范式分解

**主属性：**  包含在任一候选关键字中的属性称主属性。

**非主属性：** 不包含在主码中的属性称为非主属性。

**函数依赖：**

是指关系中一个或一组属性的值可以决定其它属性的值。函数依赖正象一个函数 y = f(x) 一样，x的值给定后，y的值也就唯一地确定了。

如果属性集合Y中每个属性的值构成的集合唯一地决定了属性集合X中每个属性的值构成的集合，则属性集合X函数依赖于属性集合Y，计为：Y→X。属性集合Y中的属性有时也称作函数依赖Y→X的决定因素（determinant）。例：身份证号→姓名。

**部分函数依赖：**

设X,Y是关系R的两个属性集合，存在X→Y，若X’是X的真子集，存在X’→Y，则称Y部分函数依赖于X。

**完全函数依赖：**

在R(U)中，如果Y函数依赖于X,并且对于X的任何一个真子集X'，都有Y不函数依赖于X'， 则称Y对X完全函数依赖。否则称Y对X部分函数依赖。

【例】；

举个例子就明白了。假设一个学生有几个属性

SNO 学号 SNAME 姓名 SDEPT系

SAGE 年龄 CNO 班级号 G 成绩

对于(SNO,SNAME,SDEPT,SAGE,CNO,G)来说，G完全依赖于(SNO, CNO), 因为(SNO,CNO)可以决定G，而SNO和CNO都不能单独决定G。

而SAGE部分函数依赖于(SNO,CNO),因为(SNO,CNO)可以决定SAGE，而单独的SNO也可以决定SAGE。

**传递函数依赖：**

设R（U）是属性集U上的关系，x、y、z是U的子集，在R（U）中，若x→y，但y→x，若y→z，则x→z，称z传递函数依赖于x，记作X→TZ。

如果X->Y, Y->Z, 则称Z对X传递函数依赖。

**计算X+ （属性的闭包） 算法：**

     a.初始化，令X+ = X;

     b.在F中依次查找每个没有被标记的函数依赖，若“左边属性集”包含于X+ ，则令 X+ = X+∪“右边属性集”, 并为访问过的函数依赖设置标记。

     c.反复执行b直到X+不改变为止。

**检验给定的任意函数依赖A1A2...An->B是否蕴含于依赖集S:**

分析：

根据属性集闭包的定义，可知A1A2...An->{A1,A2,...,An}+ 蕴含于S。只要证明B在{A1，A2，....,An}+中，那么函数依赖A1A2...An->B肯定蕴含于依赖集S中

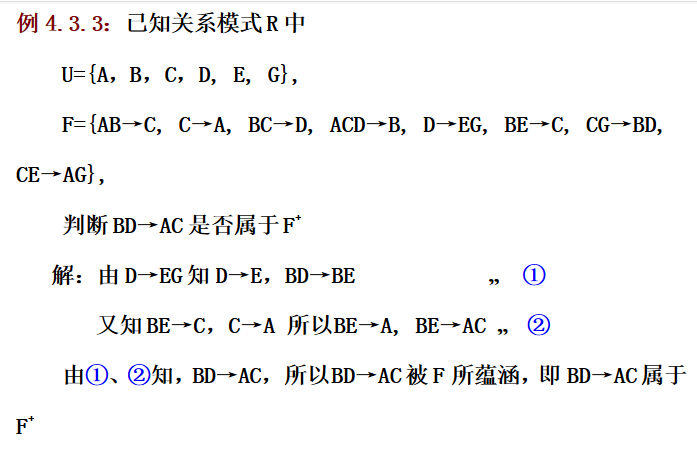
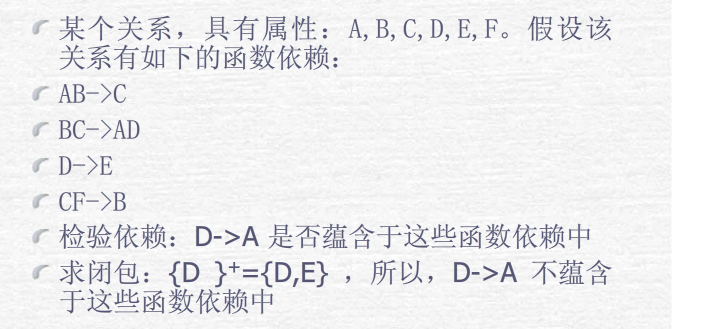
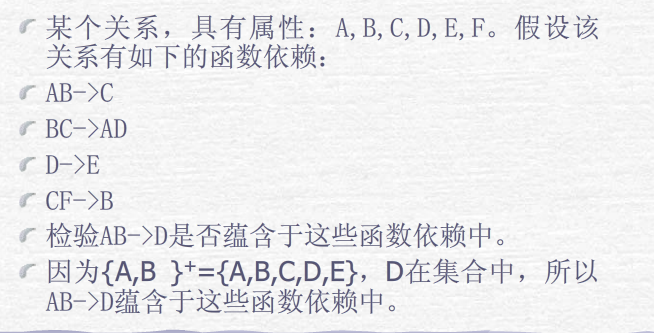
求解过程：

（1） 利用依赖集计算闭包

（2） 如果B在闭包中，则函数依赖A1A2...An->B是否蕴含于

依赖集S，否则不蕴含于S

【例】



**总结：**

判定函数依赖X→Y是否能由F导出的问题，         可转化为求X+并判定Y是否是X+子集的问题。

 即求F闭包的问题可转化为求属性集闭包的问题。

**函数依赖的闭包：**

定义：若F为关系模式R(U)的函数依赖集，我们把F以及所有被F逻辑蕴涵的函数依赖的集合称为F的闭包，记为F+

求函数依赖闭包，基于函数依赖推理规则

函数依赖推理规则：

若XY->Z，则X->Z，Y->z （错）

正确的：

若X->Y, 则XZ->YZ

若X->Y,X->Z，则X->YZ

若X->Y，Z属于Y，则X->Z

若X->Y，Y->Z，则X->Z

若X->YZ，则X->Y，X->Z //可以把每个函数依赖的右边的属性分解，从而使其右边只出现一个属性

伪传递率：

若A->B,BC->D,则AC->D

**范式**

**第一范式（1NF）**：属性，属性值，字段不可分

就是无重复的列

不满足1NF的数据库就不是关系数据库

【例】：

**第二范式（2NF）**：符合1NF，每一个非主属性**完全依赖**于码，**不能存在部分依赖**，有主键，非主键字段依赖主键; 唯一性 一个表只说明一个事物;

【例】：

不符合第二范式的例子:

表:学号, 姓名, 年龄, 课程名称, 成绩, 学分;

这个表明显说明了两个事务:学生信息, 课程信息;

存在问题:

数据冗余，每条记录都含有相同信息；

删除异常：删除所有学生成绩，就把课程信息全删除了；

插入异常：学生未选课，无法记录进数据库；

更新异常：调整课程学分，所有行都调整。

修正:

学生：Student(学号, 姓名, 年龄)；

课程：Course(课程名称, 学分)；

选课关系：SelectCourse(学号, 课程名称, 成绩)。

满足第2范式只消除了插入异常。

**第三范式（3NF）**：符合2NF，并且，消除传递依赖，非主键字段不能相互依赖; 每列都与主键有直接关系，**不存在传递依赖**;

**若所有的属性都是主属性，则属于第三范式**

要求一个数据库表中不包含已在其它表中已包含的非主关键字信息

【例】：

不符合第三范式的例子:

学号, 姓名, 年龄, 所在学院, 学院联系电话，关键字为单一关键字"学号";

存在依赖传递: (学号) → (所在学院) → (学院地点, 学院电话)

存在问题:

数据冗余:有重复值；

更新异常：有重复的冗余信息，修改时需要同时修改多条记录，否则会出现数据不一致的情况

删除异常

修正：

学生：(学号, 姓名, 年龄, 所在学院)；

学院：(学院, 地点, 电话)。

总结：1nf:不可分

2nf:一个表说明一个事物，唯一性

3nf:对字段冗余性的约束，即任何字段不能由其他字段派生出来，它要求字段没有冗余。

**bcnf:**是3NF的改进形式

**BCNF**意味着在关系模式中每一个决定因素都包含候选键，也就是说，只要属性或属性组A能够决定任何一个属性B，则A的子集中必须有候选键。BCNF范式排除了任何属性对候选键的传递依赖与部分依赖。

**满足BCNF条件**

1 所有非主属性对每一个候选键都是[完全函数依赖](http://baike.baidu.com/view/228997.htm)；

2 所有的主属性对每一个不包含它的候选键，也是完全函数依赖；

3 没有任何属性完全函数依赖于非候选键的任何一组属性。

**候选键** (又称候选码，候选关键字,码 ，candidate key)

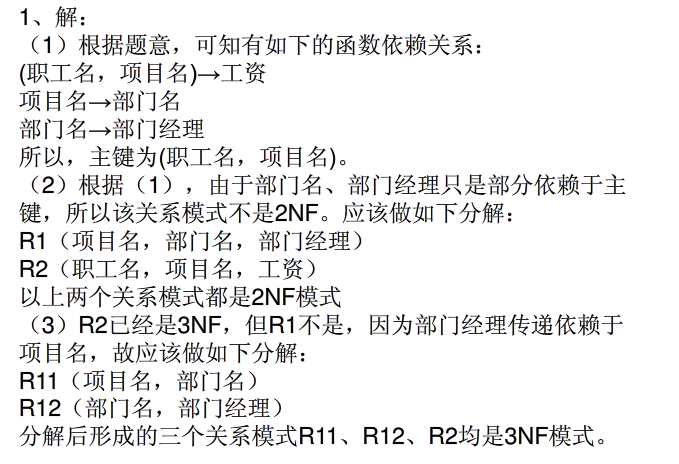
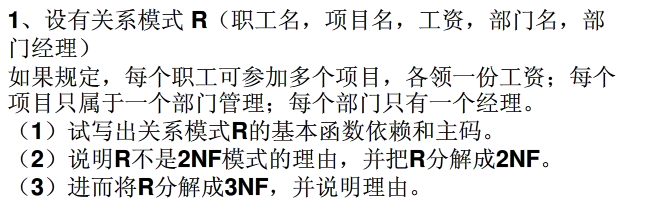
设K是一个R(U)中的属性或属性集合(注意可以是属性集合，也即多个属性的组合)，若K完全函数确定U，则K为R的候选键(Candidate key);

通俗地说就是，能够确定全部属性的某个属性或某组属性，称为候选键。若候选键多于一个，则选定其中一个作为主键。

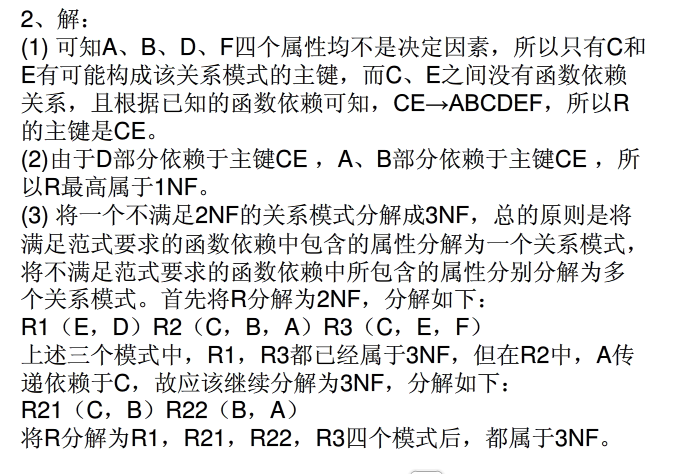
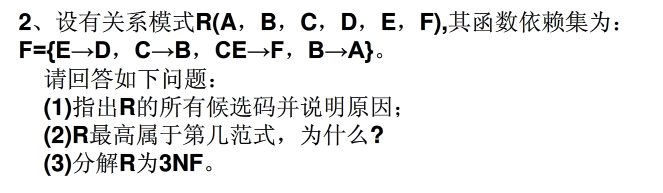
\*\*在所有依赖关系右边没有出现的属性一定是候选键的成员。

**BCNF范式排除了任何属性对候选键的传递依赖与部分依赖。**

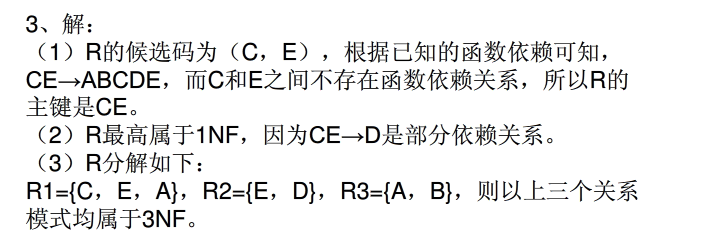
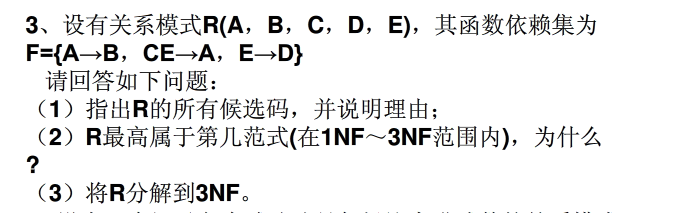
【例1】



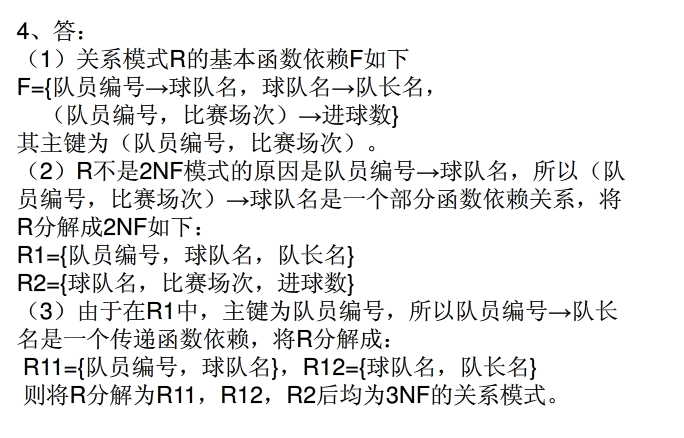
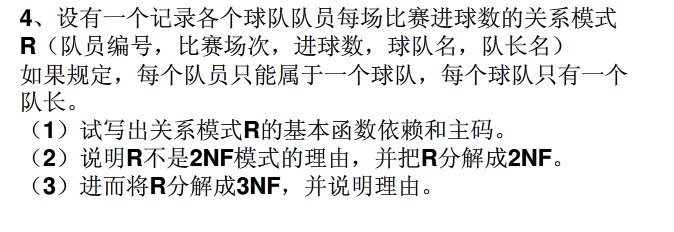
【例2】



【例3】



【例4】



**设有关系模式R（A，B，C，D，E，G）上的函数依赖集为：**

**F={ A→B，B→C，AD→G，D→E } 。求解：**

**31．求关系模式R的所有侯选键。**

解: 求出侯选键AD。（2分）

首先在F中函数依赖右边不出现的属性必在侯选键中,即AD (1分)；由于(AD)+=ABCDEG, 即AD能函数决定所有的属性, 所以侯选键只有一个 AD（1分） 。 AD+=AD BEG C

**32．分别求属性集G、AD、CD、BC的闭包。**

G+=G(1分)；      (AD)+=ABCDEG(1分)；

(CD)+= CDE(1分)；(BC)+=BC(1分)

**33．将关系模式R保持依赖地且无损地分解成3NF，要求写出分解过程。**

解：F={ A→B，B→C，AD→G，D→E }

F是最小依赖集，所有属性在F中出现，将F中是每个函数依赖组成一个关系模式得保持函数依赖的分解：{AB，BC，ADG， DE} (2分)； 并上一个侯选键{AD}得无损分解：

{AB，BC，ADG，DE}∪{AD}={ AB，BC，ADG，DE } (2分)

F={ A→B，B→C，AD→G，D→E }

**34．将关系模式R无损地分解成BCNF，要求写出分解过程。**

解：根据转换为BCNF的无损连接分解算法6.5  1）由于候选键为AD，

F中存在不符合BCNF要求的函数依赖，所以R不是BCNF，

选 A→B 分解为：R1=AB，R2=ACDEG ；(1分)

R1上保持的函数依赖集为A→B，键为A，所以是BCNF；

R2 上保持的函数依赖集为A→C，AD→G，D→E，键为AD，所以不是BCNF； (1分)

选A→C进一步分解为：R21=AC，R22=ADEG；(1分) R21上保持的函数依赖为A→C，键为A，所以是BCNF；

R22上保持的函数依赖为AD→G，D→E键为AD，所以不是BCNF； 选D→E进一步分解为：R221=DE，R222=ADG；(1分) R221上保持的函数依赖为D→E，键为D，所以是BCNF； R222上保持的函数依赖为AD→G，键为AD，所以是BCNF；

最后得保持无损连接特征的分解：{R1，R21，R221，R222}或表示为{AB，AC，DE，ADG}(1分)

注：由于选择不符合BCNF要求的函数依赖有多个，因此选择次序可有不同，最后的结果也不同，原则上按上述评分标准分步给分。

**35．说明分解ρ={R1，R2}，R1（ABC）、R2（ADEG）的范式级别并说明理由**

 答：R1是2NF (1分)，R2是1NF；(1分)

R1上的函数依赖集为：A→B，B→C，码为：A，不存在部分依赖，存在非主属性C对码A的传递依赖。(1分)

R2上的函数依赖集为：AD→G，D→E码为：AD，存在非主属性E对码AD的部分依赖。。(1分)

ρ={R1，R2}，R1（ABC）、R2（ADEG）