

表设计规范

字段命名规范

- 1. 字段语义须完整，不允许出现像status这种不明确的字段，应改为check_status
- 2. 表示是否概念的用is开头，如is_deleted（现在使用lombok生成get set方法已经不会导致序列化问题）
- 3. 每个字段都需要加上注释，尤其是枚举含义的字段，在有新数据更新时记得及时维护
- 4. 避免使用关键字，如desc，enable等

字段类型设计

- 1. 主键类型一般情况下使用 unsigned int（可以存储42亿条数据）
使用unsigned的好处有标识数据特点（没有负数），提高空间利用率，在确定不会有负数出现的字段上都应加上此标识。
- 2. 整数类型按需求使用int、unsigned int、bigint，在java中分别对应Integer、Long、Long
- 3. 小数类型使用 decimal，使用 float 和 double有精度丢失问题
- 4. 布尔类型使用 tinyint，使用1和0标识是与否，命名使用"is_XX"
- 5. 固定长度字符串使用char(x)，非固定长度使用varchar(x)，x代表字符数（包括汉字）如果长度大于5000使用text类型并另建一张表用主键关联，避免影响效率
- 6. 表必备字段：create_time, create_user, update_time, update_user，这些字段由框架从会话下文中自动取值填入
- 7. 枚举类型使用unsigned tinyint，java对应类型为Integer
- 8. 状态应说明具体是什么状态，不能命名为status，应为“xx_status”，如"check_status","sale_status"

类型说明

类型	表示范围	占用字节	java对应类型
tinyint unsigned	无符号值：0 到 255	1	int
int unsigned	无符号值：0 到约 42.9 亿	4	long
bigint unsigned	无符号值：0 到约 10 的 19 次方	8	long

- > 5.0.3版本以下，比如 varchar(100)，指的是100字节，如果存放UTF8汉字时，只能存33个（每个汉字3字节）
- > 5.0.3版本及以上，比如varchar(100)，指的是100字符，无论存放的是数字、字母还是UTF8汉字

（每个汉字3字节）， 都可以存放100个

额外说明

为方便代码生成工具识别， 以下为特别规范

bool使用tinyint(1) unsigned ， 在java中对应boolean

枚举使用tinyint unsigned ， 在java中对应int