仅供学习与交流

数据库 复习

一、 信息：新的、有用的事实和知识，其具有时效性、有用性、知识性

二、 数据：用于承载信息的物理符号

1、 有型和值之分

2、 受数据类型和取值范围的约束

3、 有定性表示和定量表示之分

4、 应具有载体和多种表现形式

三、 数据处理：数据管理（快速正确）、数据加工（有用性）、数据传播

四、 数据管理：是其他数据处理的核心和基础：组织保存数据、数据维护、提供数据查询和

数据统计功能

五、 数据库（DB）：是一个按数据结构来存储和管理数据的计算机软件系统：数据整体性、

数据共享性

六、 数据库管理系统（DBMS）：专门用于管理数据库的计算机系统软件。能够提供数据定义、

建立、维护、查询和统计等操作功能

七、 数据库应用系统（DBAS）：凡使用数据库技术管理数据的系统都称为DBAS。数据传递

系统、数据处理系统、管理信息系统

八、 管理信息系统（MIS）：以数据库技术为基础，采用功能选单方式控制程序。

九、 数据库系统：应由计算机硬件、数据库、数据库管理系统、数据库应用系统和数据库管

理员组成。数据库系统核心是数据库管理系统

十、 数据库系统管理数据特点：

1、 数据库系统以数据模型为基础

2、 数据库系统冗余度小、数据共享度高

3、 数据与程序之间有较高独立性

（1） 物理独立性：应用程序对数据存储结构的依赖程度

（2） 逻辑独立性：全局逻辑结构：全系统总体的数据逻辑结构

局部逻辑结构：具体一个用户或程序使用的数据逻辑结构

4、 通过DBMS进行数据安全性和完整性的控制

（1） 安全性：保护数据库，以防止不合法的使用造成的数据泄露、破坏和更改

（2） 完整性：保证数据的正确性、有效性、相容性，防止不符合语义的数据输入

或输出所采用的控制机制。

5、 最小存取单位是数据项

十一、数据库系统的软件组成

数据库应用系统、数据库

应用开发工具及软件

DBMS、主语言编译系统

操作系统

计算机硬件

十二、数据库系统的三级结构模式

1、 外模式（子模式）：对各个用户或程序所涉及的数据的逻辑结构和数据特征的描述

2、 模式（逻辑模式）：对数据库中数据整体的逻辑结构和特征的描述

3、 内模式（存储模式、物理模式）：对数据内部表示或底层的描述

十三、数据库管理人员的职责

1、 应参与数据库和应用系统的设计

2、 应参与决定数据库的存储结构和存取策略

3、 负责定义数据的安全性要求和完整性条件

4、 负责监视和控制数据库系统的运行，负责系统的维护和数据恢复

5、 负责数据库的改进和重组

十四、信息的三种世界描述：现实世界、信息世界、计算机世界

现实世界

实体

特征

实体集

实体间的联系

信息世界

实例

属性

对象

对象间的联系

概念模型

计算机世界

记录

数据项

数据或文件

数据间的联系

数据模型

1、 实体：现实世界中存在的可以相互区分的事物或概念

2、 实体型：用实体名及属性名集合来抽象和刻画的同类实体

3、 实体集：具有相同特征或能用相同特征描述的实体的集合

4、 属性：实体的某一方面特征的抽象表示

5、 码（关键字）：能够唯一标识一个实体

十五、概念模型：概念模型一般要求表示简单、易于理解且具有较强的语义表达能力，它应

该独立于具体的逻辑模型，并易于向数据库管理系统支持的逻辑数据模型转换。

1、 对象和实例

2、 属性

3、 码、主码、次码（不能唯一标识实体属性的码）、候选码（实体集中存在多个码时，

通常选定一个为主码，其余的为候选码）

4、 域

十六、概念模型的表示方法：E-R图法（实体-联系方法），用以描述现实世界的概念模型，

提供了表示实体集、属性和联系的方法

1、 长方形表示实体集，长方形内写实体集名

2、 用椭圆形表示实体集属性，并与实体集连线

3、 用菱形表示实体集间的联系，写上联系名，并与有关实体集连线，且标出联系的类

十七、数据模型是一组严格定义的概念集合。

数据模型三要素：数据结构，数据操作，数据约束条件。

常见数据模型：层次模型，网状模型，关系模型。

十八、数据库系统设计的意义

1、 完整的反映现实世界中信息及信息之间的联系

2、 有效的进行数据存储

3、 方便的执行各种数据检索和处理操作

4、 有利于进行数据维护和数据控制管理的工作

十九、数据库系统设计的内容

1、 结构特性设计：数据库的逻辑结构特征，又称静态结构设计

2、 行为特性设计：确定数据库用户的行为和动作，设计DMAS的层次结构、功能结构、

数据流程图，确定数据库的子模式

3、 物理模式设计：物理实现逻辑结构模型，得出其存储模式和存取方法

二十、数据库系统设计的基本步骤

1、 需求分析阶段：准确了解并分析用户对系统的需求和要求，弄清系统要达到的目标

和实现的功能

2、 概念结构设计阶段：是设计的关键，需形成一个独立于具体计算机和DBMS的概念

模型

3、 逻辑结构设计阶段：将概念结构转换为某个DBMS所支持的数据模型，并性能优化

4、 物理设计阶段：为逻辑数据模型选取一个最适合应用环境的物理结构

5、 数据库实施阶段：根据数据库的逻辑设计和物理设计的结构建立数据库、调制应用

程序、组织数据入库并进行系统试运行

6、 数据库运行和维护阶段：在试运行中，必须不断的对其结构性能进行评价、调整和

修改

二一、数据字典：各类数据描述的集合，进行详细的数据收集和数据分析后所获得的主要成

果。

1、 数据项：不可再分的数据单位

2、 数据结构：数据间的组合关系

3、 数据流：数据结构在系统内传输的路径

4、 数据存储：数据及其结构停留或保存的地方

5、 数据处理：描述处理过程的说明性信息

二二、概念结构即概念模型。

1、 特点：

（1） 是现实世界的一个真实模型

（2） 应易于理解

（3） 应易于更改

（4） 应易于向数据模型转换

2、 设计方法：

（1） 自顶向下

（2） 自底向上

（3） 逐步扩张

（4） 混合策略

3、 设计步骤

数据流程图、数据字典 → 分E-R图 → 总E-R图

二三、三种数据抽象方法

1、 分类：定义某一类概念作为现实世界中一组对象的类型，这些对象具有某些共同的

特性和行为。“职工”

2、 聚集：定义某一类型的组成部分，抽象了“组成部分”语义。“职工号”

3、 概括：定义了类型之间的一种子集联系，抽象了类型之间的“所属”语义。

二四、数据库逻辑结构的设计（概念模型 → 逻辑模型）

概念模型 → 一般数据模型 → 特定DBMS → 优化的数据模型

二五、概念模型 → 网状模型

1、 不同型实体集及其联系

（1） 每个实体集转换成一个记录

（2） 每个1：n的二元联系转换成一个系，由1指向n

（3） 每个m：n的二元联系，转换时需要一个连接记录，形成两个系，由实体记

录指向连接记录

2、 同型实体之间联系

（1）1：n---引入连接记录，转换为两个系，其方向不同

（2）m：n---引入一个连接记录，转换为两个系，方向均指向连接记录方

二六、概念模型 → 关系模型

1、 实体集的转换规则：一个实体集即为一个关系，实体属性即为关系属性，实体码即

为关系码，关系结构即为关系模式

2、 实体集间联系的转换规则：

（1）1：1

转换为一个独立的关系，也可以与任意一端实体集所对应得关系合并

（2）1：n

两种转换方法：将联系转换为一个独立关系；联系形成的关系与n端对象合并

（3）m：n

一个m：n联系转换为一个关系

二七、用户子模式的设计

1、 使用更符合用户习惯的别名

2、 对不同级别的用户可以定义不同的子模式

3、 简化用户对系统的使用

二八、数据库中基本关系的性质（6个）

1、 同一属性的数据具有相同质性：同一属性的数据应当是同质的数据

2、 同一关系的属性名具有不能重复性：同一关系中不同属性的数据可出自同一个域，

但不同属性要给予不同的属性名

3、 关系中的列位置具有顺序无关性：说明关系中的列的次序可以任意交换、重新组织，

属性的顺序不影响使用

4、 关系具有元组无冗余性：关系中任意两个元组不能完全相同

5、 关系中的元组位置具有顺序无关性：元组顺序可以任意互换

6、 关系中每一个分量都必须是不可分的数据项：关系模型要求关系必须是规范的

二九、关系操作基本内容：数据查询、数据维护、数据控制

三十、关系操作特点

1、 关系操作、语言操作一体化

2、 关系操作的方式是一次一集合方式

3、 关系操作语言是高度非过程化的语言

三一、关系代数

1、 集合运算：∩、∪、◊（广义笛卡儿积）、-

2、 专门的关系运算符：σ（选择）、π（投影）、÷、连接

3、 比较运算符：>、≥、<、≤、=、≠

4、 逻辑运算符：∧（与）、∨（或）、¬（非）

三二、域关系演算语言QBE：是一种域关系演算的关系语言，同时也指使用此语言的关系数

据库管理系统

1、 交互式语言

2、 表格语言

3、 基于例子的查询语言

三三、T-SQL创建和维护数据库功能

1、 创建数据库：

（1） 定义数据库名：CREATE DATABASE <数据库名>

（2） 定义数据文件：ON PRIMARY( NAME = <逻辑文件名>,

FILENAME = ‘<操作数据文件路径和文件名>’,

SIZE = <文件长度>,

MAXSIZE = <最大文件长度>,

FILEROWHT = <文件增长率>)

GO

（3） 定义日志文件：LOG ON语句

（4） FOR RESTORE子句说明能重建一个数据库，用于数据恢复操作

2、 选择数据库：USE <数据库名> + GO

3、 删除数据库：DROP DATABASE <数据库名组> + GO

三四、T-SQL定义和维护基本功能表

1、 基本表定义：CREATE TABLE

2、 字段约束

3、 记录约束：CONSTRAINT <约束名> <约束式>

A、[PRIMARY KEY [CLUSTERED | NOCLUSTERED] ( <列组名> ) ]

B、[FOREIGN KEY ( <外码> ) REFERENCES <参照表> ( <对应列> ) ]

C、[CHECK ( <条件表达式> ) ]

D、[UNIQUE ( <列组> ) ]

4、数据类型

5、基本表的维护

（1）修改字段定义：

ALTER TABLE <表名>

ALTER COLUMN <列名> <新类型> [NULL | NOT NULL] <约束定义>

（2）增加字段和表约束规则：

ALTER TABLE <表名>

ADD { <列定义> | [ <表约束条件>] }

（3）删除字段或约束条件

ALTER TABLE <表名>

DROP { [CONSTRAINT] <约束名> | COLUMN <列名> }

（4）使约束有效或无效

ALTER TABLE <表名>

{CHECK | NOCHECK} CONSTRAINT {ALL | <约束组名> }

三五、T-SQL数据查询语言

1、 SELECT [ALL | DISTINCT] [TOP <数值> [PERCENT] ] <查询列>：

（1） ALL|DISTINCT选项：前者返回结果集中所有行；后者仅显示结果集中唯一

一行。默认前者

（2） TOP<数值>选项：仅返回结果集中的前<数值>行。若有PERCENT，则返回

结果集中的PERCENT%行记录

（3） \*：指明返回在FROM子句中包括的表和视图的全部列

2、 WHERE <查询条件> | <旧格式外连接条件>

<旧格式外连接条件>：：= <列名> { \* = | = \* } <列名>

3、 HAVING <组或汇总筛选条件>：位于GROUP BY子句后

4、 ORDER BY <排序项> [ ASC | DESC ] [ ,…n ]：ASC，元组升序排列；DESC，元组降序

排列

5、 COMPUTE <统计函数组> [ BY<分组项>]

三六、视图是根据数据库子模式建立的虚表可以方便用户查询和保证系统数据安全

三七、视图的查询和维护：不支持对以下几种情况的视图进行更新视图操作

1、 由两个以上基本表导出的视图

2、 视图的列来自列表达式函数

3、 视图中有分组子句或使用了DISTINCT语句

4、 视图定义中有嵌套查询，且内层查询中涉及了与外层一样的导出该视图的基本表

5、 在一个不允许更新的视图上定义的视图