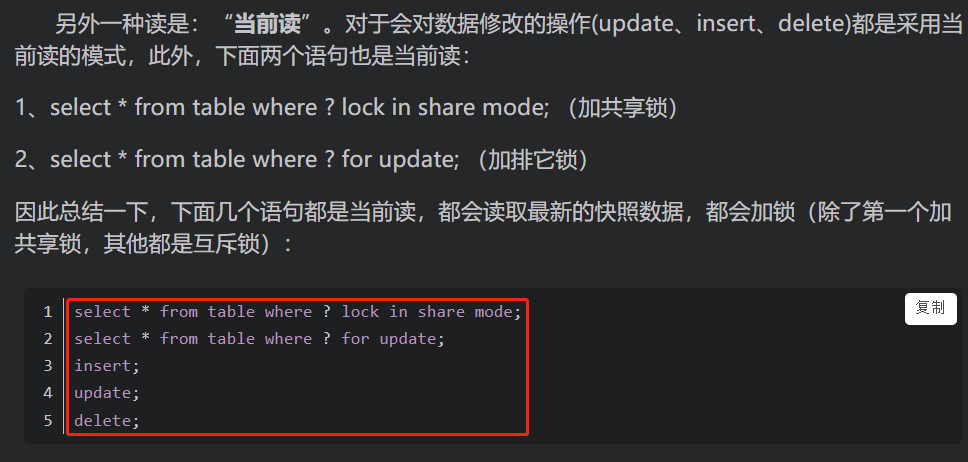
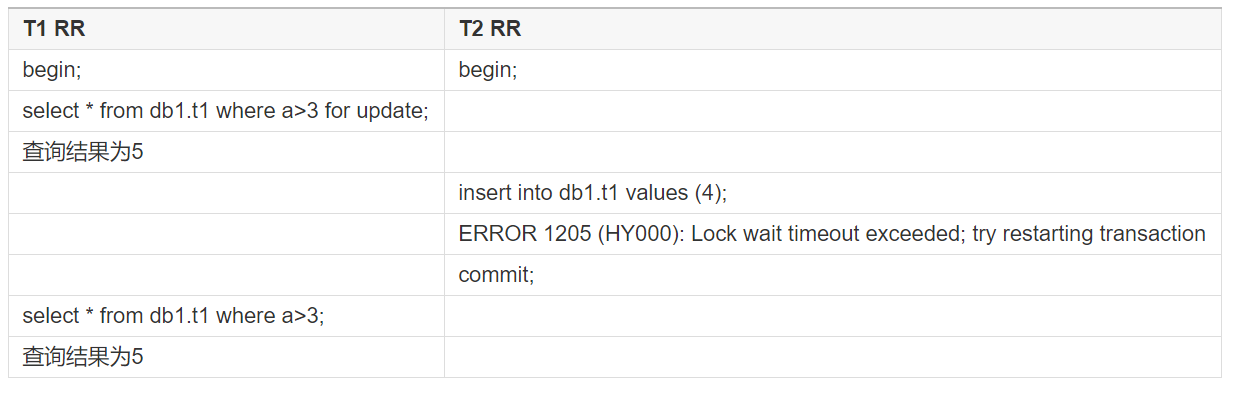


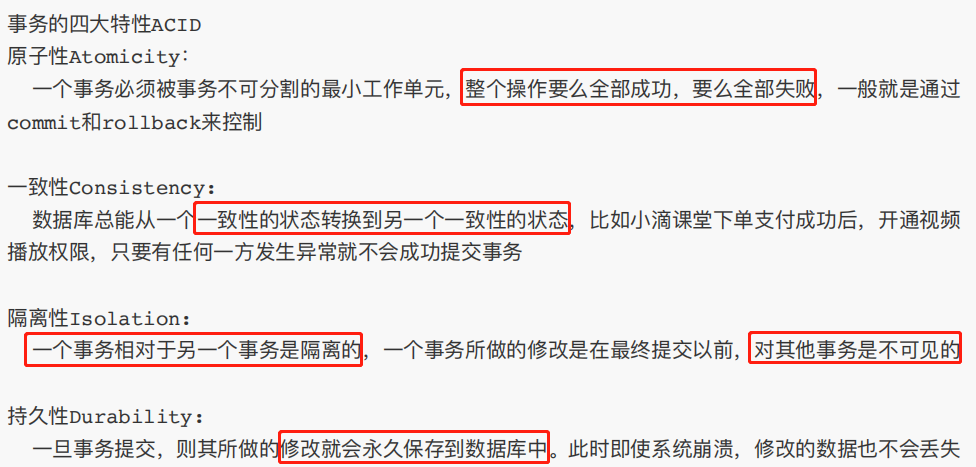
所谓的快照读就是事务只执行select语句

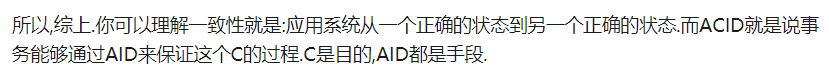


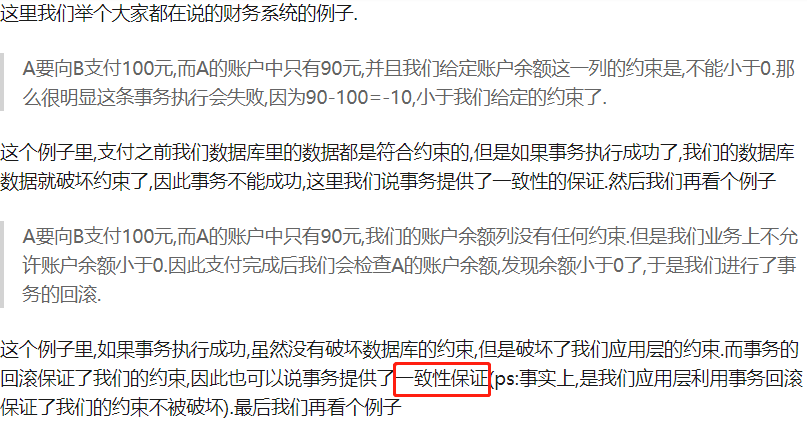


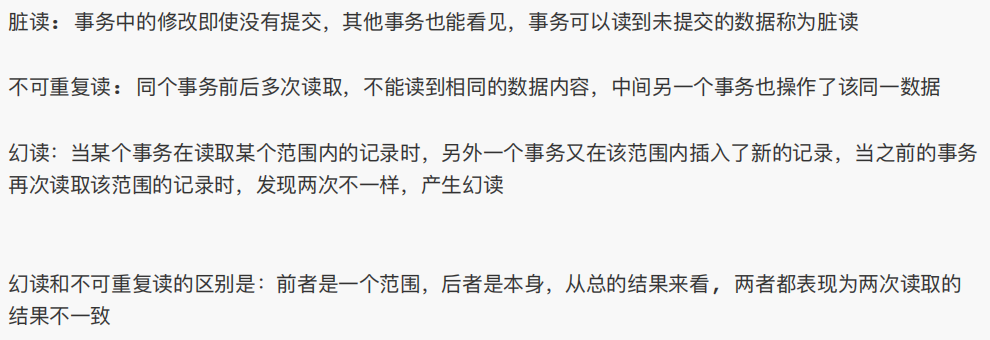


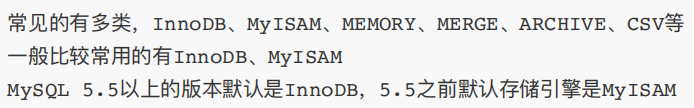










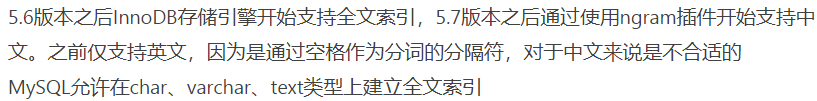


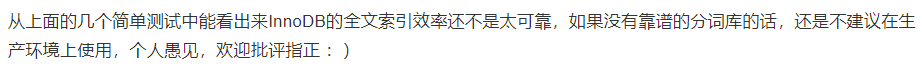
mysql的存储引擎 innodb和myisam有什么区别，应该怎么选择

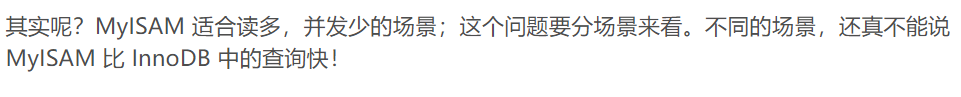


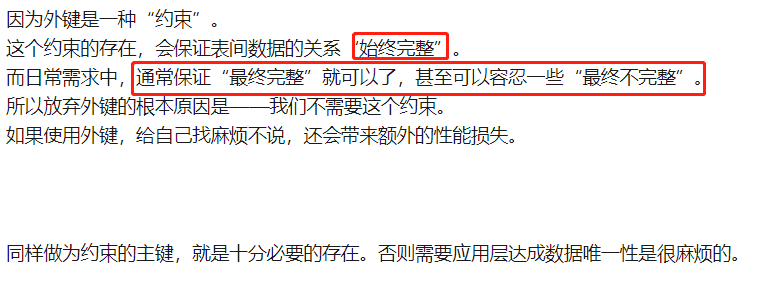
（物理外键）

Innodb会把每条语句都弄成事务









出团表，出团信息，出团表有着子表，出团人员（人员信息）。两者物理外键的话，出团人数不够，需要迁移到另一个出团表时，需要，出团人员的pid都修改完成之后再删除出团表。而逻辑删除则可以直接删除出团表，然后再进行剩下的操作

主键索引默认唯一+非空

一个表可以有多个主键索引

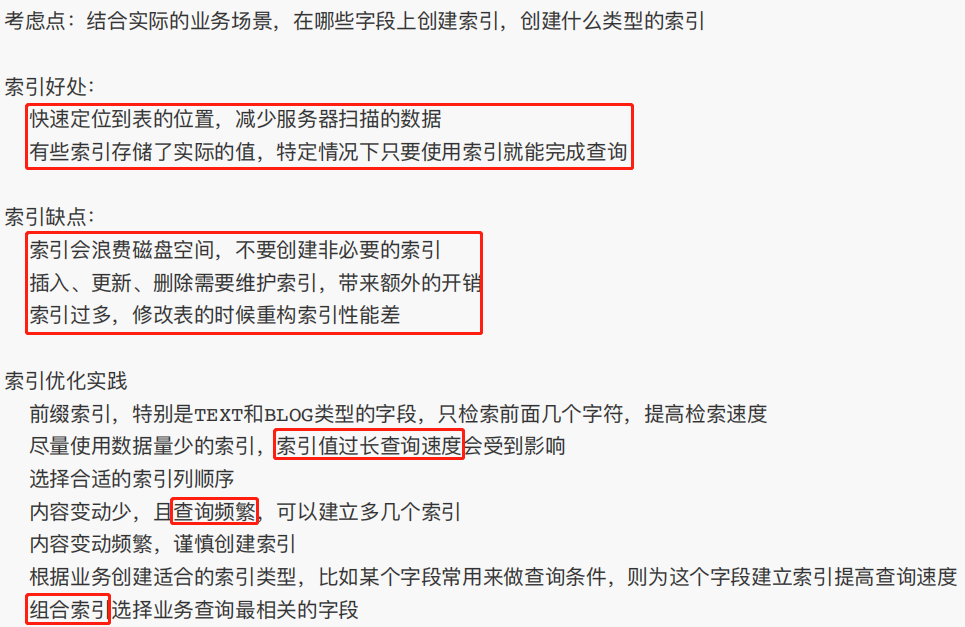




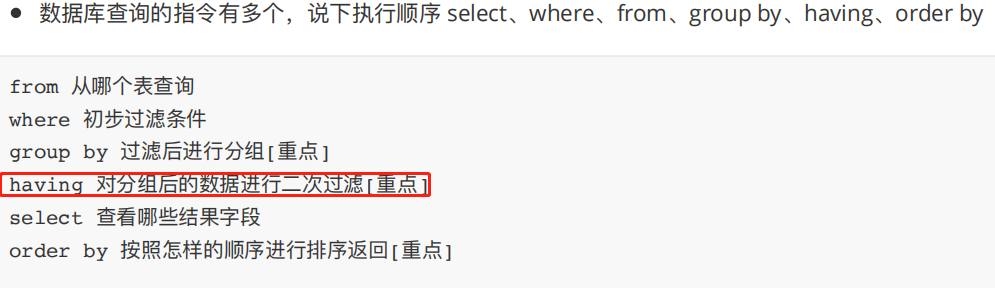


（select的数据在索引中）覆盖索引不需要回表操作，效率高

单一职责，mysql不适合太复杂的查询，效率也不好

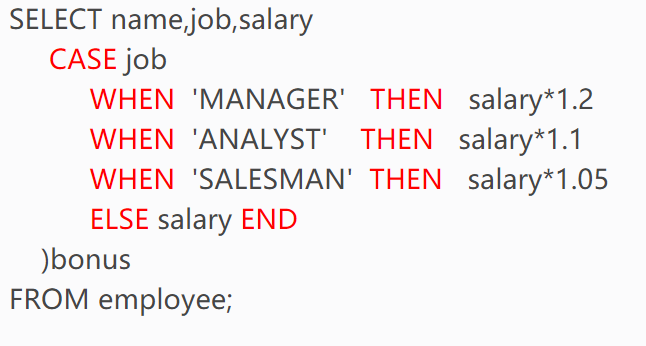


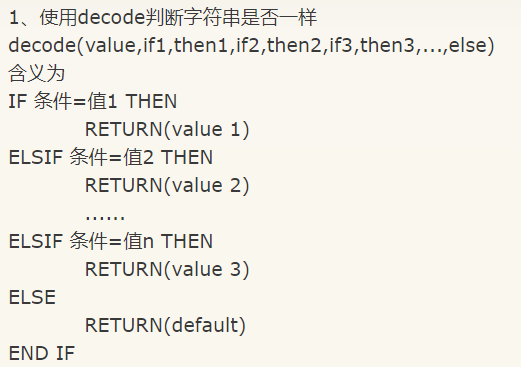
维护索引会浪费性能，索引列不能太多

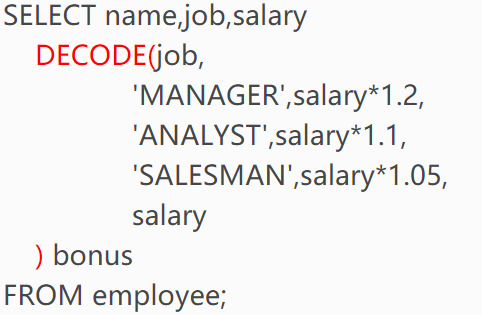


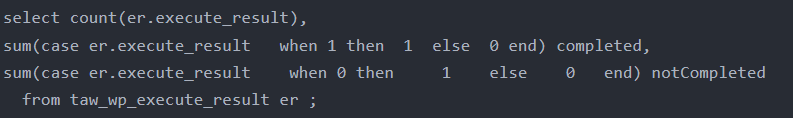
HAVING单独使用类似where

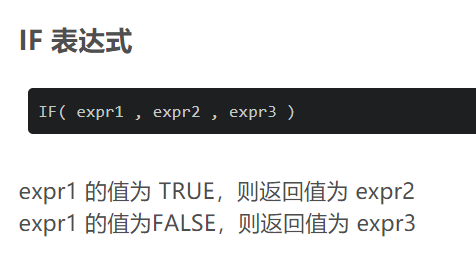








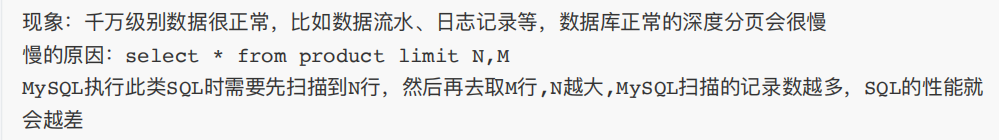




select name from role group by name having sum(if(name="supremecsp",1,0))>1;



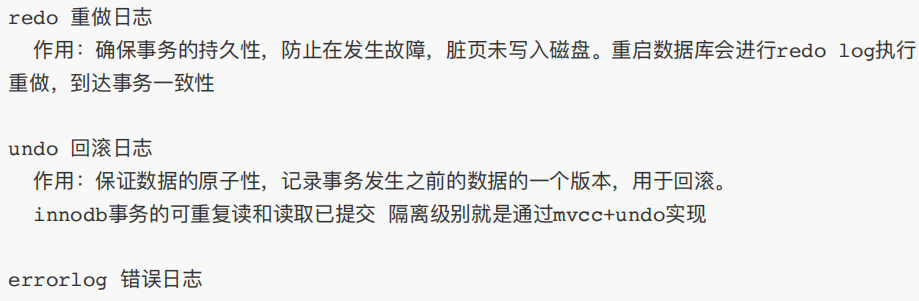
timestamp更省空间,超过时间点就插不进去了

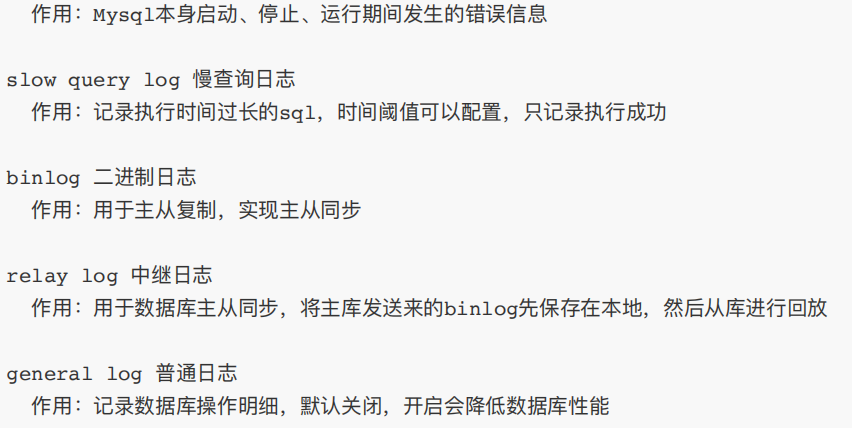


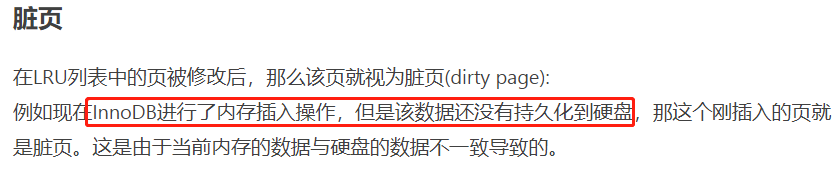


开启慢查询⽇志，定期分析慢查询⽇志

不要使用select \*









使用json就是违反了第一范式



主键是业务主键而不是伪主键

一笔订单只有一个用户，用户只依赖订单号

商家把名字改了，那么还要把订单表修改，就要多处更新，冗余

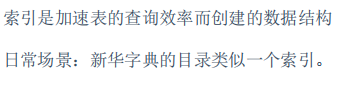
（冗余少联表）

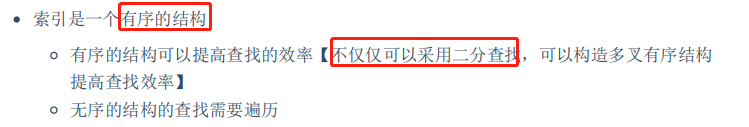


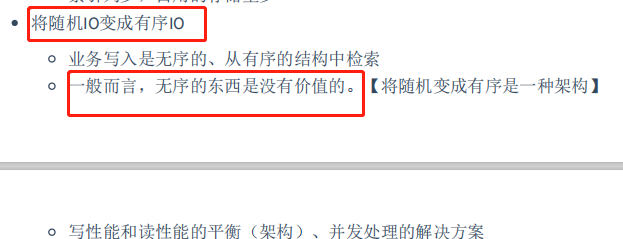
用户id只依赖于订单号，不符合第二范式，是冗余字段，但是拿到订单数据需要多次使用到订单号，此时就允许冗余字段的存在，减少联表操作

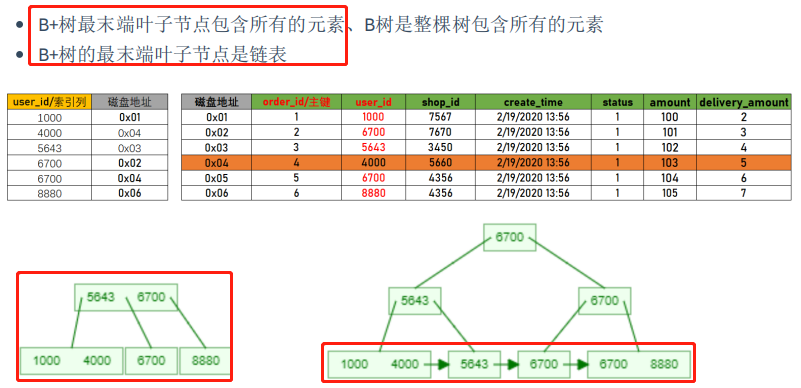


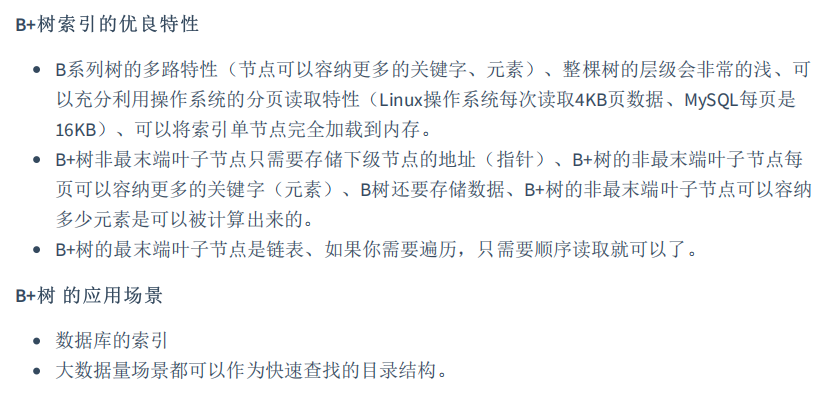
单表进行缓存非常方便，尽量业务语言join

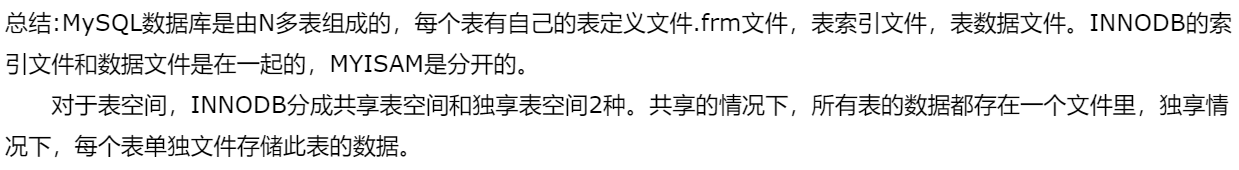


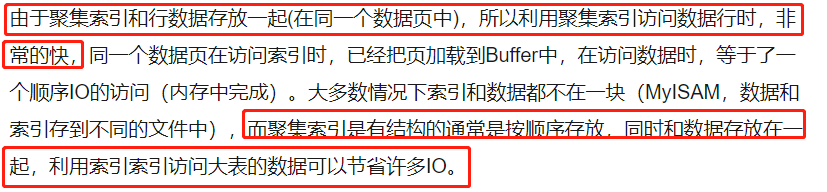


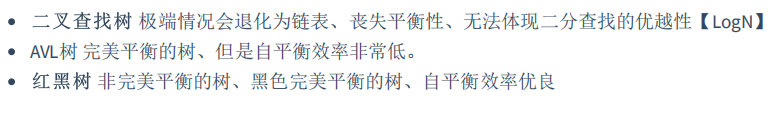


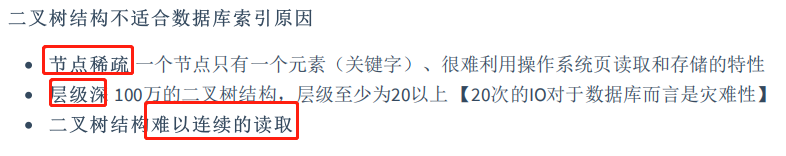




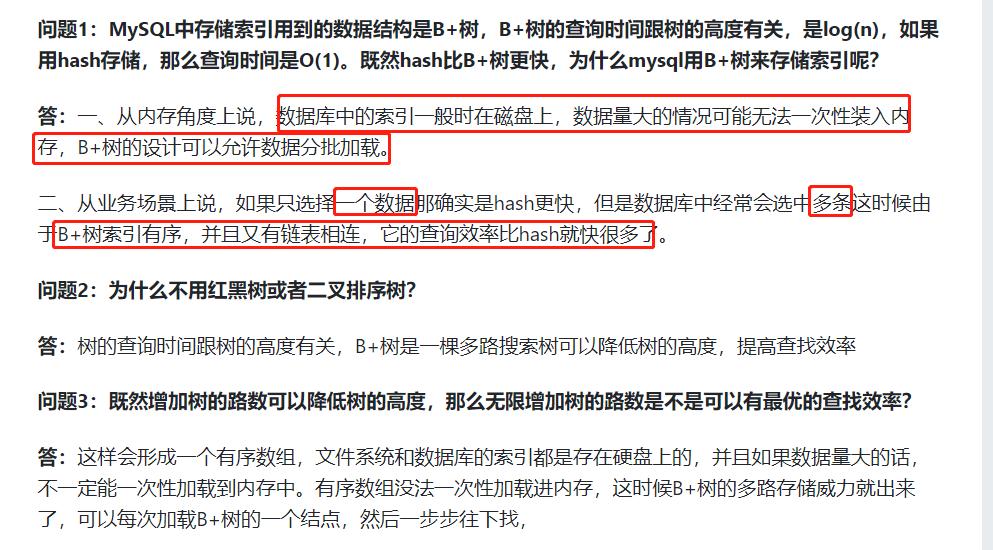






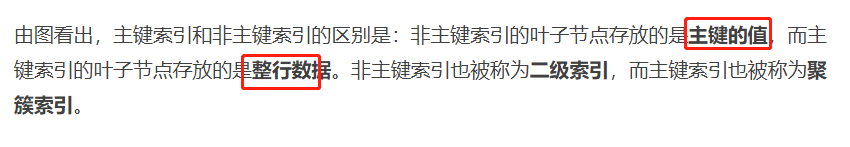


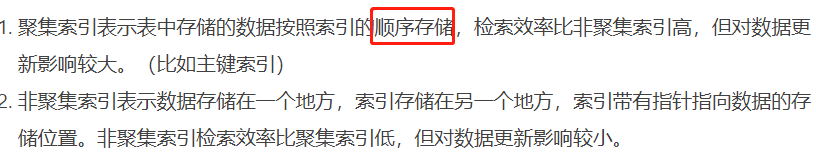
B树自平衡性差，所有节点都分散在各处，插入可能会造成旋转，读取也不太方便，没有顺序



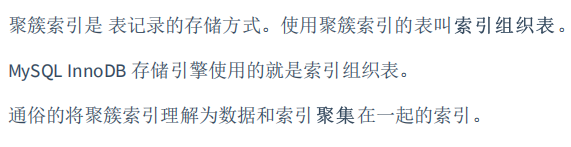
一次IO读取4kb,若是页数据量太多的话，会增加IO次数，影响性能

Mysql B+树底层是双向链表





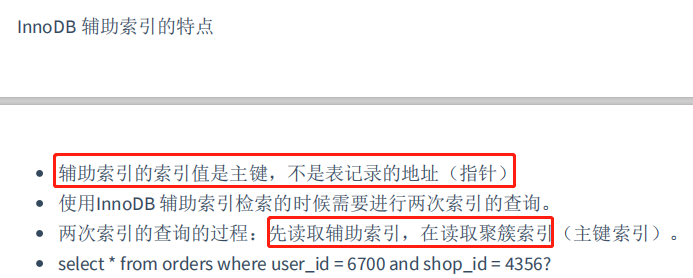
非聚簇索引叶子节点存的主键值

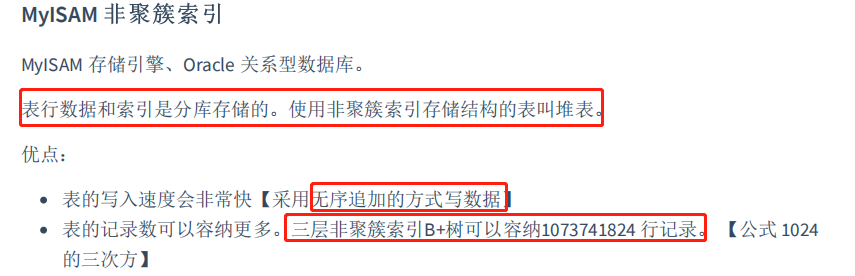


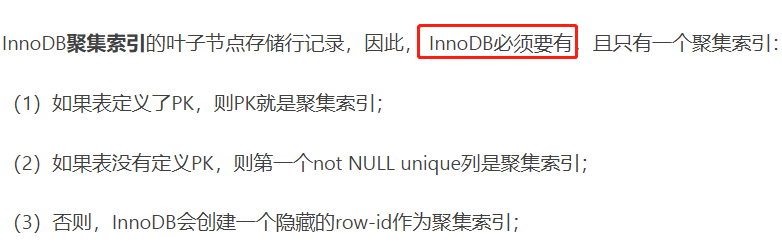
伪主键，自增主键好处，聚簇索引插入时往后节点插入，表结构变化不大



1K等于1024字节，一字节等于8bit









覆盖索引可以准确拿到主键直接到聚簇索引拿取数据