

SQL语言 - SQL语句优化

负向查询不能使用索引

```
select name from user where id not in (1,3,4);
```

应该修改为:

```
select name from user where id in (2,5,6);
```

前导模糊查询不能使用索引

如:

```
select name from user where name like '%zhangsan'
```

非前导则可以:

```
select name from user where name like 'zhangsan%'
```

建议可以考虑使用 Lucene 等全文索引工具来代替频繁的模糊查询。

数据区分不明显的不建议创建索引

如 user 表中的性别字段, 可以明显区分的才建议创建索引, 如身份证等字段。

字段的默认值不要为 null

这样会带来和预期不一致的查询结果。

在字段上进行计算不能命中索引

```
select name from user where FROM_UNIXTIME(create_time) < CURDATE();
```

应该修改为:

```
select name from user where create_time < FROM_UNIXTIME(CURDATE());
```

最左前缀问题

如果给 user 表中的 username pwd 字段创建了复合索引那么使用以下SQL 都是可以命中索引:

```
select username from user where username='zhangsan' and pwd ='axsedflsd'
```

```
select username from user where pwd ='axsedflsd' and username='zhangsan'
```

```
select username from user where username='zhangsan'
```

但是使用

```
select username from user where pwd ='axsedflsd'
```

是不能命中索引的

如果明确知道只有一条记录返回

```
select name from user where username='zhangsan' limit 1
```

可以提高效率, 可以让数据库停止游标移动。

不要让数据库帮我们做强制类型转换

```
select name from user where telno=1872222222
```

这样虽然可以查出数据, 但是会导致全表扫描。

需要修改为:

```
select name from user where telno='1872222222'
```

如果需要进行 join 的字段两表的字段类型要相同

不然也不会命中索引。