

1. 关系系统对查询优化的步骤

- ①把查询转换成某种内部表示，通常用的内部表示是语法树。
- ②把语法树转换成标准(优化)形式。即利用优化算法，把原始的语法树转换成优化的形式。
- ③选择低层的存取路径。
- ④生成查询计划，选择代价最小的。

2. 请简要说明关系系统对查询优化的一般准则

答：可以归纳为以下几条准则：

- 1) 选择运算尽可能先做。
- 2) 在执行连接前对关系适当地预处理。
- 3) 把投影运算和选择运算同时进行。
- 4) 把投影运算同其前或其后的双目运算结合起来。
- 5) 把某些选择同在它前面要执行的笛卡尔积结合起来成为一个连接运算。
- 6) 找出共同子表达式

3. 试述数据库设计过程

- 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、数据库运行和维护。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。

4. 自主存取控制与强制存取控制。

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。自主存取控制中自主的含义是：用户可以将自己拥有的存取权限“自主”地授予别人。即用户具有一定的“自主”权。

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。

5. 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（Garbage In Garbage 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。

6. 死锁的产生与预防

在数据库中，产生死锁的原因是两个或多个事务都已封锁了一些数据对象，然后又都请求已被其他事务封锁的数据加锁，从而出现死等待。

防止死锁的发生其实就是要破坏产生死锁的条件。预防死锁通常有两种方法：

①一次封锁法要求每个事务必须一次将所有要使用的数据全部加锁，否则就不能继续执行；

②顺序封锁法预先对数据对象规定一个封锁顺序，所有事务都按这个顺序实行封锁。

7. 实体完整性与参照完整性

实体完整性规则要求关系中元组在组成主键的属性上不能有空值。如果出现空值，那么主键值就起不了唯一标识元组的作用。

参照完整性规则要求“不引用不存在的实体”。其形式定义如下：如果属性集K是关系模式R1的主键，K也是关系模式R2的外键，那么R2的关系中，K的取值只允许有两种可能，或者为空值，或者等于R1关系中某个主键值。

8. 文件系统与数据库系统的区别

文件系统	数据库系统
用文件将数据长期包含在外存上	用数据库统一存储数据
程序和数据有一定的联系	程序和数据分离
用操作系统中的存取方法对数据进行管理	用DBMS统一管理和控制数据
实现以文件为单位的数据共享	实现以记录和字段为单位的数据共享

9. 等值联接与自然连接的区别：

- ①自然连接一定是等值联接，但等值连接不一定是自然连接。自然连接要求相等的分量必须是公共属性，而等值联接要求的分量不一定是公共属性。
- ②等值联接不把重复的属性去掉，而自然连接要把重复属性去掉。

10. 视图的作用：

- ①简化用户操作
- ②使用户多角度看待同一数据
- ③对于数据库的重构提供了一定程度的逻辑独立性
- ④对机密数据提供安全保护

11. 介质故障及其恢复

介质故障是指系统在运行过程中，由于某种硬件故障（磁盘损坏、磁头碰撞），或操作系统的某种潜在错误，瞬时强磁场干扰等，使存储在外存中的数据部分丢失或全部丢失的情况。

发生介质故障后，存储在磁盘上的数据被破坏，这时需要装入数据库发生介质故障之前某个时刻的数据副本，并重做自此时开始的所有成功事务，将这些事务已提交的结果重新记入数据库，具体做法：

- 1 装入最新的后备数据数据库副本，使数据库恢复到最近一次转储时刻的一致性装填。
- 2 装入有关的日志文件副本，重做已完成的事务。

12. 会破坏正在运行的数据库，并影响正在存取的这部分数据的所有事务：磁盘的磁头碰撞、瞬时的强磁场干扰、磁盘损坏（都是介质故障）

影响正在运行的所有事务，但不会破坏数据库：中央处理器故障、操作系统故障、突然停电（都是系统故障）

13. SQL的授权和回收语句

- ①grant select on SC to user1 with grant option
- 3 revoke update(Sno) on Student From user2

14. 数据库的并发控制、恢复、完整性和安全性之间的联系与区别

- 1 可恢复性防止DB被破坏或DB中的数据有错误
- 2 并发控制避免DB中数据有错误或用户读取脏数据
- ③完整性保证DB中的数据是正确的，避免非法的更新
- ④安全性防止不合法的使用，避免数据泄漏、非法更改和破坏