

1. 在表中建立索引，优先考虑 where group by 使用到的字段。
2. 查询时尽量避免使用select *，只查询需要用到到的字段。
3. 避免在where子句中使用关键字两边都是%的模糊查询，尽量在关键字后使用模糊查询。
4. 尽量避免在where子句中使用IN 和NOT IN。
 - 优化：能使用between就不用in
 - 在子查询中使用exists 子句
5. 尽量避免使用or，优化：可以用union代替or。
6. 尽量避免在where子句中使用表达式操作。
7. 尽量避免在where子句中使用null判断，优化：给字段添加默认值，对默认值判断。
8. 尽量不要在where条件中等号的左侧进行表达式.函数操作。
9. 尽量避免使用where 1=1，优化：用代码拼接sql，需要where的地方加where，需要and的地方加and
10. 尽量避免大事务操作，提高并发能力。
11. 一个表中的索引最好不要超过6个。
12. 应尽量避免在where子句中使用 != 或 <>。
13. 在使用索引字段作为条件时，如果该索引是复合索引，那么必须使用到该索引中的第一个。字段作为条件时才能保证系统使用该索引，否则该索引将不会被使用，并且应尽可能的让字段顺序与索引顺序相一致。
14. 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。

15. 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。

16. 尽量使用数字型字段，若只含数值信息的字段尽量不要设计为字符型，这会降低查询和连接的性能，并会增加存储开销。

这是因为引擎在处理查询和连接时会逐个比较字符串中每一个字符，而对于数字型而言只需要比较一次就够了。

17. 尽量避免使用游标，因为游标的效率较差，如果游标操作的数据超过1万行，那么就应该考虑改写。游标的一个常见用途就是保存查询结果，以便以后使用。

游标的结果集是由SELECT语句产生，如果处理过程需要重复使用一个记录集，那么创建一次游标而重复使用若干次，比重复查询数据库要快的多。

18. 尽量避免向客户端返回大数据量，若数据量过大，应该考虑相应需求是否合理。

19. 什么时候【要】创建索引？

表经常进行 SELECT 操作。

表很大(记录超多)，记录内容分布范围很广。

列名经常在 WHERE 子句或连接条件中出现。

什么时候【不要】创建索引？

表经常进行 INSERT/UPDATE/DELETE 操作。

表很小(记录超少)。

列名不经常作为连接条件或出现在 WHERE 子句中。

20. 索引优缺点：

索引加快数据库的检索速度。

索引降低了插入、删除、修改等维护任务的速度。

唯一索引可以确保每一行数据的唯一性，通过使用索引，可以在查询的过程中使用优化隐藏器，提高系统的性能。

索引需要占物理和数据空间，另外虽然索引可以提高查询速度，但是它们也会导致数据库系统更新数据的性能下降，因为大部分数据更新需要同时更新索引。

注：资料来源于网络。