# 数据库设计

### 前言

中国计算机学会副理事长-王珊教授的《数据库系统概论》

范式是“符合某一种级别的关系模式的集合，表示一个关系内部各属性之间的联系的合理化程度

### 范式定义

**一张数据表的表结构所符合的某种设计标准的级别**

就像家里装修买建材，最环保的是E0级，其次是E1级，还有E2级等等。数据库范式也分为1NF ，2NF，3NF，BCNF，4NF，5NF。

**意义：**

范式是为了消除重复数据减少冗余数据，从而让数据库内的数据更好的组织

**特点：**

符合高一级范式的设计，必定符合低一级范式，例如符合2NF的关系模式，必定符合1NF。

### 第一范式

1NF： 字段是最小的的单元不可再分

地址表

错：

|  |  |  |
| --- | --- | --- |
| id | 联系人 | 联系地址 |
| 1 | 郭小闹 | 北京市朝阳区幸福大街33号 13671231111 |

有点对：

|  |  |  |  |
| --- | --- | --- | --- |
| id | 联系人 | 联系人电话 | 联系人地址 |
| 1 | 郭小闹 | 13671231111 | 北京市朝阳区幸福大街33号 |

非常对：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| id | 联系人 | 联系人电话 | 省 | 市 | 区 | 地址 |
| 1 | 郭小闹 | 13671231111 | 北京 | 北京 | 朝阳区 | 幸福大街33号 |

疑问？仅仅符合第一范式是否就可以了

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 商品id | 商品名称 | 分类号 | 分类名 | 价格 | 仓库 | 仓库负责人 |
| 1 | 茜茜 | 1 | 双肩 | 100 | A | 郭小闹 |
| 1 | 茜茜 | 2 | 单肩 | 80 | A | 郭小闹 |

更新异常：今天设计师想换个名字，叫 茜茜限量版， 工作人员要想着更新两次~

插入异常：设计部&市场部的矛盾；设计部新出了 茜茜6代， 但是价格还没定，给到电商人员录入，会出现 价格null的脏数据

综上 仅仅满足第一范式还是会出现问题~

### 第二范式

2NF：满足1NF,表中的字段必须完全依赖于全部主键而非部分主键

上表 主键【商品id&分类号】

如果只想获得分类名，已知分类号即可， 即产生部分依赖

所以修改成如下：

表1 商品表

|  |  |  |  |
| --- | --- | --- | --- |
| 商品id | 商品名称 | 仓库 | 仓库负责人 |
| 1 | 茜茜 | A | 郭小闹 |

表2 分类表

|  |  |
| --- | --- |
| 分类id | 分类名 |
| 1 | 双肩 |
| 2 | 单肩 |

表3 定价表

|  |  |  |
| --- | --- | --- |
| 商品id | 分类id | 价格 |
| 1 | 1 | 100 |
| 1 | 2 | 80 |

疑问： 修改仓库时，会不会有问题？

### 第三范式

3NF：满足2NF,非主键外的所有字段必须互不依赖，即非主键字段不能有传递性依赖

商品id -> 商品名称-> 仓库 -> 仓库负责人

仓库 - > 仓库负责人

原商品表拆分如下：

1，商品表

|  |  |  |
| --- | --- | --- |
| 商品id | 商品名 | 仓库 |
| 1 | 茜茜 | A |

2，仓库表

|  |  |
| --- | --- |
| 仓库 | 仓库负责人 |
| A | 郭小闹 |
| B | 王伟超 |

### 反范式

必要的冗余可增加查询效率 【分类表记录 分类总销售数】

必要的冗余可以还原数据真实性 【订单 冗余地址详细信息】

address表

1 上海

订单表

id 创建时间 联系地址 联系人

1 2019

2020年 我修改了 地址表 id 1地址， 请问您查询19年订单的时候，幻觉