

MariaDB

对于加锁的处理，可以说就是数据库对于事务处理的精髓所在

优化点

1. 关于对MySQL的SQL_NO_CACHE的理解和用法举例
2. utf8mb4中的 **mb4** 表示**max byte 4**，即最大4个字节；
3. utf8mb4_**_ci中的ci表示**case ignore**，即大小写不敏感；
utf8mb4_**_cs中的cs表示**case sensitive**，即大小写敏感；
4. 关于索引 + 复合索引 + LIMIT

```
o -- column_a, column_b为联合索引
-- 以下四种情况，哪些会使用索引，哪些不会？
SELECT * FROM table ORDER BY column_a, column_b;
SELECT * FROM table ORDER BY column_a DESC, column_b;
SELECT * FROM table ORDER BY column_a, column_b DESC;
SELECT * FROM table ORDER BY column_a DESC, column_b DESC;

-- 添加了limit之后呐？
SELECT * FROM table ORDER BY column_a, column_b LIMIT 1;
SELECT * FROM table ORDER BY column_a DESC, column_b LIMIT 1;
SELECT * FROM table ORDER BY column_a, column_b DESC LIMIT 1;
SELECT * FROM table ORDER BY column_a DESC, column_b DESC LIMIT 1;

-- 那“ LIMIT 999999999”呐？
SELECT * FROM table ORDER BY column_a, column_b LIMIT 999999999;
SELECT * FROM table ORDER BY column_a DESC, column_b LIMIT 999999999;
SELECT * FROM table ORDER BY column_a, column_b DESC LIMIT 999999999;
SELECT * FROM table ORDER BY column_a DESC, column_b DESC LIMIT 999999999;

-- 如果是主键排序呐？如下：假设table的主键是id
SELECT * FROM table ORDER BY id DESC LIMIT 1或999999999;
```

知识点

1. InnoDB 的行锁是通过给索引上的索引项加锁来实现的；
2. 只有通过索引条件进行数据检索，InnoDB 才使用行级锁，否则，InnoDB 将使用表锁（锁住索引的所有记录）；
3. InnoDB 的表锁是通过对所有行加行锁实现的。
4. MySQL 会为这张表中所有行加行锁，没错，是所有行。但是呢，在加上行锁后，MySQL 会进行一遍过滤，发现不满足的行就释放锁，最终只留下符合条件的行。虽然最终只为符合条件的行加了锁，但是这一锁一释放的过程对性能也是影响极大的。 [MySQL 事务隔离级别和实现原理-知乎](#)
5. 意向锁（IX/IS）是一种不与行级锁冲突表级锁。**IX**，**IS**是表级锁，不会和行级的**X**，**S**锁发生冲突。只会和表级的**X**，**S**发生冲突
6. Gap（[间隙锁](#)）只在 RR 事务隔离级别存在。因为幻读问题是在 RR 事务通过临键锁和 MVCC 解决的，而临键锁=间隙锁+记录锁，所以间隙锁只在 RR 事务隔离级别存在。； [MySQL InnoDB 锁](#)
7. innodb不支持FULLTEXT类型的全文索引，但是innodb可以使用sphinx插件支持全文索引，并且效果更好。（ [sphinx](#) 是一个开源软件，提供多种语言的API接口，可以优化mysql的各种查询）
8. 清空表数据 参考[TRUNCATE TABLE](#)；

优秀博文

[Innodb中的事务隔离级别和锁的关系 | 何为幻读？ | 更多关于ORDER BY](#)
[MYSQL排序按照自定义的顺序排序 | 图文并茂讲解Mysql事务实现原理](#)

```
select * from A order by IF(ISNULL(a),1,0),a desc
```

[行级锁、表级锁、索引锁 | MySQL的锁机制 - 记录锁、间隙锁、临键锁 | 全面了解mysql锁机制（InnoDB）与问题排查 | SSD基本原理 | 聊聊MVCC和Next-key Locks | 数据库MVCC如何解决可重复读问题？ | DATETIME数据类型简介 | MySQL InnoDB 锁 | MYSQL MVCC 实现原理 | MySQL到底是怎么解决幻读的？ | mysql myisam vs innodb | 详解MySQL的InnoDB中意向锁的作用 | mysql自增长联合主键](#)