1. 说一下 MySQL 数据库存储的原理?

答: 过程是一个可编程的函数，它在数据库中创建并保存。 它可以有 SQL 语句和一些特殊的控制结构组成。当希望在不同的应用 程序或平台上执行相同的函数，或者封装特定功能时，存储过程是非 常有用的。数据库中的存储过程可以看做是对编程中面向对象方法的 模拟。它允许控制数据的访问方式。存储过程通常有以下优点:

1)存储 能实现较快的执行速度。

2)存储过程允许标准组件是编程。

3)存储过程可以用流控制语句编写，有很强的灵活性，可以完成 复杂的判断和较复杂的运算。

4)存储 可被作为一种安全机制来充分利用。

5)存储过程能过减少网络流量。

2. 事务的特性

答:1、原子性( ):事务中的全部操作在数据库中是不 可分割的，要么全部完成，要么均不执行。

2、一致性( ):几个并行执行的事务，其执行结果必 须与按某一顺序串行执行的结果相一致。

3、隔离性( ):事务的执行不受其他事务的干扰，事务 执行的中间结果对其他事务必须是透明的。

4、持久性( ):对于任意已提交事务，系统必须保证该 事务对数据库的改变不被丢失，即使数据库出现故障

4. 数据库索引

数据库索引，是数据库管理系统中一个排序的数据结构，以协助 快速查询、更新数据库表中数据。索引的实现通常使用 B\_TREE。 B\_TREE 索引加速了数据访问，因为存储引擎不会再去扫描整张表得 到需要的数据;相反，它从根节点开始，根节点保存了子节点的指针， 存储引擎会根据指针快速寻找数据。

5. 数据库怎么优化查询效率

1.储存引擎选择:如果数据表需要事务处理，应该考虑使用 InnoDB，因为它完全符合 ACID 特性。如果不需要事务处理，使用默 认存储引擎 MyISAM 是比较明智的

2.分表分库，主从，

3.对查询进行优化，要尽量避免全表扫描，首先应考虑在 where 及 order by 涉及的列上建立索引

4.应尽量避免在 where 子句中对字段进行 null 值判断，否则 将导致引擎放弃使用索引而进行全表扫描

5.应尽量避免在 where 子句中使用 != 或 <> 操作符，否则将 引擎放弃使用索引而进行全表扫描

6.应尽量避免在 where 子句中使用 or 来连接条件，如果一个 字段有索引，一个字段没有索引，将导致引擎放弃使用索引而进行全 表扫描

7.Update 语句，如果只更改 1、2 个字段，不要 Update 全部字 段，否则频繁调用会引起明显的性能消耗，同时带来大量日志

8.对于多张大数据量(这里几百条就算大了)的表 JOIN，要先 分页再 JOIN，否则逻辑读会很高，性能很差。

6. 数据库优化方案

1. 优化索引、SQL 语句、分析慢查询;

2. 设计表的时候严格根据数据库的设计范式来设计数据库;

3. 使用缓存，把经常访问到的数据而且不需要经常变化的数据 放在缓存中，能

节约磁盘 IO;

4. 优化硬件;采用 SSD，使用磁盘队列技术(RAID0,RAID1,RDID5) 等;

5. 采用 MySQL 内部自带的表分区技术，把数据分层不同的文件， 能够提高磁

盘的读取效率;

6. 垂直分表;把一些不经常读的数据放在一张表里，节约磁盘 I/O;

7. 主从分离读写;采用主从复制把数据库的读操作和写入操作 分离开来;

8. 分库分表分机器(数据量特别大)，主要的的原理就是数据 路由;

9. 选择合适的表引擎，参数上的优化;

10. 进行架构级别的缓存，静态化和分布式;

11. 不采用全文索引;

12. 采用更快的存储方式，例如 NoSQL 存储经常访问的数

7，MySQL 集群的优缺点

优点:

a) 99.999%的高可用性 b)快速的自动失效切换 c)灵活的分布式体系结构，没有单点故障 d)高吞吐量和低延迟 e)可扩展性强，支持在线扩容

缺点:a)存在很多限制，比如:不支持外键 b)部署、管理、配置很复杂c)占用磁盘空间大，内存大d)备份和恢复不方便e)重启的时候，数据节点将数据 load 到内存需要很长时间

8. 你用的 mysql 是哪个引擎，各引擎间有什么区别

主要 MyISAM 与 InnoDB 两个引擎，其主要区别如下:

1. InnoDB 支持事务，MyISAM 不支持，这一点是非常之重要。 事务是一种高 级的处理方式，如在一些列增删改中只要哪个出错还可以回滚还 原，而 MyISAM就不可以了

二、MyISAM 适合查询以及插入为主的应用，InnoDB 适合频繁修 改以及涉及到 安全性较高的应用;

三、InnoDB 支持外键，MyISAM 不支持;

四、MyISAM 是默认引擎，InnoDB 需要指定;

五、InnoDB 不支持 FULLTEXT 类型的索引;

六、InnoDB 中不保存表的行数，如 select count(\*) from table 时，InnoDB;需要 扫描一遍整个表来计算有多少行，但是 MyISAM 只要简单的读出 保存好的行数即 可。注意的是，当 count(\*)语句包含 where 条件时 MyISAM 也 需要扫描整个表;

七、对于自增长的字段，InnoDB 中必须包含只有该字段的索引， 但是在 MyISAM 表中可以和其他字段一起建立联合索引;

八、清空整个表时，InnoDB 是一行一行的删除，效率非常慢。 MyISAM 则会重 建表;

九、InnoDB 支持行锁(某些情况下还是锁整表，如 update table set a=1 where user like '%lee%'

9. 怎样结局数据库高并发的问题?

解决数据库高并发:

分表分库

数据库索引redis

缓存数据库

读写分离

负载均衡集群:将大量的并发请求分担到多个处理节点。由于单 个处理节点的故障不影响整个服务，负载均衡集群同时也实现了高可用性