

数据库设计

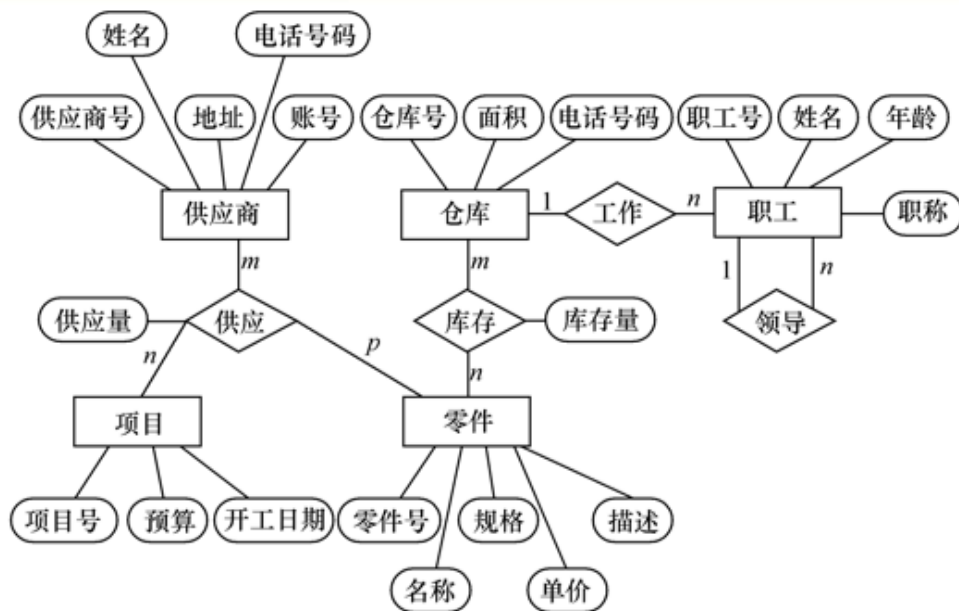
E-R模型图转成关系

授课教师：秦建斌

邮箱：qinjianbin@szu.edu.cn

深圳大学 计算机与软件学院

回顾



□ E-R (Entity-Relationship) 实体-关系模型

□ E-R图提供了表示实体型、属性和联系的方法

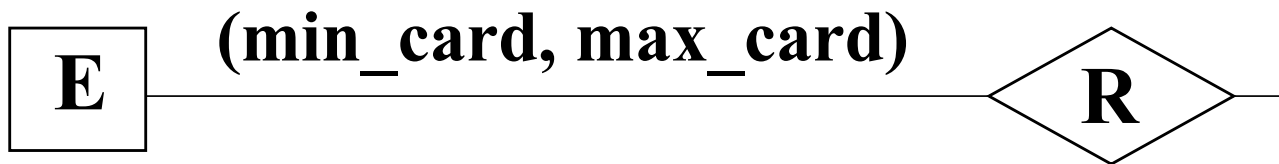
□ 实体型VS实体集

□ 属性：简单属性，复合属性，多值属性

□ 联系：度，一元，二元，三元关系；1:m, 1:1, n:m联系

回顾

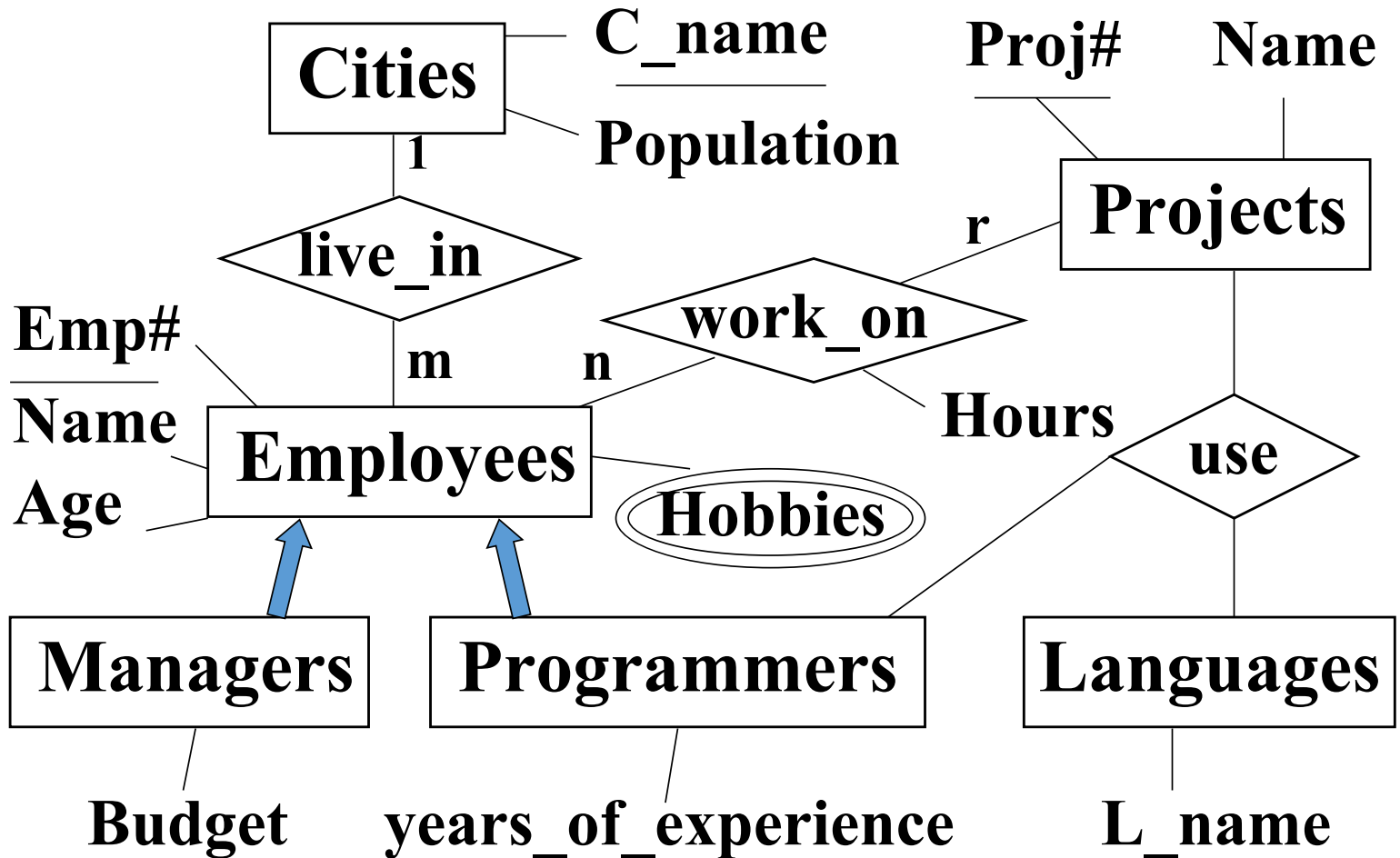
- 基数约束：说明实体型中的任何一个实体可以在联系中出现的**最少次数**和**最多次数**。



- $0 \leq \text{min_card} \leq \text{max_card}$
- 解释: E中的每个实体都可能涉及R中**min_card**和**max_card**之间的关系。

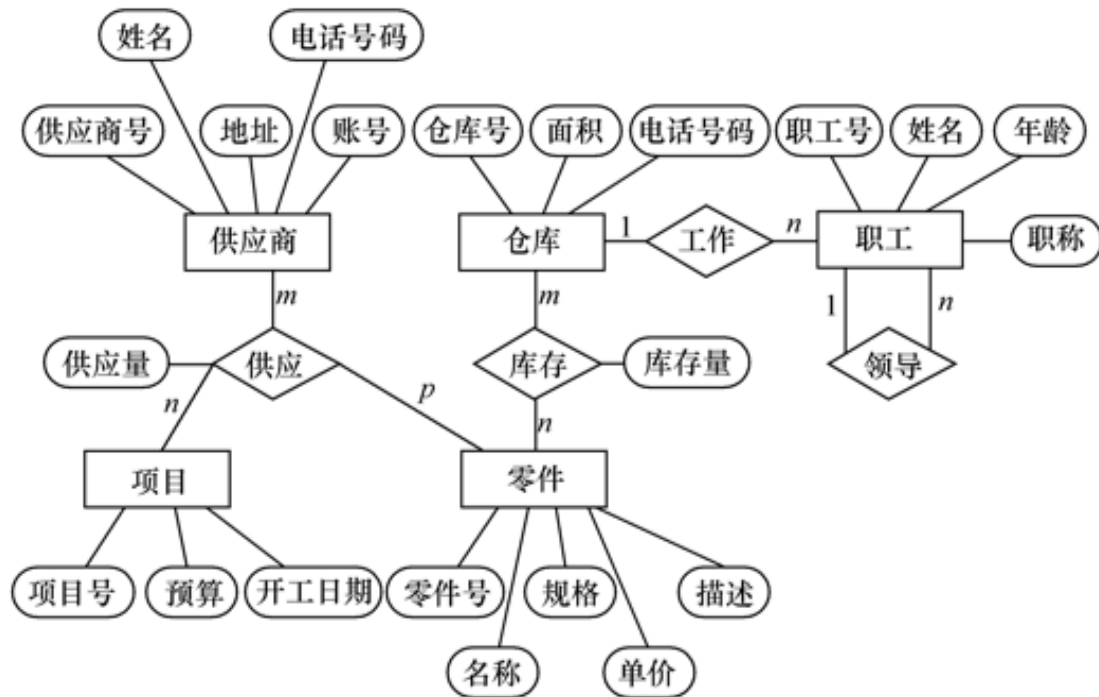


An Example EER Diagram



E-R图向关系模型的转换

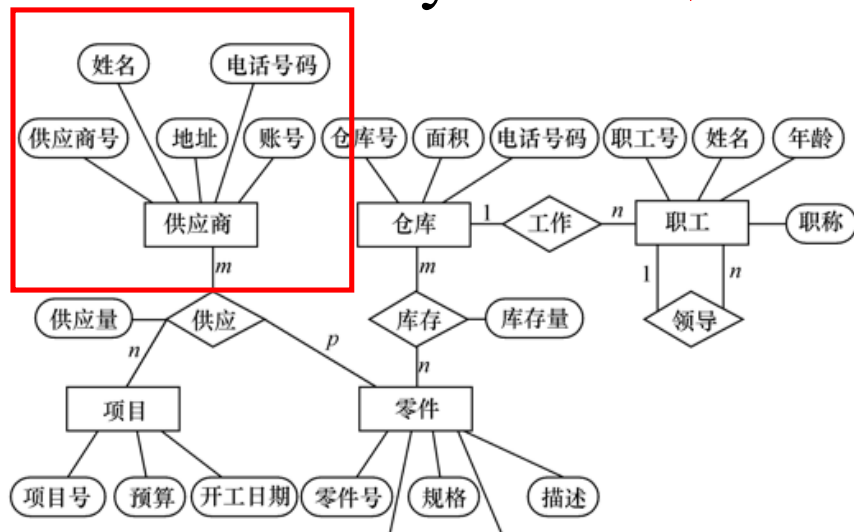
□ 将E-R图转换为关系模型：将实体型、实体的属性和实体型之间的联系转化为关系模式



E-R图向关系模型的转换

□ 转型的基本思路：实体型和属性的转换

Entity	====>	Tuple	一个实体转换为一个元组。
Entity set	====>	Relation	一个实体集转换为一个关系模式。
Attribute	====>	Attribute	属性转换为属性。
Key	====>	Key	码转换为码。

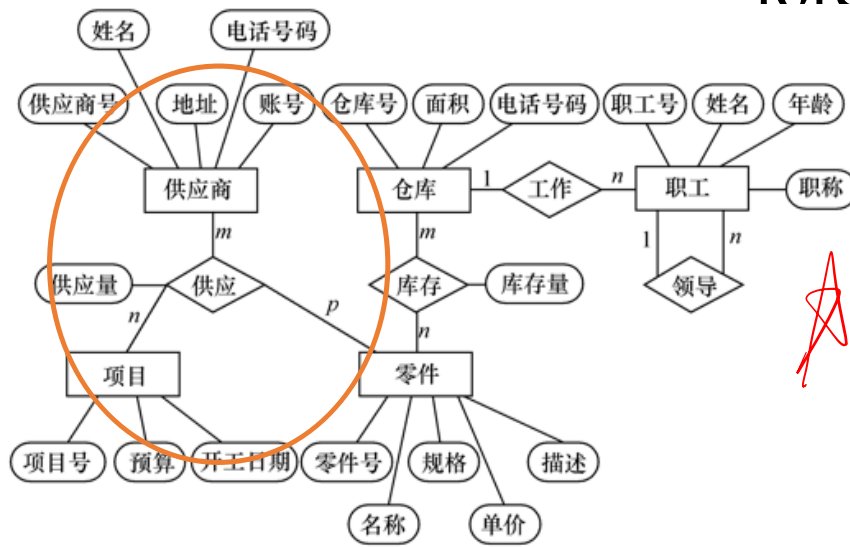


E-R图向关系模型的转换

□ 转型的基本思路：联系（关系）的转换

Relationship \Longrightarrow Tuple or foreign key value(s)

Relationship set \Longrightarrow Relation or foreign key(s)

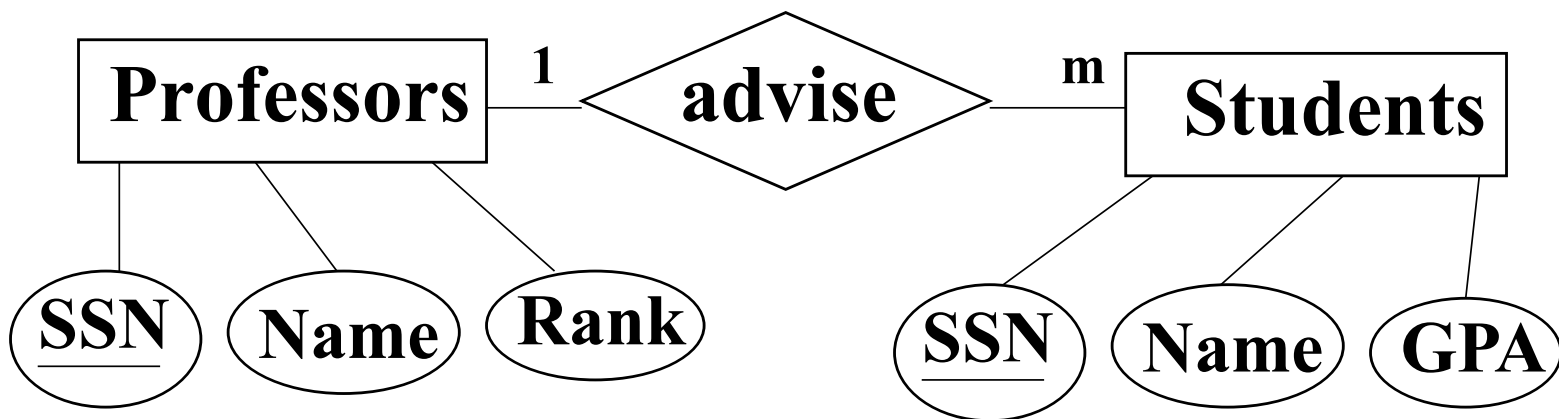


□ 一个关系转换为一个独立关系模式，

□ 或者与一端的关系合并，引用外键。

如何选择？根据关系的类型和度
如：1:1, 1:m, n:m

1 to m 关系转换



Professors	Advise	Students
p1: 123, Jack, Prof.	p1 advises s1	s1: 456, John, 3.4
p2: 234, Ann, Prof.	p1 advises s2	s2: 567, Carl, 3.2
p3: 345, Bob, Prof.	p3 advises s3	s3: 678, Ken, 3.5

一对多 { ①张表

1 to m 关系转换

创建

② 数据流

转换为3个关系表格

Professors	Advise	Students
SSN Name Rank	PSSN SSSN	SSN Name GPA
123 Jack Prof.	123 456	456 John 3.4
234 Ann Prof.	123 567	567 Carl 3.2
345 Bob Prof.	345 678	678 Ken 3.5

联系Advise 转换为一个独立的关系模式

关系的属性：与该联系相连的各实体的码以及联系本身的属性

1 to m 关系转换

转换为2个关系表格

Professors

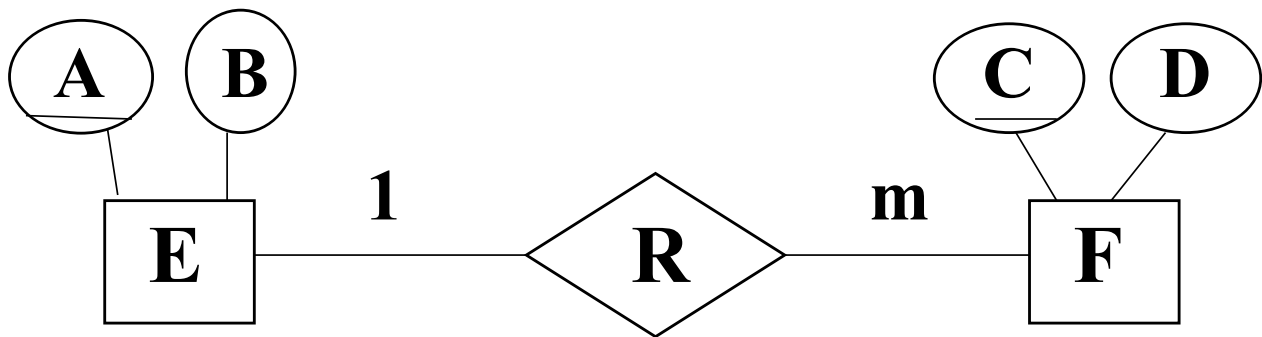
SSN	Name	Rank
123	Jack	Prof.
234	Ann	Prof.
345	Bob	Prof.

Students

SSN	Name	GPA	PSSN
456	John	3.4	123
567	Carl	3.2	123
678	Ken	3.5	345

联系Advise 与Students 合并，引入外键

1 to m 关系转换规则



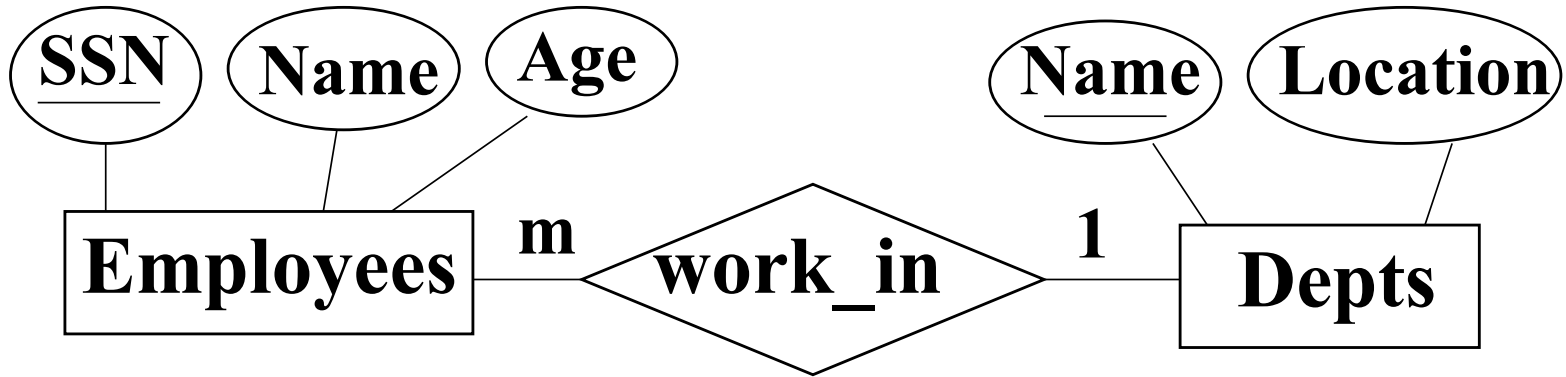
Case 1: 1 to m 关系: $x = 1$ 和 $y = m$ (或者 $x = (?, m)$ 和 $y = (?, 1)$)

$\implies E(A, B), F(C, D, A)$

❑ 关系 R 与 m 端对应的关系模式合并，在 m 端关系中加入 1 端关系的码（做为外键）

可以减少系统中的关系个数，一般情况下更倾向于采用这种方法

m to 1关系 与 (1 to m)关系相似

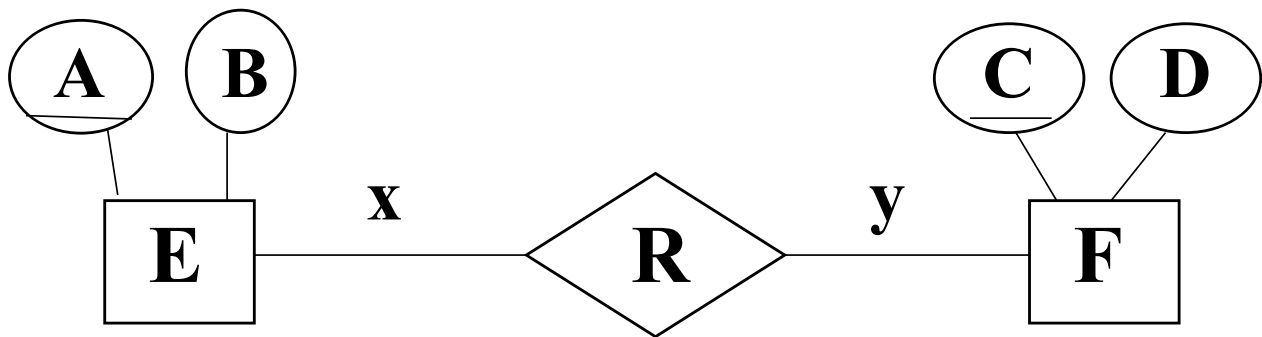


==> Depts(**Name**, Location)

Employees(**SSN**, **Name**, Age, **Dept_name**)

- ❑ 关系 R 与 m 端对应的关系模式合并，在 m 端关系中加入 1 端关系的码（做为外键）

1 to 1 关系转换



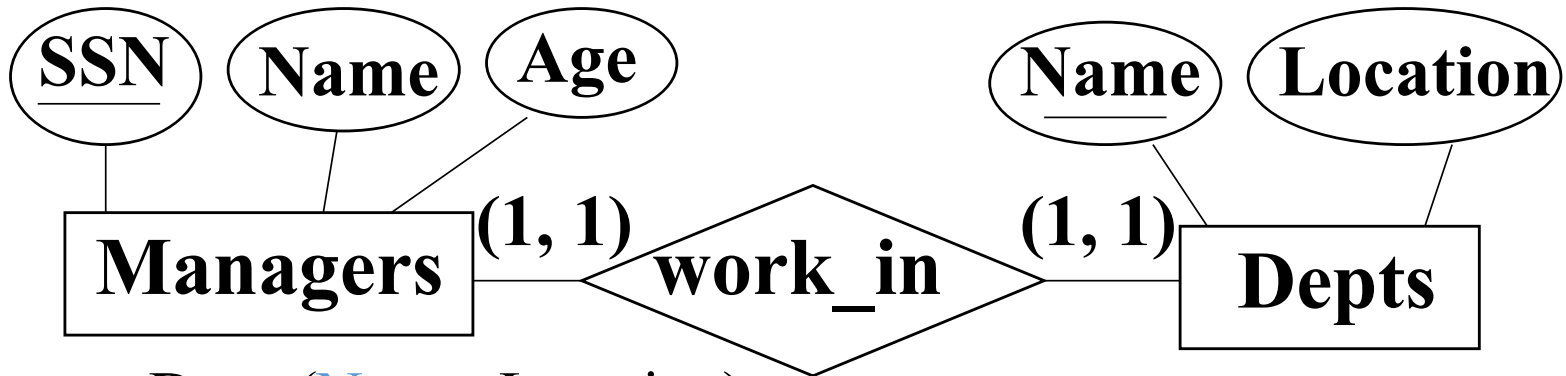
Case 1: 1 to 1 关系(强制参与) i.e $x = (1, 1)$ and $y = (1, 1)$

关系 R 与某一端实体对应的关系模式合并，加入对应关系的码做为外键

$\implies E(A, B), F(C, D, A)$ or

$\implies E(A, B, C), F(C, D)$

1 to 1 关系转换



==> Depts(**Name**, Location)

Managers(**SSN**, **Name**, **Age**, **Dept_name**)

OR

==> Depts(**Name**, Location, **Manager_SSN**)

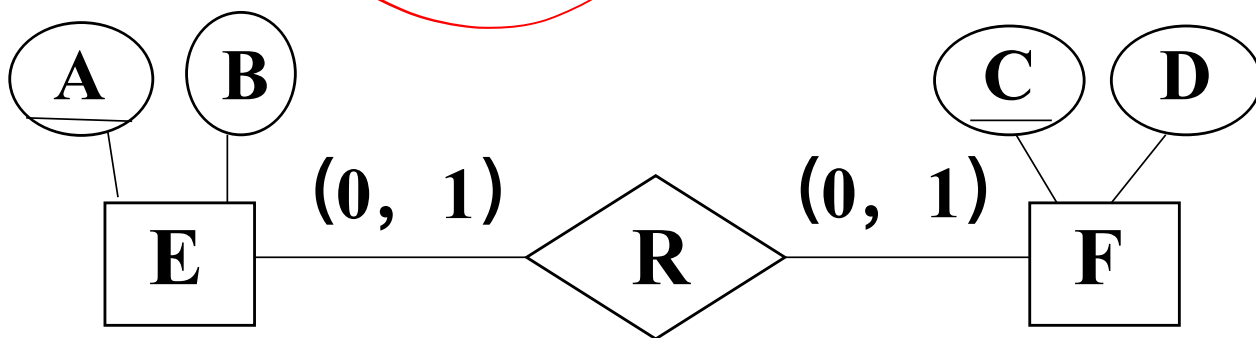
Managers(**SSN**, **Name**, **Age**)

1 to 1 关系转换

Case 2 (部分参与) : $x = (0, 1)$ and $y = (0, 1)$

(E 和 F 部分参与关系)

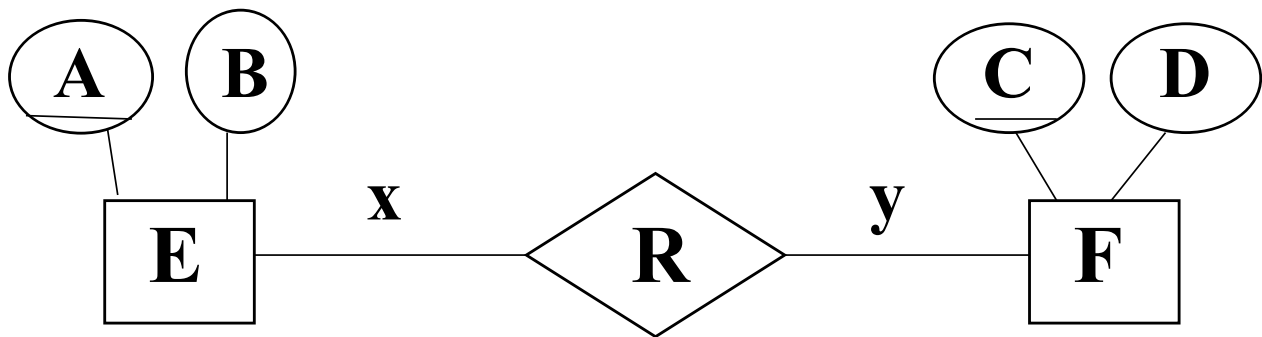
□与 1 to 1 关系(强制参与)同样处理



$\implies E(A, B), F(C, D, A)$ or

$\implies E(A, B, C), F(C, D)$

1 to 1 关系转换



Case3 (一个强制参与, 一个部分参与) : $x = (0, 1)$
and $y = (1, 1)$

$\implies E(A, B), F(C, D, A)$

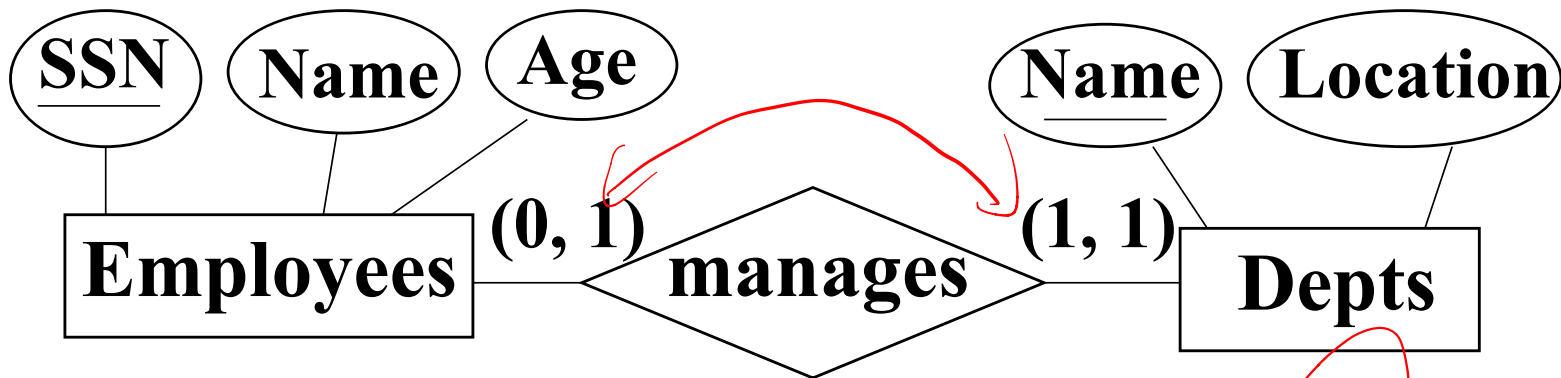
□ 关系 R 并入强制参与的一端, 引入外键。

参与多的那边。

防止产生过多空值。

1 to 1 关系转换

Case3 (一个强制参与, 一个部分参与) 举例



==> Depts(**Name**, Location, **Manager_SSN**)

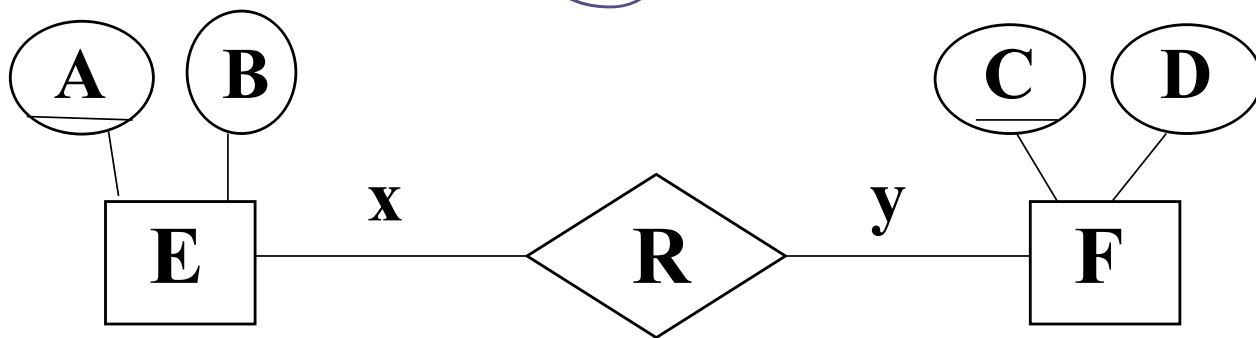
Employees(**SSN**, **Name**, **Age**)

❑ 可不可以把关系manages并入Employees?

会出现很多NULL, 有效性差

二元m to n关系转换

一个 $m:n$ 联系转换为一个独立的关系模式



$x = m, y = n$

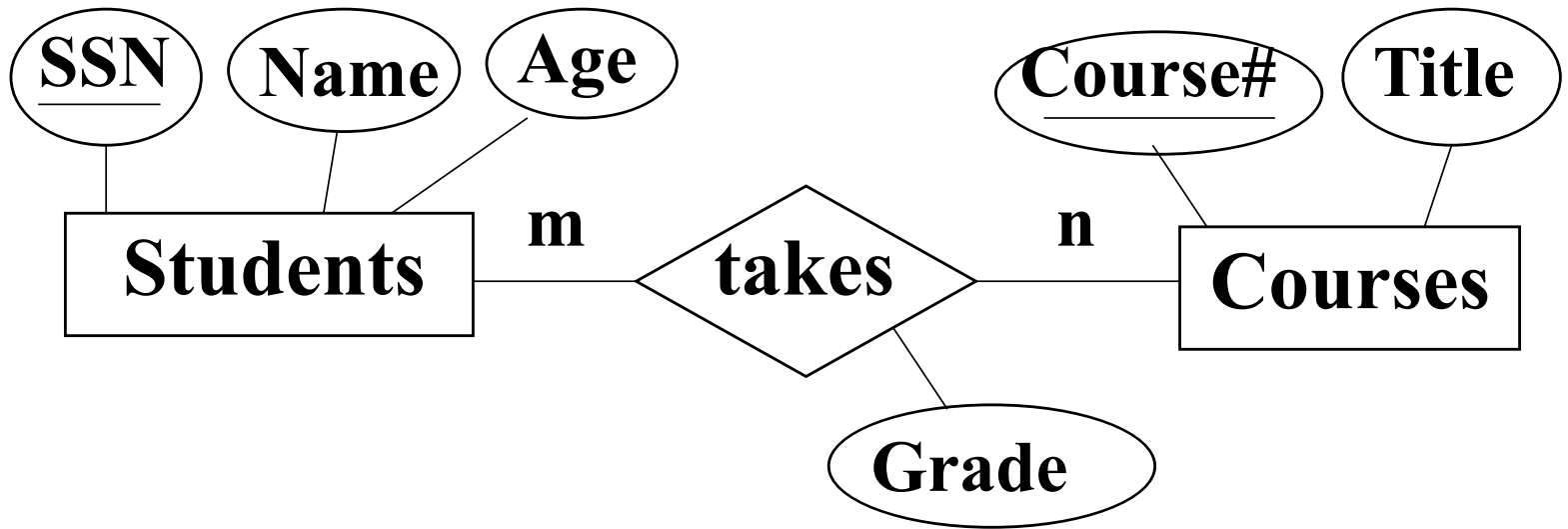
Case1: 关系没有属性 $\implies E(A, B), F(C, D), R(A, C)$

□ 一个 $m:n$ 联系转换为一个独立的关系模式.

□ 含两个外键.

2个外键

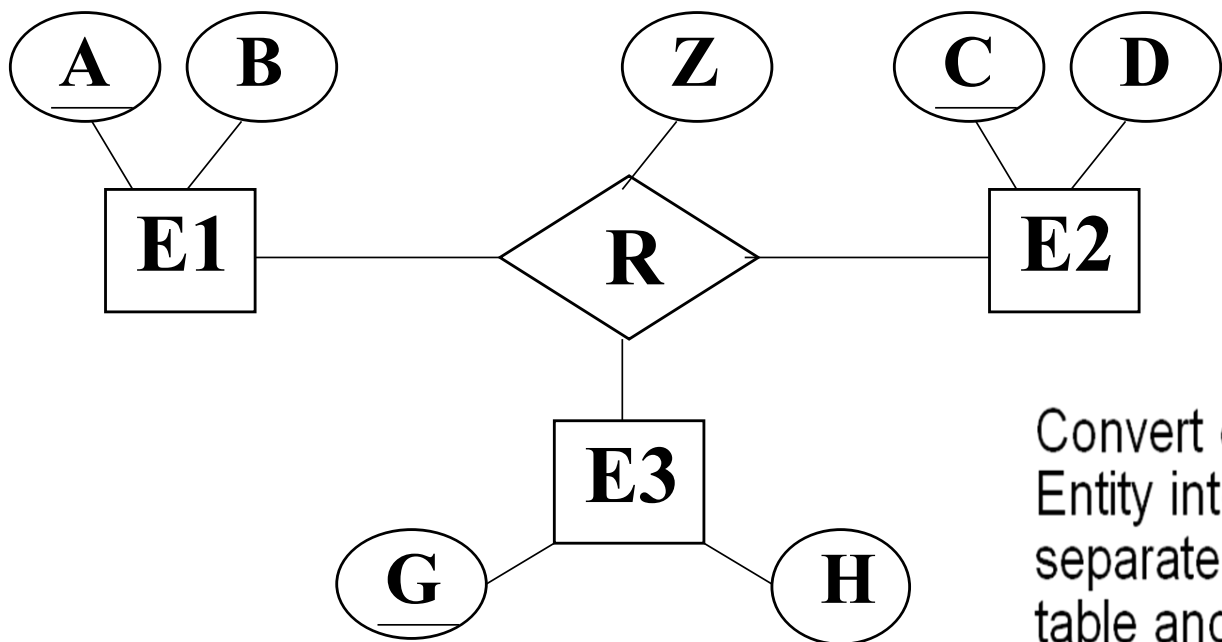
二元m to n关系转换



Case2: 关系有属性

==> Students(**SSN**, Name, Age), Course(**Course#**, Title),
takes(**SSN**, **Course#**, Grade)

三元关系转换



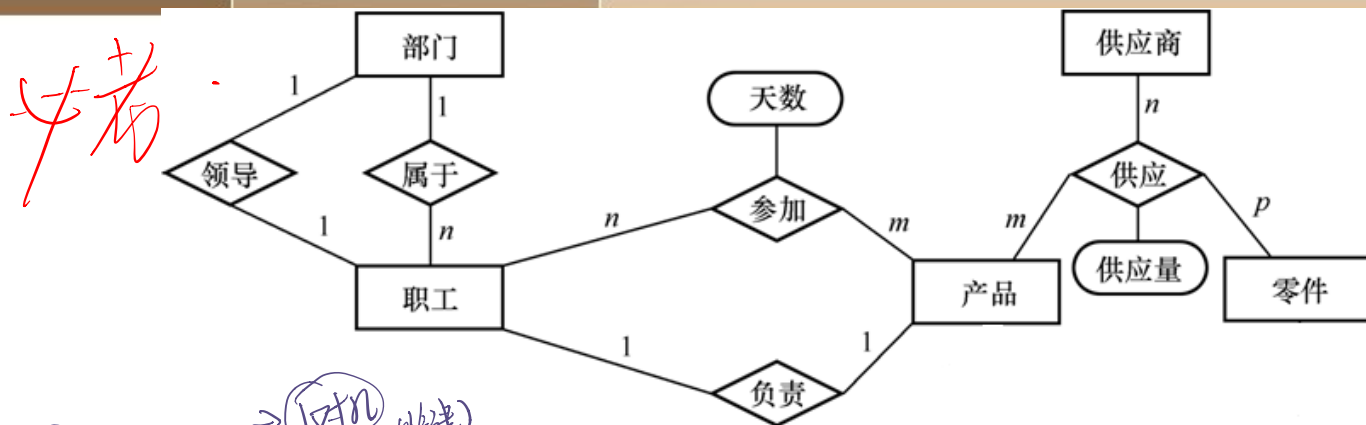
$\Rightarrow E1(A, B), E2(C, D), E3(G, H),$
 $R(\underline{A}, \underline{C}, \underline{G}, Z)$

三元转换

Convert each Entity into separate table and the relationship into a new Table

三个或三个以上实体间的一个多元联系转换为一个独立关系模式。

关系转换示例

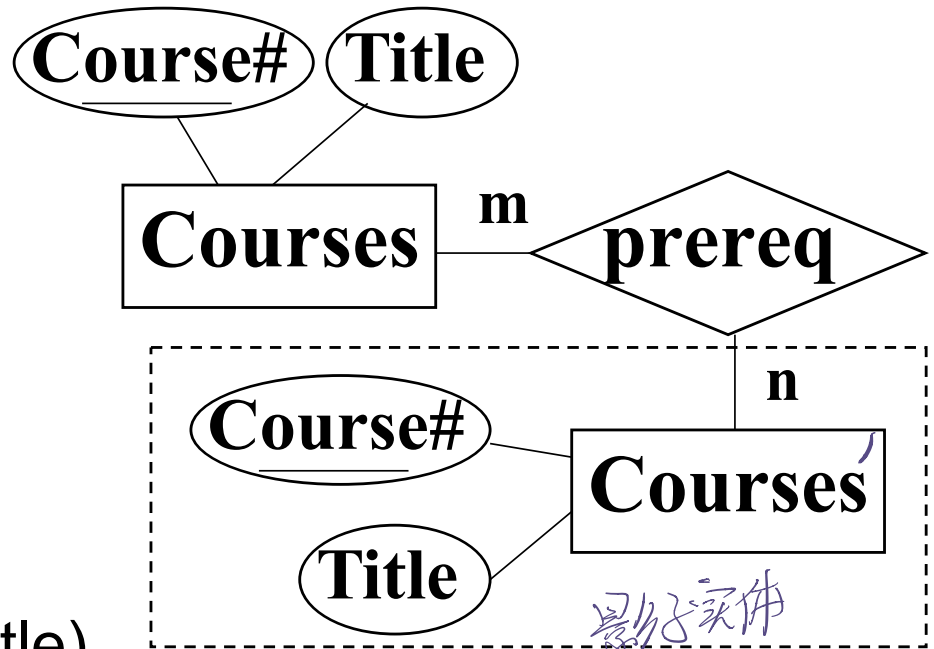
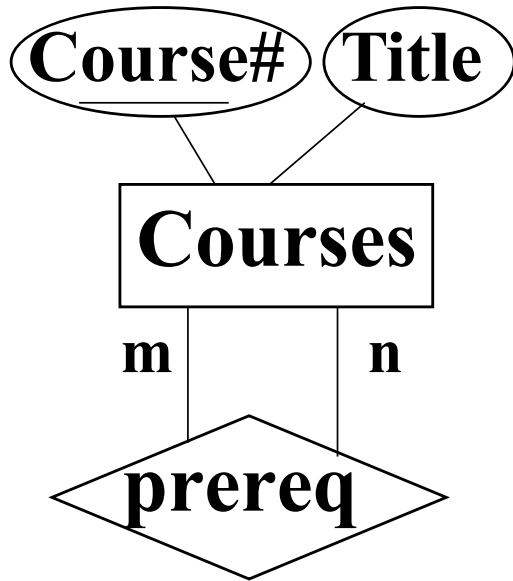


- 部门 (部门号, 部门名, 经理的职工号, ...)
- 职工 (职工号, 部门号, 职工名, 职务, ...)
- 产品 (产品号, 产品名, 产品组长的职工号, ...)
- 供应商 (供应商号, 姓名, ...)
- 零件 (零件号, 零件名, ...)
- 职工工作 (职工号, 产品号, 工作天数, ...)
- 供应 (产品号, 供应商号, 零件号, 供应量)

一元关系如何转换？

- 创建一个影子实体集
- 并将一元关系转换为二元关系，应用二元关系转换的规则。
- 转换后，删除一个冗余关系，如果没有冗余关系，则删除属性较少的关系。

一元关系转换

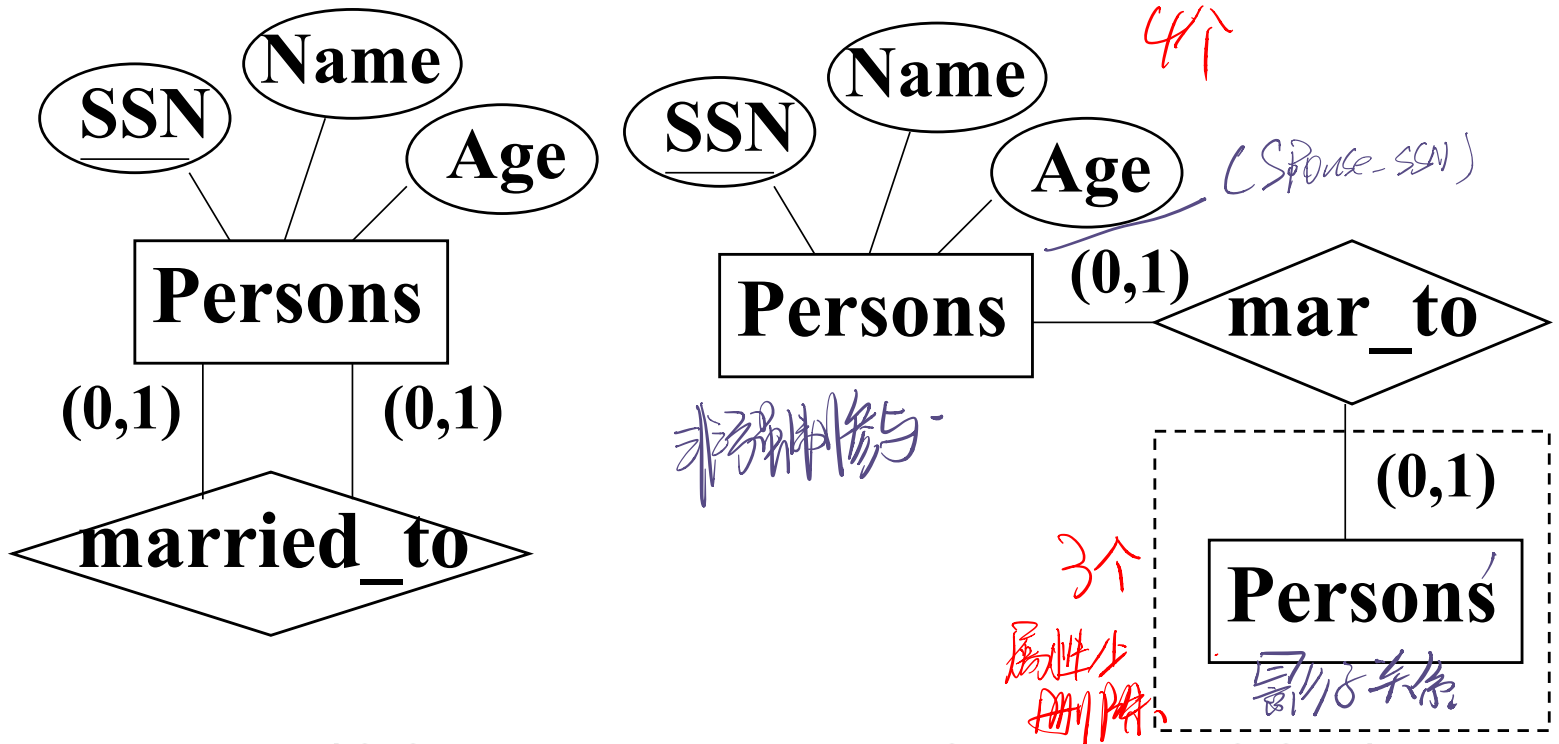


Courses(Course#, Title)

Prereq(Course#, Prereq_Course#)

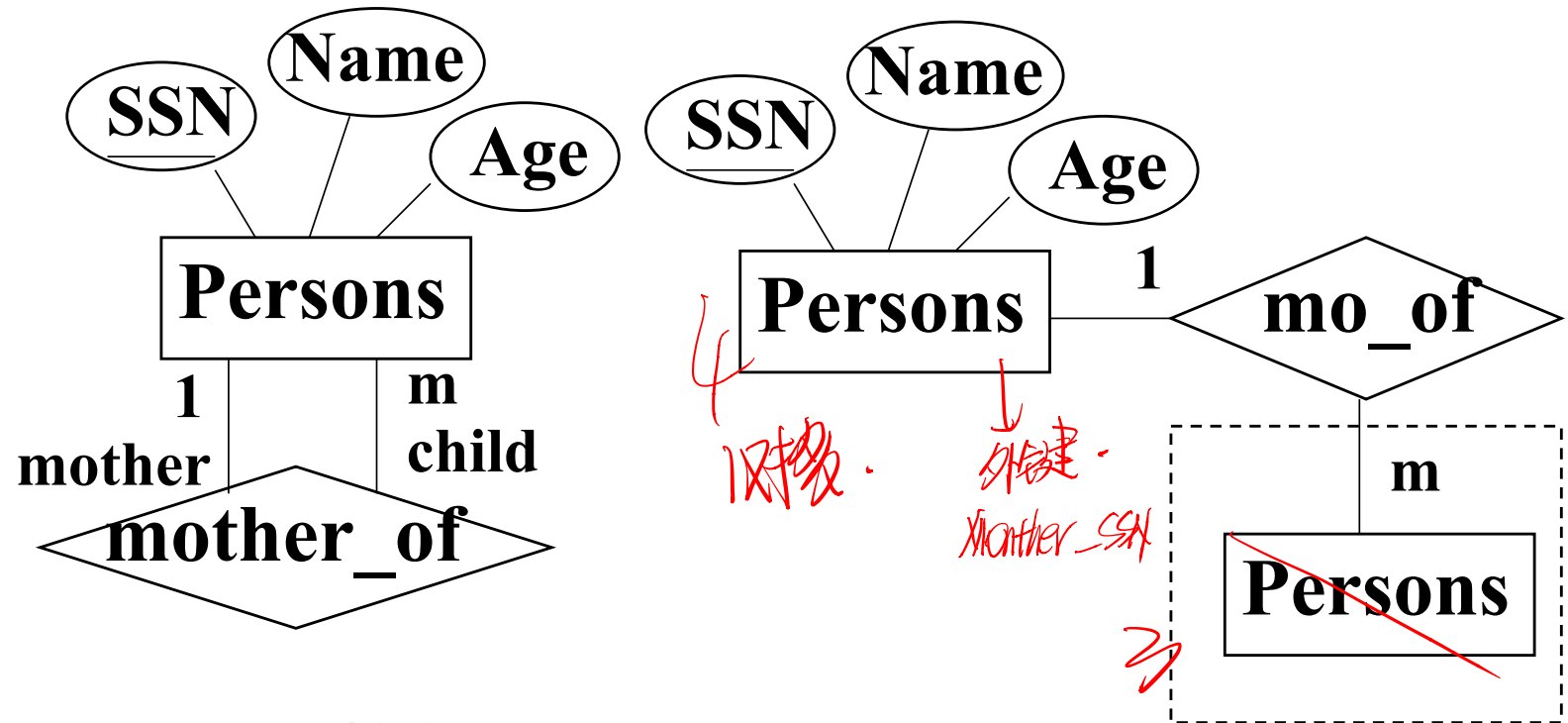
Courses'(Course#, Title) ← 删除冗余表

一元关系转换



Persons(SSN, Name, Age, Spouse_SSN)

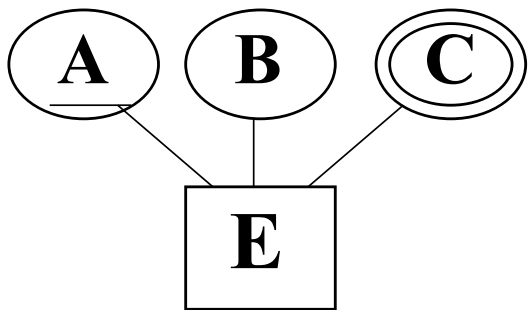
一元关系转换



Persons(SSN, Name, Age, Mother_SSN)

多值属性转换

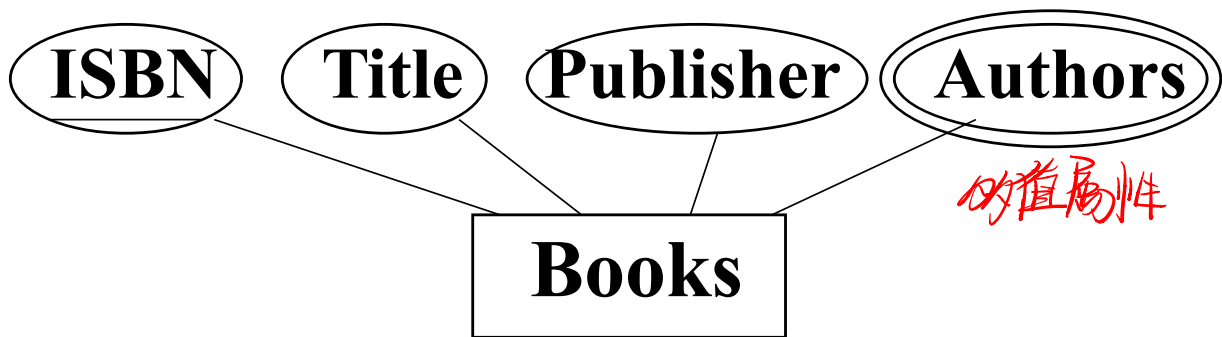
□ 给多值属性创建一个独立的关系模式



$E(\underline{A}, B), E_C(\underline{A}, C)$

□ $E_C.A$ 被定义为外键, 参考 $E.A$

多值属性转换



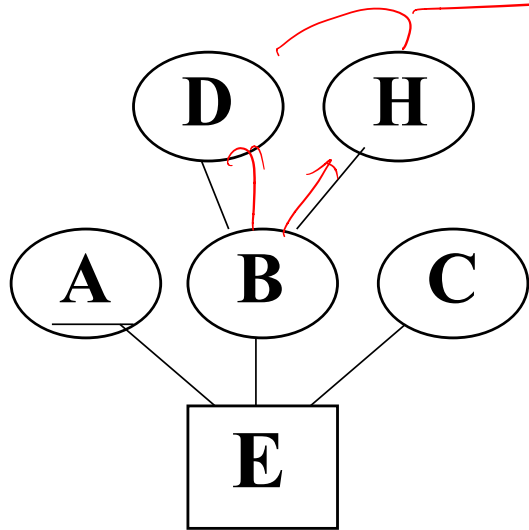
==> Books (**ISBN**, Title, Publisher)

Book_Authors (**ISBN**, **Author**) 3NF

□ Book_Authors.ISBN 是一个参考 Books.ISBN 的外键

复合属性转换

Method 1: 使用简单属性, 忽略复合属性



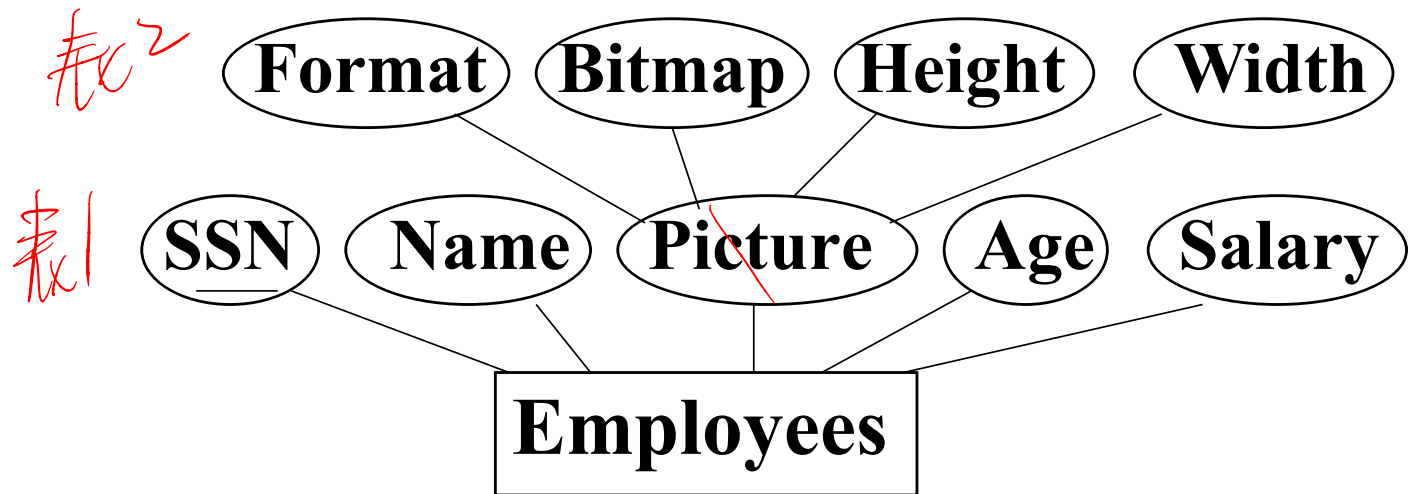
$\Rightarrow E(\textcolor{blue}{A}, D, H, C)$

分解
属性

Method 2: 把复合属性转换为一个独立的关系

$\Rightarrow E(\textcolor{blue}{A}, C),$
 $E_B(\textcolor{blue}{A}, D, H)$

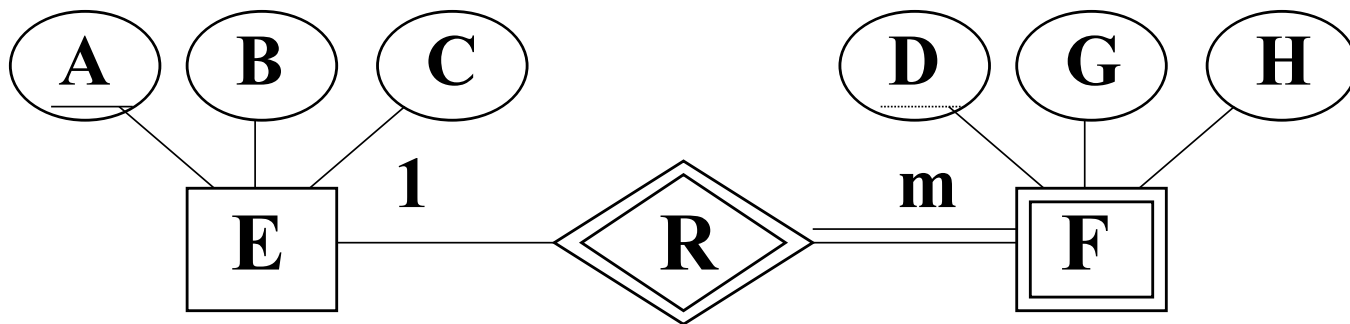
复合属性转换



Employees (SSN, Name, Age, Salary)

Emp_Pic (SSN, Bitmap, Format, Height, Width)

弱实体集转换

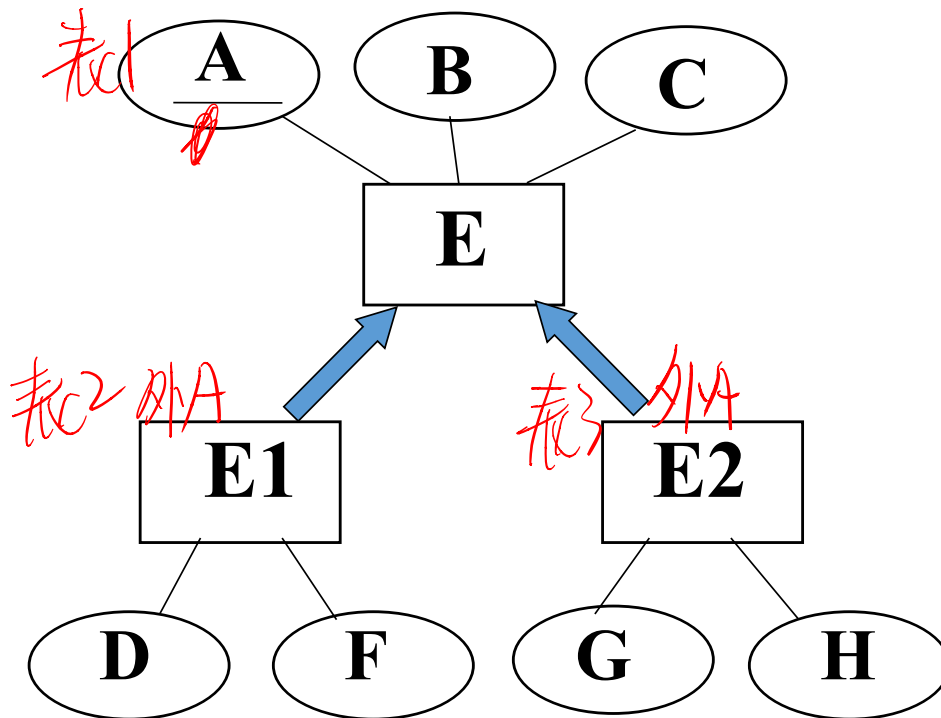


$\Rightarrow E(A, B, C), F(A, D, G, H)$

□ F 的码包含 A 和 D.

□ F.A 是参考 E.A 的外键

IS_A 层次结构转换

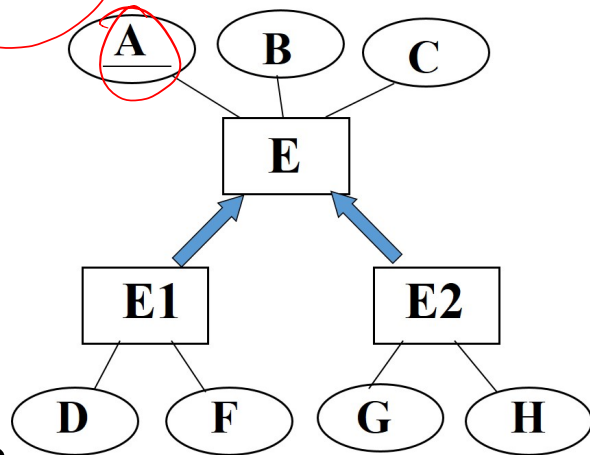


IS_A 层次结构转换

Method 1: \Rightarrow E(A, B, C), E1(A, D, F),
E2(A, G, H)

□ 子类从父类中继承码

□ E1.A 和 E2.A 是参考 E.A 的外键。



Method 2: \Rightarrow E(A, B, C),

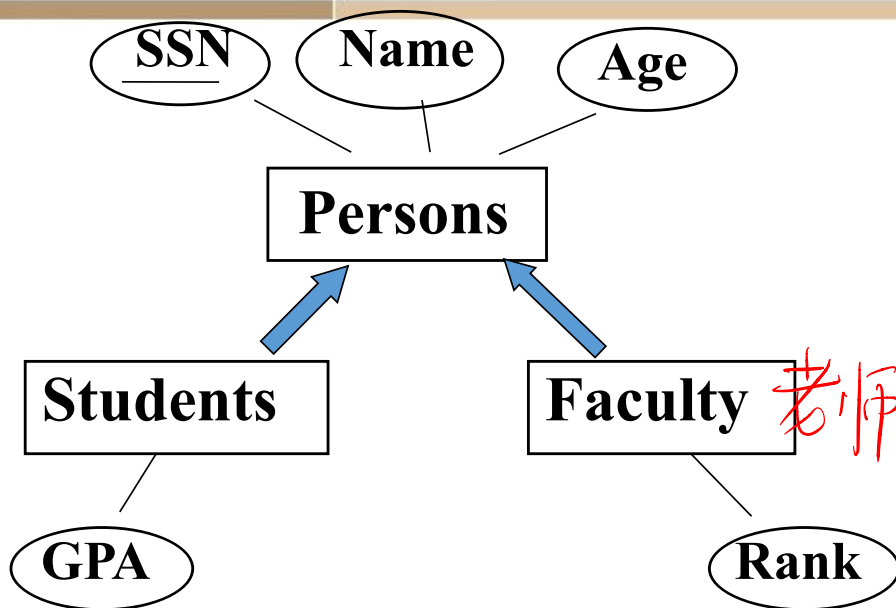
子类 E1(A, D, F, B, C), E2(A, G, H, B, C)

□ 子类从父类中继承所有属性

□ E中包含的元组不属于任何子类

□ 如果E 为空，丢弃。

IS_A 层次结构转换



□ Real world information:

	SSN	Name	Age	GPA	Rank
stud:	123456789	John	27	3.5	
facul:	234567891	Bill	43		Prof.
staff:	345678912	Mary	37		

IS_A 层次结构转换

□ Method 1:

Persons

SSN	Name	Age
123456789	John	27
234567891	Bill	43
345678912	Mary	37

Students

SSN	GPA
123456789	3.5

Faculty

SSN	Rank
234567891	Prof.

IS_A 层次结构转换

□ Method 2:

Persons

SSN	Name	Age
345678912	Mary	37

Students

SSN	Name	Age	GPA
123456789	John	27	3.5

Faculty

SSN	Name	Age	Rank
234567891	Bill	43	Prof.

复杂EER 图转换

规则:

□ 一个实体集转换为一个关系模式(不包含复合属性和多值属性).

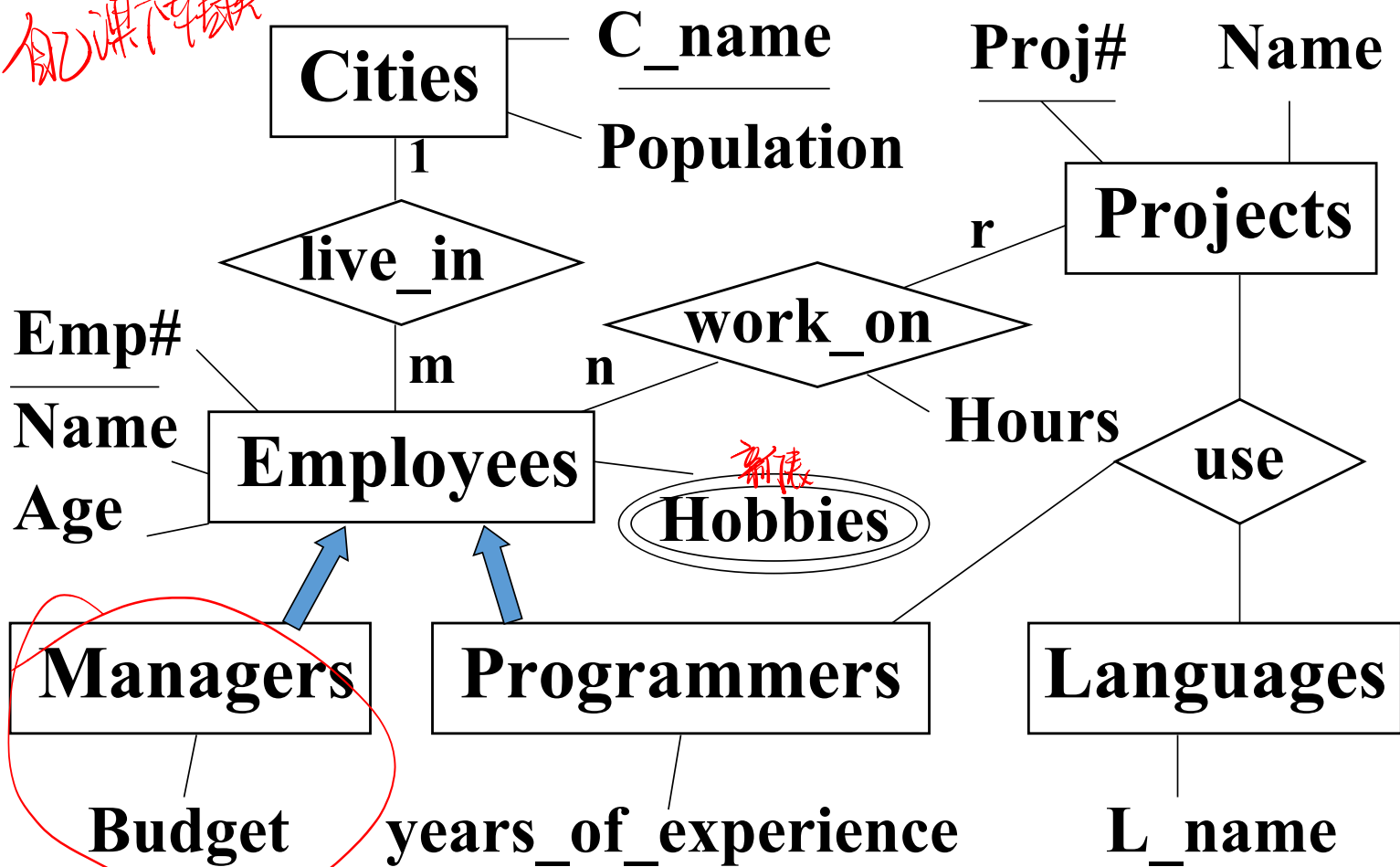
- 转换IS_A 层次结构, 从上往下.
- 转换多值属性为独立的关系模式.
- 为每个关系确定码

□ 转换联系 (关系) .

- 一元、二元1-to-1 or 1-to-m 关系, 和并关系到一端, 并添加外键.
- 多元关系 (>2) ,联系转换为独立的关系模式, 添加码和外键

复杂EER 图转换

配课下节课



复杂EER 图转换

Use method 1:

Employees(**Emp#**, Name, Age, C_name)

Employee-Hobby(**Emp#**, **Hobby**)

Managers(**Emp#**, Budget)

Programmers(**Emp#**, Years_of_experience)

Cities(**C_name**, Population)

Projects(**Proj#**, Name)

Languages(**L_name**)

Work_on(**Emp#**, **Proj#**, Hours)

Use(**Emp#**, **Proj#**, **L_name**)

复杂EER 图转换

Use method 2:

Employees(**Emp#**, Name, Age, C_name)

Employee-Hobby(**Emp#**, **Hobby**)

Managers(**Manager-Emp#**, Name, Age, Budget, C_name)

Manager-Hobby(**Manager-Emp#**, **Hobby**)

Programmers(**Programmer-Emp#**, Name, Age,
Years_of_experience, C_name)

Programmer-Hobby(**Programmer-Emp#**, **Hobby**)

复杂EER 图转换

Cities(C_name, Population)

Projects(Proj#, Name)

Languages(L_name)

Work_on(Emp#, Proj#, Hours)

Manager-Work_on(Manager-Emp#, Proj#, Hours)

Programmer-Work_on(Programmer-Emp#, Proj#, Hours)

Use(Programmer-Emp#, Proj#, L_name)

把下列E-R模型转换为关系模型

7、学校中有若干系,每个系有若干班级与教研室,每个教研室有若干教员,其中有的教授与副教授每人各带若干研究生;每个班有若干学生,每个学生选修若干课程,每门课可由若干学生选修。请用 E-R 图画出此学校的概念模型。

E-R 图中各实体的属性假设为:

系:系编号,系名

班级:班级编号,班级名

教研室:教研室编号,教研室

学生:学号,姓名,学历

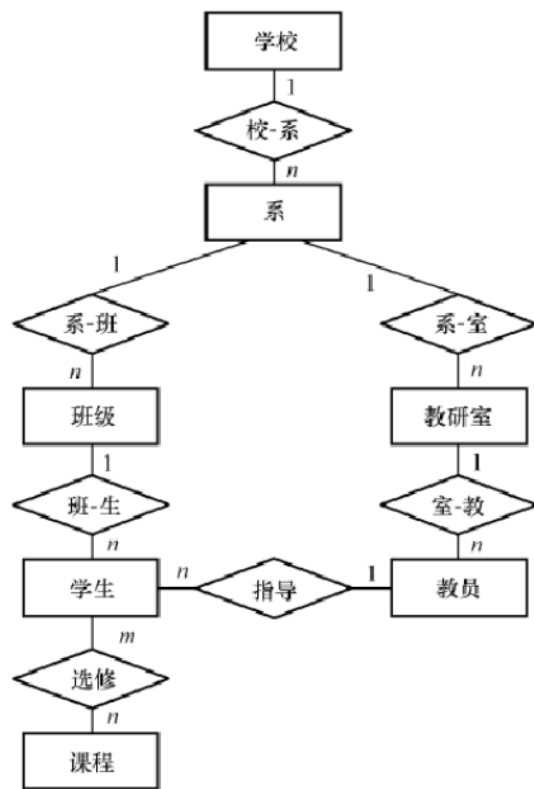
课程:课程编号,课程名

教员:职工号,姓名,职称

各联系的属性为:选修:成绩,其他联系无属性。

画E-R图
设计SQL

数据库系统概论第五版第七章习题解答及解析



习题 7 中的 E-R 图转换的关系模型如下,其中有划画线的属性就是主码属性:

系(系编号,系名,学校名)

班级(班级编号,班级名,系编号)

教研室(教研室编号,教研室,系编号)

学生(学号,姓名,学历,班级编号,导师职工号)

课程(课程编号,课程名)

教员(职工号,姓名,职称,教研室编号)

选课(学号,课程编号,成绩)

把下列E-R模型转换为关系模型

8、某工厂生产若干产品,每种产品由不同的零件组成,有的零件可用在不同的产品上。这些零件由不同的原材料制成,不同零件所用的材料可以相同。这些零件按所属的不同产品分别放在仓库中,原材料按照类别放在若干仓库中。请用 E-R 图画出此工厂产品、零件、材料、仓库的概念模型。

习题 8 中的 E-R 图转换的关系模型如下,其中有划画线的属性就是主码。

产品(产品号,产品名,仓库号)

零件(零件号,零件名)

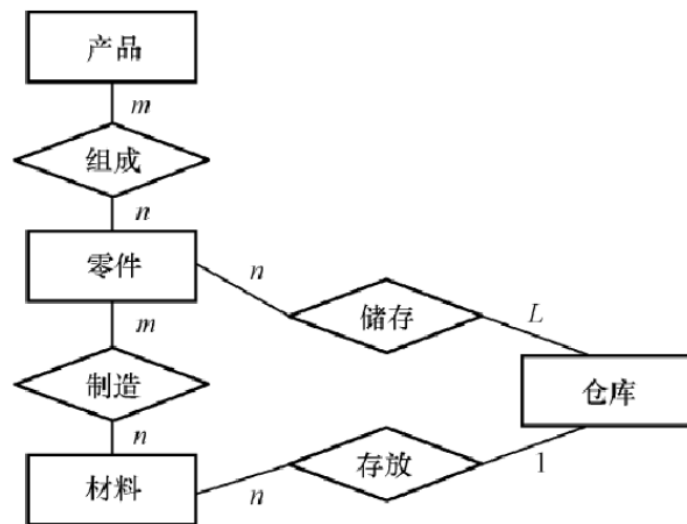
原材料(原材料号,原材料名,类别,仓库号,存放量)

仓库(仓库号,仓库名)

产品组成(产品号,零件号,使用零件量)

零件组成(零件号,原材料号,使用原材料量)

零件储存(零件号,仓库号,存储量)



E-R 图中各实体的属性假设为:

产品:产品号,产品名

零件:零件号,零件名

原材料:原材料号,原材料名,类别

仓库:仓库号,仓库名

各联系的属性为:

产品的组成:使用零件量

零件制造:使用原材料量

零件储存:存储量

材料存放:存放量