

作业_函数依赖

姓名	学号	班级	专业
王何佳	2023211603	2023211804	网络空间安全

1、关系W(工号, 姓名, 工种, 定额)，每个工种有一个定额

主码：工号

函数依赖

- 工号 \rightarrow (姓名, 工种, 定额)
- 工种 \rightarrow 定额

所有属性是原子的，满足1NF

检查是否满足2NF：

- 由于主码是单个属性，不存在非主属性部分依赖于主码的情况，因此满足2NF

检查是否满足3NF：

- 工号 \rightarrow 工种与 工种 \rightarrow 定额导致定额传递依赖于工号，不满足3NF

工种与定额的函数依赖导致非主属性对非主属性产生依赖，因此将工种与定额独立出来，将工种作为主属性，即可满足3NF。

分解后关系

- W1(工号, 姓名, 工种)
 - 主码：工号
 - 函数依赖：工号 \rightarrow (姓名, 工种)
- W2(工种, 定额)
 - 主码：工种
 - 函数依赖：工种 \rightarrow 定额

2、关系R(材料号, 材料名, 生产厂)

材料号	材料名	生产厂
M1	线材	武汉
M2	型材	武汉
M3	板材	广东
M4	型材	武汉

据图分析

- 只有材料号是唯一的，因此材料号作为**主码**
- 同一家生产厂可以生产不同的材料，不同的生产厂可以生产相同的材料，因此材料名与生产厂之间没有决定关系
- 由此确定**函数依赖**
 - 材料号 → 材料名
 - 材料号 → 生产厂

所有属性是原子的，满足1NF

检查是否满足2NF：

- 由于主码是单个属性，不存在非主属性部分依赖于主码的情况，因此满足2NF

检查是否满足3NF：

- 非主属性材料号名、生产厂之间没有函数依赖，材料名和生产厂只由主码决定，因此不存在传递依赖，满足3NF，无需分解

3、某商业集团数据库的关系模式R

R(商店编号, 商品编号, 商品库存数量, 部门编号, 负责人)

- 每个商店的每种商品只在一个部门销售
- 每个商店的每个部门只有一个负责人
- 每个商店的每种商品只有一个库存数量

(1)写出关系模式R的函数依赖关系和主码。

(2)该关系模式最高满足第几范式？举例可能的异常 (3)将该关系模式分解为3NF

函数依赖

- (商店编号, 商品编号) → 部门编号
- (商店编号, 部门编号) → 负责人
- (商店编号, 商品编号) → 商品库存数量

主码：(商店编号, 商品编号)

范式

- 所有属性是原子的，满足1NF
- 非主属性完全函数依赖于主码，满足2NF
- 存在传递依赖(商店编号, 商品编号) → 部门编号 → 负责人，负责人传递依赖于主码，违反了3NF
- 因此该关系模式最高满足2NF

异常

- **输入异常：**新部门有负责人，但无商品销售（缺商品编号，无法插入负责人）
- **删除异常：**删除某商品记录，若该商品是该部门唯一销售的，则该部门负责人信息丢失
- **更新异常：**部门负责人变更，需更新该部门所有商品记录，易导致数据不一致

分解为3NF

- 当前不满足3NF的原因在于(商店编号, 部门编号) → 负责人，部门编号为非主属性，因此产生传递。分解时将商店编号、部门编号与负责人独立出来即可

分解后关系

- R_1 (商店编号, 商品编号, 商品库存数量, 部门编号)
 - 主码：(商店编号, 商品编号)
 - 函数依赖：(商店编号, 商品编号) → (部门编号, 商品库存数量)
- R_2 (商店编号, 部门编号, 负责人)
 - 主码：(商店编号, 部门编号)
 - 函数依赖：(商店编号, 部门编号) → 负责人

4、建立一个包含系、学生、班级、学会等信息的关系数据库

- 描述学生的属性有：学号、姓名、出生日期、系名、班号、宿舍区
- 描述班级的属性有：班号、专业名、系名、人数、入校年份
- 描述系的属性有：系名、系号、系办公室地点、人数
- 描述学会的属性有：学会名、成立年份、地点、人数

有关语义如下：

- 一个系有若干专业，每个专业每年只招一个班，每个班有若干学生。
- 一个系的学生住在同一个宿舍区。
- 每个学生可参加若干学会，每个学会有若干学生。学生参加某学会有一入会年份。
请给出关系模式，写出每个关系模式的最小依赖集，指出是否存在：传递函数依赖，对于函数依赖左部是多属性的情况，讨论函数依赖是完全函数依赖还是部分函数依赖。
指出各关系的候选码、外部码并说明是否全码存在。

改进后的关系模式

改进后的设计通过新增“专业”实体、并将“宿舍区”等属性移到其逻辑上所属的“系”表中，使所有关系模式都达到3NF
原关系模式放在最后了

系(系号, 系名, 系办公室地点, 宿舍区, 人数)

- 最小依赖集：
 - 系号 \rightarrow (系名, 系办公室地点, 宿舍区, 人数)
 - 系名 \rightarrow 系号
 - 没有传递、部分函数依赖
- 候选码：系号、系名
- 外部码：无
- 全码：不存在

专业(专业名, 系号)

- 最小依赖集：
 - 专业名 \rightarrow 系号
 - 没有传递、部分函数依赖
- 候选码：专业名
- 外部码：系号
- 注：“专业”实体是为了解决班级实体中存在的依赖问题而引入的

班级(班号, 专业名, 入校年份, 人数)

- 最小依赖集：
 - 班号 \rightarrow (专业名, 入校年份, 人数)
 - (专业名, 入校年份) \rightarrow 班号
 - 没有传递、部分函数依赖
- 候选码：班号、(专业名, 入校年份)
- 外部码：专业名
- 全码：不存在
- 注：若将系名直接放入班级表，会产生**传递依赖**班号 \rightarrow 专业名 \rightarrow 系名；且因(专业名, 入校年份)为候选码，而系名只依赖于专业名，构成了**部分依赖**。因此引入“专业”实体。

学生(学号, 姓名, 出生日期, 班号)

- 最小依赖集：
 - 学号 \rightarrow (姓名, 出生日期, 班号)
 - 没有传递、部分函数依赖
- 候选码：学号

- 外部码：班号
- 全码：不存在
- 注：若将系名和宿舍区直接放入学生表，导致传递依赖
学号 → 班号 → 专业名 → 系名 → 宿舍区，因此将各属性归属到正确的实体，消除了传递依赖

学会(学会名, 成立年份, 地点, 人数)

- 最小依赖集：
 - 学会名 → (成立年份, 地点, 人数)
 - 没有传递、部分函数依赖
- 候选码：学会名
- 外部码：无
- 全码：不存在

参加(学号, 学会名, 入会年份)

- 最小依赖集：
 - (学号, 学会名) → 入会年份
 - 没有传递、部分函数依赖
- 候选码：(学号, 学会名)
- 外部码：学号、学会名
- 全码：存在

以下是原关系模式。

原关系模式（未改进的）

系(系名, 系号, 系办公室地点, 人数)

- 最小依赖集：
 - 系号 → (系名, 系办公室地点, 人数)
 - 系名 → (系号, 系办公室地点, 人数)
 - 没有传递、部分函数依赖
- 候选码：系号、系名
- 外部码：无
- 全码：不存在

班级(班号, 专业名, 系名, 人数, 入校年份)

- 最小依赖集：
 - 班号 → (专业名, 系名, 人数, 入校年份)
 - (专业名, 入校年份) → 班号
 - 专业名 → 系名

- 存在传递函数依赖：班号 \rightarrow 专业名 \rightarrow 系名
- 存在部分函数依赖：候选码为(专业名, 入校年份)，而系名仅依赖专业名
- 候选码：班号、(专业名, 入校年份)
- 外部码：无
- 全码：不存在

学生(学号, 姓名, 出生日期, 系名, 班号, 宿舍区)

- 最小依赖集：
 - 学号 \rightarrow (姓名, 出生日期, 班号)
 - 班号 \rightarrow 系名
 - 系名 \rightarrow 宿舍区
 - 存在传递函数依赖：学号 \rightarrow 班号 \rightarrow 系名和学号 \rightarrow 班号 \rightarrow 系名 \rightarrow 宿舍区
- 候选码：学号
- 外部码：班号、系名
- 全码：不存在

学会(学会名, 成立年份, 地点, 人数)

- 最小依赖集：
 - 学会名 \rightarrow (成立年份, 地点, 人数)
 - 没有传递、部分函数依赖
- 候选码：学会名
- 外部码：无
- 全码：不存在

参加(学号, 学会名, 入会年份)

- 最小依赖集：
 - (学号, 学会名) \rightarrow 入会年份
 - 没有传递、部分函数依赖
- 候选码：(学号, 学会名)
- 外部码：学号、学会名
- 全码：存在