1. 绪论（背诵）

数据库四个基本概念：数据（data）、数据库（DataBase）、数据库管理系统（DataBase Management System）、数据库系统（DataBase System）

* 数据：描述事物的符号记录，是数据库中存储的基础对象
* 数据库：数据库是长期储存在计算机内、有组织的、可共享的大量数据的集合。

基本特征：1.数据按一定的数据模型组织、描述和储存

1. 可为各种用户共享
2. 冗余度较小
3. 数据独立性较高
4. 易扩展

* 数据库管理系统：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

主要功能：1.数据定义功能：DBMS提供数据定义语言（DDL）

1. 数据组织、存储和管理：实现数据之间的联系
2. 数据操纵功能：DBMS提供数据操纵语言（DML）
3. 数据库的事务管理和运行管理
4. 数据库的建立和维护功能
5. 其他功能

* 数据库系统：数据库系统是由数据库、数据库管理系统（及其应用开发工具）、应用程序和数据库管理员组成的存储、管理、处理和维护数据的系统。

数据库发展的三个阶段：人工管理阶段、文件系统阶段、数据库系统阶段

* 人工管理阶段：

特点：1.数据不保存

1. 应用程序管理数据：没有相应的软件系统负责数据管理
2. 数据不共享
3. 数据不具有独立性

* 文件系统阶段：

特点：1.数据可以长期保存

1. 由文件系统管理数据：由专门的软件即文件系统进行数据管理
2. 数据共享性差，冗余度大
3. 数据独立性差

* 数据库系统阶段：

特点：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

1. 数据的共享性高、冗余度低且易扩充：数据共享可以大大减少数据冗余，节约存储空间。数据共享还能避免数据之间的不相容性和不一致性
2. 数据独立性高
3. 数据由数据库管理系统统一管理和控制：数据库管理系统必须提供的数据控制功能：数据的安全性保护、数据的完整性检查、并发控制、数据库恢复

数据模型是数据库系统的核心和基础，分为两大类：第一类是概念模型，第二类是逻辑模型和物理模型。

数据模型通常由数据结果、数据操作和数据的完整性约束条件三部分组成。

常用的概念模型是实体-联系方法，即E-R方法（E-R图、E-R模型）

常用的逻辑模型：层次模型、网状模型、关系模型、面向对象数据模型、对象关系数据模型、半结构化数据模型

数据库的三级模式结构是指数据库是由外模式、模式、内模式三级构成

* 模式：模式也称逻辑模式，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。

一个数据库只有一个模式，模式与数据的物理储存细节和硬件环境无关，与具体的应用程序、开发工具及高级程序设计语言无关。

* 外模式：外模式也称子模式或者用户模式，它是数据库用户能够看见和使用的局部数据的逻辑结果和特征的描述，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

模式与外模式的关系：一对多；外模式与应用的关系：一对多

* 内模式：内模式也称存储模式，一个数据只有一个内模式，他是数据物理结构和存储方式的描述，是数据在数据库内部的组织方式。

数据库系统的三级模式是数据的三个抽象级别，为了能够在系统内部实现这三个抽象层次的联系和转化，数据库管理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。

* 外模式/模式映像：当模式改变时，由数据库管理员对各个外模式/模式的映像作出相应改变，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。

同一个模式可以有任意个外模式。每一个外模式，数据库系统都有一个外模式/模式映像，定义外模式与模式之间的对应关系。

* 模式/内模式映像：当数据库的存储结构改变时，由数据库管理员对模式/内模式映像作出相应改变，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。

数据库中模式/内模式映像是唯一的，该映像定义通常包含在模式描述中。

二、关系数据库（重点是理解）

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。

关系数据结构：

* 若关系中的某一属性组的值能够唯一地标识一个元组，而其子集不能，则称该属性组为候选码。
* 若一个关系有多个候选码，则选定其中一个为主码。
* 候选码的诸属性称为主属性。不包含在任何候选码中的属性称为非主属性或非码属性。
* 若关系模式的所有属性是这个关系模式的候选码，称为全码。
* 关系可以有三种类型：基本关系（通常又称为基本表或基表）、查询表和视图表。
* 基本关系具有以下6条性质：

1. 列是同质的，即每一列中的分量是同一类型的数据，来自同一个域。
2. 不同的列可出自同一个域，称其中的每一列为一个属性，不同的属性要给予不同的属性名。
3. 列的顺序无所谓，即列的次序可以任意交换。
4. 任意两个元组的候选码不能取相同的值。
5. 行的顺序无所谓，即行的次序可以任意交换。
6. 分量必须取原子值，即每一个分量都必须是不可分的数据项。

* 关系的描述称为关系模式。它可以形式化地表示为 R（U,D,DOM,F），其中R为关系名，U为组成该关系的属性名的集合，D为U中属性所来自的域，DOM为属性向域的映像集合，F为属性间数据的依赖关系集合。

关系操作：

* 关系模型中常用的关系操作包括查询操作和插入、删除、修改操作两大部分。
* 对于查询操作，选择、投影、并、差、笛卡儿积是5种基本操作。
* 关系操作的特点是集合操作方式，即操作的对象和结果都是集合。

关系的完整性：

* 关系模型中有三种完整性约束：实体完整性、参照完整性和用户定义的完整性。其中实体完整性和参照完整性是关系模型必须满足的完整性约束条件，被称为是关系的两个不变性。
* 实体完整性规则：若属性（指一个或一组属性）A是基本关系R的主属性，则A不能取空值。所谓空值就是“不知道”或“不存在”或“无意义”的值。
* 参照完整性：

设F是基本关系R的一个或一组属性，但不是关系R的码，Ks是基本关系S的主码。如果F与Ks相对应，则称F是R的外码，并称基本关系R为参照关系，基本关系S为被参照关系或目标关系。

参照完整性规则：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：或者取空值（F的每一个属性值均为空值），或者等于S中某个元组的主码。

关系代数：

* 传统的集合运算包括并、差、交、笛卡尔积4种运算，其中笛卡尔积不要求两个关系具有相同的目。
* 专门的关系运算包括选择、投影、连接、除运算等。

四、数据库安全性

数据库的安全性是指保护数据库以防止不合法使用所造成的数据泄露、更改或破坏。

TCSEC/TDI，即紫皮书，从4个方面来描述安全性级别划分的指标，即安全策略、责任、保证和文档。

根据计算机系统对各项指标的支持情况，TCSEC/TDI将系统划分为4组7个等级，依次是D、C（C1、C2）、B（B1、B2、B3）、A（A1），按系统可靠或可信程度逐渐增高。

D级：最小保护。该级是最低级别。

C1级：自主安全保护。该级进行 自主存取控制（DAC）

C2级：受控的存取保护。

B1级：标记安全保护。对标记的主体和客体实施强制存取控制（MAC）以及审计等安全机制。B1级别的产品才被认为是真正意义上的安全产品。

B2级：结构化保护。

B3级：安全域。

A1级：验证设计。

CC安全保证要求，从EAL1至EAL7共分为7级，按保证程度逐渐增高。

EAL2：结构测试。

EAL3：系统地测试和检查。

EAL7:形式化验证的设计和测试。

粗略而言，TCSEC的C1和C2级分别相当于EAL2和EAL3；B1、B2和B3分别相当于EAL4、EAL5、EAL6；A1对应于EAL7。

数据库安全性控制，主要包括用户身份鉴别、多层存取控制、审计、视图和数据加密等安全技术。

* 用户身份鉴别：常用的用户身份鉴别方法有：

1. 静态口令鉴别
2. 动态口令鉴别
3. 生物特征鉴别
4. 智能卡鉴别

* 存取控制：存取控制机制主要包括定义用户权限和合法权限检查两部分，这两部分一起组成了数据库管理系统的存取控制子系统。C2级的数据库管理系统支持自主存取控制（DAC），B2级的数据库管理系统支持强制存取控制（MAC）。
* 数据加密：数据加密主要包括存储加密和传输加密。

五、数据库的完整性

数据库的完整性是指数据的正确性和相容性。

关系数据库管理系统使得完整性控制成为其核心支持的功能，从而能够为所有用户和应用提供一致的数据库完整性。

* 实体完整性：对单属性构成的码有两种说明实体完整性的方法，一种是定义为列级约束条件，另一种是定义为表级约束条件。对于多个属性构成的码只有一种说明方法，即定义为表级约束条件。
* 参照完整性：当可能破坏参照完整性的操作发生时，系统可以采用以下策略加以处理：

1. 拒绝执行：不允许该操作执行。该策略一般设置为默认策略。
2. 级联操作：当删除或修改被参照表的一个元组导致与参照表的不一致时，删除或修改参照表中的所有导致不一致的元组。
3. 设置为空值：当删除或修改被参照表的一个元组时造成了不一致，则将被参照表中的所有造成不一致的元组的对应属性设置为空值。

* 用户定义的完整性：包括

1. 列值非空（NOT NULL）
2. 列值唯一（UNIQUE）
3. 检查列值是否满足一个条件表达式（CHECK短语）

触发器：触发器是用户定义在关系表上的一类由事件驱动的特殊过程。

同一模式下，触发器名必须是唯一的，并且触发器名和表名必须在同一模式下。

触发器只能定义在基本表上，不能定义在视图上。

触发事件可以是INSERT, DELETE,或UPDATE，也可以是这几个事件的组合。AFTER/BEFORE是触发的时机。

触发器按照所触发动作的间隔尺寸可以分为行级触发器和语句级触发器。

触发动作体既可以是一个匿名PL/SQL过程块，也可以是对已创建存储过程的调用。

如果触发动作体执行失败，激活触发器的事件（即对数据库的增、删、改操作）就会终止执行，触发器的目标表或触发器可能影响的其他对象不发生任何变化。

1. 关系数据理论（理解+背诵）

本章把关系模型看作一个三元组：R<U,F>，其中：关系名R是符号化的元组语义；U为一组属性，F为属性组U上的一组数据依赖。

各种依赖的定义：见课本P180

候选码是最小的超码。

主码或候选码都简称为码。

包含在任何一个候选码中的属性称为主属性；不包含在任何候选码中的属性称为非主属性或非码属性。

若整个属性组是码，称为全码。

关系模型可能存在的问题：1.数据冗余

1. 更新异常
2. 插入异常
3. 删除异常

关系数据库中的关系是要满足一定要求的，满足不同程度要求的为不同范式。按级别从低到高分别是：1NF 2NF 3NF BCNF 4NF 5NF

一个低一级范式的关系模式通过模式分解可以转换成若干个高一级范式的关系模式的集合，这种过程就叫规范化。

* 第一范式（1NF）：关系的每一个分量必须是不可分的数据项。
* 第二范式（2NF）：在第一范式的基础上，不存在非主属性对码的部分函数依赖
* 第三范式（3NF）：在第二范式的基础上，不存在非主属性对码的传递依赖
* BC范式（BCNF）：在第三范式的基础上，不存在主属性对码的部分和传递函数依赖（即要求非平凡函数依赖关系的左边必须是超码）
* 第四范式（4NF）：在BC范式的基础上，不存在非平凡且非函数依赖的多值依赖（即只允许存在平凡多值依赖和函数依赖）

关于模式分解的几个重要事实是：

1. 若要求分解保持函数依赖，那么模式分离总可以达到3NF，但不一定能达到BCNF。
2. 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。
3. 若要求分解具有无损连接性，那一定可达到4NF。
4. 数据库设计

数据库设计是指对于一个给定的应用环境，构造（设计）优化的数据库逻辑模式和物理结构，并据此建立数据库及其应用系统，使之能够有效地存储和管理数据，满足各种用户的应用需求，包括信息管理要求和数据操作要求。

数据库设计的特点：

1. 数据库建设的基本规律：“三分技术，七分管理，十二分基础数据”是数据库设计的特点之一。“十二分基础数据”强调了数据的收集、整理、组织和不断更新是数据库建设中重要环节。
2. 结构（数据）设计和行为（处理）设计相结合：整个设计过程中要把数据库结构设计和对数据的处理设计密切结合起来。这是数据库设计的特点之二。

数据库设计的基本步骤：将数据库设计分为以下6个阶段：需求分析、概念结构设计、逻辑结构设计、物理结构设计、数据库实施、数据库运行和维护。在数据库设计过程中，需求分析和概念结构设计可以独立于任何数据库管理系统进行，逻辑结构设计和物理结构设计与选用的数据库管理系统密切相关。

* 需求分析：需求分析是整个设计过程的基础，是最困难和最耗费时间的一步。

需求分析需要调查，调查的重点是“数据”和“处理”，通过调查、收集和分析，获得用户对数据库的如下要求：信息要求、处理要求、安全性与完整性要求。

数据字典是进行详细的数据收集和数据分析所获得的主要成果。它是关于数据库中的数据的描述，即元数据，而不是数据本身。数据字典是在需求分析阶段建立，在数据库设计过程不断修改、充实、完善的。它在数据库设计中占有很重要的地位。

数据字典通常包括数据项、数据结构、数据流、数据的存储和处理过程几部分。其中数据项是数据的最小组成单位。

* 概念结构设计：概念结构设计是整个数据库设计的关键。描述概念模型的有力工具是E-R模型。
* 逻辑结构设计：逻辑结构设计的任务就是把E-R图转换为数据库管理系统所支持的逻辑结构。一般的转换原则如下：

1.一个实体型转化为一个关模式，关系的属性就是实体的属性，关系的码就是实体的码

2.一个1:1的联系可以转换为一个独立的关系模式，也可以与任意一端对应的关系模式合并

3.一个1：n的联系可以转换为一个独立的关系模式，也可以与n端对应的关系模式合并。

4.一个m:n的联系转化为一个关系模式

5.三个或三个实体间的一个多元联系可以转换为一个关系模式。

6.具有相同码的关系模式可以合并。

必须注意的是，并不是规范化程度越高的关系就越优。

* 物理结构设计：数据库的物理设计通常分为两步：

1. 确定数据库的物理结构：在关系数据库中主要指存取方法和存储结构
2. 对物理结构进行评价：评价的重点是时间和空间效率。

通常关系数据库物理设计的内容主要包括 为关系模式选择存取方法，设计关系、索引等数据库文件的物理存储结构。

数据库管理系统一般提供多种存取方法，常用的存取方法为索引方法和聚簇方法。

* 数据库的运行和维护：数据库应用程序的设计应该与数据库设计同时进行

1. R模型：E-R模型的主要概念包括实体、属性、实体之间的联系等。

* 实体之间的联系：

两个实体之间的联系：1.一对一联系（1:1）

2.一对多联系（1：n）

3.多对多联系（m：n）

两个以上的实体型之间的联系

单个实体型之间的联系

一般地，把参与联系的实体型称为联系的度，N个实体型之间的联系度为N，称为N元联系。

* E-R图：

1. 实体型用矩形表示
2. 属性用椭圆形表示
3. 联系用菱形表示
4. ISA联系：

不相交约束描述父类中的一个实体不能同时属于多个子类中的实体集，即一个父类中的实体最多属于一个子类实体集，用ISA联系三角形符号内加一个叉号“X”来表示。如果父类中的一个实体能同时属于多个子类中的实体，则称为可重叠约束，子类符号中没有叉号表示是可重叠的。

完备性约束描述父类中的一个实体是否必须是某一个子类中的实体。如果是，叫做完全特化，否则叫做部分特化。完全特化用父类到子类的双线连接来表示，单线连接则表示部分特化。

1. 基数约束：

基数约束是实体之间联系的细化，约束用一个数对 min..max 表示，其中\*表示无穷大，min=1的约束叫做强制参与约束，min=0的叫做非强制参与约束。

1. Part-of约束：

如果一个实体型的存在依赖于其他实体型的存在，则这个实体型叫做弱实体性，否则叫做强实体型。

* E-R图的设计与集成：

1. 为了简化E-R图的处置，现实世界的事物能作为属性对待的尽量作为属性对待。作为属性的对待的两条准则：
2. 作为属性，不能再具有需要描述的性质
3. 属性不能与其他实体具有联系
4. 各子系统的E-R图之间的冲突主要有三类：属性冲突、命名冲突、结构冲突
5. 在初步E-R图中可能存在一些冗余的数据和实体间冗余的联系。