**第一章**

3、简述数据库系统的三级模式和两级映像的含义。

答：从数据库管理系统的角度看，数据库系统的结构通常分为三级模式的总体结构，在这种模式下，形成了二级映像，实现了数据的独立性。其中三级模式结构指的是外模式、模式和内模式，二级映像指的是外模式/模式映像、模式/内模式映像。

模式也称逻辑模式和概念模式，是数据库中全体数据逻辑结构和特征的描述，描述现实世界中的实体及其性质与联系，是所有用户的公共数据视图；外模式也称子模式或用户模式，它是用以描述用户看到或使用的数据的局部逻辑结构和特性的，用户根据外模式用数据操作语句或应用程序去操作数据库中的数据；内模式也称存储模式，是整个数据库的最底层表示，它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表示方式。

对于外模式/模式映像，当模式改变时，相应的外模式/模式映像作相应的改变，以使外模式保持不变，而应用程序是依据数据的外模式来编写的，外模式不变，应用程序就没必要修改，这保证了数据与程序的逻辑独立性。对于模式/内模式映像，当数据库的存储结构变了，模式/内模式映像会作相应的改变，以使模式保持不变，而模式不变，与模式没有直接联系的应用程序也不会改变，这保证了数据与程序的物理独立性。

5、数据库管理系统的主要功能有哪些？

答：⑴、数据定义功能；

⑵、数据操纵功能；

⑶、数据组织、存取功能；

⑷、数据库运行管理功能；

⑸、数据库建立与维护功能

**第五章**

1、解释下列术语的含义：

函数依赖、平凡函数依赖、非平凡函数依赖、部分函数依赖、完全函数依赖、传递函数依赖、1NF、2NF、3NF、BCNF、多值依赖、4NF、最小函数依赖、函数依赖保持性、无损连接性。

①、函数依赖：设R(U)是属性集U上的一个关系模式，X、Y是U的子集。若对于R(U)上的任意一个可能的关系r，如果r中不存在两个元组，它们在X上的属性值相同，而在Y上的属性值不同，则称“X函数决定Y”或“Y函数依赖X”，记作X→Y。

②、平凡函数依赖：设R(U)是属性集U上的一个关系模式，X、Y是U的子集。若Y是X的子集，则称X→Y为平凡函数依赖。

③、非平凡函数依赖：设R(U)是属性集U上的一个关系模式，X、Y是U的子集。如果X→Y，且Y∉X，则称X→Y为非平凡函数依赖。

④、部分函数依赖：如果X→Y，但不完全函数依赖于X，则称Y对X部分函数依赖。

⑤、完全函数依赖：在R(U)中，如果X→Y，并且对于X的任何一个真子集X’，都有Y函数不依赖于X’，则称Y完全函数依赖于X。

⑥、传递函数依赖：在R(U)中，如果X→Y ，Y→Z，且Y∉X，X也不函数依赖于Y，则称Z传递函数依赖于X。

⑦、1NF：如果关系模式R的所有属性均为简单属性，即每个属性都是不可再分的，则称R属于第一范式。

⑧、2NF：如果关系模式R∈1NF，且每个非主属性都完全依赖于R的码，则称R属于第二范式。

⑨、3NF：如果关系模式R∈2NF，且每个非主属性都不传递函数依赖于R的候选码，则称R属于第三范式。

⑩、BCNF：如果关系模式R∈1NF，且对于所有的函数依赖X→Y(Y∉X)，决定因素X都包含了R的一个候选码，则称R属于BC范式。

、多值依赖：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。

、4NF：关系模式R<U,F>∈1NF，如果对于R的每个非平凡多值依赖X→→Y(Y∉X)，X都含有码，则称R<U,F>∈4NF。

、最小函数依赖：函数依赖集F满足以下条件：

a、F中的任何一个函数依赖的右部仅含有一个属性；

b、F中不岑仔这样一个函数依赖X→A，使得F与F-{X→A }等价；

c、F中不存在这样一个函数依赖X→A，X有真子集Z使得F-{X→A }∪{Z→A}与F等价。

、函数依赖保持性：设p={R1<U1，F1>,R2<U2，F2>,···,Rn<Un,Fn>}是关系模式R{U,F}上的一个分解。若∪Fi =F ，则称分解p具有函数依赖保持性。

、无损连接性：设p={R1<U1，F1>,R2<U2，F2>,···,Rn<Un,Fn>}是关系模式R{U,F}上的一个分解。若任何属于R{U,F}的关系r，令r1=πR1(r)，r2=πR2(r)，···，rn=πRn(r)，有r=r1∞r2∞···∞rn成立，则称分解p具有无损连接性。

**第六章**

3、简述聚集索引和非聚集索引的区别。

答：汉语字典的正文本身就是一个聚集索引。比如，我们要查“安”字，就会很自然地翻开字典的前几页，因为“安”的拼音是“an”，而按照拼音排序汉字的字典是以英文字母“a”开头并以“z”结尾的，那么“安”字就自然地排在字典的前部。如果您翻完了所有以“a”开头的部分仍然找不到这个字，那么就说明您的字典中没有这个字；同样的，如果查“张”字，那您也会将您的字典翻到最后部分，因为“张”的拼音是“zhang”。也就是说，字典的正文部分本身就是一个目录，您不需要再去查其他目录来找到您需要找的内容。正文内容本身就是一种按照一定规则排列的目录称为“聚集索引”。

如果您认识某个字，您可以快速地从自动中查到这个字。但您也可能会遇到您不认识的字，不知道它的发音，这时候，您就不能按照刚才的方法找到您要查的字，而需要去根据“偏旁部首”查到您要找的字，然后根据这个字后的页码直接翻到某页来找到您要找的字。但您结合“部首目录”和“检字表”而查到的字的排序并不是真正的正文的排序方法，比如您查“张”字，我们可以看到在查部首之后的检字表中“张”的页码是672页，检字表中“张”的上面是“驰”字，但页码却是63页，“张”的下面是“弩”字，页面是390页。很显然，这些字并不是真正的分别位于“张”字的上下方，现在您看到的连续的“驰、张、弩”三字实际上就是他们在非聚集索引中的排序，是字典正文中的字在非聚集索引中的映射。我们可以通过这种方式来找到您所需要的字，但它需要两个过程，先找到目录中的结果，然后再翻到您所需要的页码。我们把这种目录纯粹是目录，正文纯粹是正文的排序方式称为“非聚集索引”。

4、为什么一个数据文件只能有一个聚集索引？

答：由于一个数据表只能有一种实际的存储顺序，因此在一个数据表中只能建立一个聚集索引。

**第七章**

1、请简要阐述一个数据库设计的几个阶段。

答：①、需求分析阶段

②、概念设计阶段

③、逻辑设计阶段

④、物理设计阶段

⑤、数据库实现阶段

⑥、数据库的运行与维护阶段

**第八章**

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。

3、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：数据库会对用户进行标识，系统内部记录所有合法用户的标识，每次用户要求进入系统时，由系统进行核对通过鉴定以确定用户的合法性。

②、存取控制：通过用户权限定义和合法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。

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。

**第九章**

1、简述事务的概念和事务的四个特性，并解释每一个性质由DBMS的哪个子系统实现，每一个性质对DBS有什么益处。

事务的概念：事务是数据库应用中构成单一逻辑工作单元的操作集合

事务的四个特性及实现与益处：

①、事务的原子性：每个事务的所有操作要么被成功地执行，要么一个也不被执行。

原子性是由DBMS的事务管理子系统实现的。

事务的原子性保证了DBS的完整性。

②、事务的一致性：一个事务的正确执行必须数据库从一个正确状态转换为另一个正确的状态。

事务的一致性是由DBMS的完整性子系统实现的。

事务的一致性保证数据库的完整性。

③、事务的隔离性：多个并发事务之间不能相互干扰，同时并发不影响事务的执行。

事务的隔离性是由DBMS的并发控制子系统实现的。

隔离性使并发执行的事务不必关心其他事务，如同在单用户环境下执行一样。

④、事务的持久性：事务对数据库的更新必须是永久的；事务一旦提交，则永久改变数据库中的数据。

持久性是由DBMS的恢复管理子系统实现的。

持久性能保证DB具有可恢复性。

2、并发操作可能会产生哪几类数据不一致性？分别用什么方法可以避免各种不一致的情况？

①、丢失更新：采用一级封锁协议解决

②、污读：采用二级封锁协议解决

③、不可重读：采用三级封锁协议解决

3、简述封锁的概念以及基本的封锁类型。

封锁的概念：事务T在对某个数据对象操作之前，先向系统发出请求，对其加锁，加锁后事务T就对该数据对象有了一定的控制，在事务T释放它的锁之前，其他的事务不能更新此数据对象。

基本的封锁类型：①、排他锁

②、共享锁

4、什么是封锁协议？简述不同级别的封锁协议的主要区别。

封锁协议的定义：在运用X锁和S锁对数据对象加锁时，需要约定一些规则，这些规则为封锁协议

主要区别：①、一级封锁协议：事务T在修改数据R之前必须先对其加排他锁，直到事务结束才释放。它防止了丢失更新问题，但不能保证可重读和不读“脏”数据。

②、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。它防止丢失更新问题，也进一步防止读“脏”数据，但不能解决不可重读问题。

③、三级封锁协议：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，直到事务结束才释放。它防止了丢失更新问题、读“脏”数据，以及不可重读问题。

5、数据库恢复的基本原则是什么？具体实现方法是什么？

基本原则：数据库的重复存储

具体实现方法：①、转储和建立日志

②、数据恢复

6、什么是“脏”数据？如何避免读取“脏”数据？

“脏”数据定义：不正确的临时值

如何避免：采用二级封锁协议或者三级封锁协议

7、什么是活锁？试述活锁产生的原因及解决办法。

活锁的定义：多个事务申请对数据R申请加锁，而系统随机地加锁，导致某些事务长等待

活锁产生的原因：系统随机地加锁

解决办法：采用“先来先服务”的策略预防活锁的发生

8、什么是死锁？试述死锁产生的原因及解决办法。

答：死锁的定义：两个或两个以上的事务互相申请对方加锁对象的排它锁，造成了循环等待

死锁产生的原因：系统中有两个或两个以上的事务都处于等待状态，并且每个事务都在等待其中另一个事务解除封锁，它才能继续执行下去，结果造成任何一个事务都无法继续执行，这样系统就进入了死锁状态。

解决办法：①、死锁的预防：一次封锁法、顺序封锁法

②、死锁的诊断与解除：超时法、等待图法

③、选择一个处理死锁代价最小的事务，将其撤消，释放此事务持有的所有的锁，使其它事务能继续运行下去。

**选择题部分**

DB指的是数据库（DataBase），DBMS指的是数据库管理系统（DataBase Management System），DBS指的是数据库系统（DataBase System），DBA指的是数据库管理员（Database Administrator），Data指的是数据。

由书中概念易得DBS（数据库系统）包括DBMS（数据库管理系统），DBMS管理和控制DB（数据库），而DB载入、存储、重组与恢复Data（数据）。

数据库系统的特点有：⑴、实现数据共享；⑵、减少数据冗余度；⑶、保持数据的一致性；⑷、数据的独立性；⑸、安全保密性；⑹、并发控制；⑺、故障恢复

DB是长期储存在计算机内、有组织的、可共享的大量数据集合；DBS是实现有组织地、动态地存储大量关联数据，方便多用户访问计算机软件、硬件和数据资源组成的系统；DBMS是把用户对数据的操作转化为对系统存储文件的操作，有效地实现数据库三级（外模式、模式和内模式）之间的转化；MIS指的是管理信息系统（Management Information System），是一个以人为主导，利用计算机硬件、软件及其他办公设备进行信息的收集、传递、存贮、加工、维护和使用的系统。

数据的物理独立性指的是内模式改变，模式（即概念模式）不变

模式也称逻辑模式和概念模式，是数据库中全体数据逻辑结构和特征的描述，描述现实世界中的实体及其性质与联系，是所有用户的公共数据视图；外模式也称子模式或用户模式，它是用以描述用户看到或使用的数据的局部逻辑结构和特性的，用户根据外模式用数据操作语句或应用程序去操作数据库中的数据；内模式也称存储模式，是整个数据库的最底层表示，它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表示方式。

Armstrong公理系统：设关系模式R<U,F>，其中U为属性集，F是U上的一组函数依赖，则有如下推理规则：

①、自反律：若属性集Y 包含于属性集X，属性集X 包含于U，则X→Y为F所蕴涵。

②、增广律：若X→Y为F所蕴涵，且属性集Z 包含于属性集U，则XZ→YZ为F所蕴涵。

③、传递律：若X→Y，Y→Z为F所蕴涵，则X →Z为F所蕴涵。

根据以上三条推理规则又可推出下述三条推理规则：

①、合并规则：若X→Y，X→Z，则X→YZ为F所蕴涵。

②、伪传递律：若X→Y，WY→Z，则XW→Z为F所蕴涵。

③、分解规则：若X→Y，Z包含于Y，则X→Z为F所蕴涵。

多值依赖具有如下性质：

①、对称性：若X→→Y，则X→→Z，其中Z=U-X-Y

②、传递性：若X→→Y，Y→→Z，则X→→Z-Y

③、合并性：若X→→Y，X→→Z，则X→→YZ

④、分解性：若X→→Y，X→→Z，则X→→（Y∩Z），X→→Z-Y，X→→Y-Z均成立

⑤、函数依赖可看做多值依赖的特例。

物理结构设计的工作主要包括以下几点：

①、确定数据的存储结构

②、设计合适的存取路径

③、确定数据的存放位置

④、确定系统配置

属性冲突：属性值的类型、取值范围不一致。

命名冲突：a、同名异义：不同意义的对象在不同的局部应用中具有相同的名字

b、异义同名：同一意义的对象在不同的局部应用中具有不同的名字

结构冲突：a、同一对象在不同的局部应用中具有不同的身份

b、同一对象在不同的局部应用中对应的实体属性组成不完全相同

c、实体之间的联系在不同的局部应用中具有不同的类型

数据库系统安全性的主要技术和方法有以上几种：

①、存取控制技术

②、视图机制

③、审计技术

④、数据加密

若事务T对数据对象A加上X锁，则只允许T读取和修改A，其他任何事务都不能再对A加任何类型的锁，直到T释放A上的锁

关系的完整性有以下几条规则：

⑴、实体完整性规则：若属性Ａ是基本关系Ｒ的主码所包含的属性，则属性Ａ不能取空值。（注：主码不能为空，且主码整体取值也不空）

⑵、参照完整性规则：如果属性集K是关系模式R1的外键，同时K也是关系模式R2的属性，但不是R2的主键，那么K为R2的外键。在R2关系中，K的取值只允许有两种可能：①、空值；②、不为空时，等于R1关系中某个主键值。

⑶、用户自定义的完整性规则：用户针对具体的数据约束，设置完整性规则。