

# 第四章 关系规范化理论

4.1 问题的提出

4.2 函数依赖和范式

4.3 数据依赖的公理系统

4.4 关系模式的分解方法



# 问题

关系模式  $R ( U, D, \text{dom}, I, F )$  各元组的含义。

1. 为什么要进行规范化？
2. 函数依赖、函数不依赖、互相依赖、平凡函数依赖、非平凡函数依赖、完全函数依赖、部分函数依赖、传递函数依赖的定义，举例说明。
3. 候选码、主码、主属性、非主属性、全码、外码
4. 1NF, 2NF, 3NF, BCNF 的定义。举例说明。



# 范式的概念

- 范式表示关系模式满足的某种级别
- 1971年E.F.Codd 提出范式概念

1NF 2NF 3NF BCNF 4NF 5NF

$5NF \subset 4NF \subset BCNF \subset 3NF \subset 2NF \subset 1NF$



# 规范化的概念

一个低一级范式的关系模式，通过模式分解可以转换为若干个高一级范式的关系模式的集合，这种过程就叫做规范化。



## 关系模式 (Relation Schema)

关系模式是关系中信息内容结构的描述。

$$R(U, D, \text{DOM}, I, F)$$

R: 是关系名

U: 是组成关系R的全部属性的集合

D: 是U中属性取值的值域

DOM: 是属性列到域的映射

I: 是一组完整性约束条件

F: 是属性集间的一组数据依赖

简写:  $R(U)$

或  $R(U, F)$



# 关系模式

**R (U, D, DOM, I, F)**

数据依赖：关系中属性间互相依存、互相制约的关系。

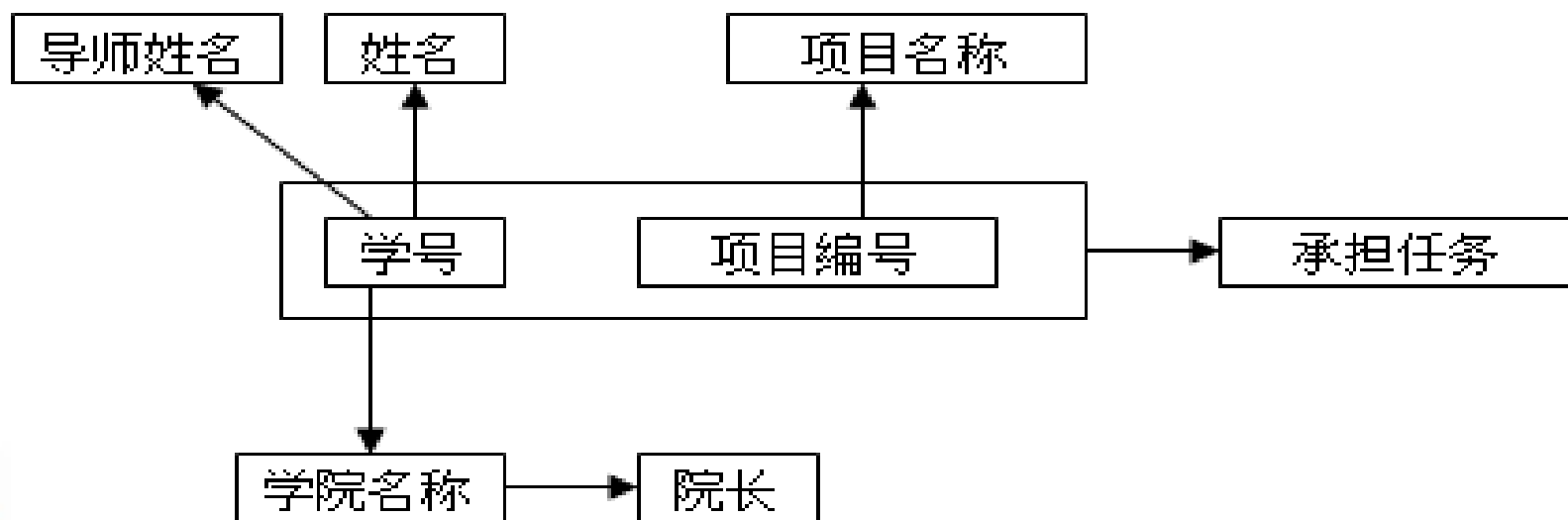
[**函数依赖**、多值依赖、连接依赖、分层依赖和相互依赖]



# 例如:

**U**= {学号、姓名、学院名称、院长、项目编号、项目名称、承担任务、导师姓名}

**F**= {学号→姓名, 学院名称→院长, 学号→学院名称,  
(学号, 项目编号)→承担任务, 项目编号→项目名称,  
学号→导师姓名}



学号	姓名	学院名称	院长	项目编号	项目名称	承担任务	导师姓名
20082401	周黎明	计算机学院	李洲彤	0042	提升机稳定性研究	实验分析	贺信维
20082402	李毅先	计算机学院	李洲彤	0042	提升机稳定性研究	系统设计	张琦
20082402	李毅先	计算机学院	李洲彤	0052	多维数据分析研究	软件编码	萨林
20083401	王鑫鑫	数学学院	吴兆民	0091	定理证明自动化研究	软件编码	刘玉琴
20083402	何飞雨	数学学院	吴兆民	0083	最大熵原理研究	软件编码	刘玉琴
20083403	杨宇奇	数学学院	吴兆民	0063	软件测试路径分析	实验分析	刘坤鹏

缺点：1、冗余太大

2、操作异常

1)插入异常    2)删除异常    3)更新异常

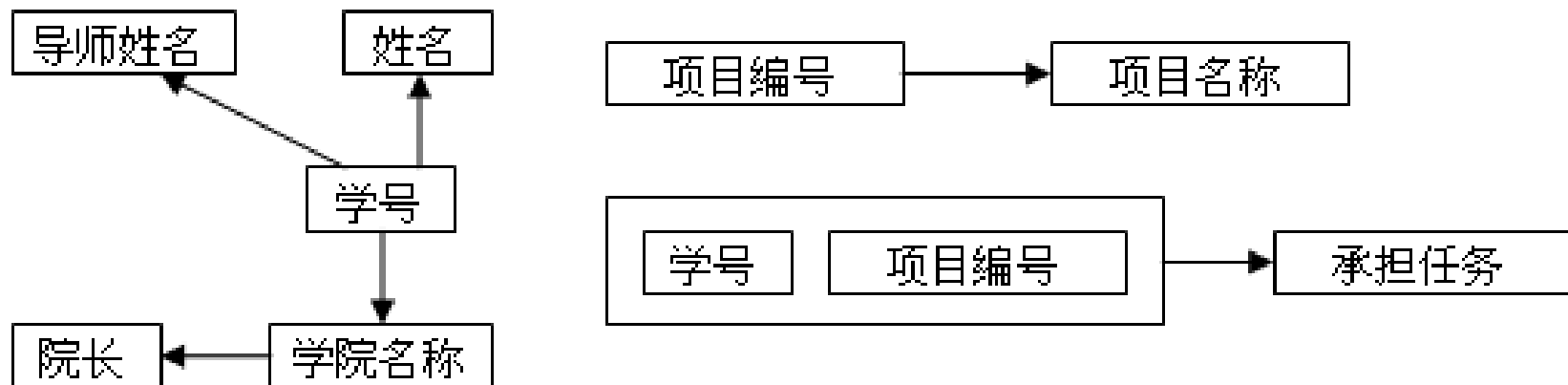


# (1) 第一种分解方法

S\_D (学号, 学生姓名, 学院名称, 院长, 导师姓名)

P (项目编号, 项目名称)

S\_P (学号, 项目编号, 承担任务)



消除部分冗余数据



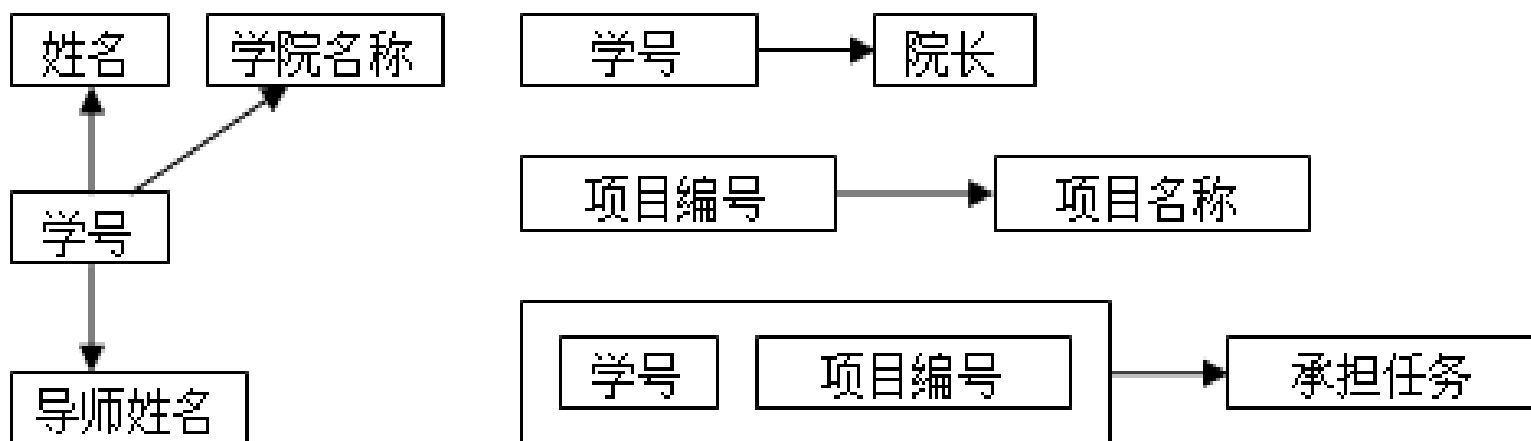
## (2) 第二种分解方法

S (学号, 学生姓名, 学院名称, 导师姓名)

P (项目编号, 项目名称)

S\_MN (学号, 院长)

S\_P (学号, 项目编号, 承担任务)



消除冗余数据, 但丢失数据依赖关系



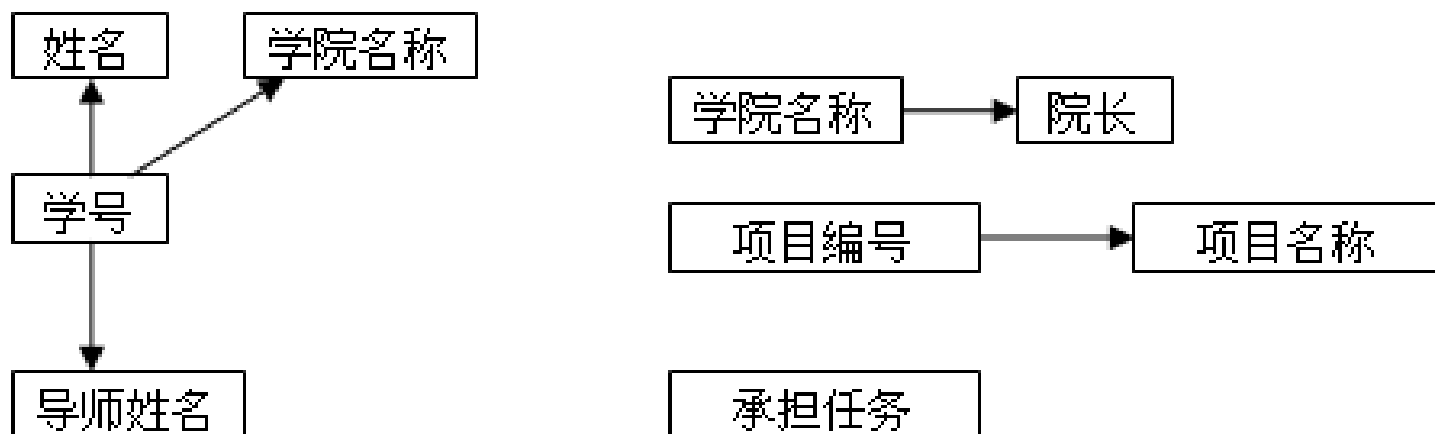
### (3) 第三种分解方法

S (学号, 学生姓名, 学院名称, 导师姓名)

P (项目编号, 项目名称)

D (学院名称, 院长)

T (承担任务)



消除冗余数据，但丢失了信息



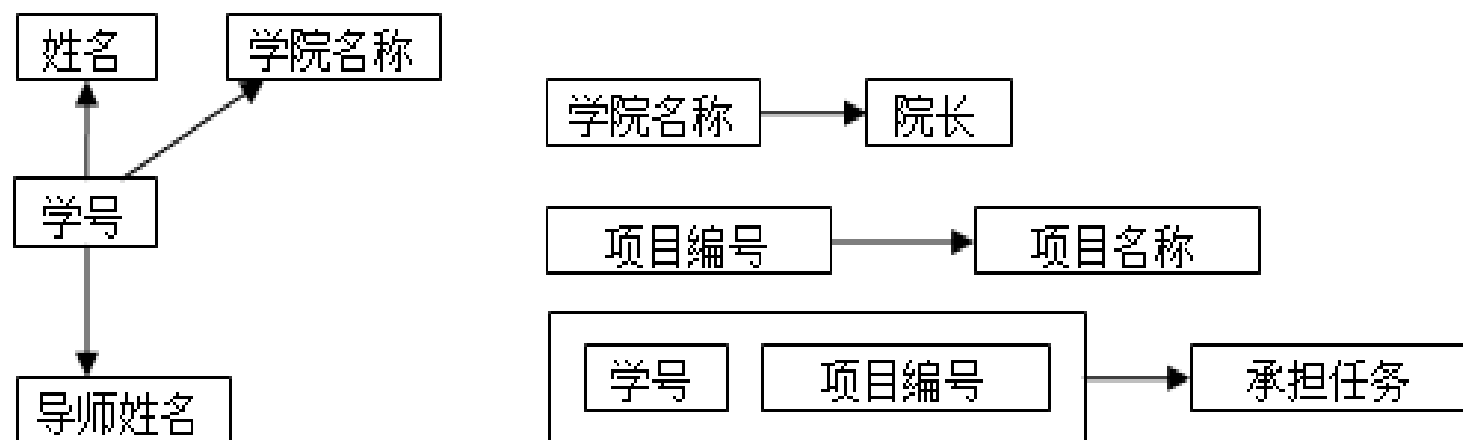
## (4) 第四种分解方法

S (学号, 学生姓名, 学院名称、导师姓名)

P (项目编号, 项目名称)

D (学院名称, 院长)

S\_P (学号, 项目编号, 承担任务)



消除冗余, 保持数据依赖, 保证信息不丢失



# 存在问题的原因

- 数据冗余和操作异常产生的重要原因就是**对数据依赖的不恰当处理**，最终导致不合理的关系模式的设计。
- 一个关系中各**属性之间**可能是**相互关联**的，而这种**关联**有“**强**”有“**弱**”，有直接关联，也有间接关联。
- 不从语义上研究和考虑属性子集间的这种关联简单地将属性**随意地编排在一起**，形成泛关系模式，就可能**产生很大程度的数据冗余**，导致“**排他**”现象，从而引发各种冲突和异常。



# 解决方法

- 解决问题的方法就是将关系模式进一步分解
- 将关系模式中的属性按照一定的约束条件重新分组，争取“**一个关系模式只描述一个独立的实体**”，使得逻辑上独立的信息放在独立的关系模式中，即进行关系模式的规范化处理。



# 规范化理论的提出

“关系规范化”理论包含两个核心的问题：

- 一、如何判断关系模式中存在的问题。
  - 通过分析关系模式中的数据依赖关系，判断关系模式的“范式”级别，从而得到这种模式中可能存在的数据冗余和操作异常问题；
- 二、如何解决关系模式中存在的问题，即对关系模式进行分解。
  - 如何分解？“关系规范化”理论为解决这些问题提供了理论依据和相应的算法。



# 4.2 函数依赖和范式

## 一、函数依赖

设 $R(U)$ 是属性集合 $U=\{A_1, A_2, \dots, A_n\}$ 上的一个关系模式， $X, Y$ 是 $U$ 上的两个子集，若对 $R(U)$ 的任意一个可能的关系 $r$ ， $r$ 中不可能有两个元组满足在 $X$ 中的属性值相等而在 $Y$ 中的属性值不等，则称“ $X$ 函数决定 $Y$ ”或“ $Y$ 函数依赖于 $X$ ”，记作 $X \rightarrow Y$ 。

示例： $U = \{\text{学号}, \text{姓名}, \text{年龄}, \text{班号}, \text{班长}, \text{课号}, \text{成绩}\}$

□ 学号  $\rightarrow$  { 姓名, 年龄 }

□ 班号  $\rightarrow$  班长

□ { 学号, 课号 }  $\rightarrow$  成绩

设计关系模式时，除给出属性全集外，还需给出数据依赖集合

注：函数依赖的分析取决于对问题领域的限定和分析，取决于对业务规则的正确理解。例如：问题领域中，学生是没有重名的，则有：“年龄”和“家庭住址”都函数依赖于“姓名”。而在另一个问题领域中，学生是有重名的，则上述函数依赖是不成立的。





## 4.2 函数依赖和范式

**或者说：**关系模式 $R(U)$ 的任一具体关系，属性集 $X$ 在任意元组上的值能唯一决定属性集 $Y$ 在该元组上的值，则称 $X$ 函数确定 $Y$ 或 $Y$ 函数依赖于 $X$ ，记作 $X \rightarrow Y$ 。

**注意：**函数依赖不是指关系模式 $R$ 的某个或某些关系满足的条件，而是指 $R$ 的一切关系均要满足的约束条件。



由定义可以导出下列概念：

1. **决定因素**：若 $X \rightarrow Y$ ，则 $X$ 叫做决定因素
2. **平凡的函数依赖**： $X \rightarrow Y$ ， $Y \subseteq X$ ，则称 $X \rightarrow Y$ 是平凡的函数依赖。
3. **非平凡的函数依赖**： $X \rightarrow Y$ ，但 $Y \not\subseteq X$ ，则称 $X \rightarrow Y$ 是非平凡的函数依赖。
4. **互相依赖**：若 $X \rightarrow Y$ ， $Y \rightarrow X$ ，则记作 $X \leftrightarrow Y$ 。
5. 若 $Y$ 不函数依赖于 $X$ ，则记作 $X \nrightarrow Y$ 。



**练习：请分析下列属性集上的函数依赖**

●**学生(学号, 姓名, 班级, 课号, 课程名, 成绩, 教师, 教师职务)**

●**客户(客户号, 客户名称, 类别, 联系电话, 产品编码, 产品名称, 数量, 要货日期)**



## 练习：请分析下列属性集上的函数依赖

### ●学生(学号, 姓名, 班级, 课号, 课程名, 成绩, 教师, 教师职务)

- ✓学号  $\rightarrow$  {姓名, 班级};      课号  $\rightarrow$  课程名;    {学号, 课号}  $\rightarrow$  成绩
- ✓教师  $\rightarrow$  教师职务
- ✓{班级, 课号}  $\rightarrow$  教师

{班级, 课号}  $\rightarrow$  教师; 课号  $\rightarrow$  教师;  
{学号, 课号}  $\rightarrow$  教师  
究竟选哪一个取决于对问题领域的理解

### ●客户(客户号, 客户名称, 类别, 联系电话, 产品编码, 产品名称, 数量, 要货日期)

- ✓客户号  $\rightarrow$  {客户名称, 类别}
- ✓产品编码  $\rightarrow$  产品名称
- ✓{ 客户号, 产品编码, 要货日期 }  $\rightarrow$  数量

本质上,函数依赖  
是对属性之间取值的一种约束,  
是一种数据依赖



## 定义4.2：完全函数依赖

在 $R(U)$ 中，如果 $X \rightarrow Y$ ，并且对于 $X$ 的任何一个真子集 $X'$ ，都有 $X' \nrightarrow Y$ ，则称 $Y$ 对 $X$ 完全函数依赖。记作：

$$X \xrightarrow{F} Y$$

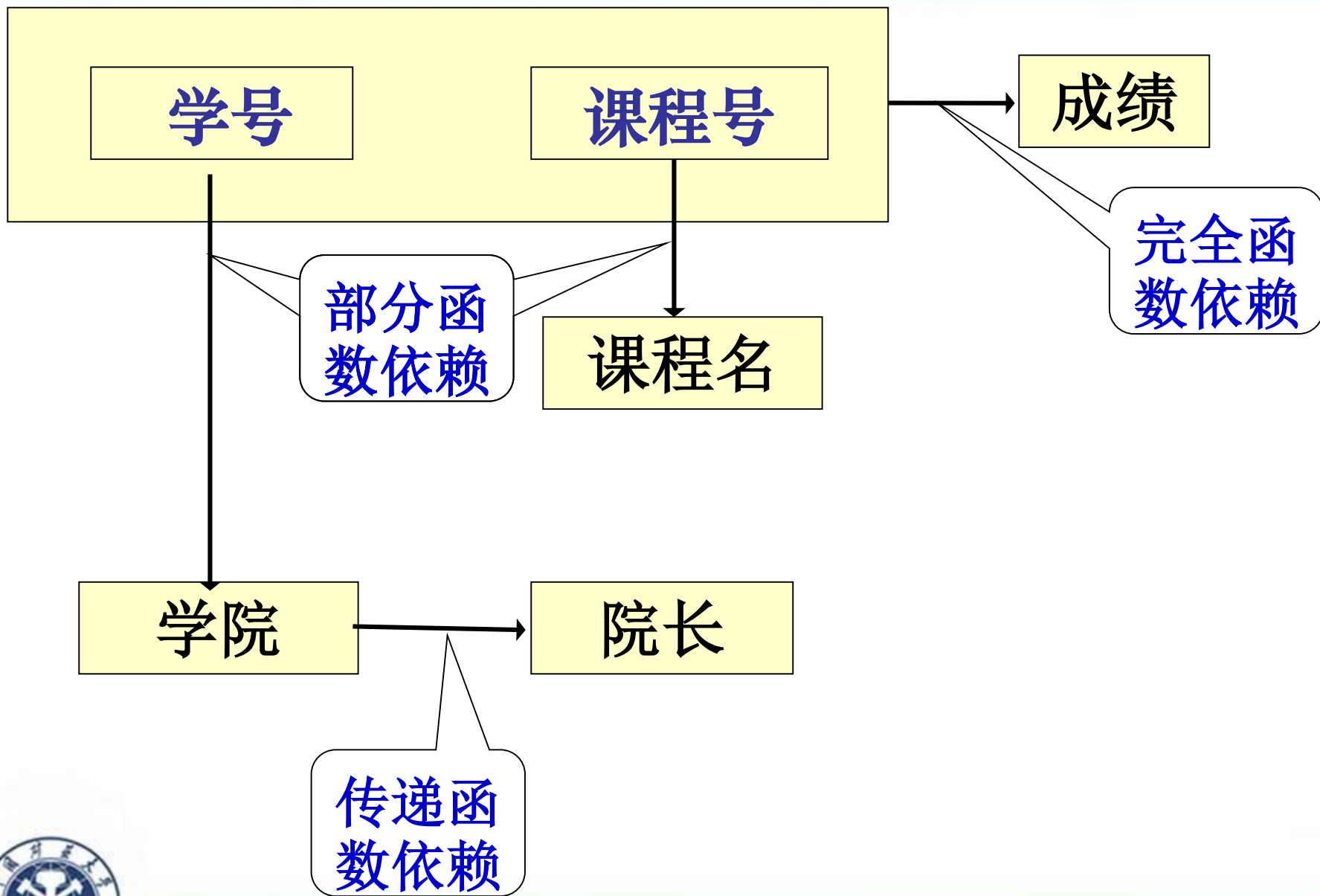
## 定义4.3：部分函数依赖

在 $R(U)$ 中，如果 $X \rightarrow Y$ ，并且对于 $X$ 的一个真子集 $X'$ ，有 $X' \rightarrow Y$ ，则称 $Y$ 对 $X$ 部分函数依赖。记作：

$$X \xrightarrow{P} Y$$

## 定义4.4：传递函数依赖

在 $R(U)$ 中，如果 $X \rightarrow Y$ ， $(Y \subsetneq X)$ ， $Y \nrightarrow X$ ， $Y \rightarrow Z$ ，则称 $Z$ 对 $X$ 传递函数依赖。



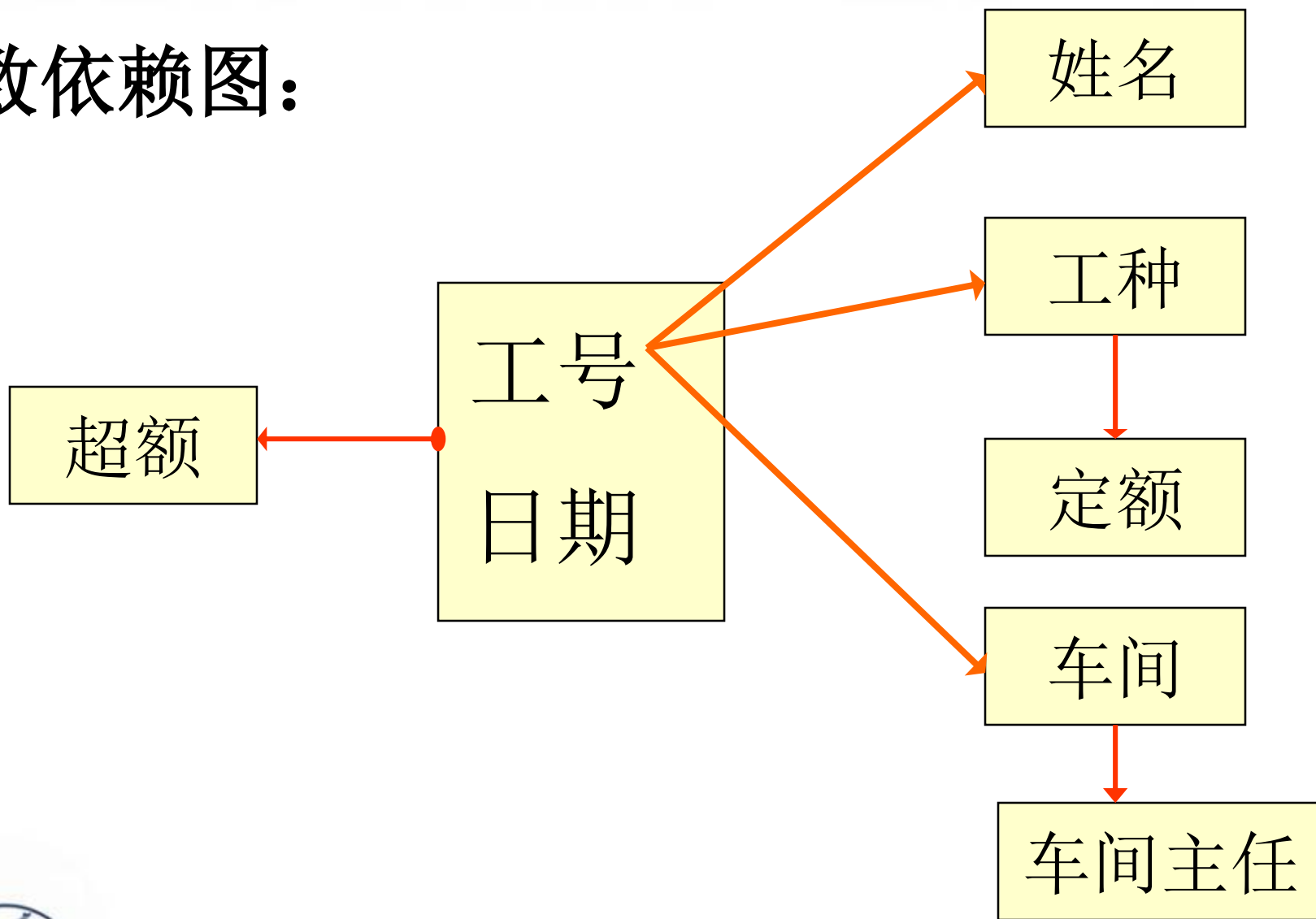
例：设车间考核职工完成生产定额关系W

W ( 日期, 工号, 姓名, 工种, 超额, 定额,  
车间, 车间主任)

F (日期、工号 → 超额, 工号 → 姓名,  
工号 → 车间, 工号 → 工种, 工种 → 定额,  
车间 → 车间主任)



# 函数依赖图:





# 分析:

日期、工号 → 超额

完全函数依赖

日期、工号 → 姓名

日期、工号 → 工种

日期、工号 → 车间

部分函数依赖

日期、工号 → 定额

日期、工号 → 车间主任

传递函数依赖



## 二、码

**定义4.5:** 设 $K$ 为 $R(U, F)$ 中的属性或属性组, 若  $K \xrightarrow{F} U$ , 则 $K$ 为 $R$ 的**候选码**。

- **主码:** 若候选码多于一个, 则选定其中的一个为主码。
- **主属性:** 包含在任何一个候选码中的属性。
- **非主属性:** 不包含在任何候选码中的属性。
- **全码:** 整个属性组是码。



**定义4.6:** 关系模式R中属性或属性组X并非R的主码，但X是另一个关系模式的主码，则称X是R的**外码**。

主码与外码提供了一个表示关系间联系的手段。



# 超码 (Super Key)

- 包含候选码的属性集合称为超码。
- 例如：学号为学生表的候选码，则包含学号的任一个属性组都是超码。



### 三、第一范式(1NF)

定义：满足关系的每一个分量是不可分的数据项  
这一条件的关系模式就属于第一范式(1NF)。

课程名称	学时数	
	讲课	实验
微机原理 化    学	65 40	15 10

课程名称	讲课时数	实验时数
微机原理 化    学	65 40	15 10



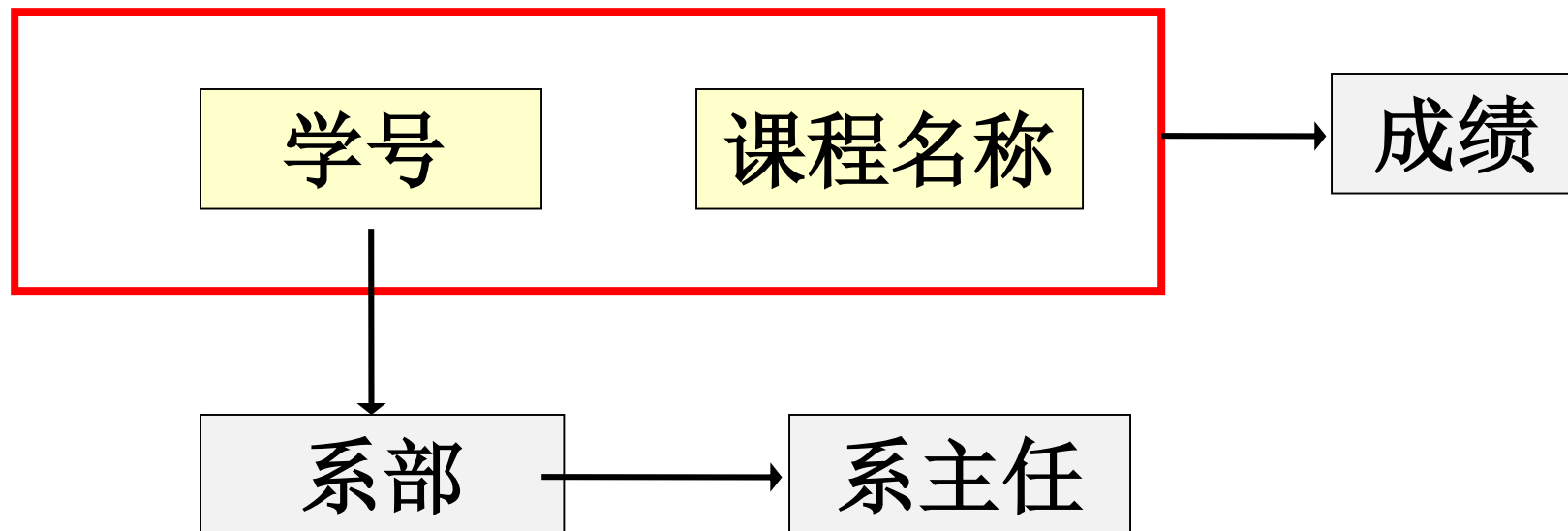
学号	系部	系主任	课程名称	成绩
99101	CS	X	AA	A
99101	CS	X	BB	B
99101	CS	X	CC	A
99102	MA	M	AA	B
99102	MA	M	DD	A
99103	MA	M	EE	C
99104	IS	J	EE	B
99105	IS	J	FF	A

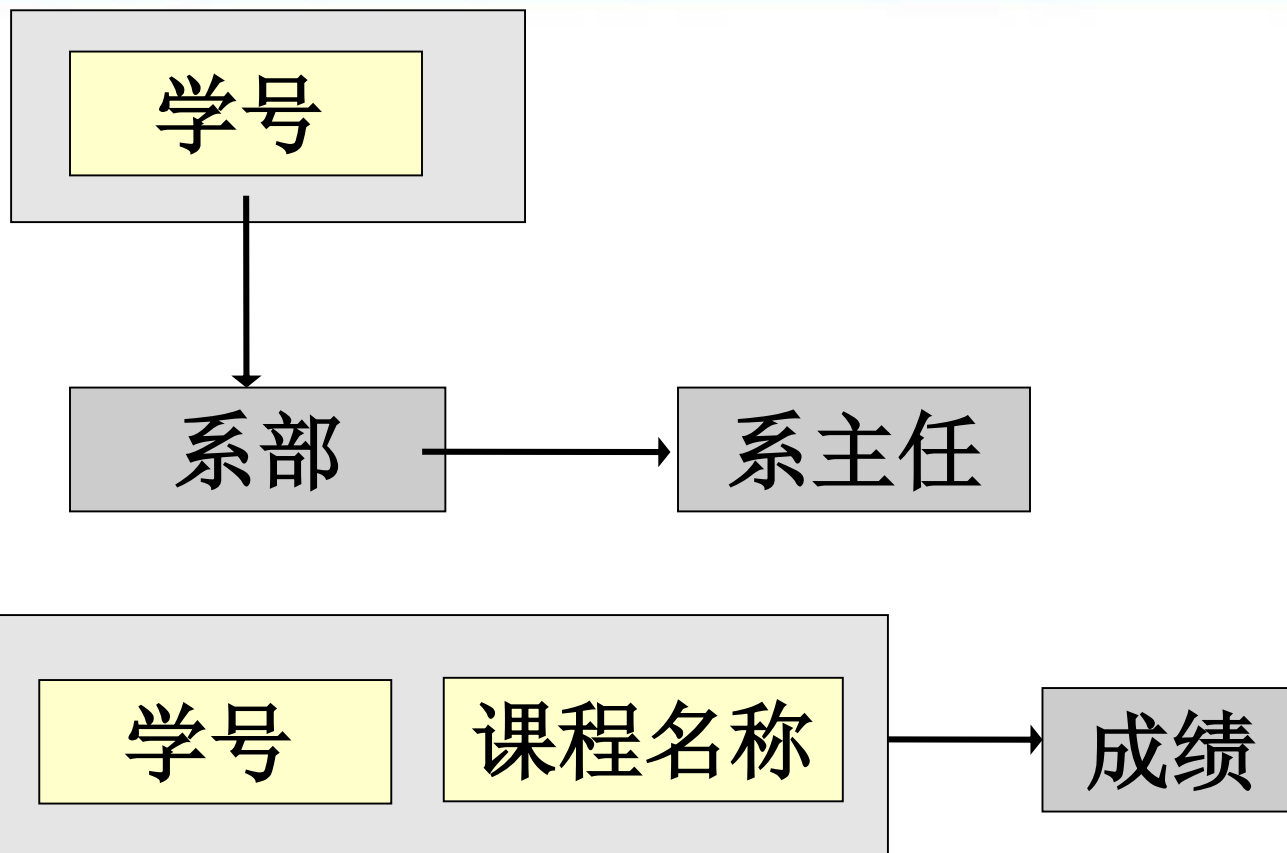
缺点：  
1.插入异常  
2.删除异常  
3.冗余太大  
4.修改复杂



## 四、第二范式(2NF)

定义：若 $R \in 1NF$ ，且每一个非主属性完全函数依赖于候选码，则 $R \in 2NF$ 。





(学号, 系部, 系主任)

(学号, 课程名称, 成绩)





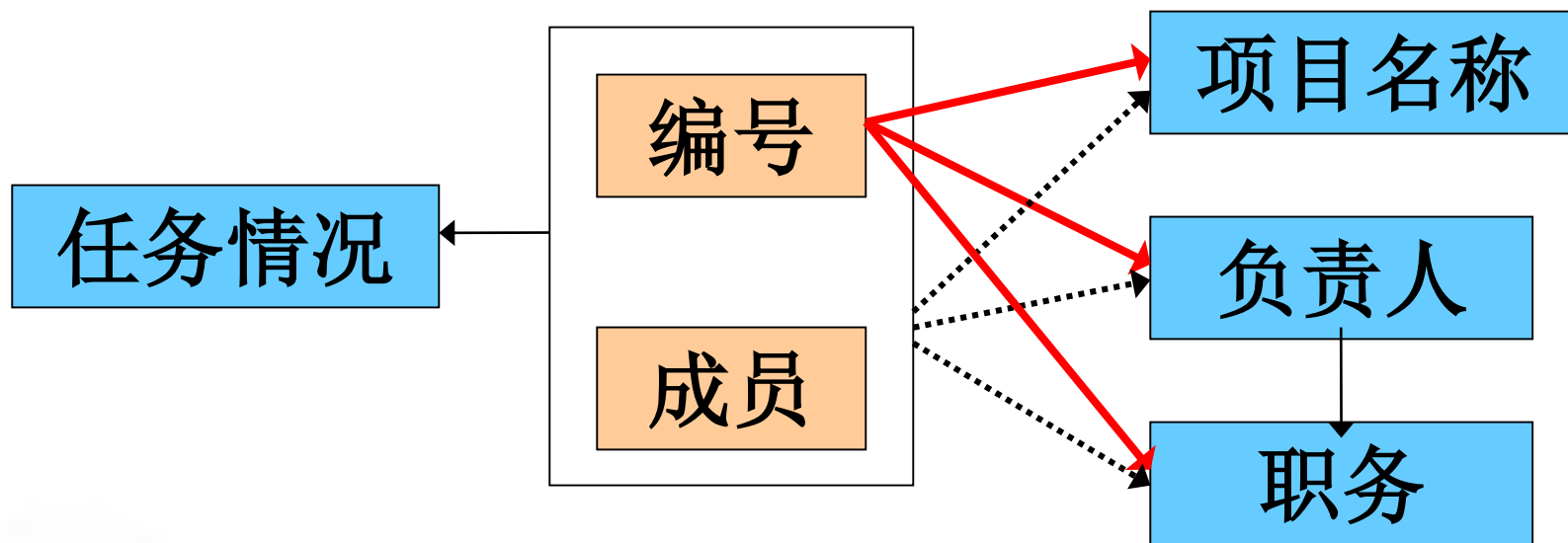
## 四、第三范式(3NF)

定义：关系模式 $R(U, F)$ 中若不存在这样的码 $X$ ，属性组 $Y$ 及非主属性组 $Z(Z \subsetneq Y)$ 使得 $X \rightarrow Y$ ， $(Y \not\rightarrow X) Y \rightarrow Z$ 成立，则称 $R(U, F) \in 3NF$ 。

若 $R \in 1NF$ ，且每一个非主属性既不部分依赖于候选码也不传递依赖于候选码，则 $R \in 3NF$ 。

若 $R \in 2NF$ ，且每一个非主属性不传递依赖于候选码，则 $R \in 3NF$ 。

**例：**项目(编号, 项目名称, 负责人, 职务, 成员, 任务情况)  
(假设:负责人无重名情况)

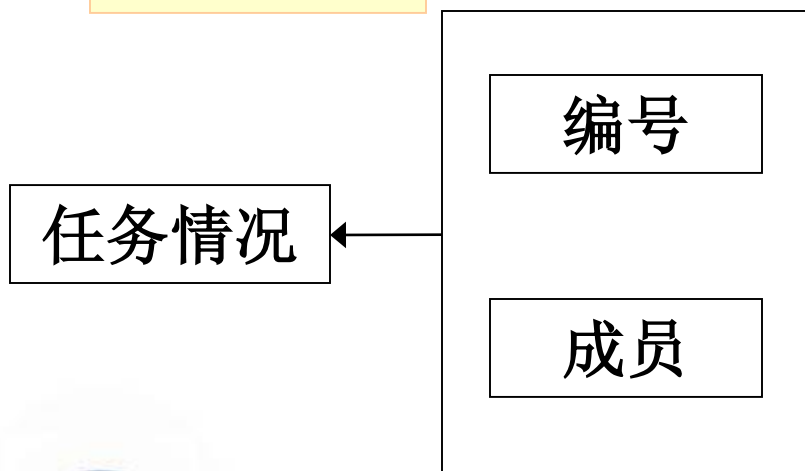


# 根据2NF要求

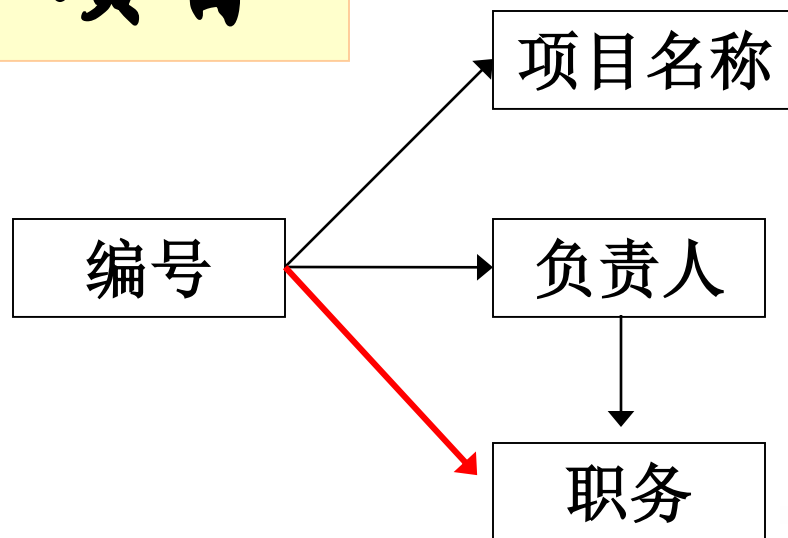
任务（编号，成员，任务情况）

项目（编号，项目名称，负责人，职务）

任务

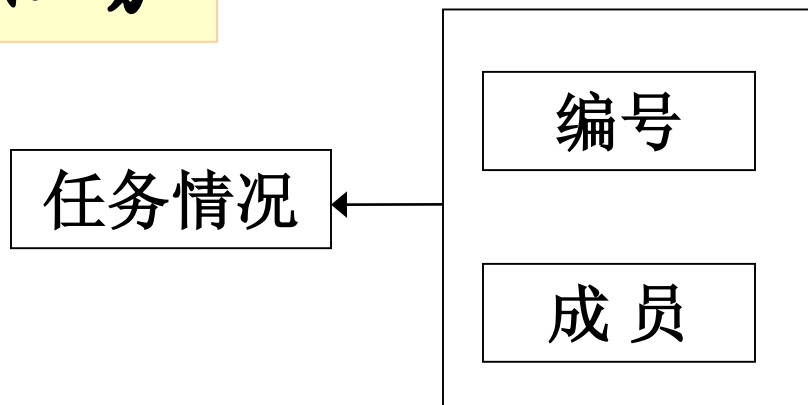


项目

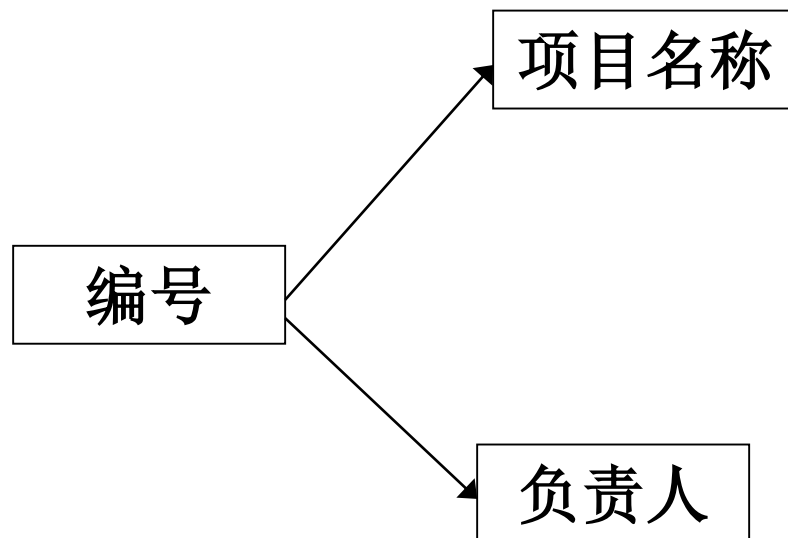


# 根据3NF要求

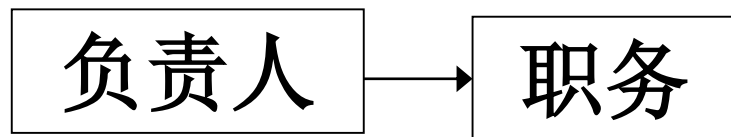
任务



项目



负责人职务

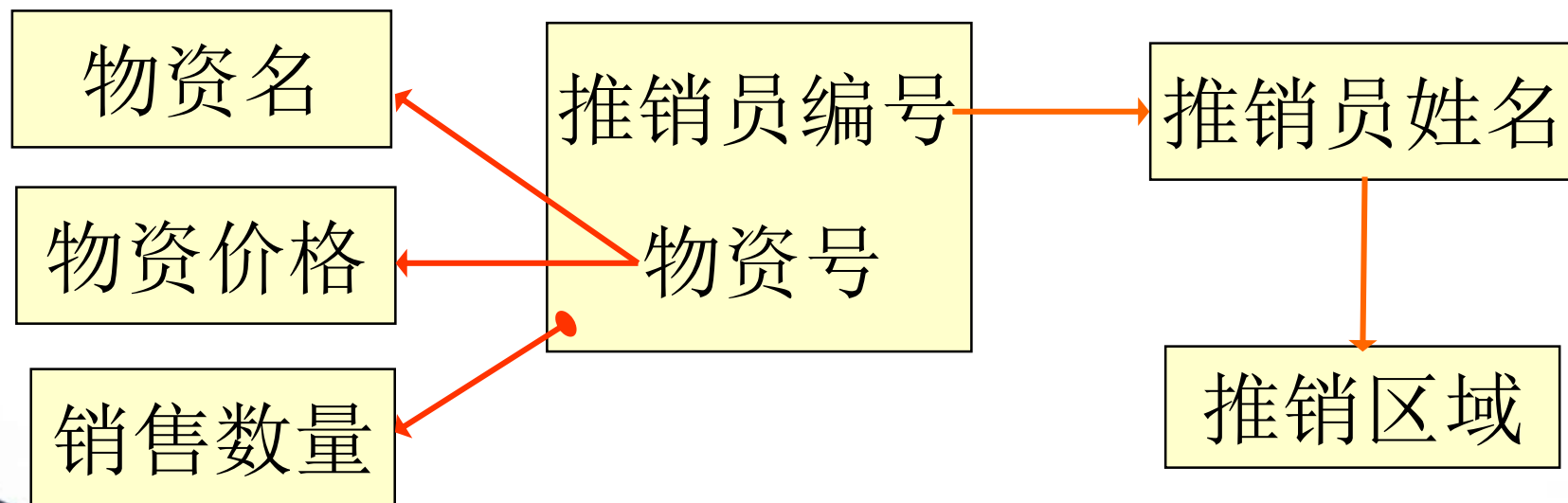


# 例：分析下列关系属于第几范式

推销员管理信息：

（推销员编号，推销员姓名，物资号，物资名，物资价格，销售数量，销售区域）假设推销员姓名无重名。

函数依赖关系如下图：



# 分析

(1) 候选码:

推销员号+物资号

(2) 存在的函数依赖:

非主属性对码存在部分依赖

(3) 达到的范式级别:

属于1NF



# 分解:

推销员编号+物资号

销售数量

物资号

推销员编号

推销员姓名

物资名

推销员姓名

推销区域

物资价格



- 凡是满足3NF的关系，一般都能获得满意的效果。但是某些情况下，3NF仍会出现问题。
- 原因是没有对主属性与候选码之间给出任何限制，如果出现主属性部分或传递依赖于候选码，则也会使关系性能变坏。





## 五、BCNF(扩充的3NF)

定义：关系模式 $R(U, F) \in 1NF$ 。若  $X \rightarrow Y$  且  $Y \subseteq X$  时 $X$ 必含有候选码，  
则 $R(U, F) \in BCNF$ 。

即：关系模式 $R(U, F)$  中，若每一个决定因素都包含候选码，则 $R(U, F) \in BCNF$ 。



# 一个满足BCNF的关系模式有：

- 所有非主属性对每一个码都是完全函数依赖。
- 所有主属性对每一个不包含它的码也是完全函数依赖。
- 没有任何属性完全函数依赖于非码的任何一组属性。



**例：**关系模式SJP (S, J, P)

S: 学生 [学生选修课程有一定的名次]

J: 课程 [每门课程中每一名次只有一个学生]

P: 名次 (名次没有并列)

函数依赖:  $(S, J) \rightarrow P$

$(J, P) \rightarrow S$

分析得知:  $SJP \in 3NF$

$SJP \in BCNF$



**例：关系模式STJ (S, T, J)**

**S：** 学生 [某一学生选定某门课，就对应一个固定教师]

**T：** 教师 [每个教师只教一门课]

**J：** 课程 [每门课有若干教师]

**函数依赖：**  $T \rightarrow J$  ,  $(S, J) \rightarrow T$  ,  $(S, T) \rightarrow J$

**候选码：**  $(S, J)$        $(S, T)$



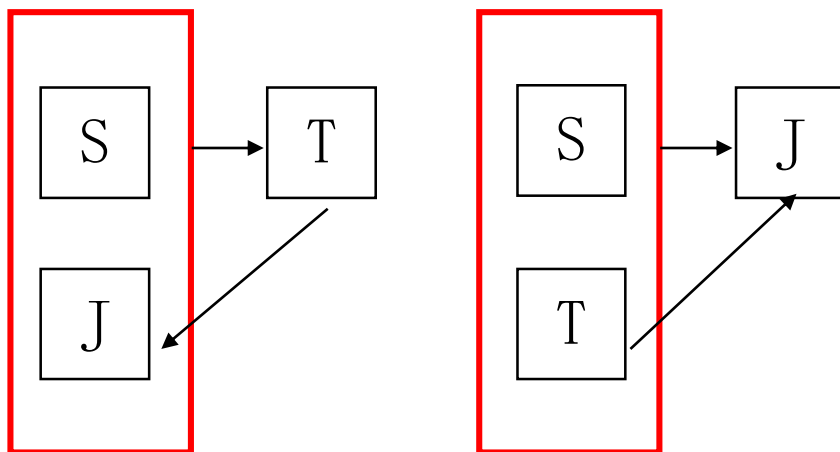
函数依赖:  $T \rightarrow J$  ,  $(S, J) \rightarrow T$  ,  $(S, T) \rightarrow J$

候选码

$(S, J)$      $(S, T)$

$STJ \in ? \text{ NF}$

$STJ \in 3\text{NF}$



$T \rightarrow J$ , 即T是决定属性集, 可是T只是主属性, 但它不是候选码。



## 存在的问题：

数据冗余度大：虽然一个教师只教一门课，但每个选修该教师该门课程的学生元组都要记录这一信息。

插入异常：如果某个教师开设了某门课程，但尚未有学生选修，则有关信息也无法存入数据库中。

删除异常：如果选修过某门课程的学生全部毕业了，在删除这些学生元组的同时，相应教师开设该门课程的信息也同时丢掉了。

修改异常：某个教师开设的某门课程改名后，所有选修了该教师该门课程的学生元组都要进行相应修改。



## BCNF的关系模式所具有的性质

所有非主属性都完全函数依赖于每个候选码。

所有主属性都完全函数依赖于每个不包含它的候选码。

没有任何属性完全函数依赖于非码的任何一组属性。

## 3NF与BCNF的关系

如果关系模式 $R \in \text{BCNF}$ ，必定有 $R \in 3\text{NF}$ 。

如果 $R \in 3\text{NF}$ ，且 $R$ 只有一个候选码，则 $R$ 必属于BCNF。

如果一个关系数据库中的所有关系模式都属于BCNF，那么在函数依赖范畴内，它已实现了模式的彻底分解，达到了最高的规范化程度，消除了插入异常和删除异常。



## 课堂练习—学习通04-1:

设有关系模式 $R(A, B, C, D)$ ，其数据依赖集:

$F = \{ (A, B) \rightarrow C, C \rightarrow D \}$ ，则关系模式 $R$ 的规范化程度最高达到（ ）。





练习：分析下列关系属于第几范式？

学生学习情况（学号，姓名，年龄，宿舍，学院，院长，课程号，课程名，先修课程，成绩）

候选码

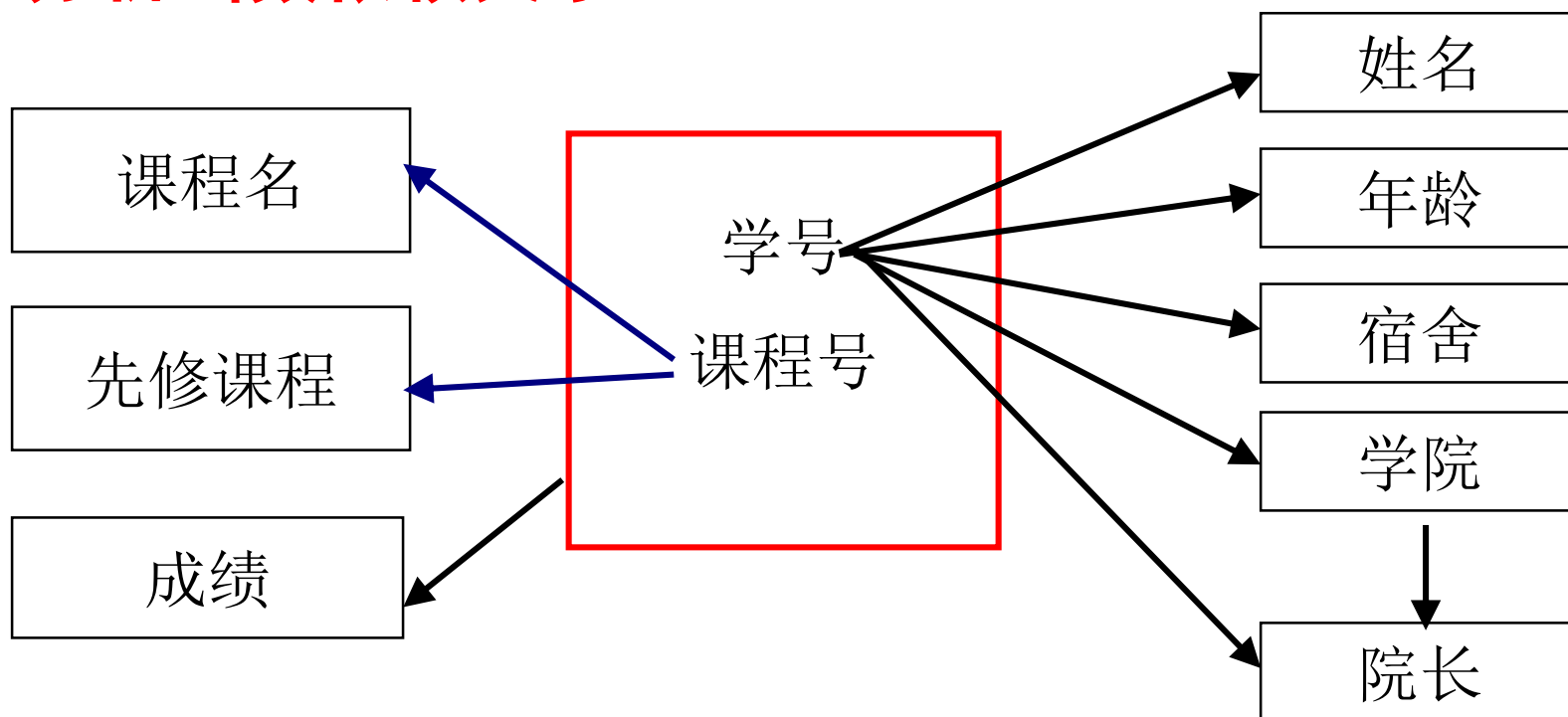
主属性

非主属性



1) 关键字： (学号, 课程号)

2) 分析函数依赖关系：



3) 判断： 属于1NF



分解:

学号+课程号

成绩

课程号

先修课程 课程名

学院

院长

学号

姓名

年龄

宿舍

学院

