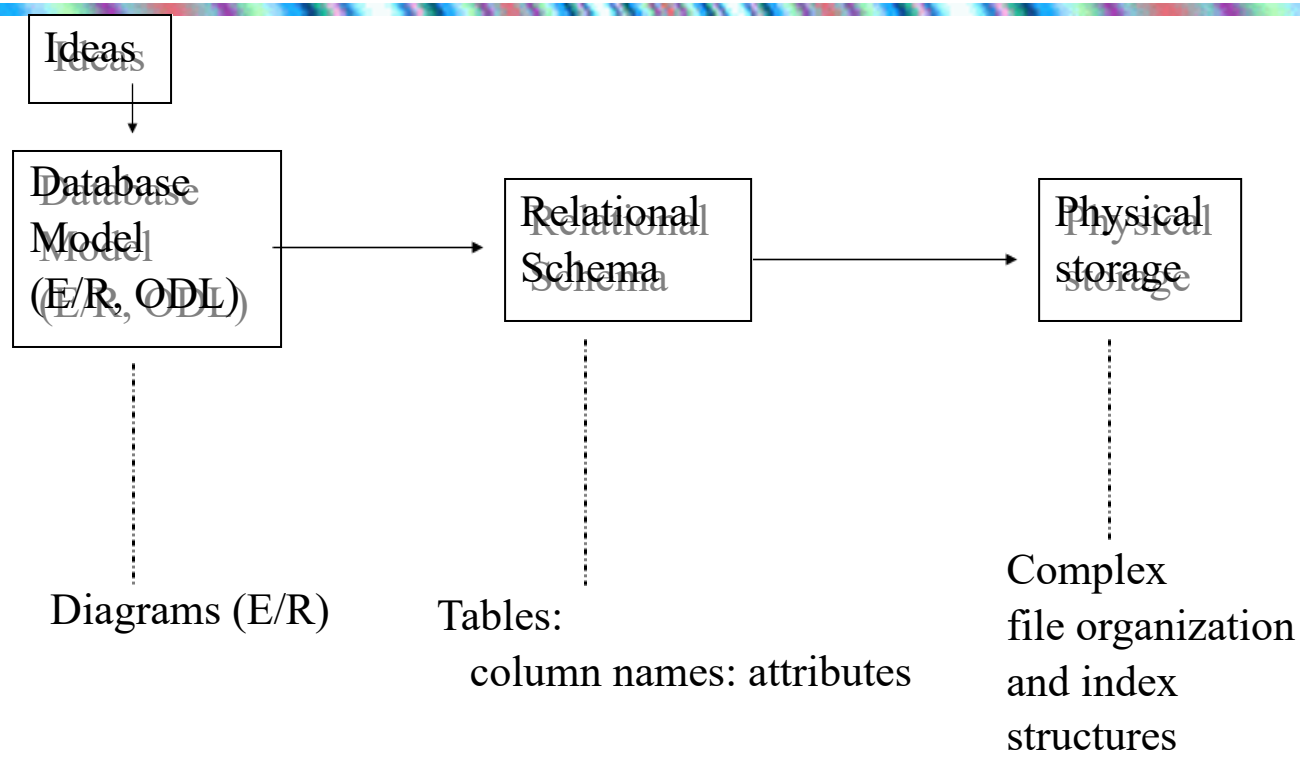
Week5_Course_part1

Relational Data Model-Basic

The Relational Data Model

- Basic stuff & Basic cases
- Special cases
  - combining two relations
  - translating weak entity sets
- Special cases
  - translating subclasses
- Integrity constraints

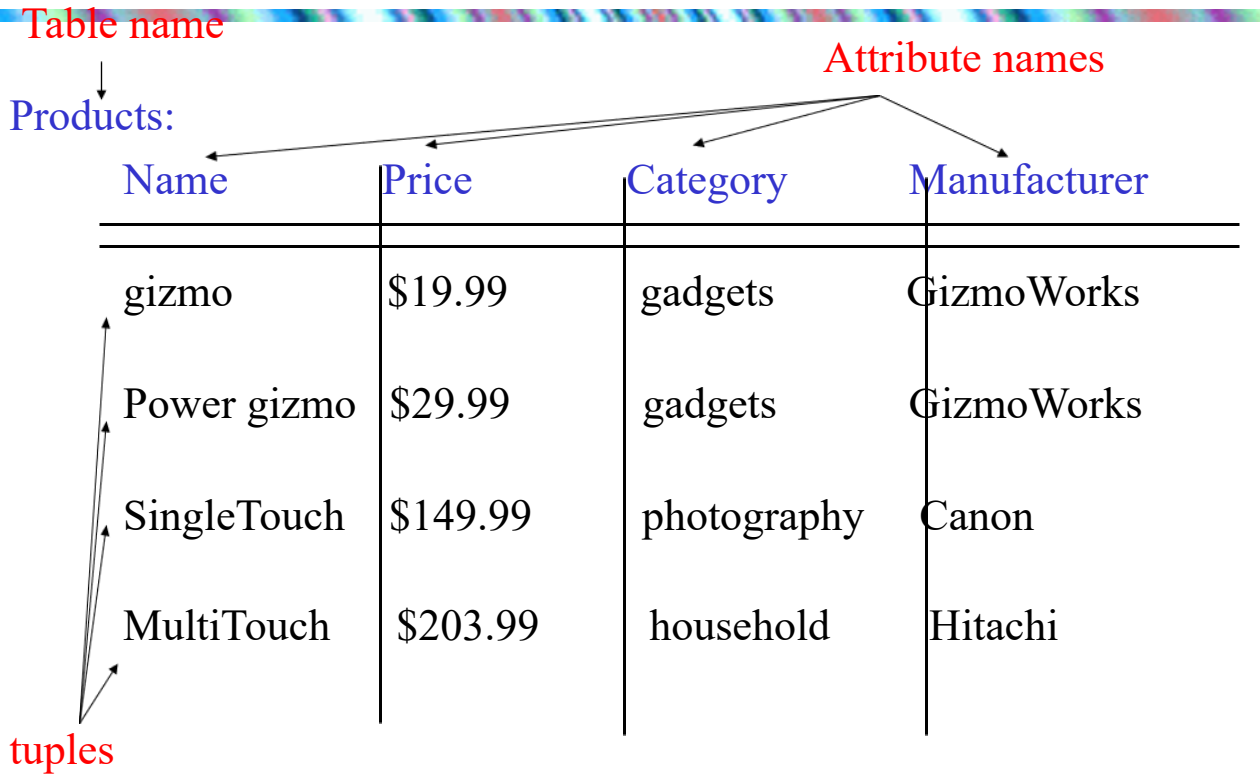
Database Modeling & Implementation



ER Model vs. Relational Model

- Both are used to model data
- ER model has many concepts
  - entities, relations, attributes, etc.
  - well-suited for capturing the app. requirements
  - not well-suited for computer implementation
- Relational model
  - has just a single concept: relation
  - world is represented with a collection of tables
  - well-suited for efficient manipulations on computers

An Example of a Relation



Domains

- Each attribute has a type, called *domain*
- Must be atomic type
- Examples:
  - Integer
  - Real
  - String
  - …

Schemas

The Schema of a Relation:

- Relation name plus attribute names
- E.g. Product(Name, Price, Category, Manufacturer)
- In practice we add the domain for each attribute
- E.g. Product(Name:string, Price:real, Category:string, Manufacturer:string)

The Schema of a Database

- A set of relation schemas
- E.g. Product(Name, Price, Category, Manufacturer),
  Vendor(Name, Address, Phone),
  . . . . . .

Instances

- **Relational schema** = R(A1,⋯,Ak):

  Instance = the set of tuples with k attributes (of “type” R)

  – values of corresponding domains

- **Database schema** = R1(⋯), R2(⋯), ⋯, Rn(⋯)

  Instance = n relations, of types R1, R2, ..., Rn

Example

Relational schema: Product(Name, Price, Category, Manufacturer)

Instance:

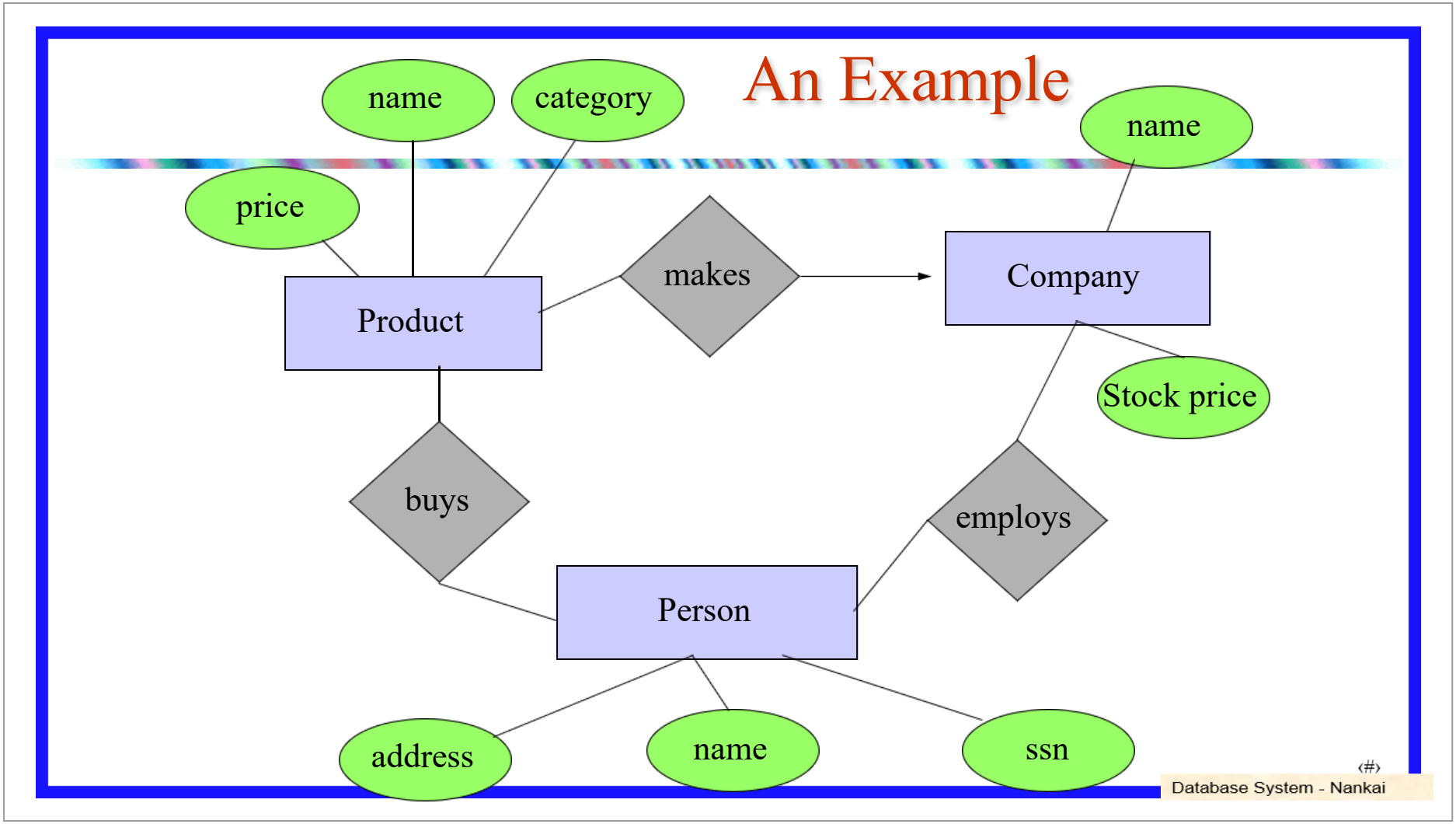
Name	Price	Category	Manufacturer
gizmo	$19.99	gadgets	GizmoWorks
Power gizmo	$29.99	gadgets	GizmoWorks
SingleTouch	$149.99	photography	Canon
MultiTouch	$203.99	household	Hitachi

Schemas and Instances

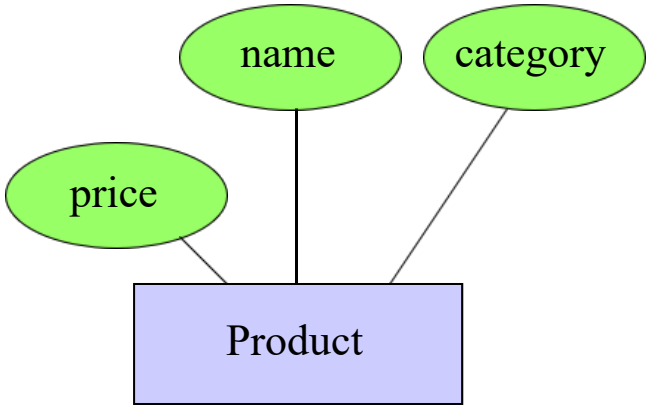
- Analogy with programming languages:
  - Schema = type
  - Instance = value
- Important distinction:
  - Database Schema = stable over long periods of time
  - Database Instance = changes constantly, as data is inserted/updated/deleted

Translating ER Diagram to Rel. Design

- Basic cases
  - entity set E = relation with attributes of E
  - relationship R = relation with attributes being keys of related entity sets + attributes of R
- Special cases
  - combining two relations
  - translating weak entity sets
  - translating is-a relationships


An Example
name
category
price
Product
makes
Company
name
Stock price
buys
employs
Person
address
name
ssn
‹#›
Database System - Nankai

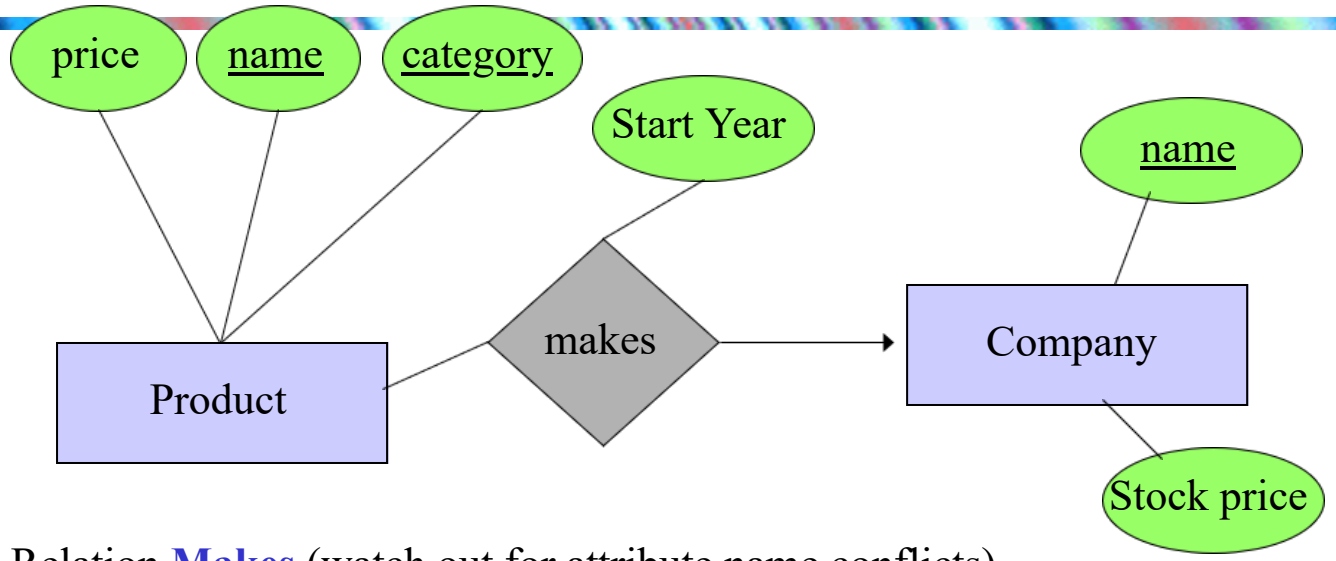
Entity Sets to Relations



Product:

Name	Category	Price
gizmo	gadgets	$19.99

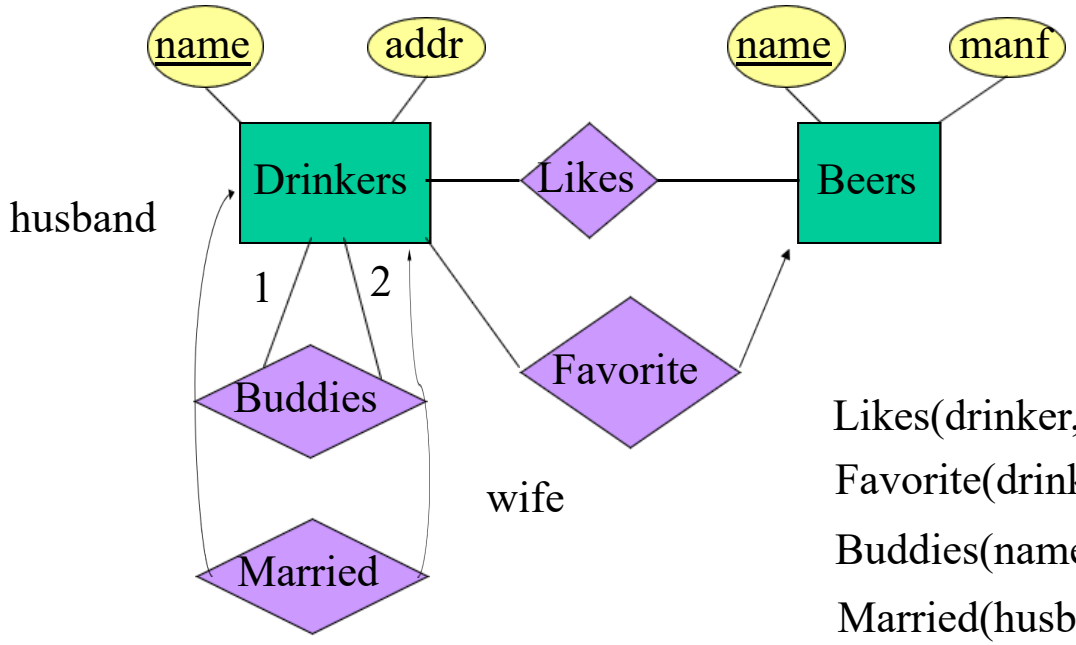
Relationships to Relations



Relation **Makes** (watch out for attribute name conflicts)

Product-name	Product-Category	Company-name	Starting-year
gizmo	gadgets	gizmoWorks	1963

Relationship to Relation: Another Example



Likes(drinker, beer)

Favorite(drinker, beer)

Buddies(name1, name2)

Married(husband, wife)

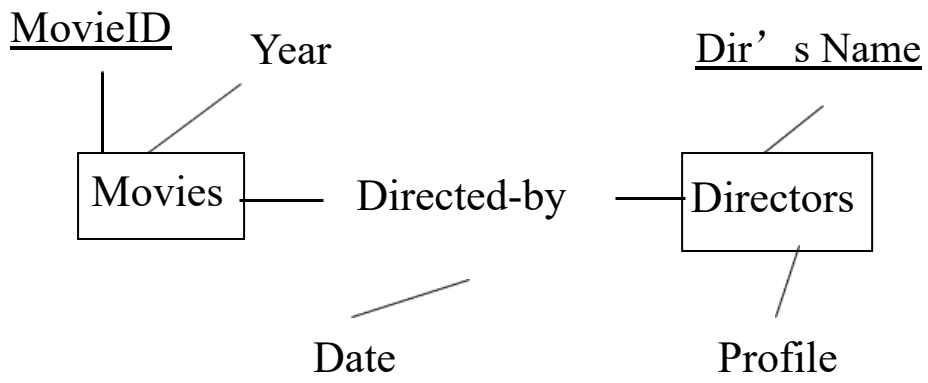
Basic stuff & Basic cases: Summary

- Basic stuff
  - Relation, schema, instance
- Basic cases
  - Entity to table, relationship to table

单选题 1分

互动交流一

下面ER图转换为关系模式时，应该生成生成几张表?



A 2

B 3

C 4

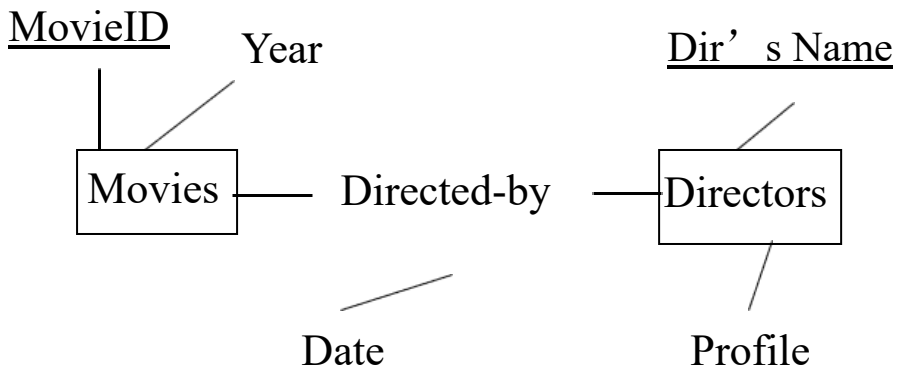
提交

Database System - Nankai

单选题 1分

互动交流二

下面ER图转换为关系模式时，Directed-by这张表由几个属性构成？请弹幕写出这张表的模式。



A 2

B 3

C 4

提交

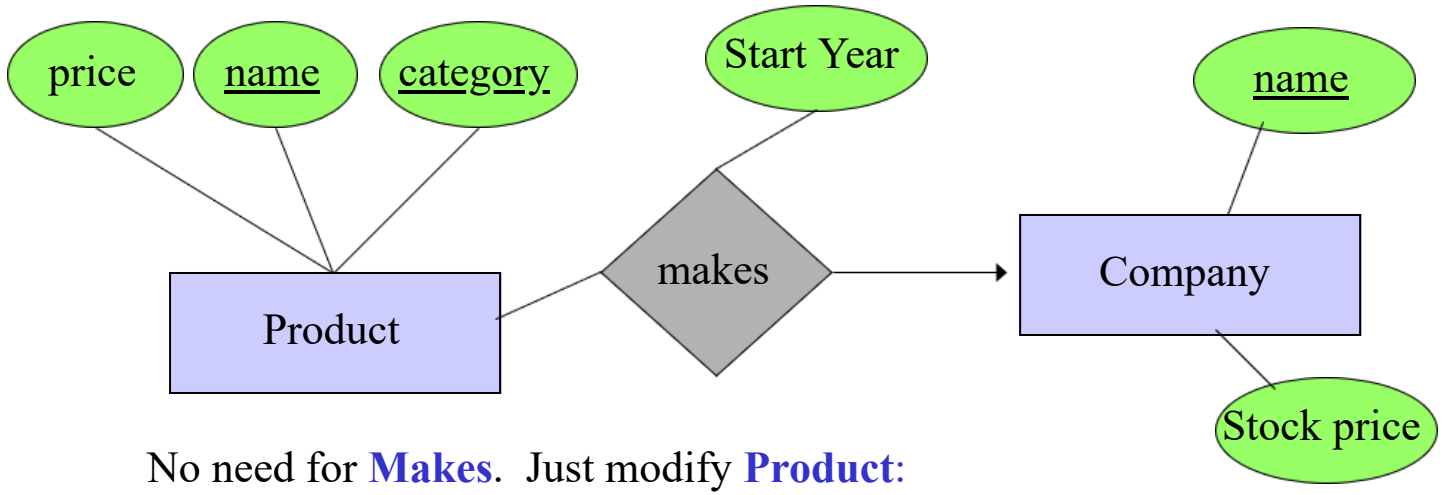
Week5_Course_part2

Relational Data Model-Special cases part I

The Relational Data Model

- Basic stuff & Basic cases
- Special cases
  - combining two relations
  - translating weak entity sets
- Special cases
  - translating subclasses
- Integrity constraints

Combining Two Relations

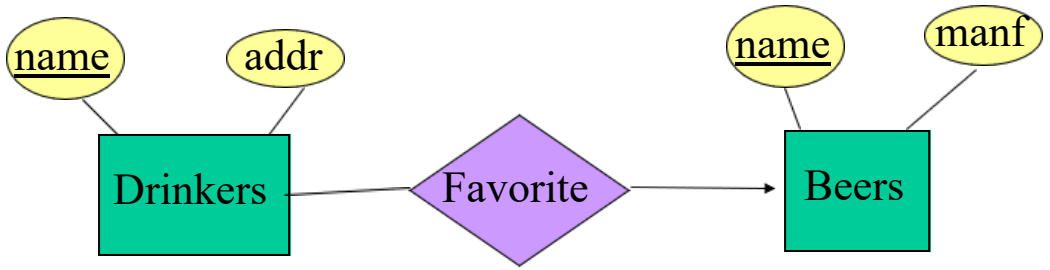


No need for **Makes**. Just modify **Product**:

name	category	price	companyName	StartYear
gizmo	gadgets	19.99	gizmoWorks	1963

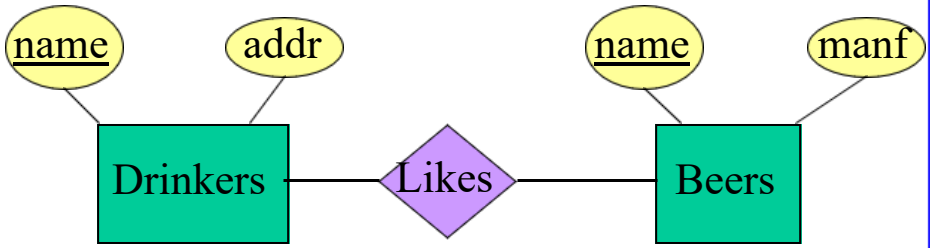
Combining Relations

- It is OK and more efficient to combine the relation for an entity set *E* with the relation *R* for a many-one relationship from *E* to another entity set.
- Example: Drinkers(name, addr) and Favorite(drinker, beer) combine to make Drinker1(name, addr, favoriteBeer).



Risk with Many-Many Relationships

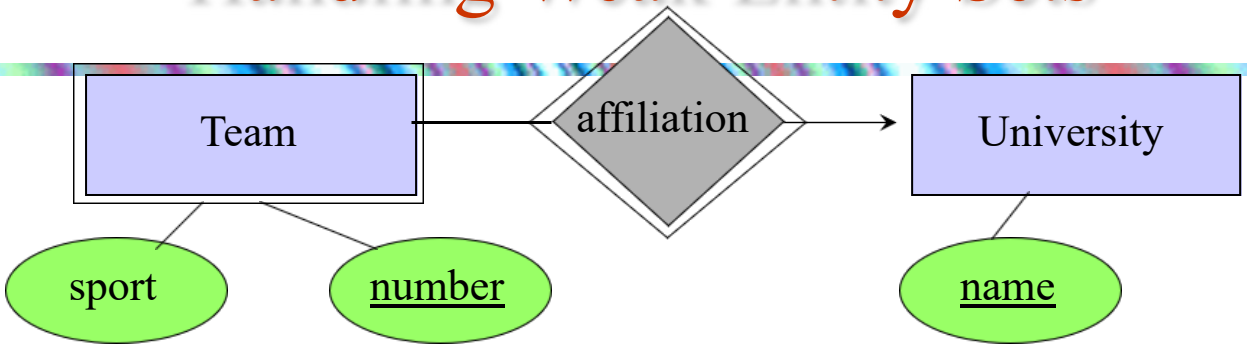
- Combining Drinkers with Likes would be a mistake. It leads to redundancy, as:



name	addr	beer
Sally	123 Maple	Bud
Sally	123 Maple	Miller

Redundancy

Handling Weak Entity Sets

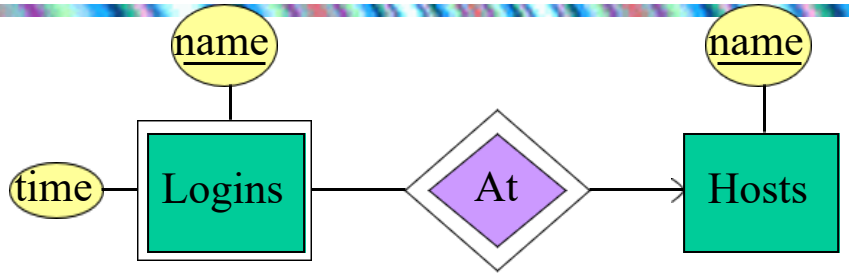


Relation Team:

Sport	Number
wrestling	15

- need all the attributes that contribute to the key of Team
- don’ t need a separate relation for Affiliation. (why ?)

Another Example



Hosts(hostName)

Logins(loginName, hostName, time)

~~At(loginName, hostName, hostName2)~~

At becomes part of Logins

Must be the same

Special cases part 1: Summary

- Combining two relations
  - Many-one relation can be merged
  - Merging many-many is dangerous
- Translating weak entity sets

多选题 1分

互动交流一

以下哪些联系类型可以与实体集合的表进行合并?

A One-one

B Many-one

C Many-many

提交

单选题 1分

互动交流二

当对many-one关系进行合并时，是与哪边的实体集合的表进行合并?

A Many 端

B One 端

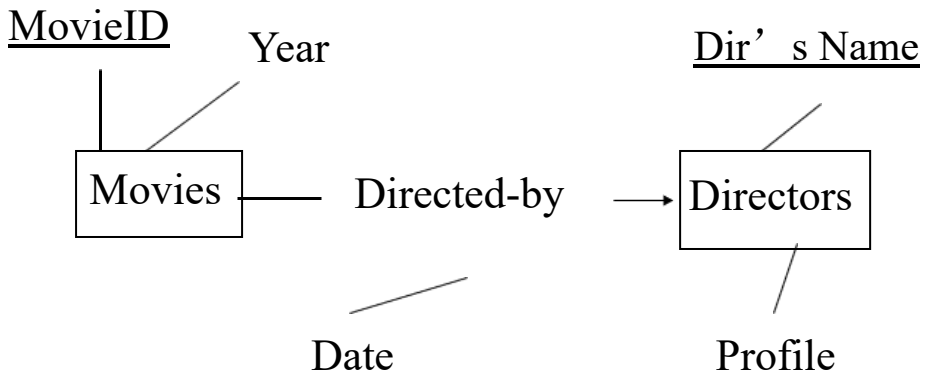
提交

单选题 1分

互动交流三

下图将联系和实体表进行合并，合并之后的表属性有几个？

请弹幕写出这张表的模式。



A 2

B 3

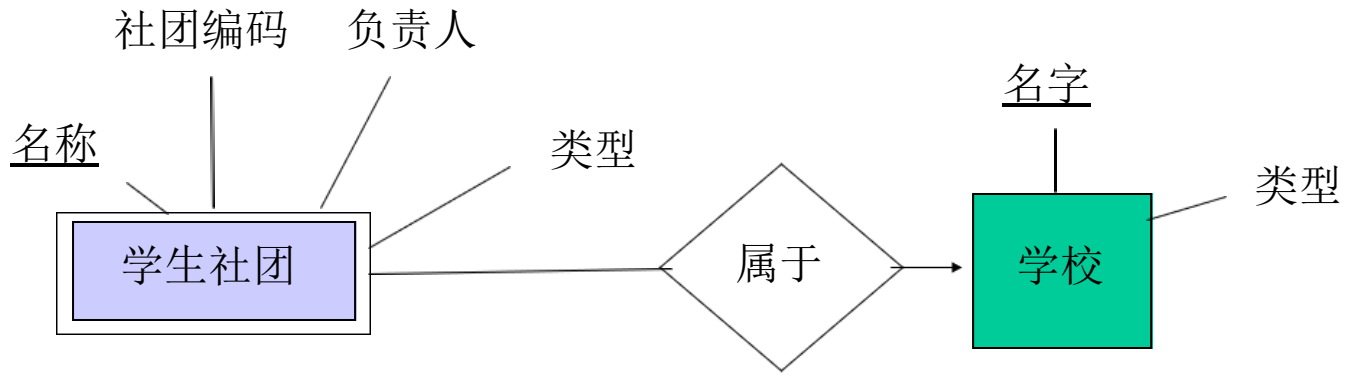
C 4

提交

单选题 1分

互动交流四

下面ER图转换为关系模式时，一共会生成几张表?



A 1

B 2

C 3

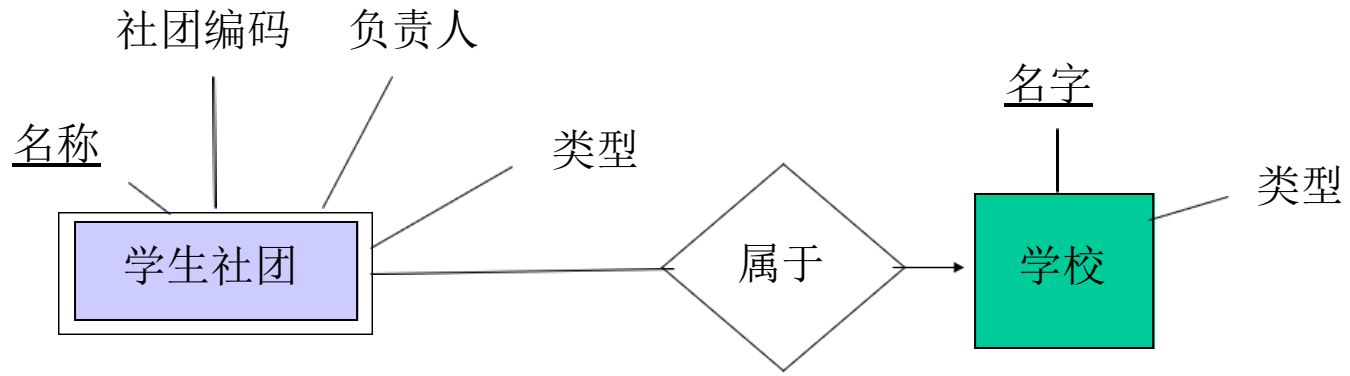
D 4

提交

单选题 1分

互动交流五

下面ER图转换为关系模式时，学生社团这张表由几个属性构成？请弹幕写出这张表的模式。



A 4

B 5

C 6

D 7

提交

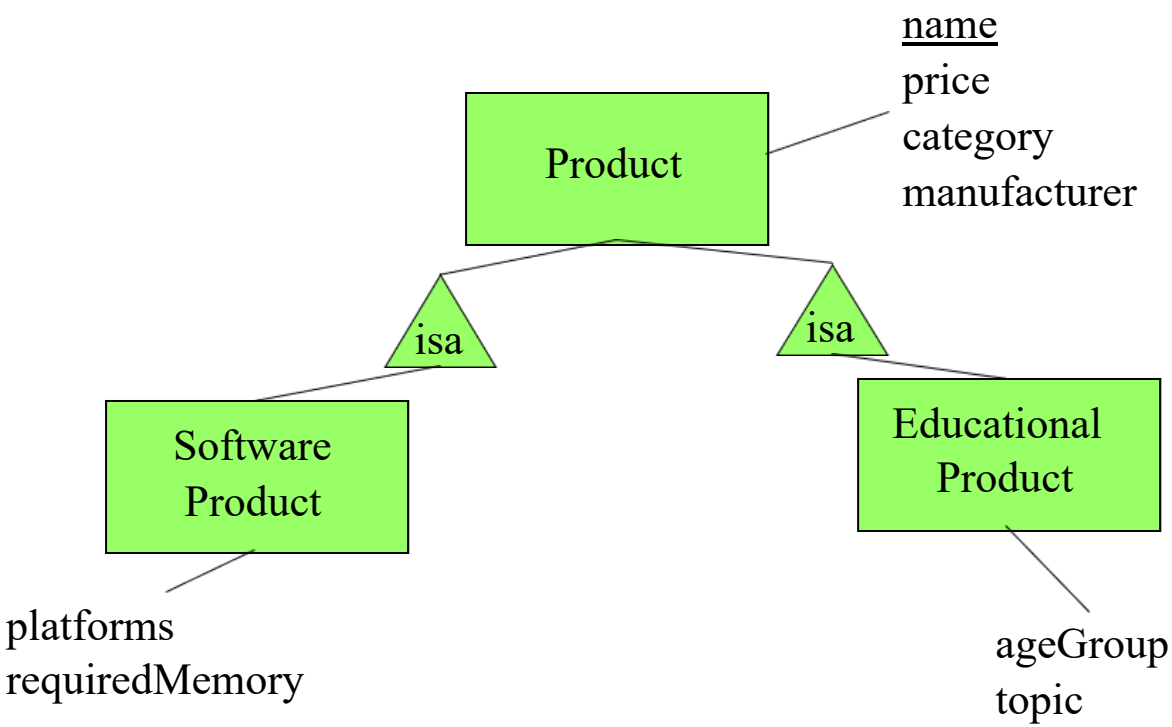
Week5_Course_part3

Relational Data Model-Special cases part II

The Relational Data Model

- Basic stuff & Basic cases
- Special cases
  - combining two relations
  - translating weak entity sets
- Special cases
  - translating subclasses
- Integrity constraints

Translating Subclass Entities



Option #1: the OO Approach

4 tables: each object can only belong to a single table

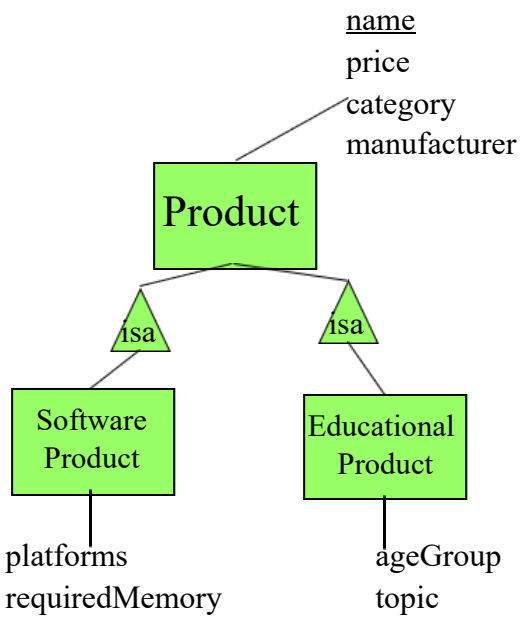
Product(name, price, category, manufacturer)

EducationalProduct(name, price, category, manufacturer, ageGroup, topic)

SoftwareProduct(name, price, category, manufacturer, platforms, requiredMemory)

EducationalSoftwareProduct(name, price, category, manufacturer, ageGroup, topic, platforms, requiredMemory)

All names are distinct



Option #2: the E/R Approach

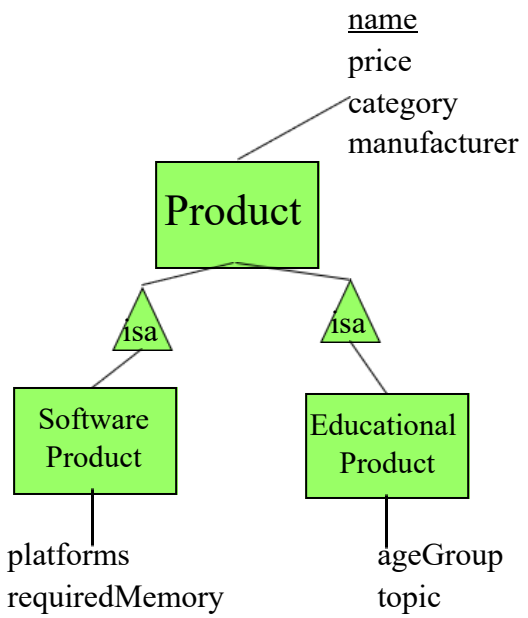
Product(name, price, category, manufacturer)

EducationalProduct(name, ageGroup, topic)

SoftwareProduct(name, platforms, requiredMemory)

No need for a relation EducationalSoftwareProduct

Same name may appear in several relations



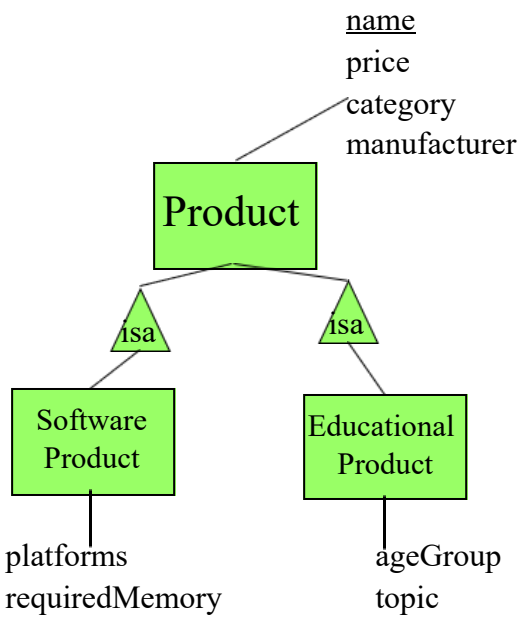
Option #3: The Null Value Approach

Have one table:

Product (name, price, category, manufacturer, age-group, topic, platforms, required-memory)

Some values in the table will be NULL, meaning that the attribute not make sense for the specific product.

Too many NULL’s

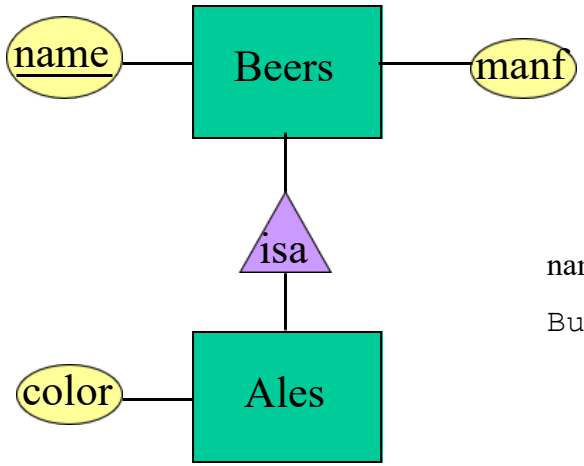


Translating Subclass Entities: The Rules

Three approaches:

1. *Object-oriented* : each entity belongs to exactly one class; create a relation for each class, with all its attributes.
2. *E/R style* : create one relation for each entity set, with only the key attribute(s) and attributes attached to that entity set; entity represented in all relations to whose entity sets it belongs.
3. *Use nulls* : create one relation; entities have null in attributes that don’t belong to them.

Object Oriented



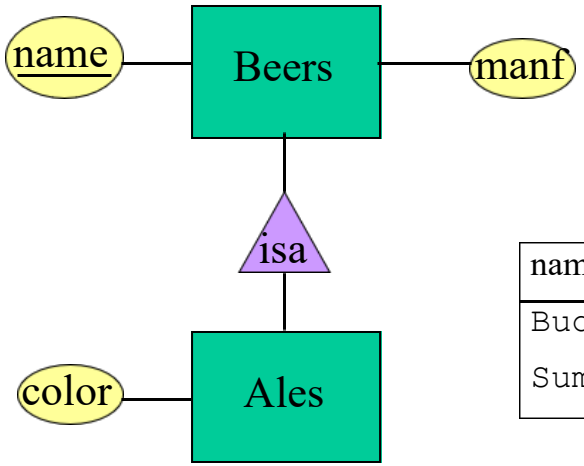
name	manf
Bud	Anheuser-Busch

Beers

name	manf	color	
Summerbrew		Pete's	dark

Ales

E/R Style



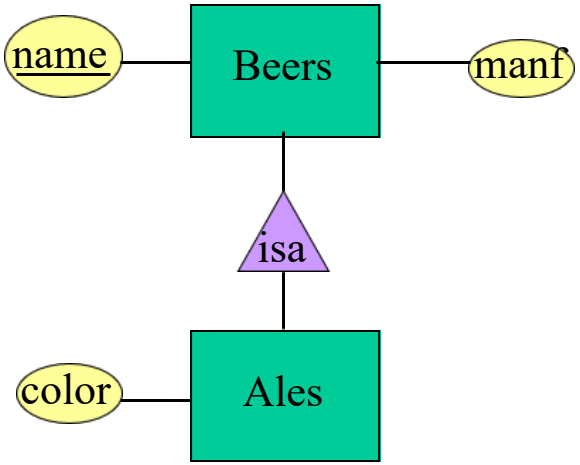
name	manf
Bud	Anheuser-Busch
Summerbrew	Pete's

Beers

name	color
Summerbrew	dark

Ales

Using Nulls



name	manf	color	
Bud	Anheuser-Busch	NULL	
Summerbrew	Pete's		dark

Beers

Comparisons

- O-O approach good for queries like “find the color of ales made by Pete’s.”
  - Just look in Ales relation.
- E/R approach good for queries like “find all beers (including ales) made by Pete’s.”
  - Just look in Beers relation.
- Using nulls wastes space if there are *lots* of attributes that are usually null.

Special cases part 2: Summary

- Translating subclasses
  - O-O
  - E/R
  - Using nulls
  - With trade-offs

单选题 1分

互动交流一

在对子类层次结构进行关系转换时，以下说法错误的是：

A O-O方法最省存储空间

B O-O方法中同一实体信息可能存在于多张表中

C E/R方法中各个表之间的某些元组的键属性取值可能相同

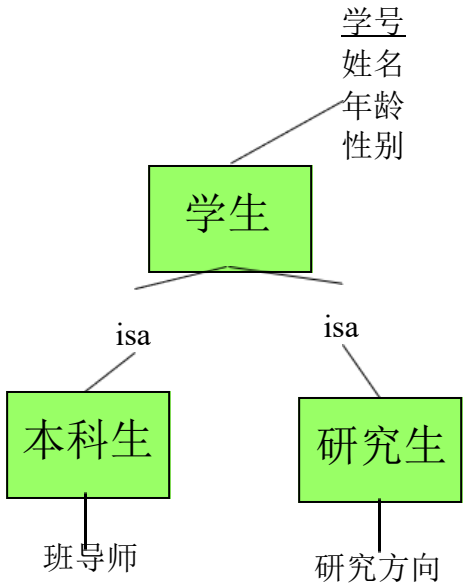
D 使用空值的方法只需要一张表来存储各类实体信息

提交

投票 最多可选2项

互动交流二

如果采用0-0的方法对该子类层次结构进行关系模式的转换，最终生成几张表？



A 2

B 3

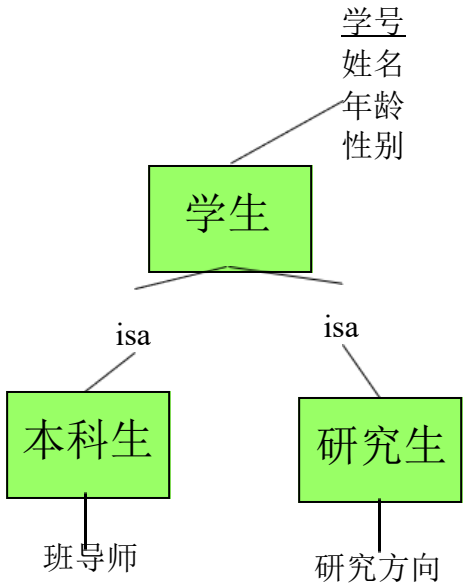
C 4

提交

单选题 1分

互动交流三

如果采用0-0的方法对该子类层次结构进行关系模式的转换，最终生成的本科生表有几个属性？



A 2

B 4

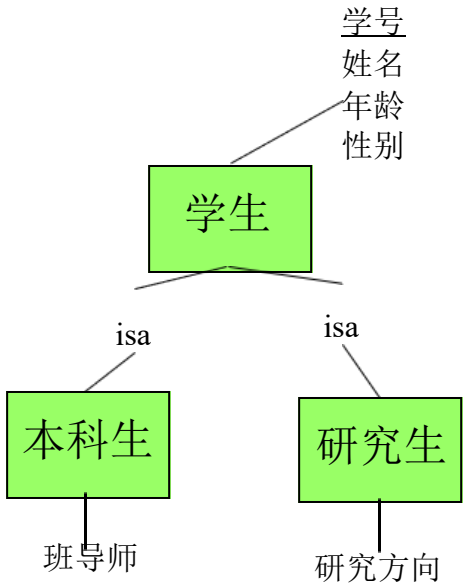
C 5

提交

单选题 1分

互动交流四

如果采用E/R的方法对该子类层次结构进行关系模式的转换，最终生成几张表？



A 2

B 3

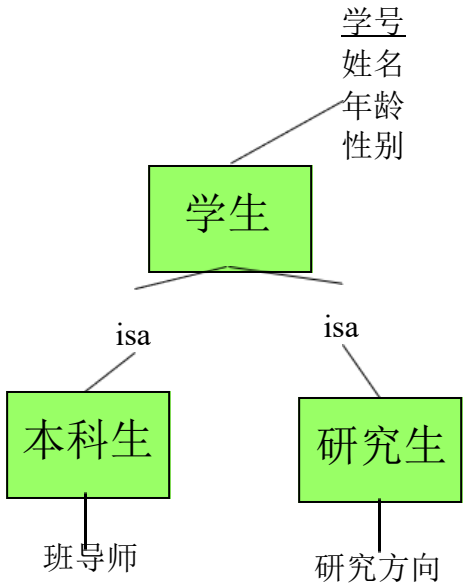
C 4

提交

单选题 1分

互动交流五

如果采用E/R的方法对该子类层次结构进行关系模式的转换，最终生成的本科生表有几个属性？



A 2

B 4

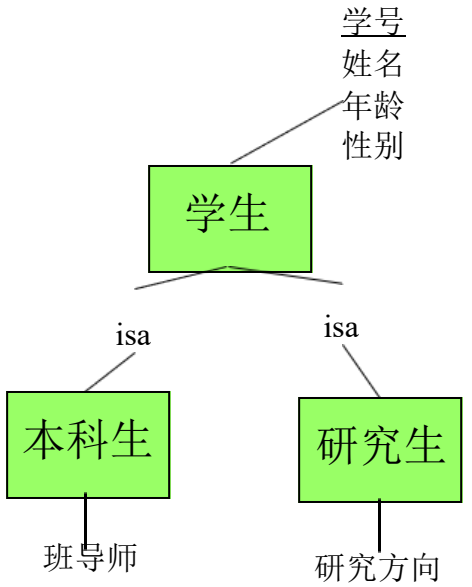
C 5

提交

单选题 1分

互动交流六

如果采用使用空值的方法对该子类层次结构进行关系模式的转换，最终生成几张表?



A 1

B 3

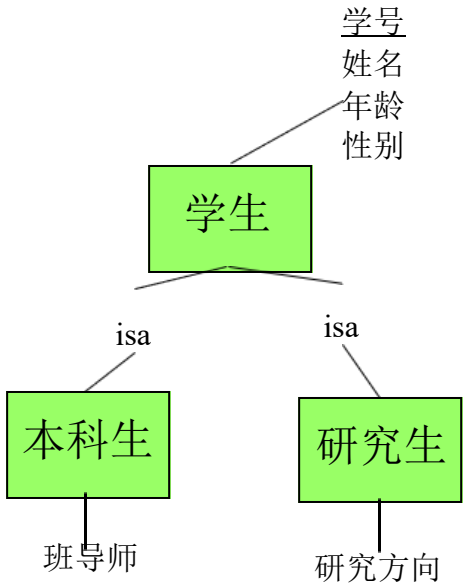
C 4

提交

单选题 1分

互动交流七

如果采用使用空值的方法对该子类层次结构进行关系模式的转换，最终生成表的属性个数?



A 4

B 5

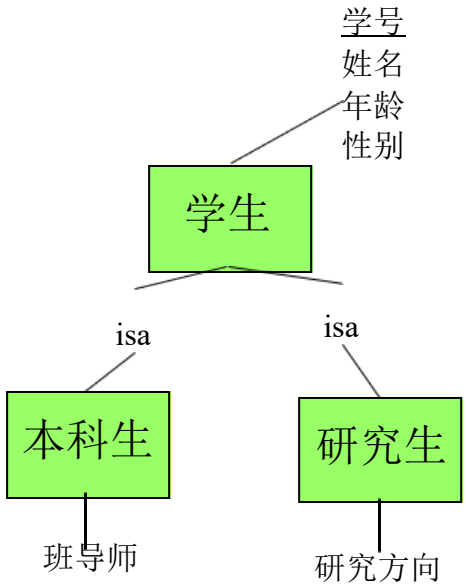
C 6

提交

单选题 1分

互动交流八

如果要查询“年龄小于23的研究生的研究方向”，采用哪种方式查询效率最低?



A O-O

B E/R

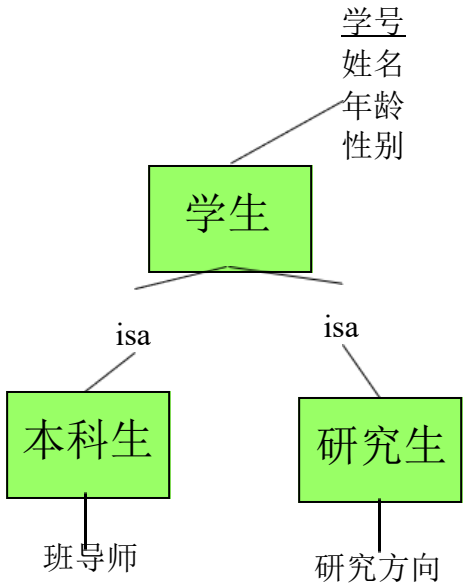
C 使用空值

提交

单选题 1分

互动交流九

如果要查询“年龄小于23的学生的姓名和学号”，采用哪种方式需要查询2张表?



A O-O

B E/R

C 使用空值

提交

Week5_Course_part4

Relational Data Model-Integrity constraints

The Relational Data Model

- Basic stuff & Basic cases
- Special cases
  - combining two relations
  - translating weak entity sets
- Special cases
  - translating subclasses
- Integrity constraints

数据模型

- **数据模型**
  - 现实世界数据特征的抽象，也是现实世界的模拟
- **数据库系统都是基于某种数据模型**
- **概念模型**
  - 按用户的观点来对数据建模，用于数据库设计
    - ER模型
- **逻辑模型**
  - 按计算机系统的观点对数据建模，用于DBMS的实现
    - 层次模型、网状模型、关系模型、面向对象模型…
- **物理模型**
  - 对数据最底层的抽象，描述数据在系统内部的表示方式和存取方法

数据模型

- 从现实世界到概念模型的转换
  - 由数据库设计人员完成
- 从概念模型到逻辑模型的转换
  - 可以由数据库设计人员完成，也可以用数据库设计工具协助设计人员完成
- 从逻辑模型到物理模型的转换
  - 一般是由DBMS完成

关系模型概述

- 单一的数据结构--关系（表）
  - 任何一个关系数据库都是由若干张互相关联的表组成
- 关系操作
  - 查询操作：选择、投影、连接、除、并、交、差
  - 更新操作：增加、删除、修改
  - 关系操作的特点是集合操作方式
- 关系的三类完整性约束
  - 实体完整性、参照完整性和用户定义的完整性
  - 实体完整性和参照完整性是关系模型必须满足的完整性约束

实体完整性(Entity Integrity)

- 主属性(主键中的属性)不能取空值
- 实体应该是可区分的，主键是区分实体的唯一性标识，因此不能为空（即不能不知道）。
- 例如：
  - 选修（学号，课程号，成绩）
  - 人（身份证号，姓名，家庭住址，出生日期，...）

参照完整性(Referential Integrity)

- 设*F*是基本关系*R*的一个或一组属性，如果*F*与基本关系*S*的主键K_S相对应，则称*F*是基本关系*R*的外键(foreign key).
- 外键或者取参照关系中的某个主键值，或者取空值（表示还没有分配）
  - Stu(sno,name,age,sex,deptno), Dept(deptno,deptname)
  - Stu(sno,name,age), Course(cno,cname), SC(sno,cno,grade)
  - (sno和cno既是主键也是外键)

用户定义完整性(User-defined Integrity)

- 用户定义的完整性就是针对某一具体关系数据库的约束条件。它反映某一具体应用所设计的数据必须满足的语义要求。

- 例如：某个属性必须取唯一值或非空值，某些属性值之间应满足一定的函数关系，某个属性的取值范围等等。

Integrity constraints: Summary

- Entity Integrity
- Referential Integrity
- User-defined Integrity

多选题 1分

互动交流一 —不定项选择

右边某张表违反了以下哪种完整性约束?

A 实体完整性

B 参照完整性

C 用户定义完整性

D 没有违反这三类完整性约束

学生:

学号	姓名	出生日期	所属学院
20161001	张三	1994.8.5	英语
20162023	李四	NULL	数学
20162023	王五	1994.11.15	计算机

学院:

学院名称	类型	人数规模
计算机	工科	800
数学	理科	600
英语	文科	420

提交

Database System - Nankai

多选题 1分

互动交流二 —不定项选择

右边某张表违反了以下哪种完整性约束?

A 实体完整性

B 参照完整性

C 用户定义完整性

D 没有违反这三类完整性约束

学生:

学号	姓名	出生日期	所属学院
20161001	张三	1994.8.5	NULL
20162023	李四	NULL	数学
20164512	王五	1994.11.15	计算机

学院:

学院名称	类型	人数规模
计算机	工科	800
数学	理科	600
英语	文科	420

提交

多选题 1分

互动交流三 —不定项选择

右边某张表违反了以下哪种完整性约束?

A 实体完整性

B 参照完整性

C 用户定义完整性

D 没有违反这三类完整性约束

学生:

学号	姓名	出生日期	所属学院
20161001	张三	1994.8.5	NULL
20162023	李四	NULL	数学
20164512	王五	1994.11.15	金融

选课:

课号	学号	成绩
1001	20161001	95
1002	NULL	70
1003	20162023	80

课程:

课号	课名	学分
1001	数据库	3
1002	英语	NULL
1003	操作系统	3

提交

Database System - Nankai

多选题 1分

互动交流四 —不定项选择

右边某张表违反了以下哪种完整性约束?

A 实体完整性

B 参照完整性

C 用户定义完整性

D 没有违反这三类完整性约束

学生:

学号	姓名	出生日期	所属学院
20161001	张三	1994.8.5	计控
20162023	李四	NULL	数学
20164512	王五	1994.11.15	金融

选课:

课号	学号	成绩
1001	20161001	95
1002	20161001	NULL
1007	20162023	80

课程:

课号	课名	学分
1001	数据库	3
1002	英语	NULL
1003	操作系统	3

提交

Database System - Nankai