

# sql

(4)对于有外键约束的表，我们在导入数据之前也可以忽略外键检查，因为innodb的外键是即时检查的，所以对导入的每一行都会进行外键的检查。

```
set foreign_key_checks = 0;
load data .....
set foreign_key_checks = 1;
```

## 2.优化INSERT语句

(1)如果同时从同一客户端插入大量数据，应该尽量使用多个值的表的INSERT 语句，这种方式将大大减少客户端与数据库服务器之间的连接，关闭等消耗，使得效率比分开执行的单个INSERT语句快(大部分情况下，使用多个值表的INSERT语句能比单个INSERT语句快上好几倍)，比如下面一次插入多行：

```
INSERT INTO VALUES ('yayun',23),('tom',26),('atlas',32),('david',25).....
```

(2)插入延迟。如果从不同客户端插入很多行，可以通过使用INSERT DELAYED语句得到更高的速度。DELAYED的意思是让INSERT语句马上执行，其实数据都被放在内存的队列中，并没有真正写入磁盘，这比每条语句分别插入要快的多；LOW\_PRIORITY刚好相反，在所有其他用户对表的读写完成后才进行插入。

(3)将索引文件和数据文件放在不同的磁盘（利用建表中的选项）

(4)如果进行批量插入，可以通过增加bulk\_insert\_buffer\_size 变量值的方法来提高速度，这只对MyISAM表有用。

(5)当从一个文本文件装载一个表时，使用LOAD DATA INFILE。通常比使用很多的INSERT语句快。

## 无法使用索引的情况

(1).以%开头的like查询

(2).数据类型出现隐式转换的时候也不会使用索引，特别是当列类型是字符串，那么一定记得在where条件中把字符串常量值用引号引起来，否则即便这个列上有索引，MySQL也不会用到，因为MySQL默认把输入的常量值进行转换以后才进行检索

(3).复合索引的情况下，如果查询条件不包含索引列的最左边部分，即不满足最左前缀原则，则不会使用索引

(4).如果mysql估计使用索引扫描比全表扫描更慢，则不使用索引。(扫描数据超过30%，都会走全表)

(5).用or分割开的条件，如果 or前的条件中的列有索引，而后面的列中没有索引，那么涉及的索引都不会被用到

(6).字段使用函数，将无法使用索引

(7).Join 语句中 Join 条件字段类型不一致的时候 MySQL 无法使用索引

添加一个字段: alter table students drop abc int(8) comment '新添加' default '123';

清空表: truncate students