#数据库第一章 绪论

&1.1数据库系统概述

1. 基础术语

数据（DATA）：数据是数据库中存储的基本对象。（描述事物的符号记录+语义）

EG.学生档案中的一条记录

(黎明，男，19970212，江苏南京，计算机，2015)

语义：姓名，性别，出生日期，籍贯，所在系别，入学时间

数据库（DB）:数据库是长期存储在计算机内，有组织的、可共享的大量数据的集合。

特征: a.数据按照一定的数据模型进行组织、存储和描述。

b.可为各种用户所用

c.冗余度(不同应用程序有些部分有相同的数据，必须是各自建立对应的文件)较小

d.数据独立性（修改逻辑结构之后是否修改应用程序）高

e.易扩展

数据库管理系统(DBMS):位于用户和操作系统之间的一层数据管理软件。

△功能：a数据定义功能：提供数据定义语言DDL可以用拉力定义数据对象

b数据操纵功能(DML)：数据的增删改查

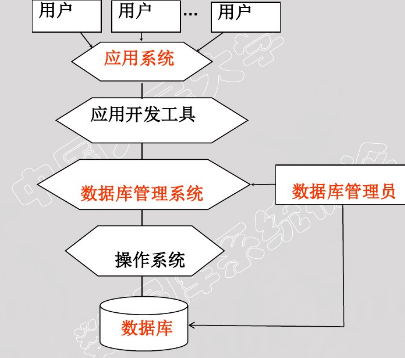
并发性(允许多个用户同时进行操作)

c 数据库的运行管理 完整性（对字段输入时的限制）

eg.性别在输入时候只接受录入男或者女

安全性(访问数据库的权限)

d 数据库的建立与维护：备份/存储(采取方式：双机异地备份)

数据库系统(DBS):由数据库、数据库管理系统、应用程序和数据管理员组成

二 数据库系统的特点

**1数据结构化**

**2数据的共享度高，冗余度低且易扩充**。

**3数据独立性高**

包括物理独立性和逻辑独立性。

物理独立性：用户的应用程序和数据库中数据的物理存储是相互独立的。当数据的 物理存储改变了，应用程序不用改变。

逻辑独立性：指用户的应用程序与数据库的逻辑结构是相互独立的。数据的逻辑结构改变了，应用程序不用改变。

△数据的独立性由数据库管理系统的二级映像来保证。

**4数据由数据库管理系统统一管理和控制**

1. 数据安全性保护

保护数据以防止不合法使用造成的数据的泄密和破坏

1. 数据完整性检查

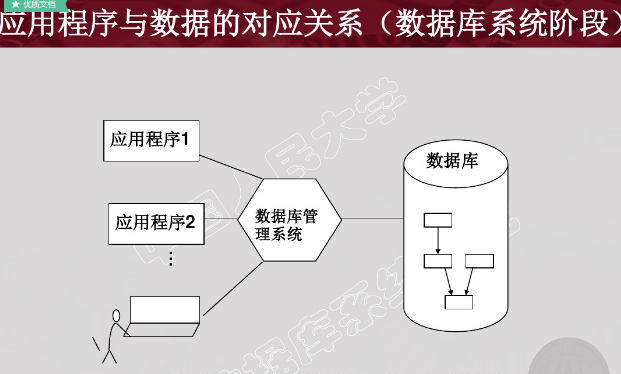
保证数据的正确性、有效性和相容性

1. 并发控制

对多用户的并发操作加以控制和协调，防止相互干扰而造成错误

1. 数据库恢复

将数据库从错误的状态恢复到某一已知正确状态



**&1.2数据模型**

1. **两类数据模型**

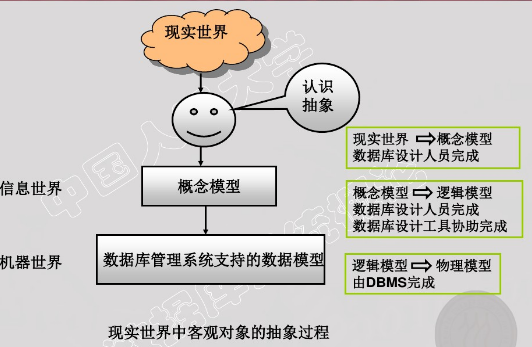
1.概念模型

按照用户观点对数据和信息进行建模，用于数据库设计。

2.逻辑模型和物理模型

逻辑模型主要包括网状模型、层次模型、关系模型、面向对象数据模型、对象关系数据模型等。

物理模型是对数据最底层的抽象，是数据在系统内部的表示和存取方法。



1. **概念模型**

**1.概念模型基本术语**

（1）实体(Entity)

客观存在并可以互相区分的事务成为实体。可以是具体的人、事、物或抽象的概念。

（2）属性(Attribute)

实体所具有的某一特征。一个实体可以有若干个属性来进行刻画。

（3）码(Key)

唯一标识实体的属性称为码。

Eg.学号是学生实体的码。

（4）域(Domain)

一组具有相同数据类型的值的集合。属性的取值范围来自某个域。

Eg.姓名的域为（男，女）

（5）实体型（Entity type）

用实体名及其属性名集合来抽象和刻画的同类实体称为实体型。

Eg.学生(学号，姓名，年龄，性别，籍贯，院系，入学时间)

（6）实体集（Entity set）

同一类型的实体的集合。

Eg.全体学生。

**二．联系**

现实世界中事物内部以及事物之间的联系在信息世界中反应为实体（型）内部的联系和实体（型）之间的联系。

实体（型）内部的联系：组成实体的各个属性之间的联系

实体（型）之间的联系：不同实体集之间的联系

实体之间的联系包括：一对一，一对多和多对多。

1. 一对一联系 1:1

对于实体集A中的每一个实体，实体集B中至多有一个（可以没有）实体与之联系，反之亦然。

Eg.一夫一妻

1. 一对多的联系1:n

对于A中的每一个实体，实体集B中可以有多个实体与之对应，反之，对于实体集B中实体A中至多只有一个实体与之对应

1. 多对多的联系m:n

对于实体集A中的每一个实体，实体集B中国有n个实体与之对应，反之，对于实体集B中的每一个实体实体集A中也有m个实体与之对应。

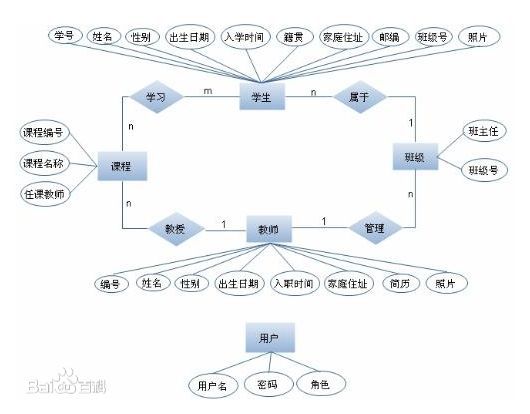
三.E-R图

矩形框：表示实体，在框中记入实体名。

菱形框：表示联系，在框中记入联系名。

椭圆形框：表示实体或联系的属性，将属性名记入框中。对于主属性名，则在其名称下划一下划线。

连线：实体与属性之间；实体与联系之间；联系与属性之间用直线相连，并在直线上标注联系的类型。（对于一对一联系，要在两个实体连线方向各写1； 对于一对多联系，要在一的一方写1，多的一方写N；对于多对多关系，则要在两个实体连线方向各写N,M。)



**三.数据模型**

**从计算机角度出发，数据模型是信息模型的数据化**

1.数据模型组成要素：p13

（1）数据结构

描述对象和对象之间的联系

（2）.数据操作

（3）.数据完整性约束条件

2.最常用的数据模型(只要求掌握关系模型)

（1）关系

一个关系通常对应一张表

（2）元祖

表中的一行即为一个元祖

（3）属性

表中的一列即为一个属性，每一个属性起一个名字即为属性名。

（4）主码

表中的某个属性组，它可以唯一的确定一个元祖。

（5）域

一组具有相同数据类型的值的集合。属性的取值范围来自某个域。

（6）分量

元祖其中一个属性值

（7）关系模式

关系名(属性1，属性2，属性3…)

2.2关系模型规范

关系要求是规范化的，关系中的每一个分量都必须是不可分项，不允许表中有表



2.3 关系数据模型的操纵与完整性约束

操纵：查询、插入、删除、更新数据

完整性约束：实体参照完整性、参照完整性和用户定义完整性

**△ 实现从概念模型🡪数据模型的转化**

**实体集A与实体集B之间：**1：m的原则

1. 实体型直接转为关系模式
2. 将实体集A中的码连同联系的属性一起放入B中

**实体集A与实体集B之间：**1：1的原则

1. 实体型直接转为关系模式
2. 将任意一个实体集中的码连同联系的属性一起放入另一个实体集中

**实体集A与实体集B之间：**m：n的原则

1. 实体型直接转为关系模式
2. 产生一个新的关系名，新的关系名是联系名，双方实体的KEY和新产生的属性放入新的关系中。

#`1.3数据库系统结构

1. **数据库系统模式的概念**
2. 型和值的概念

型：对某一类数据的结构的和属性的说明。

值:是型的一个具体实例。

EG.学生记录(学号，姓名，性别，院系，籍贯，年龄)

（150410210，彭一，女，计算机，重庆，20）

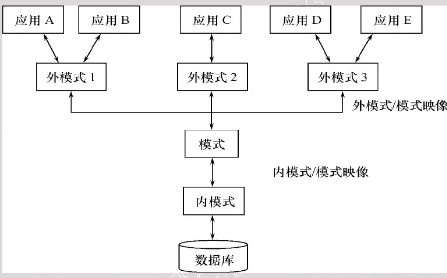
1. 模式的概念

模式数据库逻辑结构和特征的一个描述，是型的描述不涉及具体的值，反应的是数据的结构及其联系，模式是相对稳定的。

1. 实例的概念

实例是模式的一个具体的值，反映数据库某一时刻的一个状态，同一个模式可以有多个实例，实例随着数据库中的数据更新而变动。

1. **数据库系统的三级模式结构**

****

1. **模式（逻辑模式）**

数据库中全体数据的逻辑结构和特征的描述，所有用户的公共数据视图。

* 一个数据库只有一个模式

1. **外模式（子模式或者用户模式）**

数据库用户（包括应用程序员和最终用户）使用的局部数据的逻辑结构和特征描述。

△模式与外模式之间的关系是：一对多。

同一个外模式可以为某一用户的多个应用程序所使用，但是一个应用程序只能对应一个外模式

1. **内模式**

数据物理结构和存储方式的描述，是数据在数据库内部的表示方式。

△ 一个数据库只有一个内模式

**三.数据库系统的二级映像功能与数据独立性**

**1.外模式/模式映像**

每个外模式都有一个外模式/模式映像，它保证了数据的**逻辑独立性**，即：当模式改变时，数据库管理员对外模式/模式映像做出改变，而应用程序可以不用修改

**2.模式/内模式映像**

数据库中模式/内模式映像是唯一的，它保证了数据的物理独立性，即：当数据库的存储模式改变时，只需修改该映像，而应用程序无需做出修改。

#第二章 关系数据库

**&2.1关系数据结构及其形式化定义**

1. 关系（逻辑结构就是一张表）

关系是建立在集合的基础上的，从集合角度给出定义：

1. 域(Domain)

域是一组具有相同数据类型的值的集合。

1. 笛卡尔积：
2. 定义：

给定一组域D1,D2,…,Dn,允许其中某些域是相同的。D1,D2,…,Dn的笛卡尔积为

D1×D2×…×Dn={(d1,d2,…,dn)|di∈Di,i=1,,2,..n}

所有域的一个取值组合，不能重复

1. 元祖

笛卡尔积中每一个元素(d1,d2,…,dn)叫做一个n元祖或元祖

Eg.（张三，男，18,重庆）

1. 分量：

笛卡尔积元素（d1,d2,…,dn）中的每一个值di叫做一个分量

Eg.张三、男

1. 基数

若Di=(i=1,2,…,n)为有限集，其基数为mi(i=1,2,…,n)，则

D1×D2×…×Dn=

Eg.见书P42-43

1. 关系：

（1）定义

D1×D2×…×Dn的子集叫做在域D1，D2，…，Dn上的关系，表示为：

R(D1,D2,…,Dn)。其中R为关系名，n为关系的目或者度。

（2）元祖

关系中每个元素是关系中的元祖，用t来表示

（3）单元关系与二元关系

n=1时，成该关系为一元关系/单元关系；

n=2时，成该关系为二元关系

（4）关系的表示

关系也是一个二维表，表的每一行对应一个元祖，表的每一列对应一个域

1. 属性

关系中不同列可对应相同的域，为了加以区分对每一列起一个名字成为属性，n目关系必须有n个属性。

1. 码

候选码：若关系中的某一个属性值能唯一的标识一个元祖，则称该属性组为候选码。

全码：关系模式的所有属性组是这个关系模式的候选码，成为全码

主码：若一个关系有多个候选码，选择其中一个为主码。

主属性：候选码中的诸属性成为主属性

（7）（了解）三类关系

基本关系：实际存在的表，是实际存储数据的逻辑表示

查询表：查询结果对应的表

视图表：由基本表或者其他视图导出的表，不对应实际存储的数据

1. 基本关系的性质
2. 列是同质的
3. 不同的列可以出自同一个域：不同的列赋予不同属性名
4. 列顺序无所谓，列的次序可以任意交换
5. 任意两个元祖的候选码不能相同
6. 行的顺序无所谓，行的次序可以任意交换
7. 分量必须取原子值，即每一个分量都必须是不可再分项。
8. 关系模式

三.关系数据库

**&2.2关系操作**

常用的关系操作：选择、投影、连接、除、并、交、差、笛卡尔积等。

**&2.3关系的完整性**

1. 关系的三类完整性约束
2. 实体完整性

若A属性是基本关系R的主属性，那么A不能取空值。即：不能“不知道”、“不存在”、“无意义”。

1. 参照完整性
2. 关系间的引用

某个属性的取值需要参考另外一个属性的取值

1. 外码

设F为基本关系R的一个或一组属性，但不是关系R的码。如果F与基本关系S的主码Ks相对应，则称F是R的外码。

基本关系R称为参照关系，S称为目标关系（被参照关系）

PS.R和S不一定是不同的关系

外码不一定和相应的主码同名（但一般同名，便于识别）

Ks和F必须定义在同一个域上

1. 参照完整性规则

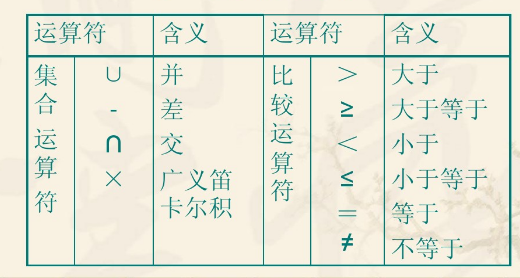
若属性（属性组）F是基本关系R的外码，它与基本关系S的主码Ks相对应（基本关系R和S不一定为不同的关系），则对于R中的每个元祖在F中的取值必须为：

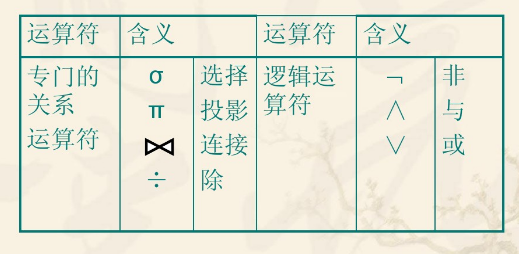
1. 空值（F中的每个属性取值均为空，表示尚未确定）
2. s中某个元祖的主码值
3. 用户定义完整性

针对某一具体关系型数据库的约束条件，反映某一具体应用所涉及的数据必须满足的语义要求。

关系模型应提供定义和检验这类完整性的机制，以便于用统一的方法来处理它们，而不需由应用程序来承担这一功能。

**&△2.4关系代数**





1. 传统的集合运算
2. 并

具有相同的目n（即两个关系都有n个属性），相应的属性取自同一个域

R∪S={t|t∈R∨t∈S}

1. 差

具有相同的目n，相应的属性取自同一个域

R－S={t|t∈R∧t∉S}

1. 交

具有相同的目n，相应属性取自同一个域

R∩S={t|t∈R∧t∈S} ;

R∩S=R-(R-S)

1. 广义笛卡尔积

R：n目关系，k1个元祖

S：m目关系，k2个元祖

R×S

列：（n+m）列元祖的集合

行：k1×k2个元祖



Eg,书P54页

二．专门的关系运算

1.几个术语

（1）R, t∈R, t[Ai]

设关系模式为R, t∈R表示t是R的一个元祖，t[Ai]表示元祖t总相应于属性Ai的一个分量。

（2）A, t[A],

若A=[Ai1, Ai2,…, Aik],其中Ai1, Ai2,…, Aik是A1,A2,…An中的一部分，则称A为属性列或者属性组。

t[A]=（t[Ai1], t[Ai2],…, t[Aik]）表示元祖在属性列A上的诸分量的集合

表示{A1,A2,…,An}中去掉{Ai1, Ai2,…, Aik}剩余的属性组

（3）

R为n目关系，S为m目关系，tr∈R，ts∈S，为元祖的连接。是一个m+n列的元祖。

1. 象集Zx

给定一个关系R(X,Z),其中X和Z为属性组。

当t[X]=x时，x在R中的象集为：

Zx={t[Z] | t∈ R,t[X]=x}

它表示R中属性组X上值为x的诸元祖在Z分量上的集合

Eg.

|  |  |  |
| --- | --- | --- |
| 学号 | 课程 | 成绩 |
| 1 | DB | 90 |
| 2 | DB | 82 |
| 1 | OS | 99 |

t[学号=“1”]的象集为{(DB,90),(OS,99)};

2专门的关系运算

1. 选择

在关系R中选择满足给定条件的诸元祖

σF(R) = {t|t∈R ∧ F(t)='真'}

F:选择条件，是一个逻辑表达式，取值为真或者假

基本形式：X1 θ Y1 ,θ表示比较运算符，它可以是>、≥、<、≤、=或≠

1. 投影

关系R上的投影是从R中选择出若干属性列组成新的关系。记作：

ΠA(R) = { t[A] | t∈R }

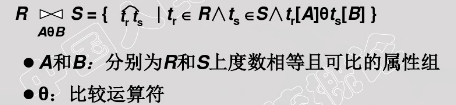
其中A为R中的属性列。

Ps.投影之后可能不仅取消了原关系中的某些列，还会取消某些元祖。因为取消某些列后可能出现重复行

1. 连接

①定义

从两个关系的笛卡尔积中选取属性间满足一定条件的元祖



连接运算从R和S的广义笛卡尔积R×S中选取R关系在属性A上的值与S关系在B属性组当中的值满足比较关系θ的元祖

②等值连接

③自然连接

④外连接

1. 除运算