



MySQL® 数据库设计与应用

索引



第3章内容

● 数据库操作

● 记录操作-插入、修改、删除记录

● 数据类型

● 索引及操作

● 数据表操作-创建、查看等

● course管理系统实例

● 数据表操作-修改表



- 理解索引
- 索引分类
- 索引设计原则
- 创建索引
- 删除索引



理解索引

1. 索引的本质是什么？
2. MySQL数据库中，数据是如何检索的？
3. 一个数据库表只能创建一个索引吗？



索引分类

- 普通索引 (index)
- 唯一性索引 (unique)
- 主键索引 (primary key)
- 全文索引 (fulltext)
- 空间索引 (spatial)
- 单列索引和组合索引



索引设计原则

- 索引并非越多越好
- 字段的离散度越高，该字段越适合选作索引的关键字
- 经常排序和分组的数据列、两个表的连接字段应该建立索引
- 更新频繁的字段不适合创建索引，不会出现在where子句中的字段不应该创建索引



»» 创建索引

- 在创建表的同时创建索引
- 在已有表上创建索引



在创建表的同时创建索引

create table 表名 (

字段名1 数据类型 [约束条件],

字段名2 数据类型 [约束条件],

...

[其他约束条件],

...

[unique | fulltext] index [索引名] (字段名 [(长度)]

[asc | desc])

) engine=存储引擎类型 default charset=字符集类型

// “[]” 表示可选项

// “()” 表示必选项

// “asc” 与 “desc” 为可选参数，分别表示升序与降序

//创建索引的类型



»» 在已有表上创建索引

方法一：

```
create [ unique | fulltext ] index 索引名 on 表名 ( 字段名 [(长度)] [ asc | desc ] )
```

创建普通索引示例

```
Create INDEX teacher_id_index ON teacher( teacher_id );
```

创建唯一性索引示例

```
Create UNIQUE INDEX teacher_id_index ON teacher( teacher_id);
```



»» 在已有表上创建索引

方法二：

```
alter table 表名 add [ unique | fulltext ] index 索引名 ( 字段名 [(长度)] [ asc | desc ] )
```



删除索引

drop index 索引名 on 表名



索引的优缺点

➤ 优点:

1. 可以大大加快数据库的检索速度。
2. 可以加速表之间的连接。
3. 在使用分组和排序字句进行检索时，同样可以显著减少查询中分组和排序的时间。
4. 通过使用索引，可以提高系统性能。

➤ 缺点:

1. 创建索引和维护索引需要耗费时间。
2. 索引是需要占用物理内存的。
3. 当对表中的数据进行增加、删除和修改的时候，索引也需要进行动态维护，这样就降低了数据库的维护速度。



总结与思考

思考：

- mysql是否允许在相同列上创建多个索引？
- 如果可以，对数据库的性能有影响吗？



谢谢！