

第七章 数据库设计和E-R模型

一、建模

- 一个数据库可以理解为一组实体的集合
- **实体(entity)**: 是现实世界中可区别于所有其他对象的一个“事物”或“对象”
- **实体集(entity set)**: 是相同类型即具有相同性质的一个实体集合
- **联系(relationship)**: 是指多个实体之间的相互关联
 - 联系也可以具有描述性属性
- **联系集(relationship set)**: 是相同类型联系的集合
 - 联系集的**度(degree)**: 参与联系集的实体集的数目

二、属性

- 属性的**域或值集**：一个属性的可取值的集合
- 属性的分类：
 - 简单属性和复合属性
 - 复合属性可以再划分为更小的部分(即其他属性)
 - 单值属性和多值属性
 - 单值属性：属性对应一个特定的实体只有一个值
 - Student_ID属性只对应于一个学生ID
 - 多值属性：属性对应一个特定的实体有一组值
 - Phone_number属性，每个实体可以有0个、1个或多个电话号码
 - 派生属性：这类属性的值可以从别的相关属性或实体派生出来。
 - 从当前的日期和date_of_birth计算出age
- 空值(null)
 - 当实体在某个属性上没有值时使用空值
 - 空值可以表示“不适用”，即该实体的这个属性不存在值

eg. 名字 → 姓+名.
复合属性.

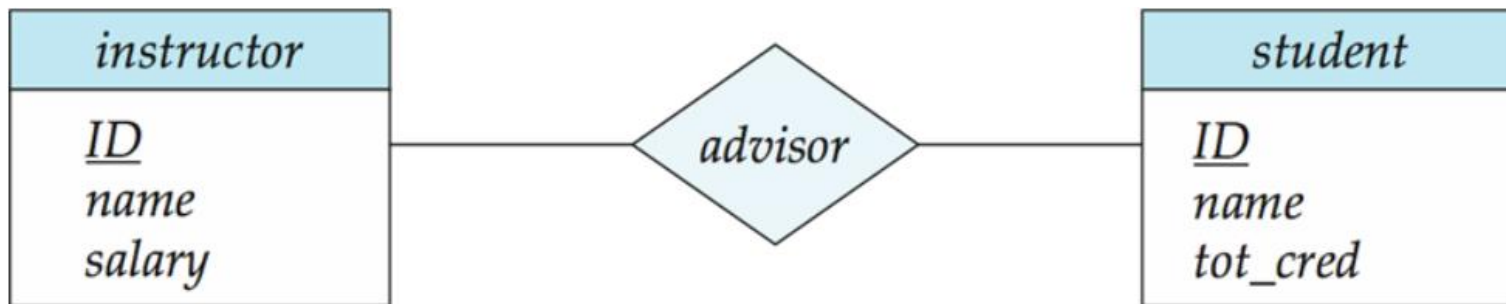
三、E-R图(实体-联系图)

1.基本结构

- 矩形代表实体集
- 菱形代表联系集
- 双菱形代表连接到弱实体集的标志性联系集
- 线段将实体集连接到联系集
- 虚线将联系集属性连接到联系集
- 双线显示实体在联系集中的参与度
- 椭圆代表属性(或者是矩形里面的项代表属性)
 - 双线椭圆代表多值属性
 - 用虚线画的椭圆代表派生属性
- 下划线代表主键

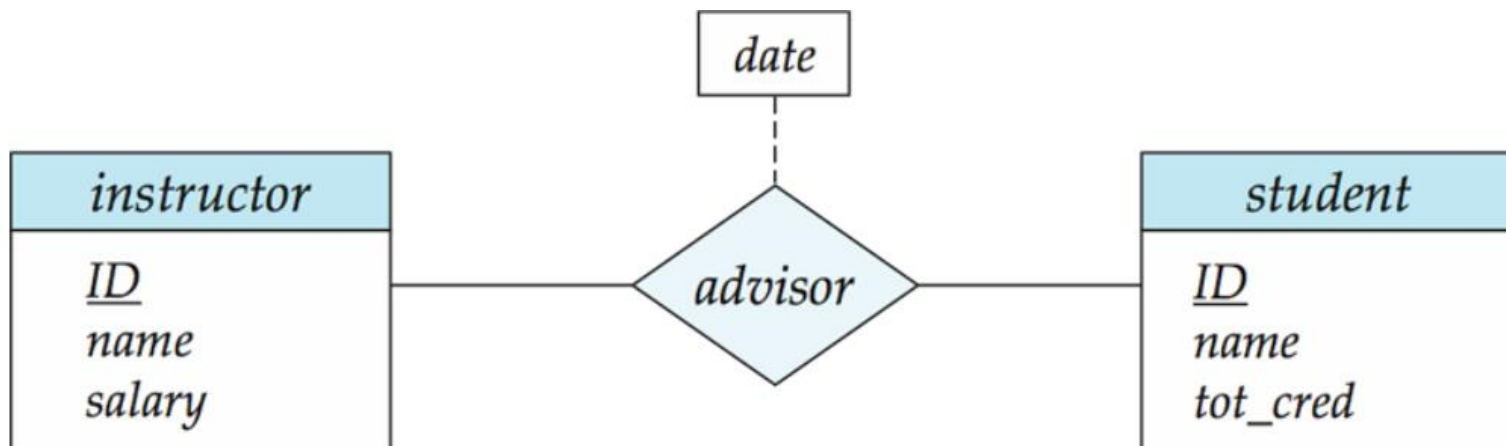
1.基本结构

- 本图包含：实体集、属性、主键、联系集



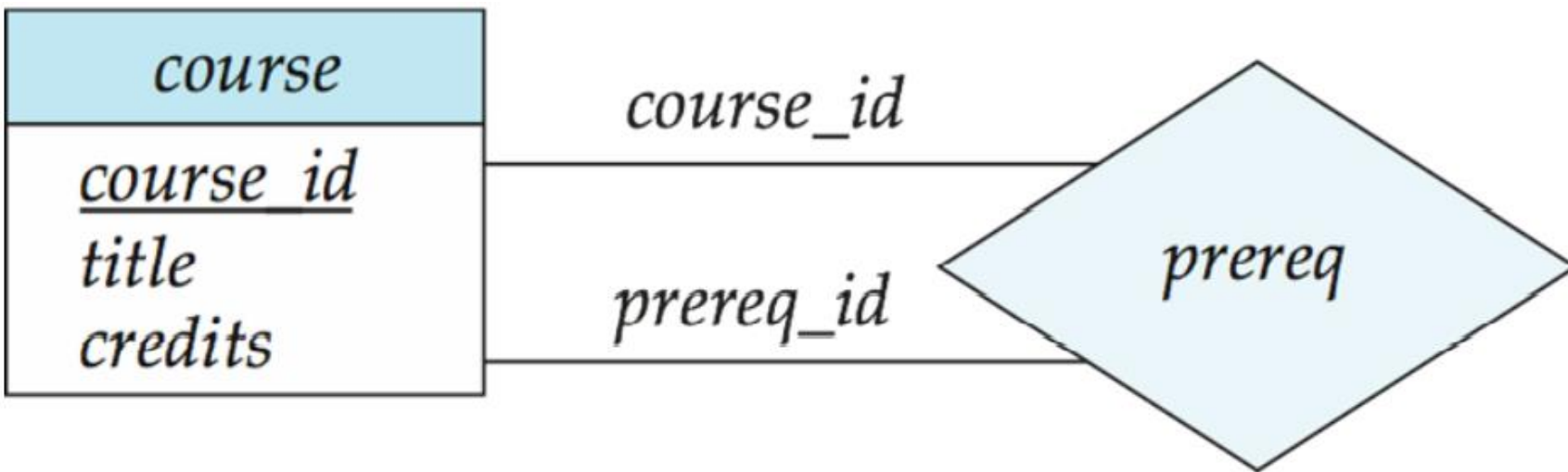
- 本图增加了联系集的属性

- 联系集的属性：如果一个联系集有关联的属性，那么我们将这些属性放入一个矩形中，并且用虚线将该矩形与代表联系集的菱形联系起来



2. 角色

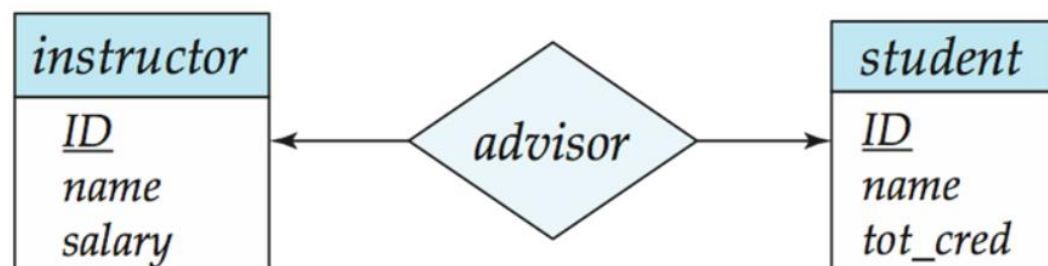
- 通过在菱形和矩形之间的连线上进行标注来表示角色
- 表示了同一实体集内部不同实体之间的联系
 - Eg.course实体集和prereq联系集之间的角色标识



3.映射基数(实体集和联系集的对应关系)

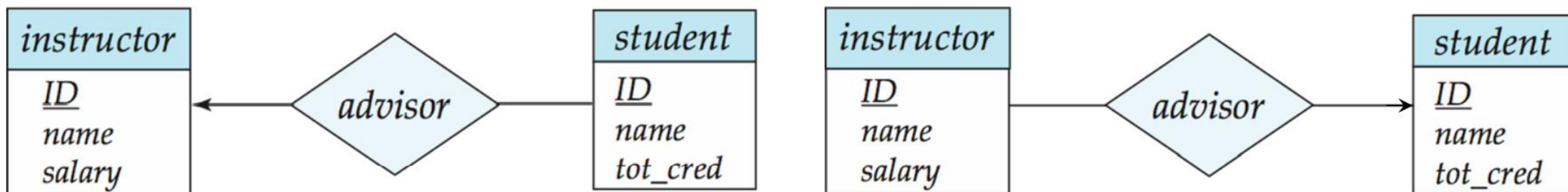
- 一对一

- 联系集向两个实体集各画一个箭头
- 例子表明一名教师至多可以指导一名学生，并且一名学生至多可以有一位导师



- 一对多/多对一(是一对多还是多对一主要是看我们当前在看哪个实体集)

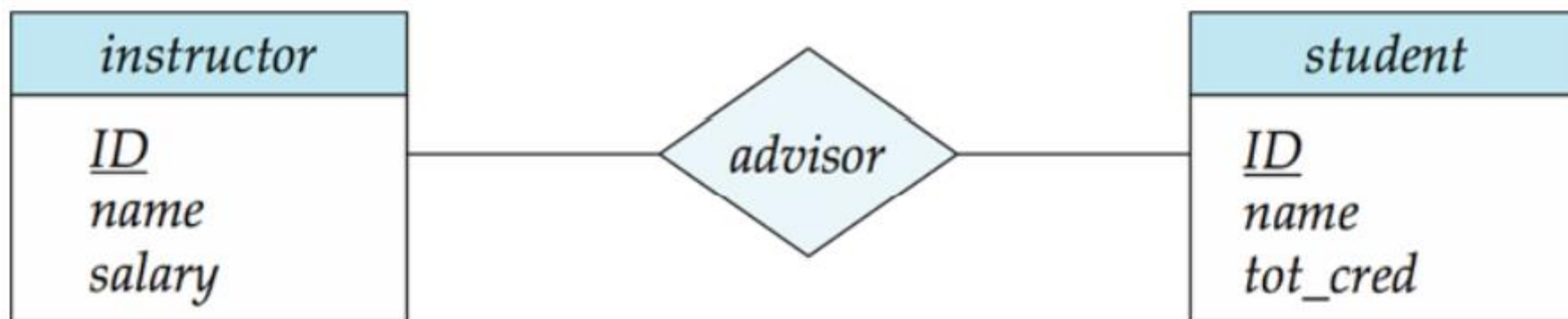
- 联系集向一个实体集画一条线段，向另一个实体画一条箭头
- 左边图的意思是一名导师可以指导多名学生，但一名学生至多可以有一位导师
- 右边图的意思是一名导师至多可以指导一名学生，但一名学生可以有多位导师



3.映射基数(实体集和联系集的对应关系)

- 多对多

- 联系集向两个实体集各画一条线段
- 一位导师可以指导多名学生，一名学生也可以拥有多位导师



4.码

一个实体的属性的值必须可以唯一标识该实体。也就是说，在一个实体集中不允许两个实体对于所有属性都具有完全相同的值。

- **超码(super key)**: 是一个或多个属性的集合，这些属性的组合可以使我们在一个关系中唯一地标识一个元组
- **候选码(candidate key)**: 最小超码
- **主码(primary key)**: 从候选码中挑选，主要用来在一个关系中区分不同元组的候选码；主码应该选择那些值从不或极少变化的属性

5.弱实体集(weak entity sets)

- **弱实体集**：没有足够的属性以形成主码的实体集
- 弱实体集必须与另一个称作**标识或属主实体集**的实体集**关联**才能有意义。
弱实体集**存在依赖于**标识实体集；
标识实体集**拥有**它所标识的弱实体集；
将弱实体集于其标识实体集相联的联系称为**标识性联系**
- **分辨符**：弱实体集的分辨符也称为该实体集的部分码
- **弱实体集的主码**：由**标识实体集的主码**加上该**弱实体集的分辨符**构成。
- 在E-R图中，弱实体集和强实体集类似，以矩形标识，但有两点区别：
 - 弱实体集的分辨符以虚下划线标明，而不是实线
 - 关联弱实体集和标识性强实体集的联系集以双菱形表示



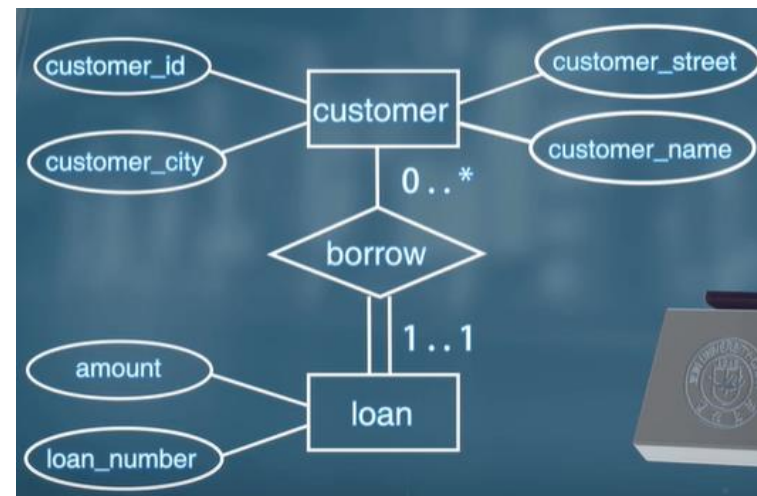
弱实体集section通过联系集sec_course依赖于强实体集course

6. 一个实体集在关系集里的参与形式

- 完全参与(用双线表示): 在实体集里的每一个实体至少要在关系集里参与一个关系
 - 例: 从下面的图可以看出, 每一个section必须有一个相关的course



- 部分参与: 实体集里的部分实体可能不会在关系集里参与任何关系
 - 例: 银行里的存款的人不一定要贷款, 因此实体集customer里的部分实体可能不会在关系集borrow里参与任何关系

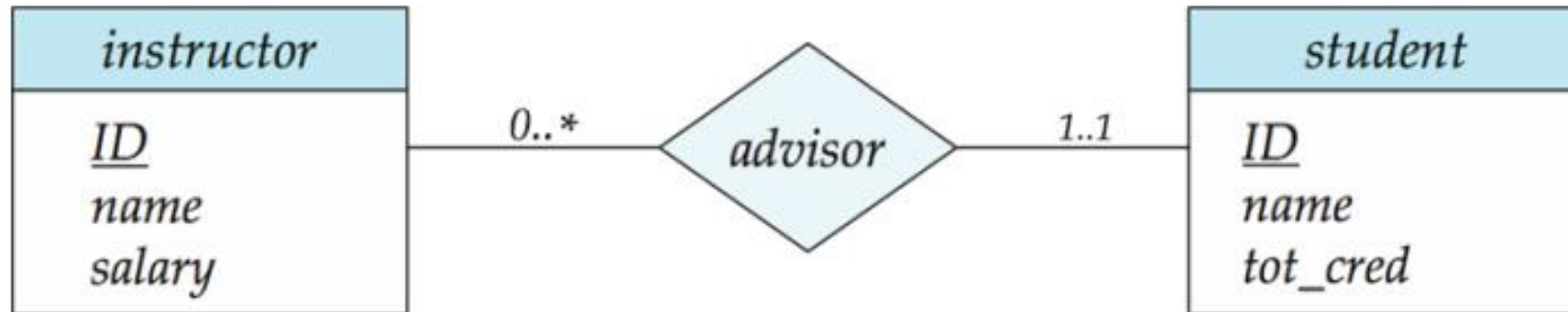


7.用映射基数极限来表示参与约束

- 映射基数用l..h表示，其中l表示最小的映射基数，h表示最大的映射基数
- 最小值为1表示这个实体在该联系集中全部参与，即实体集中的每个实体在联系集中的至少一个联系中出现。

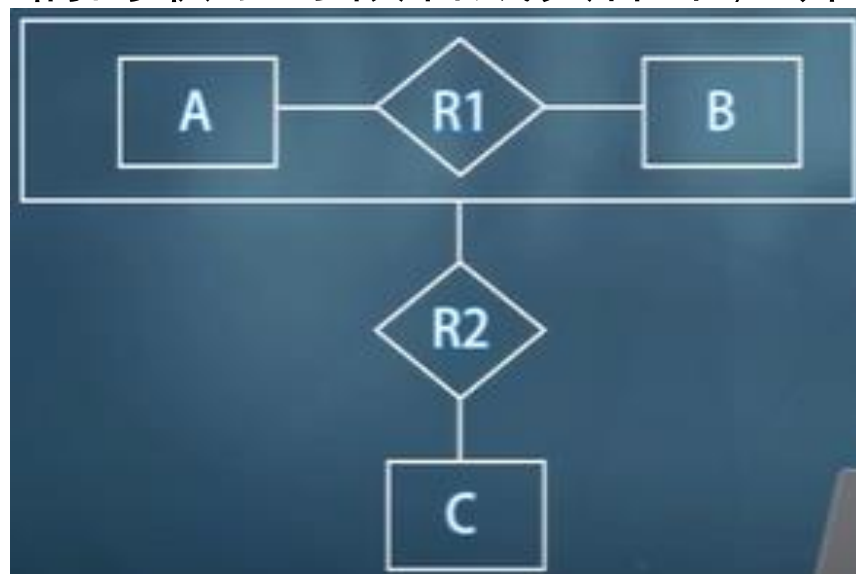
最大值为1表示这个实体参与至多一个联系，而最大值为*代表没有限制

- 例子：在advisor和student之间的边有1..1的基数约束，意味着基数的最小值和最大值都是1。也就是，每个学生必须有且仅有一个导师。从advisor到instructor边上的约束0..*说明教师可以有0个或多个学生。因此，advisor联系是从instructor到student的一对多联系。



四、聚集

- 什么是聚集？
 - 聚集是一种抽象，通过这种抽象，联系被视为高层实体。
- 何时使用聚集？
 - 为了表达联系与联系之间的联系
- 如何表示聚集？
 - 实体集A与B以及它们的联系可被看成实体集，并与另一实体集C发生联系

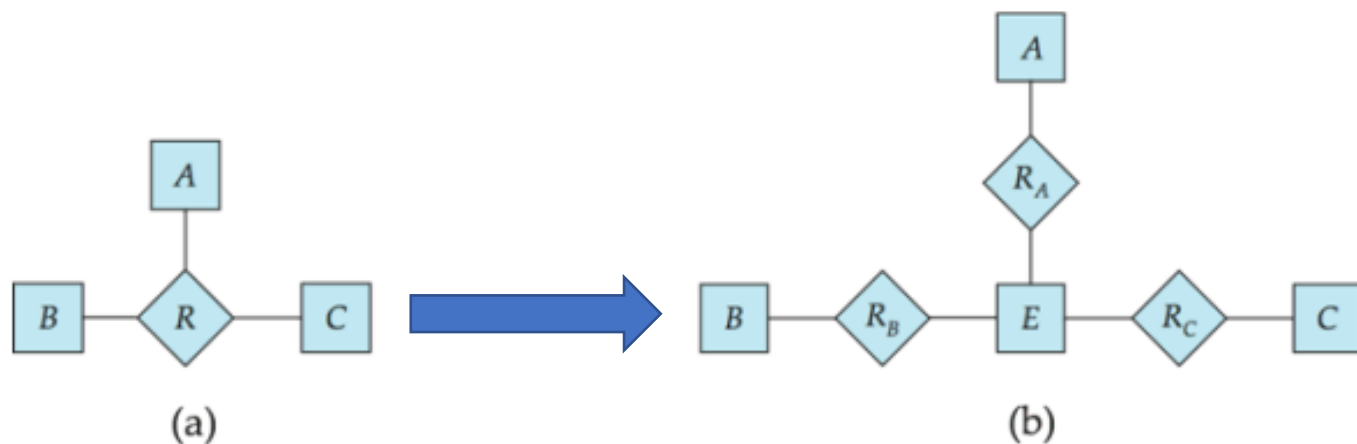


五、一些问题

- 在设计E-R模型，描述一个对象时，是把这个对象定义成一个实体集，还是定义成一个属性？
 - 取决于当前服务企业的业务模型和类型
- 在设计E-R图时，有一些对象是定义为实体集还是定义为联系集？
 - 答：如果不确定，可以先把它定义成联系集。然后为这个联系集确定它的组件。如果这个组件满足联系集的定义并满足业务需求，那么它就是联系集。
 - 较为人话一点的一个原则：当描述发生在实体间的行为时采用联系集。
- 联系集的度
 - 尽量用二元联系
- 联系集的属性问题
 - 在实际运行中，有一些两个实体集发生联系会产生一些结果。那么这个结果就可以作为联系集的属性

六、多元联系到二元联系的转换

- 如a图所示，这是一个三元联系集
- 如何从a图转换为b图？
 - 用实体集E替代联系集R，并创建三个联系集
 - R_A , 关联E和A
 - R_B , 关联E和B
 - R_C , 关联E和C
 - 针对联系集R中的每个联系 (a_i, b_i, c_i) , 在实体集E中创建一个新的实体 e_i 。然后，在三个新联系集中，分别插入新联系如下：
 - 在 R_A 中插入 (e_i, a_i)
 - 在 R_B 中插入 (e_i, b_i)
 - 在 R_C 中插入 (e_i, c_i)



七、E-R模型向关系模式的转换

- 一般情况
 - 实体->关系
 - 属性->关系的属性
 - 例子：



七、E-R模型向关系模式的转换

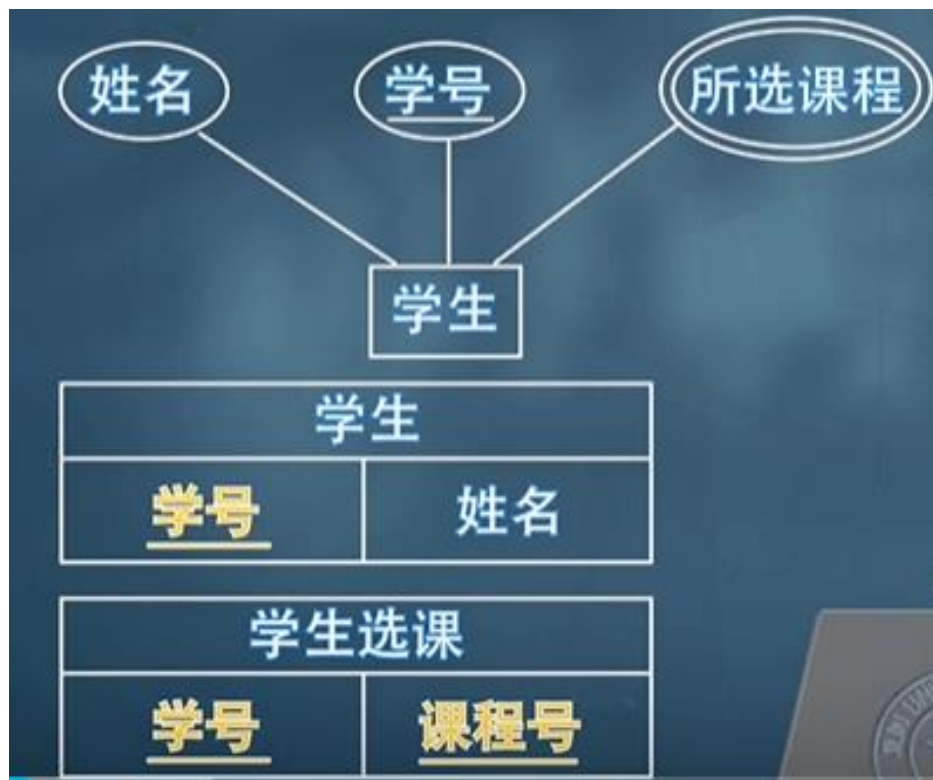
- 复合属性->将每个组合属性作为复合属性所在实体的属性
 - 例子:



复合属性不会出现在关系里，
复合属性里面的属性会依次出现在关系里

七、E-R模型向关系模式的转换

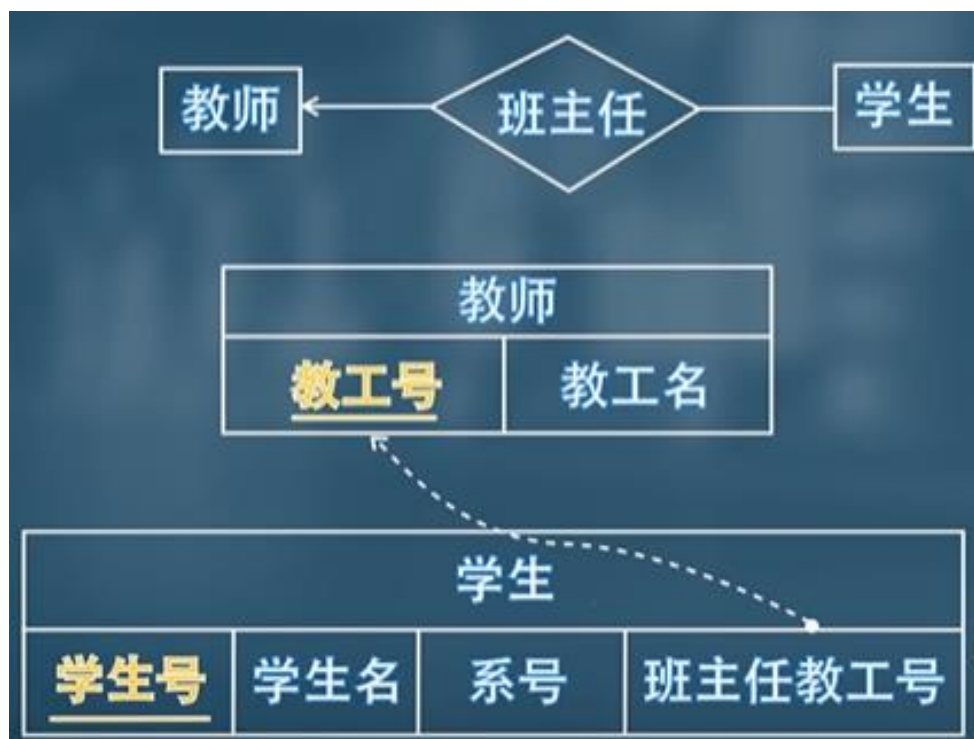
- 多值属性 -> 新的关系 + 所在实体的码
 - 例子:



一个学生所选的课程肯定不止一门，所以如果让所选课程这个属性直接作为学生关系属性里的一项显然不够，于是我们就产生了一个新的名为“学生选课”的关系，首先这个关系的属性一定要包含“学生”这个关系里的主键(也就是学号)，其次就是课程号。

七、E-R模型向关系模式的转换

- 一对多联系->将单方参与实体的码作为多方参与实体的属性
- 例子:



一名学生只有一位班主任，一位班主任管理多名学生

在“学生”这一关系里面添加“教师”关系里的主键(也就是教工号)

七、E-R模型向关系模式的转换

- 多对多的联系->将联系定义为新的关系，属性为参与双方的码

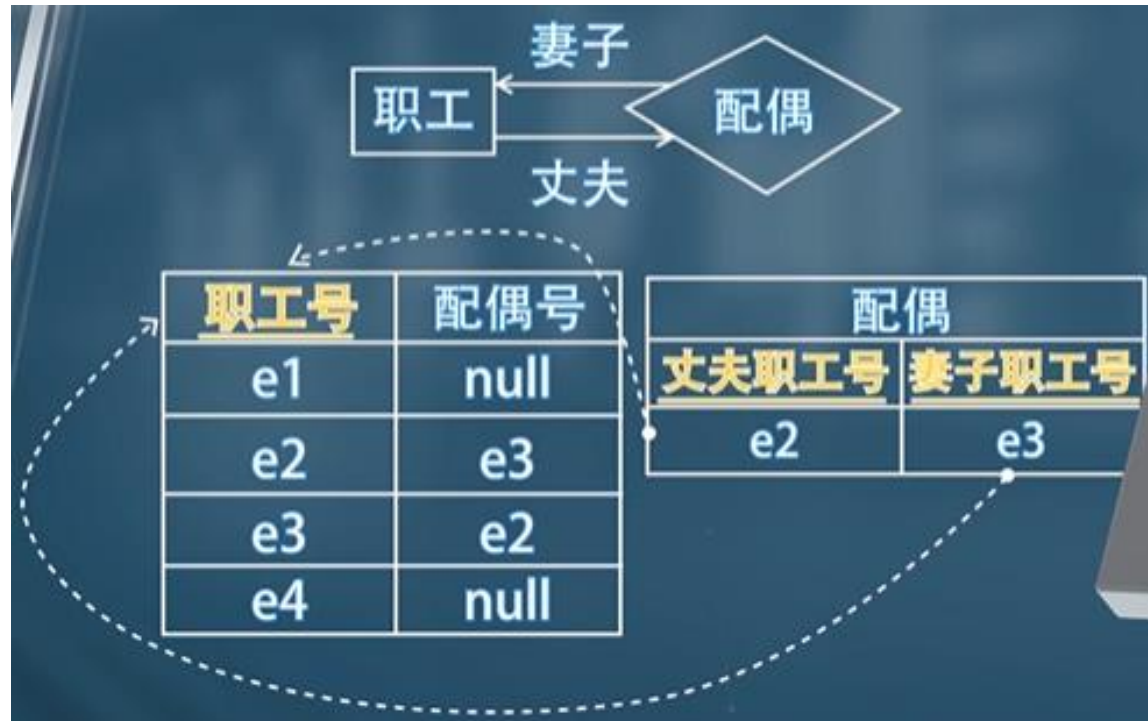
• 例：



学生与选修课之间是多对多的关系，此时需要新建立一个名为“选修”的关系，这个关系里至少要包含“学生”和“课程”关系里的主键。

七、E-R模型向关系模式的转换

- 一对一联系->若联系双方均部分参与，则将联系定义为一个新的关系，属性为参与双方的码（情况1）
 - 例子1:

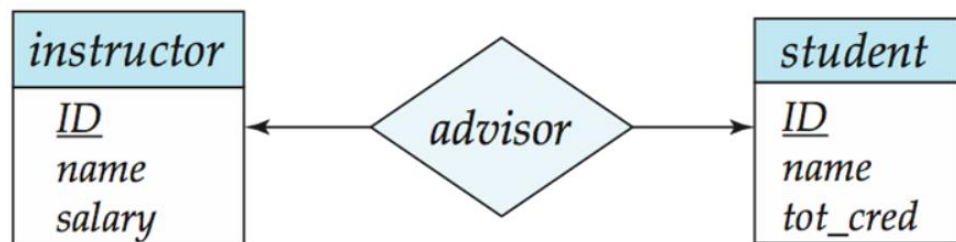


在本例中就是有的职工有配偶，有的职工没有配偶，而丈夫职工号和妻子职工号都由职工这一关系产出，因此满足联系双方均部分参与。

七、E-R模型向关系模式的转换

- 一对一联系->若联系一方全部参与，则将联系另一方的码作为全部参与一方的属性（情况2）

• 例子2:



一位学生对应一位导师，一位导师也只对应一名学生

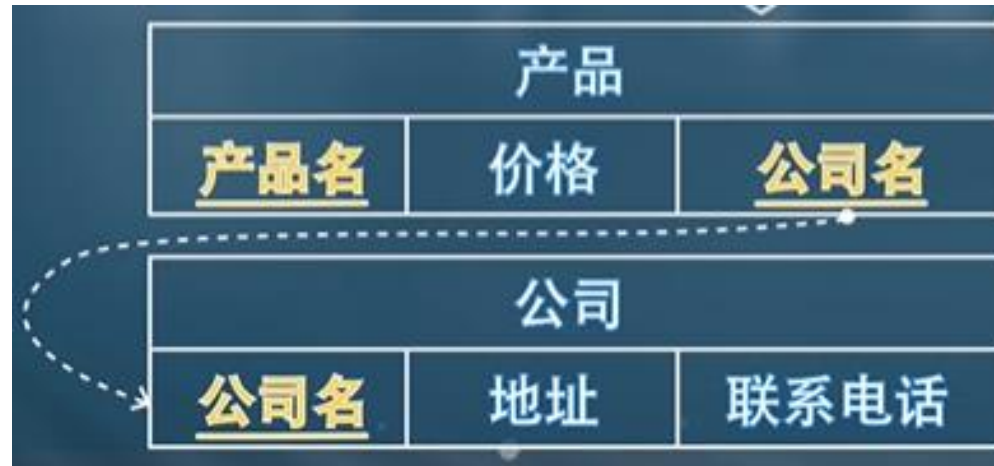
假设“导师”一方全部参与，则应该将“学生”一方的主键作为全部参与一方(即导师一方)的属性。

导师			
<u>导师编号</u>	导师姓名	薪水	指导学生编号

学生		
<u>学生编号</u>	学生姓名	tot_cred(不知道是啥，不重要)

七、E-R模型向关系模式的转换

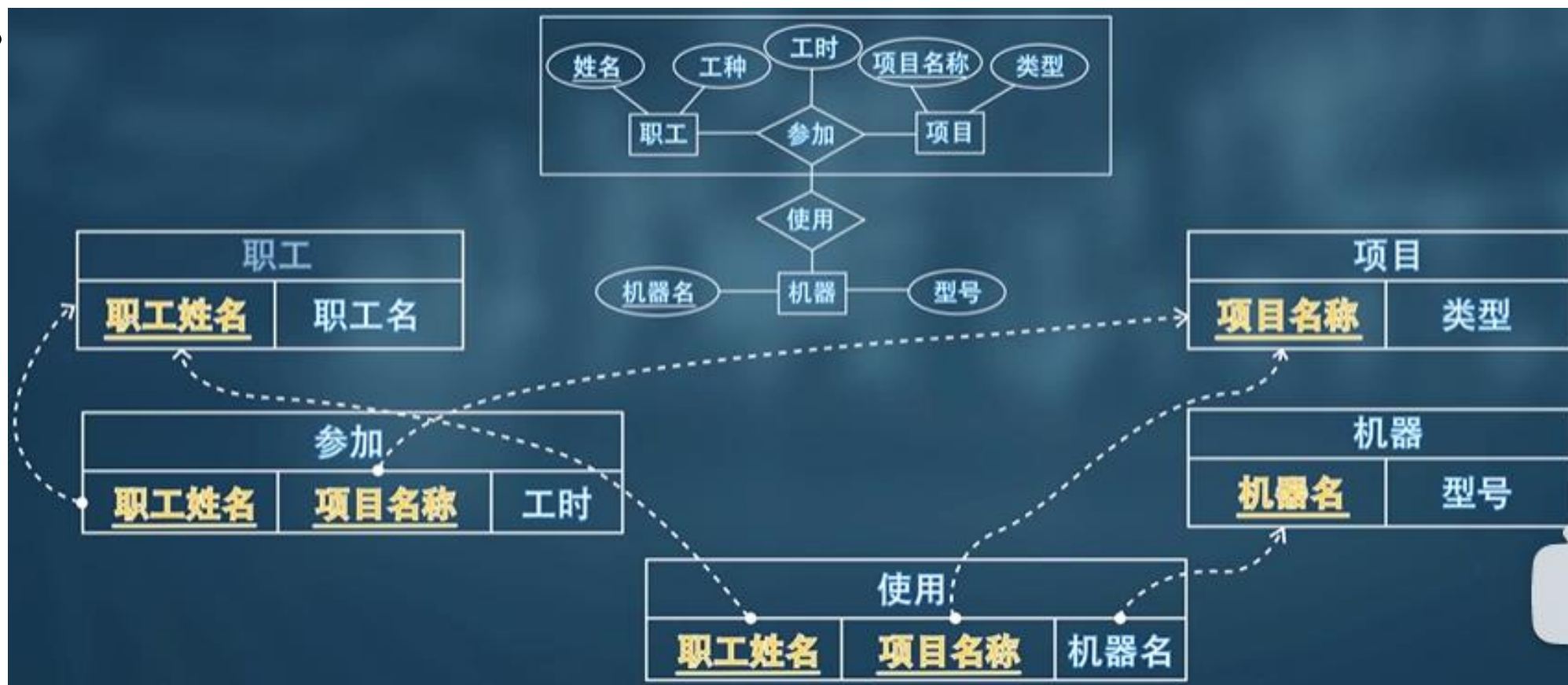
- 弱实体集->所对应的关系的码由弱实体集本身的分辦符再加上所依赖的强实体集的码
 - 例子:



七、E-R模型向关系模式的转换

- 聚集->实体集A与B以及它们的联系R被看成实体集C，C与另一个实体集D构成联系S，则S对应的关系的码，由联系R和实体集D的码构成。

• 例子：



八、E-R模型的一些拓展特性

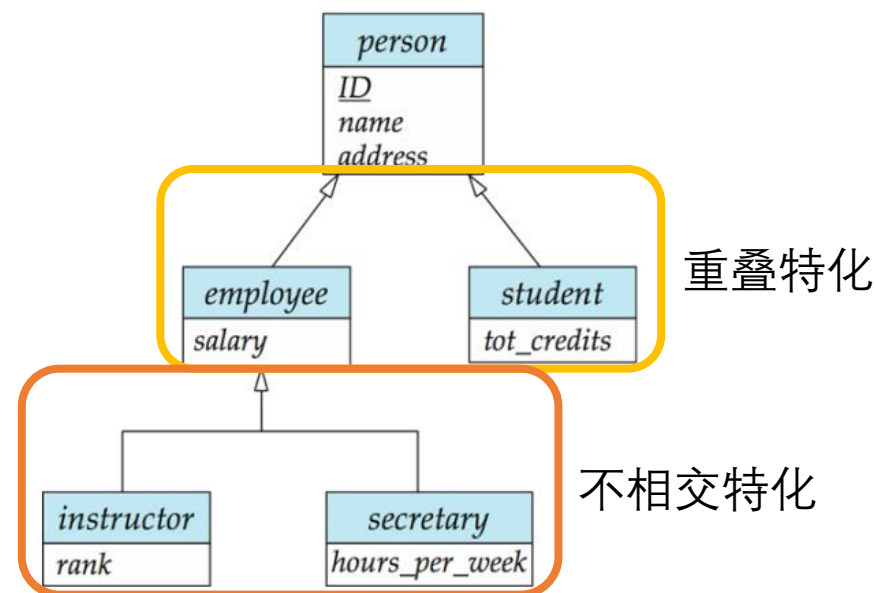
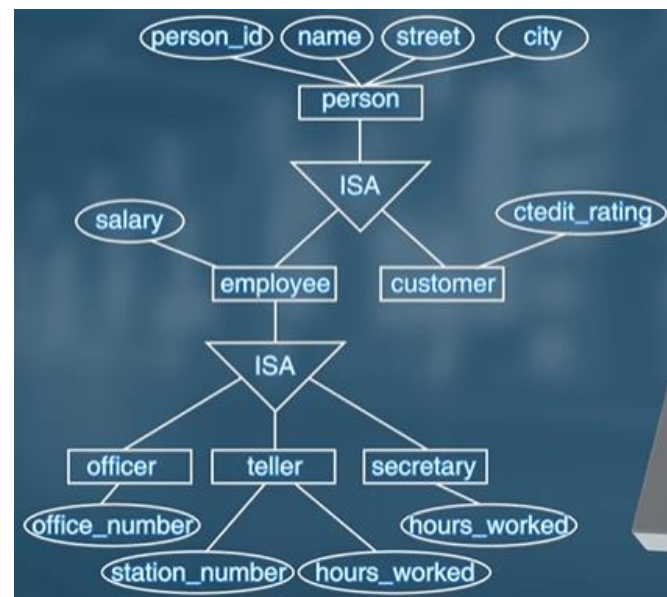
1.特化(specialization)

- 定义：在实体集内部进行分组的过程称为特化.
- 自顶向下的设计过程
- 分类：重叠特化、不相交特化
 - 分类依据：取决于一个实体集是否可能属于多个特化实体集(重叠特化)或者它是否必须属于至多一个特化实体集(不相交特化)。
- 如何表示特化？(书和视频不一样)
 - 视频里：使用一个倒三角连接父类和子类，倒三角里写"ISA",意为"is a"
 - 书上：根据特化的分类，采取不同的连线方法
 - 重叠特化，父类特化出的子类分开使用箭头；
 - 不相交特化，父类特化出的子类用一个箭头连接

八、E-R模型的一些拓展特性

1. 特化(specialization)

- 如何表示特化？（书和视频不一样）（举例）
 - 视频里：使用一个倒三角连接父类和子类，倒三角里写“ISA”，意为“is a”
- 书上：根据特化的分类，采取不同的连线方法
 - 重叠特化，父类特化出的子类分开使用箭头；身份可重叠
 - 不相交特化，父类特化出的子类用一个箭头连接；只能选一个身份



八、E-R模型的一些拓展特性

2. 概化(generalization)

- 定义：概化是高层实体集与一个或多个低层实体集间的包含关系.
- 自底向上的设计过程
- 对所有的实际应用来说，概化只不过是特化的逆过程；
- 在E-R图中，我们对概化和特化的表示不作区分。