数据库规范化

“数据库规范化，又称正规化、标准化，是数据库设计的一系列原理和技术，以减少数据库中数据冗余，增进数据的一致性。”

–Wikipedia

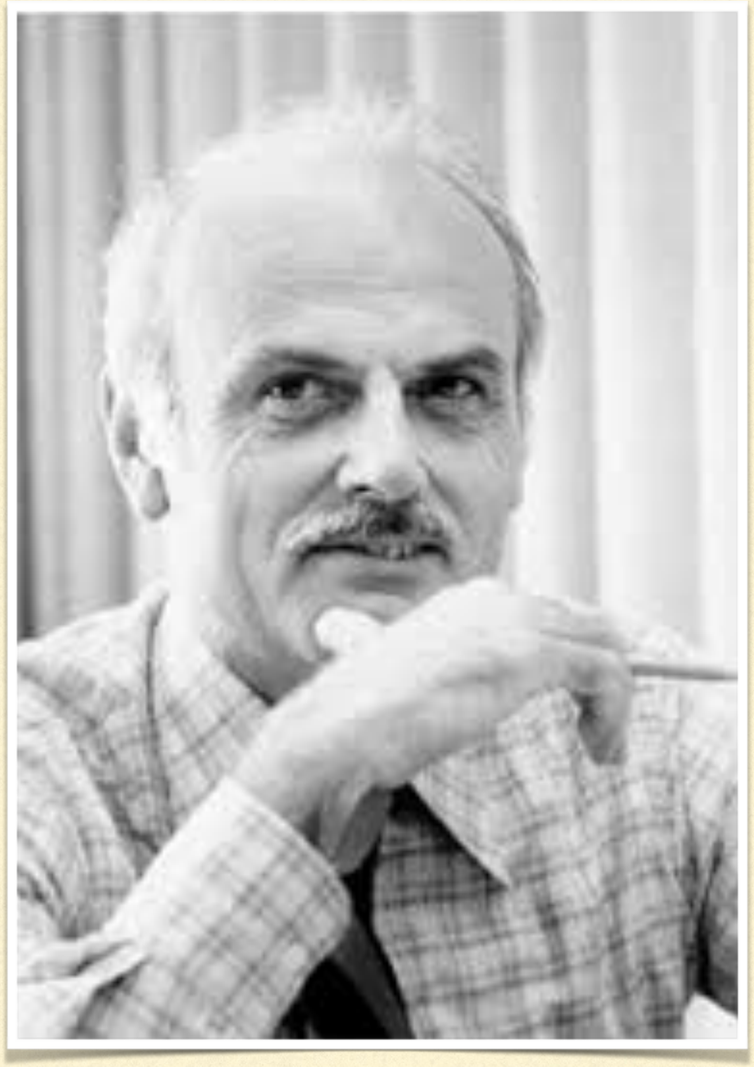
埃德加·弗兰克·科德

（Edgar Frank Codd，1923年8月23日－2003年4月18日）

关系模型的发明者埃德加·科德最早提出这一概念，并于1970年代初定义了**第一范式**、**第二范式**和**第三范式**的概念。

还与RAYMOND F. BOYCE于1974年共同定义了第三范式的改进范式——**BC范式**。

除外还包括针对多值依赖的**第四范式**，连接依赖的**第五范式**、**DK范式**和**第六范式**。



“关系实际上就是关系模式在某一时刻的状态或内容。也就是说，关系模式是型，关系是它的值。关系模式是静态的、稳定的，而关系是动态的、随时间不断变化的，因为关系操作在不断地更新着数据库中的数据。但在实际当中，常常把关系模式和关系统称为关系”

百度百科——关系模型

关系模型的基本概念和基本术语

- 关系(Relation)：一个关系对应着一个二维表，二维表就是关系名。
- 元组(Tuple)：在二维表中的一行，称为一个元组。
- 属性(Attribute)：在二维表中的列，称为属性。属性的个数称为关系的元或度。列的值称为属性值。
- （值）域(Domain)：属性值的取值范围为值域。
- 分量：每一行对应的列的属性值，即元组中的一个属性值。
- 关系模式：在二维表中的行定义，即对关系的描述称为关系模式。一般表示为（属性1，属性2，……,属性n），如老师的关系模型可以表示为教师（教师号，姓名，性别，年龄，职称，所在系）。
- 键(码)：如果在一个关系中存在唯一标识一个实体的一个属性或属性集称为实体的键，即使得在该关系的任何一个关系状态中的两个元组，在该属性上的值的组合都不同。
- **候选键(候选码)**：若关系中的某一属性组的值能唯一的标识一个元组，而其任何真子集都不能再标识，则称该属性组为候选键。
- **主键**：所谓主键就是在实体集中区分不同实体的候选键。一个实体集中只能有一个主键，但可以有多个候选键。
- **主属性**：包含在任一候选键中的属性称主属性。
- **非主属性**：不包含在候选键中的属性。

第一范式（1NF）

- 在关系模型中，所有的域都应该是原子性的，即数据库表的每一列都是不可分割的原子数据项，而不能是集合，数组，记录等非原子数据项。
- 如果出现重复的属性，就可能需要定义一个新的实体，新的实体由重复的属性构成，新实体与原实体之间为一对多关系。
- 简而言之，第一范式就是无重复的列。

第二范式（2NF）

- 它的规则是要求数据表里的所有非主属性都要和该数据表的主键有完全依赖关系。
- 每个表必须有且仅有一个数据元素为主关键字(Primary key)，其他数据元素与主关键字一一对应。
- 通常称这种关系为函数依赖(Functional dependence)关系，即表中其他数据元素都依赖于主关键字，或称该数据元素惟一地被主关键字所标识。
- 如果一个数据表只有一个字段的话，它就一定符合第二范式。

一个数据表符合第二范式当且仅当：

* 它符合第一范式
* 所有非键的字段都一定是候选键全体字段的函数

第二范式的例子

货物类型	货物ID	货物名称	注意事项
瓷碗	1	白色瓷碗	易碎品
瓷碗	2	青花瓷碗	易碎品
瓷碗	3	雕花瓷碗	易碎品
三合板	1	普通三合板	易燃物品，注意防火

在该表中**主键为（货物类型，货物ID）**，货物名称字段完全依赖于这个主键，换句话说，货物的名称完全是取决于这个主键的值的。但“注意事项”这一列，仅依赖于一个主键中”货物类型“这一个属性。简单地说，第二范式要求每个非主属性完全依赖于主键，而不是仅依赖于其中一部分属性。

该表中存在一个对主键不是完全依赖的字段，所以不符合第二范式。

第二范式的例子

货物类型	货物ID	货物名称
瓷碗	1	白色瓷碗
瓷碗	2	青花瓷碗
瓷碗	3	雕花瓷碗
三合板	1	普通三合板

在该表中的**主键依然是（货物类型、货物ID）**，非主键字段“货物名称”，完全依赖于这两个主键，那么我们就可以说，该表是符合数据库第二范式的。

第三范式（3NF）

- 要求所有非主键属性都只和候选键有相关性，也就是说非主键属性之间应该是独立无关的。
- 表中的所有数据元素不但要能唯一地被主关键字所标识，而且它们之间还必须相互独立，不存在其他的函数关系。

第三范式的例子

组件编号	制造商名称	制造商地址
1000	Toyota	Park Avenue
1001	Mitsubishi	Lincoln Street
1002	Toyota	Park Avenue

本例中制造商地址很明显地不该被列在这个关系里面，因为和组件本身比起来，制造商地址应该和制造商比较有关系；正确的做法应该是把制造商独立成为一个新的数据表。

第三范式的例子

制造商名称	制造商地址
Toyota	Park Avenue
Mitsubishi	Lincoln Street
Toyota	Park Avenue

组件编号	制造商名称
1000	Toyota
1001	Mitsubishi
1002	Toyota

本例中制造商地址很明显地不该被列在这个关系里面，因为和组件本身比起来，制造商地址应该和制造商比较有关系；正确的做法应该是把制造商独立成为一个新的数据表。

订单编号 (Order Number)(主键)	客户名称 (Customer Name)	单价 (Unit Price)	数量 (Quantity)	小计 (Total)
1000	David	$35.00	3	$105.00
1001	Jim	$25.00	2	$50.00
1002	Bob	$25.00	3	$75.00

在本例中，非主键字段完全依赖于主键订单编号，也就是说唯一的订单编号能导出唯一非主键字段值，符合第二范式。

第三范式要求非主键字段之间不能有依赖关系，显然本例中小计依赖于非主键字段“单价”和“数量”，不符合第三范式。

小计不应该放在这个数据表里面，只要把单价乘上数量就可以得到小计了；如果想要符合第三范式的话，就把小计拿掉吧

BC范式（BCNF）

- 如果对于关系模式R中存在的任意一个非平凡函数依赖X->A，都满足X是R的一个超键，那么关系模式R就属于BCNF。
- BCNF去除了属性间的不必要的函数依赖。
- 在第三范式的基础上加上稍微更严格约束，每个BCNF关系都满足第三范式。

！平凡函数依赖关系是指，如果属性集合X包含了属性集合A，那么就一定有X->A

任何一个BCNF必然满足：

* 所有非主属性都完全函数依赖于每个候选键
* 所有主属性都完全函数依赖于每个不包含它的候选键
* 没有任何属性完全函数依赖于非候选键的任何一组属性

BC范式的例子

Id	名称	类型名称	类型值
1000	Toyota	Park Avenue	1
1001	Mitsubishi	Lincoln Street	2
1002	Toyota	Park Avenue	1

第四范式

- 设关系R（X，Y，Z），其中X，Y，Z是成对的、不相交属性的集合。若存在非平凡多值依赖，则意味着对R中的每个属性A存在有函数依赖x -> a（X必包含键）。那么关系R属于第四范式。
- 换句话说，当关系R的属性集合X是非平凡多值依赖的域，它就包含关系R的键。
- 这个定义和BCNF定义唯一的不同点是后者研究非平凡多值依赖的域。

过高的范式

- 现在数据库设计最多满足3NF，普遍认为范式过高，虽然具有对数据关系更好的约束性，但也导致数据关系表增加而令数据库IO更易繁忙，原来交由数据库处理的关系约束现更多在数据库使用程序中完成。