## 题目 01- 请你说一说 MySQL 的锁机制

**要求：**

**按照锁的粒度，锁的功能来分析？**

从锁的粒度来分。分为全局锁，表锁，和行锁。

从锁的功能分，分为读锁（共享锁），写锁（排他锁）

全局锁：锁的是库，让整个库只能读，一般用于整库备份

表锁：锁的是某个表。又分为读锁和写锁。

表读锁中，当前事务可select，不可delete update insert，其他事务可以select。

表写锁中，当前事务select,insert,delete,update都可以，其他事务都不可以。

元数据锁，不需要显式指定，Mysql默认支持。

一般情况下，select,insert,delete,update加读，create，alter表结构加写锁。

行锁：锁的是某行。innodb存储引擎的行锁锁的是索引，如果没有索引，行锁会退化成表锁。

innodb 行级锁又分为：

记录锁，锁索引树上的一个节点。

间隙锁，锁索引节点的前后，开区间。间隙锁可以解决幻读问题，因此RC隔离级别没有间隙锁。

临键，锁索引的前后，左开右闭区间。

插入意向锁，间隙锁的一种，insert时产生。

**什么是死锁，为什么会发生，如何排查？**

死锁出现的标志是某条sql语句无限等待某个锁。死锁出现的原因是循环等待，事务A持有锁1等待锁2的释放，事务B持有锁2等待锁1的释放。

排查：SHOW ENGINE INNODB STATUS;

**行锁是通过加在什么上完成的锁定？**

MySQL 中 InnoDB 引擎中是加在索引上，锁住的是索引或者索引区间。一个表上的索引不能过多，索引多了可能会死锁

**详细说说这条 SQL 的锁定情况：delete from tt where uid = 666;**

1、如果uid是主键，在主键索引上加行级写锁（记录锁）。

2、如果uid是非主键唯一索引，在uid和主键索引上都加行级写锁（记录锁）。

3、如果uid是非唯一索引，RC隔离级别，在uid和主键索引的若干条记录上都加行级写锁（记录锁）。RR级别及以上，都要满足可重复读，因此会多出一个uid上的间隙锁。

4、uid如果不是索引，RC隔离级别，会先在全表加行级写锁（记录锁），然后不满足的条件释放锁，满足的保留。RR隔离级别及以上，类似上一条，也会多出间隙锁。

## 题目 02- 请你说一说 MySQL 的 SQL 优化

通过执行计划explain来分析现有语句的状态，看这条sql的效率高不高，需不需要优化。

1、查询时尽量避免使用 select \*。

2、尽量避免在where子句中使用 or 作为查询条件。

3、where条件中尽量不要出现null值的比较。

5、避免在where中使用 != 或 <> 操作符。

6、使用like查询条件时应尽量避免前缀使用‘%’。

7、执行查询时尽量采用最左匹配原则

8、避免在查询条件中使用一些内置的SQL函数

9、假如in表达式后面的数据太多(一般不建议超过200），尽量避免使用in作为查询条件

10、当有多个查询条件、分组条件、排序条件时，尽量使用联合索引

11、表连接时优先使用内连接(inner join)，使用小表驱动大表