的存储结构和存取方法

        + 确定数据库的存储结构

            + 确定存放位置

                + 经常存取部分与和存取频率较低部分分开存放

                + 数据和日志备份放于不同的磁盘上

                + 确定系统配置

            + 确定系统配置变量、存储分配参数，进行物理优化：比如“页”的大小

        + 选择关系的存取方法

            + 索引方法

            + 聚集方法

            + HASH方法

    + 数据库实施

    + 数据库运行和维护

## 题目

1. 需求分析阶段（自制）

需求分析阶段的目标是收集支持系统应用目标的基础数据及其处理。调查的重点是“\_\_”和“\_\_”，包括：\_\_\_\_，\_\_\_\_和\_\_\_\_。

数据;处理；处理要求；信息要求；安全性和完整性的要求。

1. 数据流图的英文名是\_\_\_\_，是以图形方式来表达系统的功能、数据在系统内部的\_\_\_\_和\_\_\_\_过程。

Data Flow Diagram（DFD图）；逻辑流向；逻辑变换

1. E-R图

E-R图的组成包括\_\_\_\_，\_\_\_\_，\_\_\_\_。它们如何表达？

实体；属性；联系。

实体：用长方形表示实体型，在框内写上实体名。

属性：用椭圆形表示实体的属性，并用无向边把实体与其属性连接起来。

联系：用菱形表示实体间的联系，菱形框内写上联系名。用无向边把菱形分别与有关实体相连，在无向边旁标上联系的类型。若联系也具有属性，则属性和菱形也用无向边连接上。

1. 实体模型调整的准则有哪些？

作为属性，不能再具有需要描述的性质。属性必须是不可分的数据项，

属性不能与其他实体具有联系

实体和描述它的属性之间保持**1**：**1**或**n**：**1**的联系

1. 初步E-R图的冲突包括

冲突主要包括：属性冲突、命名冲突和结构冲突

（1）属性冲突：属性的类型、取值范围或取值集合不同，或属性取值单位冲突。

（2）命名冲突：包括属性名、实体名、联系名之间的同名异义，异名同义。

（3）结构冲突：同一对象在不同应用中有不同抽象。如在一应用中为实体，在另一应用中为属性；同一实体在不同分E-R图中属性个数、次序不同。

1. 初步E-R图的冲突内容（2020秋-期末-2.7）

在合并E-R图形成初步E-R图中，下列各项必须解决的冲突包括

Ⅰ同一个实体同一个属性名称不同

Ⅱ同一个概念在一个分E-R图为实体，另一个E-R图为属性

Ⅲ冗余的联系

Ⅳ实体之间的联系在不同分E-R图中呈现不同类型

A.ⅠⅣ B.ⅡⅢⅣ

C. ⅠⅡ D. I II IV

D

冗余的联系是在组合局部ER图形成总ER图的“修改和重构”这一步完成的，所以含有III的B是错误的。

1. 2012秋-期末-7

(共13分）某学校要建立一个教学管理系统，管理学生、课程、教师、教室以及教学活动相关的信息。

课程信息，包括课程编号、课程名、先行课号，一门课程可以有多门先行课，一门课程也可作为多门课程的先行课;

学生信息，包括学号、姓名、年龄、住址;

教师信息，包括职工编号、姓名、年龄、职称；

教室信息，包括教室编号、容纳人数、是否有电教设备。

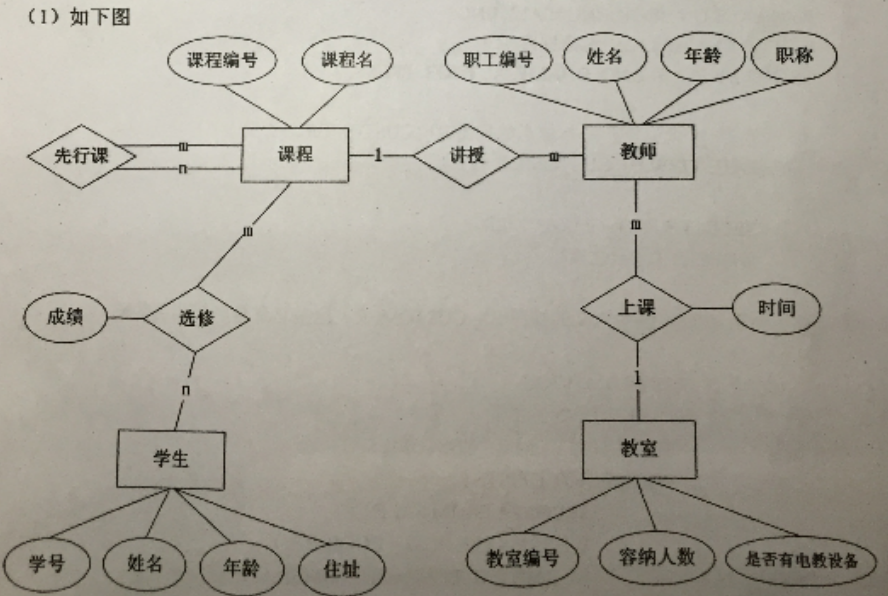
一个学生可以选修多门课程，一门课程可由多个学生选修，学生选修一门课程有一个成绩;

一门课程可在不同的时间由多个教师在不同的教室讲授，

一个教师在固定的教室上课。要求:

(1)画出此管理系统的E-R图。(8分)

(2）将以上E-R图转换为关系模式，并指出每个关系模式的主码和外部码。(5分)



1. 逻辑结构设计包括哪些步骤？（自制）

形成初始关系数据库模式

关系模式规范化

关系模式优化

子模式定义

1. 关系模式的优化包括什么？（自制）

（1）按应用系统的处理要求，确定是否进行模式合并或分解;

（2）为了提高存取效率和存储空间的利用率，可以对关系模式进行必要的分解，包括水平分解和垂直分解。

水平分解：是把关系的元组分为若干子集合，定义每个子集合为一个子关系，以提高系统效率。可以把经常使用的那一部分数据分解出来作为一个关系，其他数据作为另一个关系。

垂直分解：是把关系模式R的属性分解为若干子集合，形成若干子关系模式。

1. 设计用户子模式的优点有哪些（自制）

符合用户习惯的别名

不同视图保证安全性

降低查询的难度

1. 2011秋-期末-8

一个研究所要对科研项目进行计算机管理，该研究所有若干科研人员，

与科研人员有关的信息包括:

职工号(唯一的)、姓名、性别、年龄，专业，研究方向，参加工作时间，工作年限,工资，所在办公室等，

与每个科研项目有关的信息包括:

研究项目编号（唯一的)、项目名称、起始时间和完成时间、项目经费额，经费来源、项目负责人、参加项目的每个人员所承担的任务等信息。

该研究所规定，一个科研项目可以有多名研究人员参加，一个研究人员也可以参加多个研究项目，每个项目由一个科研人员担任负责人。

每个办公室有房间编号（唯一的)，面积和办公电话，一个办公室可以有多个科研人员办公,而一个科研人员只能在一个办公室里办公。

要求:

(1）请画出此管理系统的E-R图。(8分)

(2）将以上E-R 图转换为关系模式，并指出每个关系模式的主码和外部码。

(3）写出创建科研人员表的SOL 语句,表名和属性名可以用中文，相关各属性项的类型与约束如下:(3.分)

职工号:整型,长度6,唯一,非空;

姓名:字符型,长度4，非空;

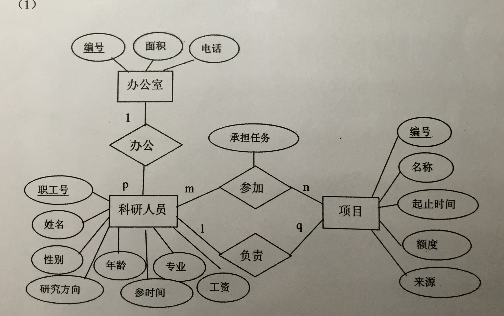
性别:字符型,长度1,默认值为:’男’;出生年月:日期型;

专业:字符型，长度20;

研究方向:字符型,长度15，非空;参加工作时间:日期型;

工作年限:整型，长度2,取值大于0;工资:整型,长度5;

所在办公室:字符型，3;



办公室（编号,面积，电话);

科研人员（职工号，姓名，性别，年龄，专业，工资，参时间,研究方向,办公室编号);

项目(编号,名称,起止时间,额度,来源，负责人职工编号);

参加项目(职工号,项目编号,承担任务)。

（3）

Creatle table教师(

职工号char(7) not null unique,

姓名char(20) not null,

性别char(2)

年龄 int,

专业 char(8) not null,

研究方向 char (10)

工资int ,

参加工作时间 date,

办公室编号char(6),

Primary key(职工编号),

Foreign key (办公室编号))

1. （2012秋-期末-2.3）

关系模式规范化属于数据库设计的

A.概念设计阶段 B.逻辑设计阶段

C物理设计阶段 D.数据库运行维护阶段

B

1. 聚集方法的缺点（自制）

建立与维护聚集系统开销很大，对于更新操作远远多于连接操作的关系不应使用聚集方法。（）

对

1. 存取方法（2011秋-期末-3.1）

在进行数据库物理设计时，一般有三种存取方法可供选择，它们分别是:索引存取方法、HSAH存取方法和聚簇存取方法。

对

1. 什么是索引方法，什么是聚集方法？（自制）

索引，类似于B+树。

把关系中某个属性/组(聚集键)值相同的记录集中存放在连续的物理块，称为聚集。能够提高该属性的查询速度

1. 一个关系不能参加多个聚集，索引越多，反而可能造成效率下降。（）

对

# 存储管理和索引

1. 2021秋-期末-3.x

最小化磁盘和主存间传输存储块的数量，即最小化磁盘存取次数；实现手段是在主存中保持尽量多的块（）

对

1. 2021秋-期末-2.X

下列说法错误的是

稀疏索引只有部分索引域值有索引记录。

B和B+树限制了每个节点放置关键字与指针的最小和最大个数

Hash索引不适用于区间值的检索，以及部分匹配检索

B和B+树的关键字散布在各层上。

D

# 关系查询处理与查询优化

## 目录

+ 查询的步骤

    + 查询分析

    + 查询检查

    + 查询优化

    + 查询执行

+ 查询执行

    + 选择运算的实现

    + 排序运算的实现

    + 连接的实现

        + 嵌套循环

        + 索引连接法

        + 排序-合并

+ Hash Join法

+ 查询优化

    + 必要性

    + 目标

    + 代数优化

    + 物理优化

## 题目

1. 选择操作有哪些方法，具体如何操作？（自制）

全表扫描法和索引扫描法。

全表扫描法：按照物理顺序读表的M块到内存，检查内存的每个元组t，如果满足条件则输出t，直到表所有块都经过上述检查

索引扫描法：如果在选择条件的属性上有索引，先通过索引找到目标索引项，再通过索引项找到元组

1. 连接操作包括哪些方法？

嵌套循环；索引连接；排序-合并；hash join法

1. 2012秋-期末-3.2

在执行不相关子查询时，如果外部查询的候选行有20行，则内部的嵌套子查询需要执行20次。

答案：错。

1. 建立索引（2020秋-期末-6）

对BORROW表的（SNO，BNO）建立Unique类型的索引

CREATE UNIQUE INDEX index\_name

ON BORROW (SNO, BNO)

1. 表达式的执行有哪些方法？

可选方法包括物化(materialized)方法和流水线

1. 2012秋-期末-1.10

按照优化的层次，查询优化一般可以分为:\_\_\_\_和\_\_\_\_。

代数优化，物理优化

1. \_\_\_\_是优化策略中最重要、最基本的一条。

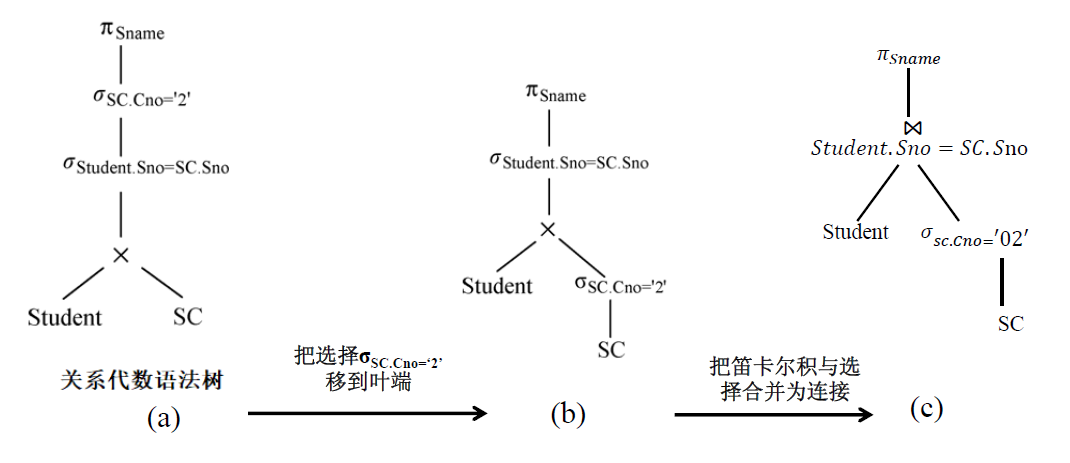
选择运算尽早执行

1. 课件（类似2021秋-4.1）

SELECT Student.Sname

FROM Student，SC

WHERE Student.Sno=SC.Sno AND SC.Cno=‘2’；



1. 为下列优化选择对应的方法：

选择操作的启发式规则

–对于小关系，使用\_\_\_\_。

–对于大关系：可以采用\_\_\_\_（如结果的元组数目较小）。

连接操作的启发式规则

–如果两个表都已经按照连接属性排序——\_\_\_\_；

–如果一个表在连接属性上有索引——\_\_\_\_；

–如果连接属性上未排序且未建索引，且其中一个表较小——\_\_\_\_；

–最后可选用\_\_\_\_，并选择较\_\_\_\_（大/小）的表为外循环表。

全表扫描; 索引扫描法

排序-合并法; 索引连接法; Hash join法; 嵌套循环法; 小

# 事务处理技术

## 目录

+ 事务

    + 特性

+ 数据库恢复技术

    + 恢复的实现技术

        + 数据转储

          + 转储状态：静态转储、动态转储

          + 转储方式：海量转储、增量转储

        + 日志

        + 检查点（checkpoint）

    + 故障的恢复策略

        + 事务的故障：UNDO

        + 系统的故障

        + 介质故障

+ 并发控制技术

    + 必要性

        + 丢失更新

        + 脏读

        + 重复读

    + 基本手段——封锁

        + 三级协议

        + 多粒度封锁

    + 死锁与活锁

        + 预防

        + 死锁的检测

    + 事务调度

## 题目

1. 事务在哪两种情况下可能对原子性造成破坏？（自制）

（1）多个事务并行运行时，不同事务的操作交叉进行；

（2）事务在运行过程中被强行停止

并发和强制停止

1. \_\_\_\_是数据库恢复和并发控制的基本单位。

事务

1. sql中事务定义语句:



1. 2012秋-期末-4.1

事务的4个特性在什么情况下会遭到破坏，数据库系统中采用什么技术保让事务的这些特性不被破坏?

事务的特性分别为:原子性,一致性，隔离性和持续性。

原子性是指:事务是数据库工作的最基本单位，事务中的诸操作要么全做，要么全不做。造成原子性破坏的原因主要是由于程序故障、系统故障和介质故障等。需要由数据库使用日志、转储等技术实现的恢复机制加以保护。

一致性是指:事务的执行结果必须使数据库从一个一致性状态变为另外一个一致性状态。造成一致性破坏的原因是事务的原子性被破坏等,需要使用数据库恢复和完整性约束机制来加以保护。

隔离性是指一个事务的执行不能被其他事务干扰。造成隔离性被破坏的原因主要是事务的并发执行所产生的错误,需要利用并发控制机制，如封锁、锁协议等加以保护。

事务的持续性是指，一个事务一旦被提交，它对数据库中数据的改变就是永久的。造成持续性被破坏的原因主要是原子性遭到了破坏,|需要使用数据库恢复机制加以保护。

1. 什么是数据库的恢复？其基本原理是\_\_\_\_。（自制）

数据库管理系统必须具有把数据库从错误状态恢复到某一已知正确状态的功能。

数据库恢复的基本原理为冗余，利用存储在别处的数据恢复现有数据。

1. 数据转储分为两种转储状态分别是\_\_\_\_和\_\_\_\_。

静态转储; 动态转储

1. 2012秋-期末-2.5

在数据库系统中，日志文件用于记录:

A.对数据库的所有操作。

B.对数据库的所有更新操作。

C.所有访问数据库的程序名。

D.所有访问数据库的用户名和口令。

B

1. 2011秋-期末-1.8

在进行数据库系统故障的恢复时，对于已提交的事务，应当\_\_\_\_扫描日志文件,将日志记录中\_\_\_\_的值写回数据库。

正向，更新

1. 事务故障如何恢复？

UNDO，即撤消事务

在不影响其它事务的情况下，强行回滚，撤消已做的修改。具体步骤：

反向扫描日志文件，查找该事务的更新操作；

对该事务的更新操作（插入、删除、修改）执行逆操作，即将日志记录中的“更新前的值”写入数据库；

如此处理下去，直到读到该事务的开始标志。

1. 系统故障如何恢复？（自制）

UNDO+REDO

系统故障造成数据库不一致状态的原因有两个：

一是未完成的事务对数据库的更新可能已经写入数据库；

二是已提交事务对数据库的更新可能还留在缓冲区未写入数据库。

因此恢复操作就是要撤销(UNDO)故障发生时未完成的事务，重做(REDO)已完成的事务

具体步骤：

正向扫描日志文件，找出故障发生前已经提交（也就是写道日志里的）的事务，将其事务标识记入重做（REDO）队列。

同时找出故障发生时尚未完成的事务，将其事务标识记入撤销（UNDO）队列；

对撤销队列中的各个事务进行UNDO处理；

对重做队列中的各个事务进行REDO处理。

1. 数据库系统故障恢复（2010秋-期末-2.8）

对数据库的系统故障进行恢复时,应当\_\_\_\_。

A.正向扫描日志文件，建立Undo队列和 Redo队列。

B.反向扫描日志文件,建立Undo队列和 Redo队列。

C.正向扫描日志文件，建立Redo队列;反向扫描日志文件，建立Undo队列。

D.反向扫描日志文件，建立Redo队列;正向扫描日志文件,建立Undo队列。

A

1. 介质故障的如何恢复？

恢复的方法：

–装入最新的数据库后备副本，使数据库恢复到最近一次转储时的一致状态。对于动态转储的副本，还需要装入转储开始时刻的日志文件副本，将数据库恢复到一致状态；

–装入转储以后的日志文件副本，重做已经写入到日志的事务。

1. 简述检查点技术（hqy）

系统用检查点方法进行恢复的步骤是：

（1）在重新开始文件中找到最后一个检查点记录在日志文件中的地址，由该地址在日志文件中找到最后一个检查点记录。

（2）由该检查点记录得到检查点建立时刻所有正在运行的事务清单**ACTIVE-LIST**(暂时放入**UNDOLIST**)

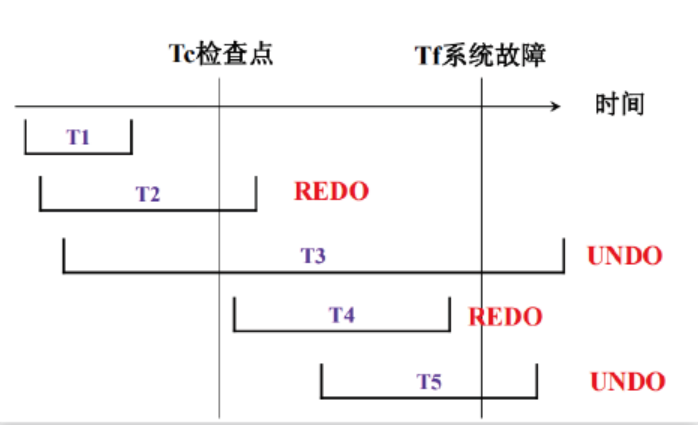
（3）从检查点开始正向扫描日志文件：

如果有新开始的事务Ti，把Ti暂时放入UNDO-LIST;

如果有提交的事务Tj ，把Tj从UNDO-LIST队列移入到REDO-LIST队列；

直到文件结束，对UNDO-LIST中的每一个事务执行UNDO操作，对REDO-LIST中的每个事务

执行REDO操作。



从检查点到故障点，已经提交的就redo，有开始却没提交就undo

1. 如何利用检查点技术恢复数据？

（1）利用重新开始文件定位最近检查点记录

（2）找到检查点时刻运行事务清单：由该检查点记录得到检查点建立时刻所有正在运行的事务清单ACTIVE-LIST，把ACTIVE-LIST 暂时放入UNDO-LIST 。

（3）确定需要撤消和重做的事务：从检查点开始正向扫描日志文件，做如下处理，直到文件结束。

如果有新开始的事务Ti，把Ti暂时放入UNDO-LIST;

如果有提交的事务Tj ，把Tj从UNDO-LIST队列移入到REDO-LIST队列；

（4）执行撤消或重做动作：对UNDO-LIST中的每一个事务执行UNDO操作，对REDO-LIST中的每个事务执行REDO操作

1. 2012秋-期末-1.6

在进行数据库系统故障的恢复时,对于已提交的事务应当进行\_REDO操作,对于未提交的事务应当进行\_UNDO操作。

1. 并发若不做控制，可能发生\_\_\_\_、\_\_\_\_和\_\_\_\_。

丢失更新；脏读；重复读。

1. 2012秋-期末-2.10

在数据库中，产生数据不一致的根本原因是

A.数据库保护机制不完善

B.数据库安全控制机制不完善

C.数据冗余

D.数据存储量大

C

1. 三级封锁协议的内容：

一级封锁协议：事务T在修改数据R之前必须对其加X锁，直到事务结束才释放。

二级封锁协议：一级封锁协议加上事务T在读取数据R之前必须先对其加S锁，读完后即可释放S锁。

三级封锁协议：一级封锁协议加上事务T在读取R之前必须对其加S锁，直到事务结束才释放

1. 2011秋-期末-2.10

二级锁协议可以

I.防止读“脏数据”。

II.保证可重复读。

II.防止丢失更新。

IV.保证可串行化。

A.只有III

B.II和IV

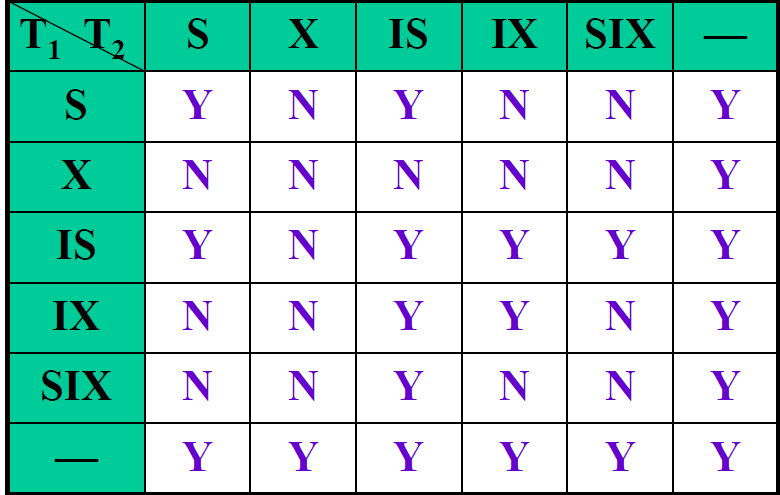
C.I和III

D.I和II

C

可重复读是三级，可串行化是二段锁协议。

1. 确保会下面这张表



IS和IX之间都是Y

IS+IX：IS说，我下面某一个结点被读，来个IX，没关系，结点不一定冲突，冲突它们之间会处理的。

IX+IS：IX说，我下面某一个结点被修改，来个IS没关系，IS不一定读我修改的那个。

参考资料：<https://www.zhihu.com/question/41172525>

1. 2012秋-期末-2.7

若事务T对数据对象A加上IS 锁，则其他事务不能在A上加\_C-。

A.IS锁

B.X锁

C.X锁

D.SIX锁

1. 2011秋-期末-3.5

在使用多粒度封锁技术的情况下,如果要对多粒度树中的一个节点加排他锁，则应对其上级节点加X锁。

错误。

对其子节点加排他锁。

1. 死锁的检测包括\_\_\_\_和\_\_\_\_\_。

超时法、等待图法

1. 2012-期末-2.6

在数据库中为了发现和解除死锁，可以采用。

A.一次封锁法

B.顺序封锁法

C.等待图法

D.两阶段锁协议

C

1. 2011秋-期末-1.7

预防死锁的方法主要有：\_\_\_\_\_和\_\_\_\_。

一次封锁法;顺序封锁法。

1. 多个事务的并发执行是正确的，当且仅当其结果与按\_\_\_\_时的结果相同，我们称这种调度策略为\_\_\_\_。\_\_\_\_是并行事务正确性的准则。一个给定的并发调度，当且仅当它是\_\_\_\_的，才认为是正确调度。

某一次序串行执行; 可串行化调度; 可串行性; 可串行化

1. 判定下列调度是不是可串行化的（郎波课件）

Sc2=R1(A)W1(A) R1(B) W1(B)R2(A)W2(A) R2(B)W2(B)

经交换操作得到

Sc1=R1(A)W1(A)R2(A)W2(A)R1(B)W1(B)R2(B)W2(B)

是可串行化调度。

1. 丢失更新（2010秋-期末-2.9）

(9)在并发调度中，事务T1和T2的下列操作顺序可能导致丢失更新的是:.

A. R1(X).W2(X),Rl(X)

B.R1(X).W2(Y),Rl(Y)

C. W1(Y),W2(X).R2(X)

D. W1(X),W2(X)，R2(Y)

D

1. 两段锁协议(Two-phase Locking)可保证并行事务的\_\_\_\_。

可串行性

1. 两段锁协议的内容

（1）在对任何数据进行读、写操作之前，事务首先要获得对该数据的封锁。

（2）在释放一个封锁之后，事务不再获得任何其它封锁。

1. 事务分为两个阶段，第一个阶段是获得封锁，也称为\_\_\_\_；第二个阶段是释放封锁，也称为\_\_\_\_。

扩展阶段; 收缩阶段

1. 2012秋-期末-1.7

多个事务并发事务调度的可串行化，是指这个并发调度的执行结果与按某一次序\_\_\_\_它们时的结果相同。\_\_\_\_可保证事务并发调度的正确性。

串行执行，两段锁协议。

# 数据库保护

## 目录

+ 数据库安全性控制

    + 用户标识与鉴别

    + 存取控制

        + 自主存取控制

            + sql实例

                + 用户级权限

                + 关系级权限

        + 强制存取控制

    + 视图

    + …

+ 数据完整性控制

    + 完整性内容

        + 正确性

        + 相容性

    + 约束条件

        + 静态

    + 控制方法

        + 定义功能

        + 检查功能

        + 违约响应

## 题目

1. 数据库保护分为\_\_\_\_和\_\_\_\_。

数据库安全性控制；数据库完整性控制。

1. 列出安全性控制的技术。

用户标识与认证

访问控制（存取控制）

加密技术

审计

1. \_\_\_\_是系统提供的最外层安全保护措施。 常用的方法是采用\_\_\_\_和\_\_\_\_。

用户标识和认证；用户名；口令

1. 存取控制的方法分为\_\_\_\_和\_\_\_\_。

自主存取控制：

用户对于不同的数据对象拥有不同的存取权限，不同的用户对同一对象也有不同的权限，而且用户还可以将其拥有的权限转授给其他用户。

强制存取控制**：**每一个数据对象被标以一定的密级，每一个用户也被授予某一个级别的许可证。对于任一个对象，只有具有合法许可证的用户才可以存取

1. sql中可以授予用户两类权限：\_\_\_\_和\_\_\_\_。

用户级权限；关系级权限。

1. SQL使用哪些语句为用户授予权限，举例说明。（自制）

通过Grant语句授予用户用户级权限或角色

Grant <用户级权限>|<角色> [{,<用户级权限>|<角色>}]

To <用户名>|<角色>|public [{,<用户名>|<角色>}]

[With Grant Option]

当要取消一个用户或角色的权限时，可以使用Revoke语句将其收回：

Revoke <用户级权限>|<角色> [{,<用户级权限>|<角色>}]

From <用户名**>|<**角色**>|public [{,<**用户名**>|<**角色**>}]**

1. 关系级权限含义

是数据库管理员或数据库对象的拥有者为用户授予的与\_\_\_\_有关的权限。这种权限是对用户使用关系和视图权限的限定。

关系或视图

1. (2)数据库的存取控制方法主要有:强制存取控制和自主存取控制。
2. SQL使用哪些语句为关系或视图授予权限，举例说明。（自制）

Grant **ALL| <**权限**> [{,<**权限**>}]**

**On <**表名> | <视图名> [{,<表名> | <视图名>}]

To {<用户> [{, <用户>}] | public}

[With Grant Option]

Revoke ALL|<表级权限> [{,<表级权限>}]

On <表名>|<视图名> [{,<表名> | <视图名>}]

From {<用户>[{,<用户>}]|PUBLIC}

–收回权限时，若该用户已将权限授予其它用户，则也一并收回



1. SQL用户级权限和关系级权限区分（2020秋-期末-3.5）

语句”Grant All On Course To PUBLIC”定义权限为关系级权限（）

对。因为把一个表（关系）的权限给了别人

1. 在**MAC**中，**DBMS**所管理的全部实体被分为\_\_和\_\_两类。前者是系统中的活动实体，既包括**DBMS**所管理的实际用户，也包括代表用户的各进程。 后者是系统中的被动实体，是受主体操纵的，包括文件、基本表、索引、视图等。

主体；客体

1. 对于主体和客体，**DBMS**为他们每个实例指定\_\_\_\_。敏感度标记被分为若干级别，如绝密、机密、秘密、公开等。主体的敏感度标记称为\_\_\_\_，客体的敏感度标记称为\_\_\_\_。 **MAC**机制通过对比\_\_\_\_和\_\_\_\_，最终确定主体是否能够存取客体。

一个敏感度标记**(Label);** 许可证级别; 密级;主体的**Label;** 客体的**Label**

1. 写入如下sql语句：

限制王平只能检索**Student**表中计算机系学生的学号和姓名。

Create View CS\_Student

As Select Sno, Sname From Student

Where Sdept = ‘CS’;

Grant Select On CS\_Student To Wangping

1. 数据完整性是指数据的正确性和相容性。它们的含义是什么？

正确性；相容性。

正确性是指数据应具有合法的类型，并在有效的取值范围之内。

相容性是指表示同一个事实的两个数据应该相同。

1. 施加在数据库数据之上的\_\_\_\_称为数据库完整性约束条件。 完整性约束条件作用的对象可以是\_\_\_\_三种。

语义约束条件; 列、元组、关系

1. 约束分为哪些

静态列级约束：对一个列的取值域的说明，包括对数据类型（包括数据类型、长度、单位、精度等） 、数据格式、取值范围或取值集合、空值等的约束。

静态元组约束：

规定了组成一个元组的各个列之间的约束关系。

静态关系约束：

规定了一个关系的若干元组或者若干关系之间常常存在的各种联系或约束。包括：实体完整性约束、参照完整性约束、函数依赖、统计约束等。

动态约束

•动态列级约束是修改列定义或列值时应满足的约束条件;

•动态元组约束指修改元组值时元组中各个字段间需要满足的约束;

•动态关系约束是加在关系变化前后状态上的限制条件。

1. 维护完整性

为维护数据库的完整性，DBMS 应当:提供定义完整性约束条件的机制，提供完整性检查的方法,违约处理\_。

1. 数据库完整性控制包括三个方面的功能： \_\_\_\_，提供定义完整性约束条件的机制；\_\_\_\_，检查用户发出的操作请求是否违背了完整性约束条件；\_\_\_\_，若违背了完整性约束条件，则采取一定措施来保证数据的完整性。

定义功能；检查功能；违约响应。

1. 完整性规则如何表示？

一条完整性规则可以用一个五元组 ( D，O，A，C，P ) 来描述，其中：

D ( Data ) 约束所作用的数据对象

O ( Operation ) 触发完整性检查的数据库操作。

A ( Assertion ) 数据对象必须满足的断言或语义约束。

C ( Condition ) 选择A作用的数据对象值的谓词。

P ( Procedure ) 违反完整性规则时触发的过程。

1. 完整性约束条件按照完整性检查的时机分为\_\_\_\_和\_\_\_\_。

立即执行约束; 延迟执行约束

1. 说一说SQL的完整性支持，并举例。

create table，create assertion，create trigger

**C**REATE TABLE 语句

Create Table <表名>

(<列名><数据类型>[<列级完整性约束>]

[{,<列名><数据类型>[<列级完整性约束>]}]

[{, [<表级完整性约束>]}]);

–完整性约束

•NULL/NOT NULL

•UNIQUE

•PRIMARY KEY

•FOREIGN KEY

•CHE**CK**

1. 为SC表建立断言，满足约束每一门课程最多**60**人选修 。（郎波PPT）

CREATE ASSERTION ASSE-SC-SNUM1

CHECK (

60>= ALL( SELECT COUNT(\*)

FROM SC

GROUP BY C#) );

触发器（**Trigger)**是用户定义在关系上的一类由事件驱动的特殊过程

•对于用户对表的更新操作，系统自动激活相应触发器，执行完整性控制

•定义触发器

–**CREATE TRIGGER <**触发器名称**>**

**{BEFORE|AFTER}<**触发器事件**> ON <**表名**>**

**REFERENCING NEW | OLD ROW AS <**变量**>**

**FOR EACH {ROW|STATEMENT}**

**[WHEN <**触发条件**>]**

**<**触发动作体**>**

# 分布式数据库

## 目录

## 题目

1. 分布式数据库以“\_\_\_\_”为前提，强调\_\_\_\_（局部应用）以及\_\_\_\_（全局应用），两者缺一不可

数据分布; 场地自治性; 自治场地之间的协作性

1. 分布式数据库，数据独立性除了逻辑独立性和物理独立性，还具有\_\_\_\_，也称为\_\_\_\_。

分布独立性；分布透明性。

1. 分布式数据库透明性实现（2010秋-1.10）

在分布式数据库系统中,通过\_\_\_\_和\_\_\_\_之间的映像可以实现分片透明性。

全局模式；分片模式

1. (10)分布式数据库以数据分布为前提,强调场地自治性与场地之间协作性、在分布式数据库中，分布透明性包括：分片透明性，位置透明性和局部数据模型透明性。
2. (9)分布透明性包括:分片透明性,位置透明性和局部数据模型透明性
3. 简述分布数据库的特点

（1）数据独立性

•数据的逻辑独立性和物理独立性

•数据的分布独立性（也称分布透明性）：

（2）集中与自治相结合的控制结构

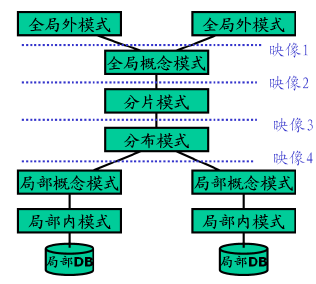
（3）适当增加数据冗余

（4）全局的一致性、可串行性和可恢复性

1. 列出分布式数据库的模式结构。

全局外模式；全局概念模式；分片模式；分布模式；

局部概念模式；局部内模式



1. 全局外模式全局应用的\_\_\_\_，是全局概念模式的\_\_\_\_。

全局概念模式定义分布式数据库中数据的\_\_\_\_，使得数据如同没有分布一样。

用户视图;子集；整体逻辑结构

1. 水平分片的概念（2021秋-期末-1.x）

不会做。

分片的方式有多种：

–水平分片，垂直分片**—**两种基本的分片方式

–混合分片，导出分片**—** 较复杂的分片方式

水平分片

–将关系依照一定条件按行分为不相交的若干子集，每个子集称为一个水平片段。

•垂直分片

–将关系按列分为若干属性子集，每个子集称为一个垂直片段。垂直分片的片段通过连接的方法恢复原关系。因此垂直分片的诸片段通常都包含关系的码。

•导出分片

–导出水平分片，分片的条件不是关系本身属性条件，而是其它关系的属性条件。如**SC**（**SNO**，**CNO**，**G**）按**S**关系中学生所在的系分片。

•混合分片

–指按上述三种分片方式得到的片段，继续按另一种方式分片。

1. 数据分片的约束有：

完全性：一个全局关系中的数据必须完全划分为若干片段

不相交性 ：不允许一个全局关系的某些数据既属于该全局关系的某一个片段，又属于另一个片段（垂直分片的码属性除外） 。

可重构性 ：可以由片段重构全局关系

1. 分布透明性（分布独立性）包括：\_\_\_\_，\_\_\_\_，\_\_\_\_。

\_\_\_\_是分布透明性的最高层次。

分片透明性; 位置透明性; 局部数据模型透明性;分片透明性

1. 分布式数据库由哪些部分组成

LDBMS(局部场地上的数据库管理系统）

•功能：建立和管理局部数据库，提供场地自治能力，执行局部应用及全局查询的子查询。

–GDBMS（全局数据库管理系统）

•功能：提供分布透明性，协调全局事务的执行，协调各局部DBMS以完成全局应用，并保证数据库的全局一致性，执行并发控制，实现更新同步，提供全局恢复功能。

–GDD（全局数据字典）

•存放全局概念模式、分片模式、分布模式的定义以及各模式之间映象的定义。

•存放有关用户存取权限的定义，以保证全局用户的合法权限和数据库的安全性。

•存放数据完整性约束条件的定义。

–CM（通信管理）

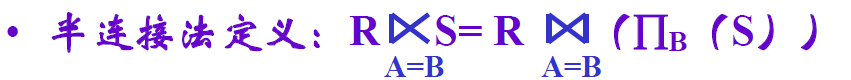
1. 查询优化（自制）

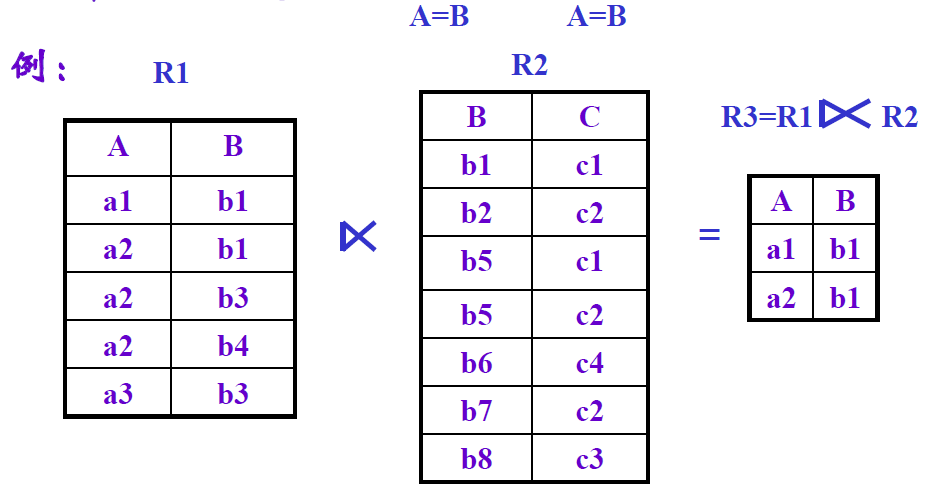
不同结点之间的\_\_\_\_和\_\_\_\_是数据传输的主要原因，因此连接查询的优化是优化中研究的重要问题。

连接操作;并操作

1. 给出半连接的定义并举例说明

半连接：使用半连接来缩减关系（或片段）进而节省传输开销。

定义：



1. 当**R**中参与连接的元组足够少时采用半连接策略是有利的（）

对

1. 半连接（2011秋-期末-3.4）

分布式数据库中的半连接，是通过缩减关系(或片段）来节省通信开销，从而实现连接运算优化的一种方法。

答案：对

1. 数据库查询优化（2010秋-期末-9）

在分布式数据库系统中，\_\_\_\_和\_\_\_\_可以使用集中式数据库的查询优化技术，而\_\_\_\_查询是分布式数据库查询优化的主要对象,连接查询的优化通常采用的技术是\_\_\_\_。

局部查询; 远程查询; 全局;半连接

1. 在分布式数据库系统中，一个全局事务被划分为\_\_\_\_。

分布事务的原子性是\_\_\_\_。 在多用户系统中，还必须保证分布式事务的\_\_\_\_。

许多结点上的子事务；组成该事务的所有子事务要么一致地全部提交，要么一致地全部回滚。 可串行性

1. 分布事务管理主要包括：事务的恢复和并发控制。
2. 简述两段提交协议的内容

协调者（一个），参与者。

协调者：负责作出该事务是提交还是撤消的最后决定。

参与者：负责管理相应子事务的执行以及在各自局部数据库上执行写操作。

第一阶段：协调者征求意见作决定

第二阶段：参与者执行决定

# 参考资料

范式：<https://blog.csdn.net/wyh7280/article/details/83350722>