

第1章 绪论

1. 试述数据、数据库、数据库系统、数据库管理系统的概念。

答：

(1) 数据 (Data)：描述事物的符号记录称为数据。数据的种类有数字、文字、图形、图像、声音、正文等。数据与其语义是不可分的。解析在现代计算机系统中数据的概念是广义的。早期的计算机系统主要用于科学计算，处理的数据是整数、实数、浮点数等传统数学中的数据。现代计算机能存储和处理的对象十分广泛，表示这些对象的数据也越来越复杂。数据与其语义是不可分的。500 这个数字可以表示一件物品的价格是 500 元，也可以表示一个学术会议参加的人数有 500 人，还可以表示一袋奶粉重 500 克。

(2) 数据库 (DataBase，简称 DB)：数据库是长期储存在计算机内的、有组织的、可共享的数据集合。数据库中的数据按一定的数据模型组织、描述和储存，具有较小的冗余度、较高的数据独立性和易扩展性，并可为各种用户共享。

(3) 数据库系统 (DataBase System，简称 DBS)：数据库系统是指在计算机系统中引入数据库后的系统构成，一般由数据库、数据库管理系统 (及其开发工具)、应用系统、数据库管理员构成。解析数据库系统和数据库是两个概念。数据库系统是一个人一机系统，数据库是数据库系统的一个组成部分。但是在日常工作中人们常常把数据库系统简称为数据库。希望读者能够从人们讲话或文章的上下文中区分“数据库系统”和“数据库”，不要引起混淆。

(4) 数据库管理系统 (DataBase Management system，简称 DBMS)：数据库管理系统是位于用户与操作系统之间的一层数据管理软件，用于科学地组织和存储数据、高效地获取和维护数据。DBMS 的主要功能包括数据定义功能、数据操纵功能、数据库的运行管理功能、数据库的建立和维护功能。解析 DBMS 是一个大型的复杂的软件系统，是计算机中的基础软件。目前，专门研制 DBMS 的厂商及其研制的 DBMS 产品很多。著名的有美国 IBM 公司的 DB2 关系数据库管理系统和 IMS 层次数据库管理系统、美国 Oracle 公司的 Oracle 关系数据库管理系统、Sybase 公司的 Sybase 关系数据库管理系统、美国微软公司的 SQL Server 关系数据库管理系统等。

2. 使用数据库系统有什么好处？

答：

使用数据库系统的好处是由数据库管理系统的特点或优点决定的。使用数据库系统的好处很多，例如，可以大大提高应用开发的效率，方便用户的使用，减轻数据库系统管理人员维护的负担，等等。使用数据库系统可以大大提高应用开发的效率。因为在数据库系统中应用程序不必考虑数据的定义、存储和数据存取的具体路径，这些工作都由 DBMS 来完成。用一个通俗的比喻，使用了 DBMS 就如有了一个好参谋、好助手，许多具体的技术工作都由这个助手来完成。开发人员就可以专注于应用逻辑的设计，而不必为数据管理的许许多多复杂的细节操心。还有，当应用逻辑改变，数据的逻辑结构也需要改变时，由于数据库系统提供了数据与程序之间的独立性，数据逻辑结构的改变是 DBA 的责任，开发人员不必修改应用程序，或者只需要修改很少的应用程序，从而既简化了应用程序的编制，又大大减少了应用程序的维护和修改。使用数据库系统可以减轻数据库系统管理人员维护系统的负担。因为 DBMS 在数据库建立、运用和维护时对数据库进行统一的管理和控制，包括数据的完整性、安全性、多用户并发控制、故障恢复等，都由 DBMS 执行。总之，使用数据库系统的优点是很多的，既便于数据的集中管理，控制数据冗余，提高数据的利用率和一致性，又有利于应用程序的开发和维护。读者可以在自己今后的工作中结合具体应用，认真加以体会和总结。

3. 试述文件系统与数据库系统的区别和联系。

答：

文件系统与数据库系统的区别是：文件系统面向某一应用程序，共享性差，冗余度大，数据独立性差，记录内有结构，整体无结构，由应用程序自己控制。数据库系统面向现实世界，共享性高，冗余度小，具有

较高的物理独立性和一定的逻辑独立性，整体结构化，用数据模型描述，由数据库管理系统提供数据的安全性、完整性、并发控制和恢复能力。

文件系统与数据库系统的联系是：文件系统与数据库系统都是计算机系统中管理数据的软件。解析文件系统是操作系统的重要组成部分；而 DBMS 是独立于操作系统的软件。但是 DBMS 是在操作系统的基础上实现的；数据库中数据的组织和存储是通过操作系统中的文件系统来实现的。

4. 举出适合用文件系统而不是数据库系统的例子；再举出适合用数据库系统的应用例子。答：

(1) 适用于文件系统而不是数据库系统的应用例子数据的备份、软件或应用程序使用过程中的临时数据存储一般使用文件比较合适。早期功能比较简单、比较固定的应用系统也适合用文件系统。

(2) 适用于数据库系统而非文件系统的应用例子目前，几乎所有企业或部门的信息系统都以数据库系统为基础，都使用数据库。例如，一个工厂的管理信息系统（其中会包括许多子系统，如库存管理系统、物资采购系统、作业调度系统、设备管理系统、人事管理系统等），学校的学生管理系统，人事管理系统，图书馆的图书管理系统，等等，都适合用数据库系统。希望读者能举出自己了解的应用例子。

5. 试述数据库系统的特点。

答：

数据库系统的主要特点有：

(1) 数据结构化数据库系统实现整体数据的结构化，这是数据库的主要特征之一，也是数据库系统与文件系统的本质区别。解析注意这里的“整体”两个字。在数据库系统中，数据不再针对某一个应用，而是面向全组织，具有整体的结构化。不仅数据是结构化的，而且数据的存取单位即一次可以存取数据的大小也很灵活，可以小到某一个数据项（如一个学生的姓名），大到一组记录（成千上万个学生记录）。而在文件系统中，数据的存取单位只有一个：记录，如一个学生的完整记录。

(2) 数据的共享性高，冗余度低，易扩充数据库的数据不再面向某个应用而是面向整个系统，因此可以被多个用户、多个应用以多种不同的语言共享使用。由于数据面向整个系统，是有结构的数据，不仅可以被多个应用共享使用，而且容易增加新的应用，这就使得数据库系统弹性大，易于扩充。解析数据共享可以大大减少数据冗余，节约存储空间，同时还能够避免数据之间的不相容性与不一致性。所谓“数据面向某个应用”是指数据结构是针对某个应用设计的，只被这个应用程序或应用系统使用，可以说数据是某个应用的“私有资源”。所谓“弹性大”是指系统容易扩充也容易收缩，即应用增加或减少时不必修改整个数据库的结构，只需做很少的改动。可以取整体数据的各种子集用于不同的应用系统，当应用需求改变或增加时，只要重新选取不同的子集或加上一部分数据，便可以满足新的需求。

(3) 数据独立性高数据独立性包括数据的物理独立性和数据的逻辑独立性。数据库管理系统的模式结构和二级映像功能保证了数据库中的数据具有很高的物理独立性和逻辑独立性。

(4) 数据由 DBMS 统一管理和控制数据库的共享是并发的共享，即多个用户可以同时存取数据库中的数据甚至可以同时存取数据库中同一个数据。为此，DBMS 必须提供统一的数据控制功能，包括数据的安全性保护、数据的完整性检查、并发控制和数据库恢复。解析 DBMS 数据控制功能包括四个方面：数据的安全性保护：保护数据以防止不合法的使用造成的数据的泄密和破坏；数据的完整性检查：将数据控制在有效的范围内，或保证数据之间满足一定的关系；并发控制：对多用户的并发操作加以控制和协调，保证并发操作的正确性；数据库恢复：当计算机系统发生硬件故障、软件故障，或者由于操作员的失误以及故意的破坏影响数据库中数据的正确性，甚至造成数据库部分或全部数据的丢失时，能将数据库从错误状态恢复到某一已知的正确状态（亦称为完整状态或一致状态）。下面可以得到“什么是数据库”的一个定义：数据库是长期存储在计算机内有组织的大量的共享的数据集合，它可以供各种用户共享，具有最小冗余度和较高的数据独立性。DBMS 在数据库建立、运用和维护时对数据库进行统一控制，以保证数据的完整性、安全性，并在多用户同时使用数据库时进行并发控制，在发生故障后对系统进行恢复。数据库系统的出现使信息系统从以加工数据的程序为中心转向围绕共享的数据库为中心的新阶段。

6. 数据库管理系统的主要功能有哪些？

答：

- (1) 数据库定义功能；
- (2) 数据存取功能；
- (3) 数据库运行管理；
- (4) 数据库的建立和维护功能。

7. 什么是概念模型？试述概念模型的作用。

答：概念模型，也称信息模型，是按用户的观点来对数据和信息建模，主要用于数据库设计。

概念模型实际上是现实世界到机器世界的一个中间层次。概念模型用于信息世界的建模，是现实世界到信息世界的第一层抽象，是数据库设计人员进行数据库设计的有力工具，也是数据库设计人员和用户之间进行交流的语言。

8. 定义并解释概念模型中以下术语：实体，实体型，实体集，实体之间的联系。

答：

实体：客观存在并可相互区分的事物叫实体。实体型：具有相同属性的实体具有相同的特征和性质，用实体名及其属性名集合来抽象和刻画同类实体，称为实体型。实体集：同型实体的集合称为实体集；实体之间的联系：通常是指不同实体型的实体集之间的联系，实体之间的联系有一对一，一对多和多对多等多种类型。

9. 试述数据模型的概念、数据模型的作用和数据模型的三个要素。

答：

数据模型是数据库中用来对现实世界进行抽象的工具，是数据库中用于提供信息表示和操作手段的形式构架。一般地讲，数据模型是严格定义的概念的集合。这些概念精确描述了系统的静态特性、动态特性和完整性约束条件。因此数据模型通常由数据结构、数据操作和完整性约束三部分组成。

(1) 数据结构：是所研究的对象类型的集合，是对系统静态特性的描述。

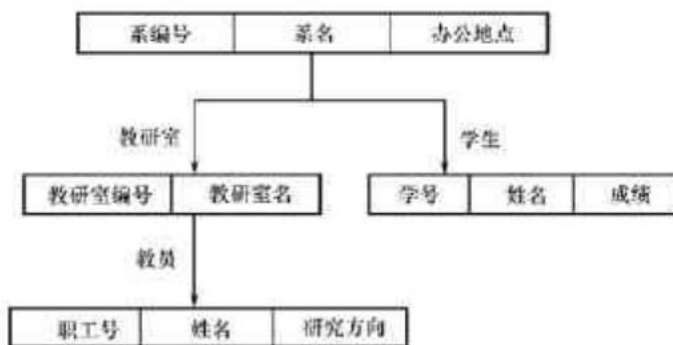
(2) 数据操作：是指对数据库中各种对象（型）的实例（值）允许进行的操作的集合，包括操作及有关的操作规则，是对系统动态特性的描述。

(3) 数据的约束条件：是一组完整性规则的集合。完整性规则是给定的数据模型中数据及其联系所具有的制约和依存规则，用以限定符合数据模型的数据库状态以及状态的变化，以保证数据的正确、有效、相容。解析数据模型是数据库系统中最重要的概念之一。必须通过《概论》的学习真正掌握数据模型的概念和作用。数据模型是数据库系统的基础。任何一个 DBMS 都以某一个数据模型为基础，或者说支持某一个数据模型。数据库系统中，模型有不同的层次。根据模型应用的不同目的，可以将模型分成两类或者说两个层次：一类是概念模型，是按用户的观点来对数据和信息建模，用于信息世界的建模，强调语义表达能力，概念简单清晰；另一类是数据模型，是按计算机系统的观点对数据建模，用于机器世界，人们可以用它定义、操纵数据库中的数据，一般需要有严格的形式化定义和一组严格定义了语法和语义的语言，并有一些规定和限制，便于在机器上实现。

10. 试述层次模型的概念，举出三个层次模型的实例。

答：

- (1) 教员学生层次数据库模型



(2) 行政机构层次数据库模型



(3) 行政区域层次数据库模型



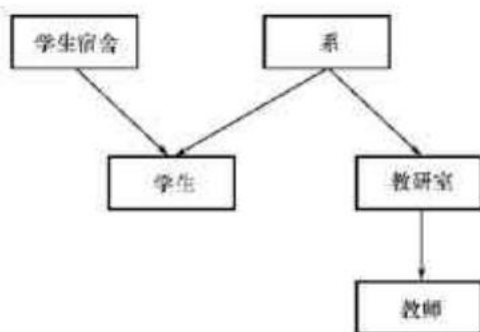
11. 试述网状模型的概念，举出三个网状模型的实例。

答：

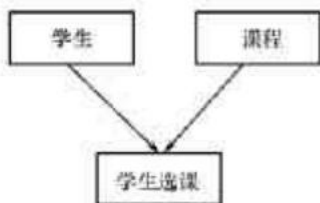
满足下面两个条件的基本层次联系集合为网状模型。

(1) 允许一个以上的结点无双亲； (2) 一个结点可以有多于一个的双亲。

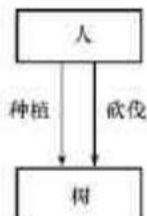
实例 1 ：



实例 2 ：



实例 3：



12. 试述网状、层次数据库的优缺点。

答：

层次模型的优点主要有：（1）模型简单，对具有一对多层次关系的部门描述非常自然、直观，容易理解，这是层次数据库的突出优点；（2）用层次模型的应用系统性能好，特别是对于那些实体间联系是固定的且预先定义好的应用，采用层次模型来实现，其性能优于关系模型；（3）层次数据模型提供了良好的完整性支持。

层次模型的缺点主要有：（1）现实世界中很多联系是非层次性的，如多对多联系、一个结点具有多个双亲等，层次模型不能自然地表示这类联系，只能通过引入冗余数据或引入虚拟结点来解决；（2）对插入和删除操作的限制比较多；（3）查询子女结点必须通过双亲结点。

网状数据模型的优点主要有：（1）能够更为直接地描述现实世界，如一个结点可以有多个双亲；（2）具有良好的性能，存取效率较高。

网状数据模型的缺点主要有：（1）结构比较复杂，而且随着应用环境的扩大，数据库的结构就变得越来越复杂，不利于最终用户掌握；（2）其 DDL、DML 语言复杂，用户不容易使用。由于记录之间联系是通过存取路径实现的，应用程序在访问数据时必须选择适当的存取路径。因此，用户必须了解系统结构的细节，加重了编写应用程序的负担。

13. 试述关系模型的概念，定义并解释以下术语：（1）关系（2）属性（3）域（4）元组（5）主码（6）分量（7）关系模式

答：

关系模型由关系数据结构、关系操作集合和关系完整性约束三部分组成。在用户观点下，关系模型中数据的逻辑结构是一张二维表，它由行和列组成。（1）关系：一个关系对应通常说的一张表；（2）属性：表中的一列即为一个属性；（3）域：属性的取值范围；（4）元组：表中的一行即为一个元组；（5）主码：表中的某个属性组，它可以惟一确定一个元组；（6）分量：元组中的一个属性值；（7）关系模式：对关系的描述，一般表示为关系名（属性 1，属性 2，…，属性 n）

14. 试述关系数据库的特点。

答：

关系数据模型具有下列优点：（1）关系模型与非关系模型不同，它是建立在严格的数学概念的基础上的。（2）关系模型的概念单一，无论实体还是实体之间的联系都用关系表示，操作的对象和操作的结果都是关系，所以其数据结构简单、清晰，用户易懂易用。（3）关系模型的存取路径对用户透明，从而具有更高的数据独立性、更好的安全保密性，也简化了程序员的工作和数据库开发建立的工作。当然，关系数据模型也有缺点，其中最主要的缺点是，由于存取路径对用户透明，查询效率往往不如非关系数据模型。因此为了提高性能，必须对用户的查询请求进行优化，增加了开发数据库管理系统的难度。

15. 试述数据库系统三级模式结构, 这种结构的优点是什么?

答:

数据库系统的三级模式结构由外模式、模式和内模式组成。(参见书上图 1.29) 外模式, 亦称子模式或用户模式, 是数据库用户(包括应用程序员和最终用户)能够看见和使用的局部数据的逻辑结构和特征的描述, 是数据库用户的数据视图, 是与某一应用有关的数据的逻辑表示。模式, 亦称逻辑模式, 是数据库中全体数据的逻辑结构和特征的描述, 是所有用户的公共数据视图。模式描述的是数据的全局逻辑结构。外模式涉及的是数据的局部逻辑结构, 通常是模式的子集。内模式, 亦称存储模式, 是数据在数据库系统内部的表示, 即对数据的物理结构和存储方式的描述。数据库系统的三级模式是对数据的三个抽象级别, 它把数据的具体组织留给 DBMS 管理, 使用户能逻辑抽象地处理数据, 而不必关心数据在计算机中的表示和存储。为了能够在内部实现这三个抽象层次的联系和转换, 数据库系统在这三级模式之间提供了两层映像: 外模式/模式映像和模式/内模式映像。正是这两层映像保证了数据库系统中的数据能够具有较高的逻辑独立性和物理独立性。

16. 定义并解释以下术语: 模式、外模式、内模式、DDL、DML。

模式、外模式、内模式, 亦称逻辑模式, 是数据库中全体数据的逻辑结构和特征的描述, 是所有用户的公共数据视图。模式描述的是数据的全局逻辑结构。外模式涉及的是数据的局部逻辑结构, 通常是模式的子集。内模式, 亦称存储模式, 是数据在数据库系统内部的表示, 即对数据的物理结构和存储方式的描述。DDL: 数据定义语言, 用来定义数据库模式、外模式、内模式的语言。DML: 数据操纵语言, 用来对数据库中的数据进行查询、插入、删除和修改的语句。

17. 什么叫数据与程序的物理独立性? 什么叫数据与程序的逻辑独立性? 为什么数据库系统具有数据与程序的独立性?

答:

数据与程序的逻辑独立性: 当模式改变时(例如增加新的关系、新的属性、改变属性的数据类型等), 由数据库管理员对各个外模式/模式的映像做相应改变, 可以使外模式保持不变。应用程序是依据数据的外模式编写的, 从而应用程序不必修改, 保证了数据与程序的逻辑独立性, 简称数据的逻辑独立性。数据与程序的物理独立性: 当数据库的存储结构改变了, 由数据库管理员对模式/内模式映像做相应改变, 可以使模式保持不变, 从而应用程序也不必改变, 保证了数据与程序的物理独立性, 简称数据的物理独立性。数据库管理系统在三级模式之间提供的两层映像保证了数据库系统中的数据能够具有较高的逻辑独立性和物理独立性。

18. 试述数据库系统的组成。

答:

数据库系统一般由数据库、数据库管理系统(及其开发工具)、应用系统、数据库管理员和用户构成。

19. 数据库管理员、系统分析员、数据库设计人员、应用程序员的职责是什么?

答:

数据库管理员的职责: (1) 决定数据库中的信息内容和结构;

(2) 决定数据库的存储结构和存取策略;

(3) 定义数据的安全性要求和完整性约束条件;

(4) 监控数据库的使用和运行;

(5) 数据库的改进和重组、重构。

系统分析员负责应用系统的需求分析和规范说明, 系统分析员要和用户及 DBA 相结合, 确定系统的硬件、软件配置, 并参与数据库系统的概要设计。数据库设计人员负责数据库中数据的确定、数据库各级模式的设计。数据库设计人员必须参加用户需求调查和系统分析, 然后进行数据库设计。在很多情况下, 数据库设计人员就由数据库管理员担任。应用程序员负责设计和编写应用系统的程序模块, 并进行调试和安装。

1. 试述关系模型的三个组成部分。

答：关系模型由关系数据结构、关系操作集合和关系完整性约束三部分组成。

2. 试述关系数据语言的特点和分类。

答：关系数据语言可以分为三类：

关系代数语言。

关系演算语言：元组关系演算语言和域关系演算语言。

SQL：具有关系代数和关系演算双重特点的语言。

这些关系数据语言的共同特点是，语言具有完备的表达能力，是非过程化的集合操作语言，功能强，能够嵌入高级语言中使用。

3. (略)

4.

5. 述关系模型的完整性规则。在参照完整性中，为什么外部码属性的值也可以为空？什么情况下才可以为空？

答：实体完整性规则是指若属性 A 是基本关系 R 的主属性，则属性 A 不能取空值。

若属性(或属性组)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 不一定是不同的关系)，则对于 R 中每个元组在 F 上的值必须为：或者取空值(F 的每个属性值均为空值)；或者等于 S 中某个元组的主码值。即属性 F 本身不是主属性，则可以取空值，否则不能取空值。

6. 设有一个 SPJ 数据库，包括 S, P, J, SPJ 四个关系模式：

1) 求供应工程 J1 零件的供应商号码 SNO：

$\pi_{SNO}(\sigma_{JNO='J1'}(SPJ))$

2) 求供应工程 J1 零件 P1 的供应商号码 SNO：

$\pi_{SNO}(\sigma_{JNO='J1' \wedge PNO='P1'}(SPJ))$

3) 求供应工程 J1 零件为红色的供应商号码 SNO：

$\pi_{SNO}(\pi_{SNO, PNO}(\sigma_{JNO='J1'}(SPJ)) \cap \pi_{PNO}(\sigma_{COLOR='红'}(P)))$

4) 求没有使用天津供应商生产的红色零件的工程号 JNO：

$\pi_{JNO}(SPJ) - \pi_{JNO}(\sigma_{CITY='天津' \wedge COLOR='红'}(S \cap SPJ \cap P))$

5) 求至少用了供应商 S1 所供应的全部零件的工程号 JNO：

$\pi_{JNO, PNO}(SPJ) \div \pi_{PNO}(\sigma_{SNO='S1'}(SPJ))$

7. 试述等值连接与自然连接的区别和联系。

答：连接运算符是“=”的连接运算称为等值连接。它是从关系 R 与 S 的广义笛卡尔积中选取 A, B 属性值相等的那些元组。

自然连接是一种特殊的等值连接，它要求两个关系中进行比较的分量必须是相同的属性组，并且在结果中把重复的属性列去掉。

8. 关系代数的基本运算有哪些？如何用这些基本运算来表示其他运算？

答：并、差、笛卡尔积、投影和选择 5 种运算为基本的运算。其他 3 种运算，即交、连接和除，均可以用这 5 种基本运算来表达。

第三章 关系数据库语言 SQL

1. 试述 SQL 语言的特点。

答：

(1) 综合统一。SQL 语言集数据定义语言 DDL、数据操纵语言 DML、数据控制语言 DCL 的功能于一体。

(2) 高度非过程化。用 SQL 语言进行数据操作，只要提出“做什么”，而无需指明“怎么做”，因此无需了解存取路径，存取路径的选择以及 SQL 语句的操作过程由系统自动完成。

(3) 面向集合的操作方式。SQL 语言采用集合操作方式，不仅操作对象、查找结果可以是元组的集合，而且一次插入、删除、更新操作的对象也可以是元组的集合。

(4) 以同一种语法结构提供两种使用方式。SQL 语言既是自含式语言，又是嵌入式语言。作为自含式语言，它能够独立地用于联机交互的使用方式；作为嵌入式语言，它能够嵌入到高级语言程序中，供程序员设计程序时使用。

(5) 语言简捷，易学易用。

2.

3 (1) select * from S where A='10';

(2) select A,B from S;

(3) select A,B,S.C,S.D,E,F from S,T where S.C=T.C and S.D=T.D;

(4) select * from S,T where S.C=T.C;

(5) select * from S,T where S.A<T.E;

(6) select S.C,S.D,T.* from S,T ;

4. 用 SQL 语句建立第二章习题 6 中的 4 个表。

答：

对于 S 表： S (SNO , SNAME , STATUS , CITY) ;

建 S 表：

```
CREATE TABLE S ( Sno C(2) UNIQUE, Sname C(6) , Status C(2), City C(4));
```

对于 P 表： P (PNO , PNAME , COLOR , WEIGHT) ;

建 P 表：

```
CREATE TABLE P(Pno C(2) UNIQUE, Pname C(6), COLOR C(2), WEIGHT INT);
```

对于 J 表： J (JNO , JNAME , CITY) ;

建 J 表：

```
CREATE TABLE J(Jno C(2) UNIQUE, JNAME C(8), CITY C(4))
```

对于 SPJ 表： SPJ (SNO , PNO , JNO , QTY) ;

建 SPJ 表： SPJ(SNO,PNO,JNO,QTY)

```
CREATE TABLE SPJ(Sno C(2), Pno C(2), JNO C(2), QTY INT)
```

针对建立的 4 个表用 SQL 语言完成第二章习题 6 中的查询。

(1) 求供应工程 J1 零件的供应商号码 SNO ;

```
SELECT DIST SNO FROM SPJ WHERE JNO=' J1'
```

(2) 求供应工程 J1 零件 P1 的供应商号码 SNO ;

```
SELECT DIST SNO FROM SPJ WHERE JNO=' J1' AND PNO=' P1'
```

(3) 求供应工程 J1 零件为红色的供应商号码 SNO ;

```
SELECT SNO FROM SPJ,P WHERE JNO=' J1' AND SPJ.PNO=P.PNO AND COLOR=' 红'
```

(4) 求没有使用天津供应商生产的红色零件的工程号 JNO ;

```
SELECT DIST JNO FROM SPJ WHERE JNO NOT IN (SELE JNO FROM SPJ,P,S WHERE S.CITY=' 天津' AND  
COLOR=' 红' AND S.SNO=SPJ.SNO AND P.PNO=SPJ.PNO)。
```

(5) 求至少用了供应商 S1 所供应的全部零件的工程号 JNO ;

由于 VFP 不允许子查询嵌套太深，将查询分为两步

A、查询 S1 供应商供应的零件号

```
SELECT DIST PNO FROM SPJ WHERE SNO=' S1' 结果是 (P1, P2)
```

B、查询哪一个工程既使用 P1 零件又使用 P2 零件。

```
SELECT JNO FROM SPJ WHERE PNO=' P1'
```

```
AND JNO IN (SELECT JNO FROM SPJ WHERE PNO=' P2')
```


5. 针对上题中的四个表试用 SQL 语言完成以下各项操作:

(1) 找出所有供应商的姓名和所在城市。

```
SELECT SNAME, CITY FROM S
```

(2) 找出所有零件的名称、颜色、重量。

```
SELECT PNAME, COLOR, WEIGHT FROM P
```

(3) 找出使用供应商 S1 所供应零件的工程号码。

```
SELECT DIST JNO FROM SPJ WHERE SNO='S1'
```

(4) 找出工程项目 J2 使用的各种零件的名称及其数量。

```
SELECT PNAME, QTY FROM SPJ, P
```

```
WHERE P.PNO=SPJ.PNO AND SPJ.JNO='J2'
```

(5) 找出上海厂商供应的所有零件号码。

```
SELECT PNO FROM SPJ, S WHERE S.SNO=SPJ.SNO AND CITY='上海'
```

(6) 出使用上海产的零件的工程名称。

```
SELECT JNAME FROM SPJ, S, J
```

```
WHERE S.SNO=SPJ.SNO AND S.CITY='上海' AND J.JNO=SPJ.JNO
```

(7) 找出没有使用天津产的零件的工程号码。

注意: SELECT DISP JNO FROM SPJ WHERE JNO NOT IN (SELECT DIST JNO FROM SPJ, S WHERE S.SNO=SPJ.SNO AND S.CITY='天津') 适用于 JNO 是唯一或不唯一的情况。

注意: SELECT DIST JNO FROM SPJ, S WHERE S.SNO=SPJ.SNO AND S.CITY<>'天津' 适用于 JNO 是唯一的情况

(8) 把全部红色零件的颜色改成蓝色。

```
UPDATE P SET COLOR='蓝' WHERE COLOR='红'
```

(9) 由 S5 供给 J4 的零件 P6 改为由 S3 供应。

```
UPDATE SPJ SET SNO='S3' WHERE SNO='S5' AND JNO='J4' AND PNO='P6'
```

(10) 从供应商关系中删除供应商号是 S2 的记录, 并从供应情况关系中删除相应的记录。

```
A、DELETE FROM S WHERE SNO='S2'
```

```
B、DELETE FROM SPJ WHERE SNO='S2'
```

(11) 请将 (S2, J6, P4, 200) 插入供应情况关系。

```
INSERT INTO SPJ VALUES ('S2', 'J6', 'P4', 200)
```

6. 什么是基本表? 什么是视图? 两者的区别和联系是什么?

答: 基本表是本身独立存在的表, 在 SQL 中一个关系就对应一个表。视图是从一个或几个基本表导出的表。视图本身不独立存储在数据库中, 是一个虚表。即数据库中只存放视图的定义而不存放视图对应的数据, 这些数据仍存放在导出视图的基本表中。视图在概念上与基本表等同, 用户可以如同基本表那样使用视图, 可以在视图上再定义视图。

7. 试述视图的优点。

答

(1) 视图能够简化用户的操作; (2) 视图使用户能以多种角度看待同一数据; (3) 视图对重构数据库提供了一定程度的逻辑独立性; (4) 视图能够对机密数据提供安全保护。

8. 哪类视图是可以更新的? 哪类视图是不可更新的? 各举一例说明。

答: 基本表的行列子集视图一般是可更新的。若视图的属性来自集合函数、表达式, 则该视图肯定是不可

以更新的。

所有的视图是否都可以更新？为什么？

答：不是。视图是不实际存储数据的虚表，因此对视图的更新，最终要转换为对基本表的更新。因为有些视图的更新不能惟一有意义地转换成对相应基本表的更新，所以，并不是所有的视图都是可更新的。

9．请为三建工程项目建立一个供应情况的视图，包括供应商代码(SNO)、零件代码(PNO)、供应数量(QTY)。

```
CREATE VIEW VSP AS SELECT SNO, SPJ.PNO, QTY FROM SPJ, J
WHERE SPJ.JNO=J.JNO AND J.JNAME='三建'
```

针对该视图 VSP 完成下列查询：

(1) 找出三建工程项目使用的各种零件代码及其数量。

```
SELECT DISTINCT PNO, QTY FROM VSP
```

(2) 找出供应商 S1 的供应情况。

```
SELECT DISTINCT * FROM VSP WHERE SNO='S1'
```

第 4 章 数据库安全性

1．什么是数据库的安全性？

答：数据库的安全性是指保护数据库以防止不合法的使用所造成的数据泄露、更改或破坏。

2．数据库安全性和计算机系统的安全性有什么关系？

答：安全性问题不是数据库系统所独有的，所有计算机系统都有这个问题。只是在数据库系统中大量数据集中存放，而且为许多最终用户直接共享，从而使安全性问题更为突出。

系统安全保护措施是否有效是数据库系统的主要指标之一。

数据库的安全性和计算机系统的安全性，包括操作系统、网络系统的安全性是紧密联系、相互支持的，

4．试述实现数据库安全性控制的常用方法和技术。

答：实现数据库安全性控制的常用方法和技术有：

(1) 用户标识和鉴别：该方法由系统提供一定的方式让用户标识自己的名字或身份。每次用户要求进入系统时，由系统进行核对，通过鉴定后才提供系统的使用权。

(2) 存取控制：通过用户权限定义和合法权检查确保只有合法权限的用户访问数据库，所有未被授权的人员无法存取数据。例如 CZ 级中的自主存取控制(DAC)，B1 级中的强制存取控制(MAC)。

(3) 视图机制：为不同的用户定义视图，通过视图机制把要保密的数据对无权存取的用户隐藏起来，从而自动地对数据提供一定程度的安全保护。

(4) 审计：建立审计日志，把用户对数据库的所有操作自动记录下来放入审计日志中，DBA 可以利用审计跟踪的信息，重现导致数据库现有状况的一系列事件，找出非法存取数据的人、时间和内容等。

(5) 数据加密：对存储和传输的数据进行加密处理，从而使得不知道解密算法的人无法获知数据的内容。

5. 什么是数据库中的自主存取控制方法和强制存取控制方法？

答：

自主存取控制方法：定义各个用户对不同数据对象的存取权限。当用户对数据库访问时首先检查用户的存取权限。防止不合法用户对数据库的存取。

强制存取控制方法：每一个数据对象被(强制地)标以一定的密级，每一个用户也被(强制地)授予某一个级别的许可证。系统规定只有具有某一许可证级别的用户才能存取某一个密级的数据对象。

6. (1) GRANT ALL PRIVILEGES ON Student, Class

TO U1
WITH GRANT OPTION ;

(2)GRANT SELECT,UPDATE(家庭住址),DELETE ON Student TO U2;

(3)GRANT SELECT ON Class TO PUBLIC;

(4)GRANT SELECT,UPDATE ON Student TO R1;

(5)GRANT R1 TO U1 WITH ADMIN OPTION ;

7 .SQL 语言中提供了哪些数据控制（自主存取控制）的语句？请试举几例说明它们的使用方法。

答：

SQL 中的自主存取控制是通过 GRANT 语句和 REVOKE 语句来实现的。如：

GRANT SELECT , INSERT ON Student
TO 王平

WITH GRANT OPTION ;

就将 Student 表的 SELECT 和 INSERT 权限授予了用户王平，后面的“WITH GRANT OPTION ”子句表示用户王平同时也获得了“授权”的权限，即可以把得到的权限继续授予其他用户。

REVOKE INSERT ON Student FROM 王平 CASCADE ;

就将 Student 表的 INSERT 权限从用户王平处收回，选项 CASCADE 表示，如果用户王平将 Student 的 INSERT 权限又转授给了其他用户，那么这些权限也将从其他用户处收回。

7. 请用 SQL 的 GRANT 和 REVOKE 语句(加上视图机制)完成以下授权定义或存取控制功能：

(a) 用户王明对两个表有 SELECT 权力。

GRANT SELECT ON 职工, 部门
TO 王明

(b) 用户李勇对两个表有 INSERT 和 DELETE 权力。

GRANT INSERT,DELETE ON 职工, 部门
TO 李勇

(c) 每个职工只对自己的记录有 SELECT 权力。

GRANT SELECT ON 职工
WHEN USER()=NAME
TO ALL;

(d) 用户刘星对职工表有 SELECT 权力，对工资字段具有更新权力。

GRANT SELECT,UPDATE(工资) ON 职工
TO 刘星

(e) 用户张新具有修改这两个表的结构权力。

GRANT ALTER TABLE ON 职工, 部门
TO 张新;

(f) 用户周平具有对两个表所有权力（读，插，改，删数据），并具有给其他用户授权的权力。

GRANT ALL PRIVILIGES ON 职工, 部门
TO 周平
WITH GRANT OPTION;

(g) 用户杨兰具有从每个部门职工中 SELECT 最高工资、最低工资、平均工资的权力，他不能查看每个人的工资。

CREATE VIEW 部门工资 AS

```

SELECT 部门. 名称, MAX(工资), MIN(工资), AVG(工资)
FROM 职工, 部门
WHERE 职工. 部门号=部门. 部门号
GROUP BY 职工. 部门号
GRANT SELECT ON 部门工资
TO 杨兰;

```

8. 把习题 8 中 (1) --- (7) 的每一种情况, 撤销各用户所授予的权力

```

(1) REVOKE SELECT ON 职工, 部门 FROM 王明;
(2) REVOKE INSERT, DELETE ON 职工, 部门 FROM 李勇;
(3) REVOKE SELECT ON 职工
WHEN USER ( ) =NAME
FROM ALI ;
(4) REVOKE SELECT, UPDATE ON 职工
FROM 刘星;
(5) REVOKE ALTER TABLE ON 职工, 部门
FROM 张新;
(6) REVOKE ALL PRIVILEGES ON 职工, 部门
FROM 周平;
(7) REVOKE SELECT ON 部门工资
FROM 杨兰;
DROP VIEW 部门工资;

```

9. 理解并解释 MAC 机制中主体、客体、敏感度标记的含义。

答:

主体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。

客体是系统中的被动实体, 是受主体操纵的, 包括文件、基表、索引、视图等。对于主体和 客体, DBMS 为它们每个实例 (值) 指派一个敏感度标记 (Label)。

敏感度标记被分成若干级别, 例如绝密 (Top Secret)、机密 (Secret)、可信 (Confidential)、公开 (Public) 等。主体的敏感度标记称为许可证级别 (Clearance level), 客体的敏感度标记称为密级 (Classification Level)。

11. 什么是数据库的审计功能, 为什么要提供审计功能?

答: 审计功能是指 DBMS 的审计模块在用户对数据库执行操作的同时把所有操作自动记录到系统的审计日志中。

因为任何系统的安全保护措施都不是完美无缺的, 蓄意盗窃破坏数据的人总可能存在。利用数据库的审计功能, DBA 可以根据审计跟踪的信息, 重现导致数据库现有状况的一系列事件, 找出非法存取数据的人、时间和内容等。

第 5 章 数据库完整性

1 什么是数据库的完整性?

答:

数据库的完整性是指数据的正确性和相容性。

2. 数据库的完整性概念与数据库的安全性概念有什么区别和联系?

答:

数据的完整性和安全性是两个不同的概念,但是有一定的联系。前者是为了防止数据库中存在不符合语义的数据,防止错误信息的输入和输出,即所谓垃圾进垃圾出 (Garba : e In Garba : e out) 所造成的无效操作和错误结果。后者是保护数据库防止恶意的破坏和非法的存取。也就是说,安全性措施的防范对象是非法用户和非法操作,完整性措施的防范对象是不合语义的数据。

3. 什么是数据库的完整性约束条件?可分为哪几类?

答

完整性约束条件是指数据库中的数据应该满足的语义约束条件。一般可以分为六类:静态列级约束、静态元组约束、静态关系约束、动态列级约束、动态元组约束、动态关系约束。静态列级约束是对一个列的取值域的说明,包括以下几个方面:(1)对数据类型的约束,包括数据的类型、长度、单位、精度等;(2)对数据格式的约束;(3)对取值范围或取值集合的约束;(4)对空值的约束;(5)其他约束。静态元组约束就是规定组成一个元组的各个列之间的约束关系,静态元组约束只局限在单个元组上。静态关系约束是在一个关系的各个元组之间或者若干关系之间常常存在各种联系或约束。

常见的静态关系约束有:(1)实体完整性约束;(2)参照完整性约束;(3)函数依赖约束。

动态列级约束是修改列定义或列值时应满足的约束条件,包括下面两方面:(1)修改列定义时的约束;(2)修改列值时的约束。动态元组约束是指修改某个元组的值时需要参照其旧值,并且新旧值之间需要满足某种约束条件。动态关系约束是加在关系变化前后状态上的限制条件,例如事务一致性、原子性等约束条件。

4. DBMS 的完整性控制机制应具有哪些功能?

答:

DBMS 的完整性控制机制应具有三个方面的功能:(1)定义功能,即提供定义完整性约束条件的机制;(2)检查功能,即检查用户发出的操作请求是否违背了完整性约束条件;(3)违约反应:如果发现用户的操作请求使数据违背了完整性约束条件,则采取一定的动作来保证数据的完整性。

5. RDBMS 在实现参照完整性时需要考虑哪些方面?

答

RDBMS 在实现参照完整性时需要考虑以下几个方面:

(1)外码是否可以接受空值。

(2)删除被参照关系的元组时的考虑,这时系统可能采取的作法有三种:1)级联删除 (CASCADES); 2)受限删除 (RESTRICTED); 3)置空值删除 (NULLIFIES)。 (3)在参照关系中插入元组时的的问题,这时系统可能采取的作法有:1)受限插入; 2)递归插入。

(4)修改关系中主码的问题。一般是不能用 UPDATE 语句修改关系主码的。如果需要修改主码值,只能先删除该元组,然后再把具有新主码值的元组插入到关系中。如果允许修改主码,首先要保证主码的惟一性和非空,否则拒绝修改。然后要区分是参照关系还是被参照关系。

6. 假设有下面两个关系模式:职工 (职工号,姓名,年龄,职务,工资,部门号),其中职工号为主码;部门 (部门号,名称,经理名,电话),其中部门号为主码。用 sQL 语言定义这两个关系模式,要求在模式中完成以下完整性约束条件的定义:定义每个模式的主码;定义参照完整性;定义职工年龄不得超过 60 岁。

答

CREATE TABLE DEPT

```

(Deptno NUMBER(2),
 Deptname VARCHAR(10),
 Manager VARCHAR(10),
 PhoneNumber Char(12)
 CONSTRAINT PK_SC PRIMARY KEY(Deptno));

CREATE TABLE EMP
(Empno NUMBER(4),
 Ename VARCHAR(10),
 Age NUMBER(2),
 CONSTRAINT C1 CHECK (Age <= 60),
 Job VARCHAR(9),
 Sal NUMBER(7, 2),
 Deptno NUMBER(2),
 CONSTRAINT FK_DEPTNO
 FOREIGN KEY(Deptno)
 REFERENCES DEPT(Deptno));

```

7. 关系系统中, 当操作违反实体完整性、参照完整性和用户定义的完整性约束条件时, 一般是如何分别进行处理的?

答:

对于违反实体完整性和用户定义的完整性的操作一般都采用拒绝执行的方式进行处理。而对于违反参照完整性的操作, 并不都是简单地拒绝执行, 有时要根据应用语义执行一些附加的操作, 以保证数据库的正确性。

第6章 关系数据库理论

1. 理解并给出下列术语的定义:

函数依赖、部分函数依赖、完全函数依赖、传递依赖、候选码、主码、外码、全码 (All key)、1NF、2NF、3NF、BCNF、多值依赖、4NF。

定义1: 设 $R(U)$ 是属性集 U 上的关系模式。 X, Y 是属性集 U 的子集。若对于 $R(U)$ 的任意一个可能的关系 r , r 中不可能存在两个元组在 X 上的属性值相等, 而在 Y 上的属性值不等, 则称 X 函数确定 Y 或 Y 函数依赖于 X , 记作 $X \rightarrow Y$ 。(即只要 X 上的属性值相等, Y 上的值一定相等。)

术语和记号:

$X \rightarrow Y$, 但 Y 不是 X 的子集, 则称 $X \rightarrow Y$ 是非平凡的函数依赖。若不特别声明, 总是讨论非平凡的函数依赖。

$X \rightarrow Y$, 但 Y 是 X 的子集, 则称 $X \rightarrow Y$ 是平凡的函数依赖。

若 $X \rightarrow Y$, 则 X 叫做决定因素 (Determinant)。

若 $X \rightarrow Y, Y \rightarrow X$, 则记作 $X \leftrightarrow Y$ 。

若 Y 不函数依赖于 X , 则记作 $X \nrightarrow Y$ 。

定义2: 在 $R(U)$ 中, 如果 $X \rightarrow Y$, 并且对于 X 的任何一个真子集 X' , 都有 $X' \nrightarrow Y$, 则称 Y 对 X 完全函数依赖

若 $X \rightarrow Y$, 但 Y 不完全函数依赖于 X , 则称 Y 对 X 部分函数依赖

定义3: 若关系模式 R 的每一个分量是不可再分的数据项, 则关系模式 R 属于第一范式 (1NF)。

定义4: 若关系模式 $R \in 1NF$, 且每一个非主属性完全函数依赖于码, 则关系模式 $R \in 2NF$ 。(即 1NF 消除了非主属性对码的部分函数依赖则成为 2NF)。

定义5: 关系模式 $R(U, F)$ 中若不存在这样的码 X 、属性组 Y 及非主属性 Z (Z 不是 Y 的子集) 使得 $X \rightarrow Y, Y \rightarrow Z$

$X, Y \rightarrow Z$ 成立, 则称 $R\langle U, F \rangle \in 3NF$ 。

定义 6: 关系模式 $R\langle U, F \rangle \in 1NF$ 。若 $X \rightarrow Y$ 且 Y 不是 X 的子集时, X 必含有码, 则 $R\langle U, F \rangle \in BCNF$ 。

定义 7: 关系模式 $R\langle U, F \rangle \in 1NF$, 如果对于 R 的每个非平凡多值依赖 $X \twoheadrightarrow Y$ (Y 不是 X 的子集, $Z = U - X - Y$ 不为空), X 都含有码, 则称 $R\langle U, F \rangle \in 4NF$ 。

2. 建立一个关于系、学生、班级、学会等诸信息的关系数据库。

学生: 学号、姓名、出生年月、系名、班号、宿舍区。

班级: 班号、专业名、系名、人数、入校年份。

系: 系名、系号、系办公地点、人数。

学会: 学会名、成立年份、办公地点、人数。

语义如下: 一个系有若干专业, 每个专业每年只招一个班, 每个班有若干学生。一个系的学生住在同一宿舍区。每个学生可参加若干学会, 每个学会会有若干学生。学生参加某学会会有一个入会年份。

请给出关系模式, 写出每个关系模式的极小函数依赖集, 指出是否存在传递函数依赖, 对于函数依赖左部是多属性的情况讨论函数依赖是完全函数依赖, 还是部分函数依赖。指出各关系模式的候选码、外部码, 有没有全码存在?

2. 答:

关系模式: 学生 $S(S\#, SN, SB, DN, C\#, SA)$

班级 $C(C\#, CS, DN, CNUM, CDATE)$

系 $D(D\#, DN, DA, DNUM)$

学会 $P(PN, DATE1, PA, PNUM)$

学生-学会 $SP(S\#, PN, DATE2)$

其中, $S\#$ —学号, SN —姓名, SB —出生年月, SA —宿舍区

$C\#$ —班号, CS —专业名, $CNUM$ —班级人数, $CDATE$ —入校年份

$D\#$ —系号, DN —系名, DA —系办公室地点, $DNUM$ —系人数

PN —学会名, $DATE1$ —成立年月, PA —地点, $PNUM$ —学会会员人数,

$DATE2$ —入会年份

依据上面给出的语义, 写出每个关系模式的极小函数依赖集:

$S: S\# \rightarrow SN, S\# \rightarrow SB, S\# \rightarrow C\#, C\# \rightarrow DN, DN \rightarrow SA$

/*一个系的学生住在同一宿舍区*/

$C: C\# \rightarrow CS, C\# \rightarrow CNUM, C\# \rightarrow CDATE, CS \rightarrow DN, (CS, CDATE) \rightarrow C\#$

/*因为每个专业每年只招一个班*/

$D: D\# \rightarrow DN, DN \rightarrow D\#, D\# \rightarrow DA, D\# \rightarrow DNUM$

/*按照实际情况, 系名和系号是一一对应的*/

$P: PN \rightarrow DATE1, PN \rightarrow PA, PN \rightarrow PNUM$

$SP: (S\#, PN) \rightarrow DATE2$

/*学生参加某学会会有一个入会年份*/

S 中存在的传递函数依赖:

因为 $S\# \rightarrow C\#$, $C\# \rightarrow DN$, 所以存在传递函数依赖 $S\# \rightarrow DN$,
 因为 $C\# \rightarrow DN$, $DN \rightarrow SA$, 所以存在传递函数依赖 $C\# \rightarrow SA$,
 因为 $S\# \rightarrow C\#$, $C\# \rightarrow DN$, $DN \rightarrow SA$, 所以存在传递函数依赖 $S\# \rightarrow SA$ 。

C 中存在的传递函数依赖:

因为 $C\# \rightarrow CS$, $CS \rightarrow DN$, 所以存在传递函数依赖 $C\# \rightarrow DN$ 。

函数依赖左部是多属性的情况:

$(S\#, PN) \rightarrow DATE2$ 和 $(CS, CDATE) \rightarrow C\#$ 函数依赖左部具有 2 个属性, 他们都是完全函数依赖, 没有部分函数依赖的情况。

关系	候选码	外部码	全码
S	S#	C#, DN	无
C	C#和(CS,CDATE)	DN	无
D	D#和 DN	无	无
P	PN	无	无
SP	(S#, PN)	S#, PN	无

关系模式 C 和 D 都有 2 个候选码。

解: (1) 关系模式如下:

学生: S(Sno, Sname, Sbirth, Dept, Class, Rno)

班级: C(Class, Pname, Dept, Cnum, Cyear)

系: D(Dept, Dno, Office, Dnum)

学会: M(Mname, Myear, Maddr, Mnum)

(2) 每个关系模式的最小函数依赖集如下:

A、学生 S (Sno, Sname, Sbirth, Dept, Class, Rno) 的最小函数依赖集如下: $Sno \rightarrow Sname$, $Sno \rightarrow Sbirth$, $Sno \rightarrow Class$, $Class \rightarrow Dept$, $DEPT \rightarrow Rno$

传递依赖如下:

由于 $Sno \rightarrow Dept$, 而 $Dept \rightarrow Sno$, $Dept \rightarrow Rno$ (宿舍区)

所以 Sno 与 Rno 之间存在着传递函数依赖。

由于 $Class \rightarrow Dept$, $Dept \rightarrow Class$, $Dept \rightarrow Rno$

所以 Class 与 Rno 之间存在着传递函数依赖。

由于 $Sno \rightarrow Class$, $Class \rightarrow Sno$, $Class \rightarrow Dept$

所以 Sno 与 Dept 之间存在着传递函数依赖。

B、班级 C(Class, Pname, Dept, Cnum, Cyear)的最小函数依赖集如下:

$Class \rightarrow Pname$, $Class \rightarrow Cnum$, $Class \rightarrow Cyear$, $Pname \rightarrow Dept$.

由于 $Class \rightarrow Pname$, $Pname \rightarrow Class$, $Pname \rightarrow Dept$

所以 Class 与 Dept 之间存在着传递函数依赖。

C、系 D(Dept, Dno, Office, Dnum)的最小函数依赖集如下:

$Dept \rightarrow Dno$, $Dno \rightarrow Dept$, $Dno \rightarrow Office$, $Dno \rightarrow Dnum$

根据上述函数依赖可知, Dept 与 Office, Dept 与 Dnum 之间不存在传递依赖。

D、学会 M(Mname, Myear, Maddr, Mnum)的最小函数依赖集如下:

$Mname \rightarrow Myear$, $Mname \rightarrow Maddr$, $Mname \rightarrow Mnum$

该模式不存在传递依赖。

(3) 各关系模式的候选码、外部码，全码如下：

- A、学生 S 候选码：Sno；外部码：Dept、Class；无全码
- B、班级 C 候选码：Class；外部码：Dept；无全码
- C、系 D 候选码：Dept 或 Dno；无外部码；无全码
- D、学会 M 候选码：Mname；无外部码；无全码

7. 下面的结论哪些是正确的？哪些是错误的？对于错误的请给一个反例说明之。

(1) 任何一个二目关系是属于 3NF。

答：正确。因为关系模式中只有两个属性，所以无传递。

(2) 任何一个二目关系是属于 BCNF。

答：正确。按 BCNF 的定义，若 $X \rightarrow Y$ ，且 Y 不是 X 的子集时，每个决定因素都包含码，对于二目关系决定因素必然包含码。详细证明如下：（任何二元关系模式必定是 BCNF）。

证明：设 R 为一个二目关系 R(A1, A2)，则属性 A1 和 A2 之间可能存在以下几种依赖关系：

A、 $A1 \rightarrow A2$ ，但 $A2 \not\rightarrow A1$ ，则关系 R 的码为 A1，决定因素都包含码，所以，R 是 BCNF。

B、 $A1 \rightarrow A2$ ， $A2 \rightarrow A1$ ，则关系 R 的码为 A2，所以决定因素都包含码，R 是 BCNF。

包含码。R 是 BCNF。C、R 的码为 (A1, A2)（即 $A1 \rightarrow A2$ ， $A2 \rightarrow A1$ ），决定因素都

第七章 数据库设计

1. 试述数据库设计过程。

答：这里只概要列出数据库设计过程的六个阶段：（1）需求分析；（2）概念结构设计；（3）逻辑结构设计；（4）数据库物理设计；（5）数据库实施；（6）数据库运行和维护。这是一个完整的实际数据库及其应用系统的设计过程。不仅包括设计数据库本身，还包括数据库的实施、运行和维护。设计一个完善的数据库应用系统往往是上述六个阶段的不断反复。

2. 试述数据库设计过程各个阶段上的设计描述。

答：各阶段的设计要点如下：（1）需求分析：准确了解与分析用户需求（包括数据与处理）。（2）概念结构设计：通过对用户需求进行综合、归纳与抽象，形成一个独立于具体 DBMS 的概念模型。（3）逻辑结构设计：将概念结构转换为某个 DBMS 所支持的数据模型，并对其进行优化。（4）数据库物理设计：为逻辑数据模型选取一个最适合应用环境的物理结构（包括存储结构和存取方法）。（5）数据库实施：设计人员运用 DBMS 提供的数据库语言、工具及宿主语言，根据逻辑设计和物理设计的结果建立数据库，编制与调试应用程序，组织数据入库，并进行试运行。（6）数据库运行和维护：在数据库系统运行过程中对其进行评价、调整与修改。

3. 试述数据库设计过程中结构设计部分形成的数据库模式。

答：数据库结构设计的不同阶段形成数据库的各级模式，即：（1）在概念设计阶段形成独立于机器特点，独立于各个 DBMS 产品的概念模式，在本篇中就是 E—R 图；（2）在逻辑设计阶段将 E—R 图转换成具体的数据库产品支持的数据模型，如关系模型，形成数据库逻辑模式，然后在基本表的基础上再建立必要的视图（Vi 娜），形成数据的外模式；（3）在物理设计阶段，根据 DBMS 特点和处理的需要，进行物理存储安排，建立索引，形成数据库内模式。

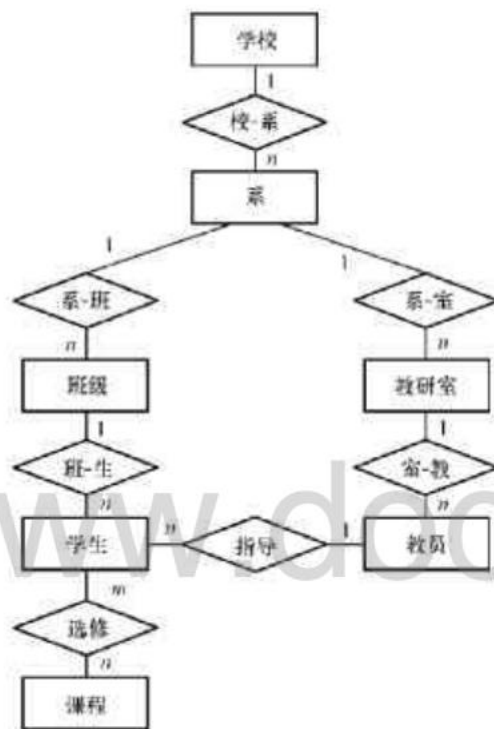
5. 什么是数据库的概念结构？试述其特点 and 设计策略。

答：概念结构是信息世界的结构，即概念模型，其主要特点是：（1）能真实、充分地反映现实世界，包

括事物和事物之间的联系，能满足用户对数据的处理要求，是对现实世界的一个真实模型；（2）易于理解，从而可以用它和不熟悉计算机的用户交换意见，用户的积极参与是数据库设计成功的关键；（3）易于更改，当应用环境和应用要求改变时，容易对概念模型修改和扩充；（4）易于向关系、网状、层次等各种数据模型转换。概念结构的设计策略通常有四种：1）自顶向下，即首先定义全局概念结构的框架，然后逐步细化；2）自底向上，即首先定义各局部应用的概念结构，然后将它们集成起来，得到全局概念结构；3）逐步扩张，首先定义最重要的核心概念结构，然后向外扩充，以滚雪球的方式逐步生成其他概念结构，直至总体概念结构；4）混合策略，即将自顶向下和自底向上相结合，用自顶向下策略设计一个全局概念结构的框架，以它为骨架集成由自底向上策略中设计的各局部概念结构。

7. 学校中有若干系，每个系有若干班级和教研室，每个教研室有若干教员，其中有的教授和副教授每人各带若干研究生；每个班有若干学生，每个学生选修若干课程，每门课可由若干学生选修。请用 E—R 图画出此学校的概念模型。

答：



各实体属性为：

系：系编号，系名

班级：班级号，班级名

教研室：教研室号，教研室

学生：学号，姓名，学历

课程：课程号，课程名

教员：职工号，姓名，职称

联系的属性：“选修”的属性为“成绩”

转换为关系模型如下：

系（系编号，系名，学校名）

班级（班级号，班级名，系编号）

教研室（教研室号，教研室，系编号）

学生（学号，姓名，学历，班级号，导师职工号）

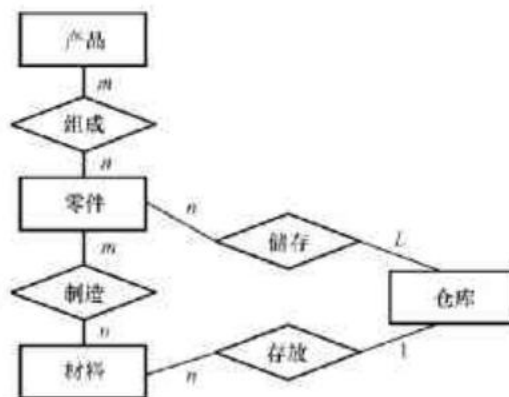
课程（课程号，课程名）

教员（职工号，姓名，职称，教研室号）

选修（学号，课程号，成绩）

8. 某工厂生产若干产品，每种产品由不同的零件组成，有的零件可用在不同的产品上。这些零件由不同的原材料制成，不同零件所用的材料可以相同。这些零件按所属的不同产品分别放在仓库中，原材料按照类别放在若干仓库中。请用 E—R 图画出此工厂产品、零件、材料、仓库的概念模型。

答：



各实体属性为：

产品：产品号，产品名

零件：零件号，零件名

原材料：原材料号，原材料名，类别

仓库：仓库号，仓库名

各联系的属性为：

产品组成：使用零件量

零件制造：使用原材料量

零件储存：存储量

材料存放：存储量

转换为关系模型如下：

产品（产品号，产品名，仓库号）

零件：零件号，零件名

原材料：原材料号，原材料名，类别，仓库号，存放量）

仓库（仓库号，仓库名）

产品组成（产品号，零件号，使用零件量）

零件组成（零件号，原材料号，使用原材料量）

零件储存（零件号，仓库号，存储量）

9. 什么是数据库的逻辑结构设计？试述其设计步骤。

答：数据库的逻辑结构设计就是把概念结构设计阶段设计好的基本 E—R 图转换为与选用的 DBMS 产品所支持的数据模型相符合的逻辑结构。设计步骤为：（1）将概念结构转换为一般的关系、网状、层次模

型；（2）将转换来的关系、网状、层次模型向特定 DBMS 支持下的数据模型转换；（3）对数据模型进行优化。

11、第七题中设计的关系模型中的各个关系模式都只有一个码，且都是唯一的决定因素，所以属于 BCNF，不会发生更新异常。