

1.数据库的事务隔离级别有哪些，各有什么优缺点

什么是数据库事务，数据库事务的性质（ACID）

- 数据库事务：是访问并可能操作各种数据项的一个数据库操作序列，是一个不可分割的工作单位。数据库事务由事务开始和事务结束中间所有的数据库操作组成。这些操作要么全部执行，要么全部不执行。
- 事务的性质（ACID）
 - 原子性 Atomicity 事务不可分割，要么全部执行，要么全部不执行
 - 一致性 Consistency 几个并行执行的事务其执行结果必须与按某一顺序串执行结果相一致。以银行转账事务为例。在事务开始之前，所有账户余额的总额处于一致状态。在事务进行的过程中，一个账户余额减少了，而另一个账户余额尚未修改。因此，所有账户余额的总额处于不一致状态。事务完成以后，账户余额的总额再次恢复到一致状态。
 - 隔离性 Isolation 对数据修改的所有并发事务都是隔离的，不应该以任何方式依赖或影响其他事务
 - 持久性 Durability 不管系统是否发生故障，只要事务处理完成，对数据库所做的改变就是永久的。

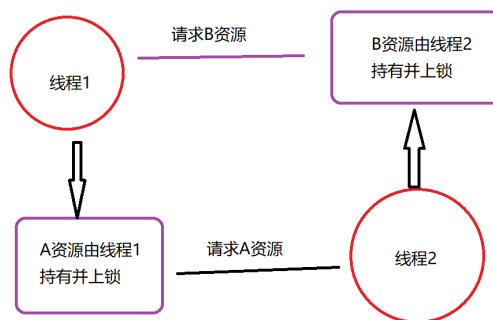
数据库的隔离级别应用和优缺点

- 数据库的四种隔离级别
 - Read Uncommitted 读未提交：级别最低，任何事情都可能发生
 - 表示还没有提交的操作被读取到
 - Read Committed 读已提交：避免脏读发生
 - 另一事务改变了数据库信息并且已经成功提交
 - Repeatable Read：避免脏读，不可重复读发生（MYSQL 默认隔离级别）
 - 一旦事务开始，其他事务不能对该记录进行修改
 - 但其他事务可以新增记录insert，会出现幻读
 - Serializable：避免脏读，不可重复读，幻读发生
 - 代价花费最高性能很低
- 数据库的四种隔离级别优缺点

- 读未提交是级别最低的隔离，仅保证不读取物理损坏的数据
- 读已提交保证一个并行事务不会读到另一个事务未提交的修改，避免了脏读。
- 可重复读，保证了在任何情况下，同一个事务中前后数次所读取的数据都是一致的

2.什么情况下会发生死锁，如何解决死锁

死锁是什么，如何产生



当多个进程访问同一个数据库，其中每个进程拥有的锁都是其他进程所需的，比如A进程执行完成需要等B释放资源，B又等A释放资源，就互相形成死锁。

MySQL有三种锁的级别：**页级、表级、行级。**

- 一般出现死锁都是行级锁 **死锁产生的必要条件**

- 1.互斥：资源互斥使用
- 2.请求与保持：多个进程保持一定资源又请求新的资源
- 3.不剥夺：资源不可被剥夺
- 4.循环等待：多个进程循环等待

数据库产生死锁：**当表进行了分区并且ALTER TABLE的 LOCK_ESCALATION 设置设置为AUTO时也会发生死锁**

如何解决死锁问题

- 破坏请求和保持条件
 - 一次性申请全部资源，用完释放
 - 只申请初期运行资源，运行完释放资源再申请下一部分资源
- 破坏不剥夺条件
 - 若一个进程获得不可抢占资源，则再申请新资源时如果失败，则释放所有资源。之后需要的话再次申请
- 破坏循环等待条件

- 对资源进行排号，按照序号递增的条件申请资源