

## 数据库设计范式

什么是范式：简言之就是，数据库设计对数据的存储性能，还有开发人员对数据的操作都有莫大的关系。所以建立科学的，规范的数据库是需要满足一些

规范的来优化数据数据存储方式。在关系型数据库中这些规范就可以称为范式。

什么是三大范式：

第一范式：当关系模式 R 的所有属性都不能在分解为更基本的数据单位时，称 R 是满足第一范式的，简记为 1NF。满足第一范式是关系模式规范化的最低要

求，否则，将有很多基本操作在这样的关系模式中实现不了。

第二范式：如果关系模式 R 满足第一范式，并且 R 得所有非主属性都完全依赖于 R 的每一个候选关键属性，称 R 满足第二范式，简记为 2NF。

第三范式：设 R 是一个满足第一范式条件的关系模式，X 是 R 的任意属性集，如果 X 非传递依赖于 R 的任意一个候选关键字，称 R 满足第三范式，简记为 3NF。

注：关系实质上是一张二维表，其中每一行是一个元组，每一列是一个属性

## 理解三大范式

### 第一范式

- 1、每一列属性都是不可再分的属性值，确保每一列的原子性
- 2、两列的属性相近或相似或一样，尽量合并属性一样的列，确保不产生冗余数据。

	Name	Code	Data Type	Length	Precision	P	F	M
1	留言编号	MsgId	int			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	留言人姓名	Name	nvarchar(30)	30		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	留言人座机	Telephone	varchar(20)	20		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	留言人手机	MobilePhone	varchar(20)	20		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	邮箱	Mail	varchar(50)	50		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	内容	Content	nvarchar(800)	800		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7	时间	Time	datetime			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8	标题	Title	nvarchar(50)	50		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9	地址	Address	nvarchar(80)	80		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

	Name	Code	Data Type	Length	Precision	P	F	M
	留言编号	MsgId	int			<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	留言人姓名	Name	nvarchar(30)	30		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	留言人座机	Telephone	varchar(20)	20		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	留言人手机	MobilePhone	varchar(20)	20		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	邮箱	Mail	varchar(50)	50		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	内容	Content	nvarchar(800)	800		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	时间	Time	datetime			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	标题	Title	nvarchar(50)	50		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	省	Province	nvarchar(20)	20		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	市	City	nvarchar(30)	30		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	详细地址	Address	nvarchar(80)	80		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

如果需求知道那个省那个市并按其分类，那么显然第一个表格是不容易满足需求的，也不符合第一范式。

	Name	Code	Data Type	Length	Precision	P	F	M
1	室号	RoomId	int			<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	物品1	ProName1	nvarchar(50)	50		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	物品1数量	proNum1	int			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	物品1价格	ProPrice1	money			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	物品2	ProName2	nvarchar(30)	30		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	物品2数量	proNum2	int			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7	物品2价格	ProPrice2	money			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

	Name	Code	Data Type	Length	Precision	P	F	M
1	室号	RoomId	int			<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	物品	ProName	nvarchar(50)	50		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	物品数量	proNum1	int			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	物品价					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

显然第一个表结构不但不能满足足够多物品的要求，还会在物品少时产生冗余。也是不符合第一范式的。

## 第二范式

每一行的数据只能与其中一列相关，即一行数据只做一件事。只要数据列中出现数据重复，就要把表拆分

开来。一张表只做一件事情；

冗余

	Name	Code	Data Type	Length	Precision	P	F	M
1	订单编号	orderId	varchar(30)	30		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	房间号	roomNum	int			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	联系人	name	nvarchar(30)	30		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	联系人电话	phone	varchar(15)	15		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	身份证号	cardNum	varchar(20)	20		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

一个人同时订几个房间，就会出来一个订单号多条数据，这样子联系人都是重复的，就会造成数据冗余。我们应该把他拆开来。

	Name	Code	Data Type	Length	Precision	P	F	M	▲
1	订单编号	= orderId	varchar(30)	30		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
2	房间号	roomNum	int			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
3	联系人编号	Peold	id			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

	Name	Code	Data Type	Length	Precision	P	F	M	▲
→	联系人编号	Peold	numeric			<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
2	联系人	name	nvarchar(30)	30		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
3	联系人电话	phone	varchar(15)	15		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
4	身份证号	cardNum	varchar(20)	20		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

这样便实现啦一条数据做一件事，不掺杂复杂的关系逻辑。同时对表数据的更新维护也更易操作。

### 第三范式

每个列只能和主键相关, 和其他列无关;

数据不能存在传递关系, 即没个属性都跟主键有直接关系而不是间接关系。像:  $a \rightarrow b \rightarrow c$  属性之间含有这样的关系, 是不符合第三范式的。

比如 Student 表 (学号, 姓名, 年龄, 性别, 所在院校, 院校地址, 院校电话)

这样一个表结构, 就存在上述关系。学号  $\rightarrow$  所在院校  $\rightarrow$  (院校地址, 院校电话)

这样的表结构, 我们应该拆开来, 如下。

(学号, 姓名, 年龄, 性别, 所在院校) — (所在院校, 院校地址, 院校电话)