

# 阿里巴巴Java开发手册数据库部分的一些最佳实践总结

## 模糊查询

对于模糊查询阿里巴巴开发手册这样说到：

**【强制】** 页面搜索严禁左模糊或者全模糊，如果需要请走搜索引擎来解决。

说明:索引文件具有 B-Tree 的最左前缀匹配特性，如果左边的值未确定，那么无法使用此索引。

## 外键和级联

对于外键和级联，阿里巴巴开发手册这样说到：

**【强制】** 不得使用外键与级联，一切外键概念必须在应用层解决。

说明:以学生和成绩的关系为例，学生表中的 student\_id 是主键，那么成绩表中的 student\_id 则为外键。如果更新学生表中的 student\_id，同时触发成绩表中的 student\_id 更新，即为级联更新。外键与级联更新适用于单机低并发，不适合分布式、高并发集群;级联更新是强阻塞，存在数据库更新风暴的风险;外键影响数据库的插入速度

为什么不要用外键呢？大部分人可能会这样回答：

1. **增加了复杂性：** a.每次做DELETE 或者UPDATE都必须考虑外键约束，会导致开发的时候很痛苦,测试数据极为不方便;b.外键的主从关系是定的，假如那天需求有变化，数据库中的这个字段根本不需要和其他表有关联的话就会增加很多麻烦。
2. **增加了额外工作：** 数据库需要增加维护外键的工作，比如当我们做一些涉及外键字段的增，删，更新操作之后，需要触发相关操作去检查，保证数据的的一致性和正确性，这样会不得不消耗资源；（个人觉得这个不是不用外键的原因，因为即使你不使用外键，你在应用层面也还是要保证的。所以，我觉得这个影响可以忽略不计。）
3. 外键还会因为需要请求对其他表内部加锁而容易出现死锁情况；
4. **对分库分表不友好：** 因为分库分表下外键是无法生效的。
5. ....

我个人觉得上面这种回答不是特别的全面，只是说了外键存在的一个常见的问题。实际上，我们知道外键也是有很多好处的，比如：

1. 保证了数据库数据的一致性和完整性；
2. 级联操作方便，减轻了程序代码量；
3. ....

所以说，不要一股脑的就抛弃了外键这个概念，既然它存在就有它存在的道理，如果系统不涉及分库分表，并发量不是很高的情况还是可以考虑使用外键的。

我个人是不太喜欢外键约束，比较喜欢在应用层去进行相关操作。

## 关于@Transactional注解

对于@Transactional事务注解，阿里巴巴开发手册这样说到：

【参考】@Transactional事务不要滥用。事务会影响数据库的QPS，另外使用事务的地方需要考虑各方面的回滚方案，包括缓存回滚、搜索引擎回滚、消息补偿、统计修正等。