

第一章 绪论

1. 简述数据、数据库、数据库管理系统、数据库系统的含义及其联系？
2. 信息模型与数据模型的有何区别，试述数据模型的三要素。
3. 简述数据库 SPARC 模式结构的内容。
4. 什么是数据独立性？数据独立性包含哪两种？
5. 关系模型相对于层次和网状模型来说有哪些优缺点？
6. 有一个记录球队、球员和球迷信息的数据库，包括：
 - 1) 球队：球队编号、名称、球员、队长（队员之一）、队服颜色；
 - 2) 球员：球员编号、姓名、年龄、籍贯、所属球队；
 - 3) 球迷：姓名、年龄、喜爱的球队、喜爱的球员、喜爱的颜色。

用 E-R 图画该数据库的信息模型。

第二章 关系数据库

1. 比较以下概念的区别：
 - 1) 关系与关系模式
 - 2) 笛卡尔积与连接
 - 3) 等值连接与自然连接
 - 4) 自然连接与外连接

2. 设有以下关系

R	S	T	U	V																																														
<table><tr><td>X</td><td>Y</td></tr><tr><td>a</td><td>d</td></tr><tr><td>b</td><td>a</td></tr><tr><td>c</td><td>c</td></tr></table>	X	Y	a	d	b	a	c	c	<table><tr><td>X</td><td>Y</td></tr><tr><td>d</td><td>a</td></tr><tr><td>b</td><td>a</td></tr><tr><td>d</td><td>c</td></tr></table>	X	Y	d	a	b	a	d	c	<table><tr><td>Y</td><td>Z</td></tr><tr><td>b</td><td>b</td></tr><tr><td>b</td><td>e</td></tr><tr><td>c</td><td>d</td></tr></table>	Y	Z	b	b	b	e	c	d	<table><tr><td>X</td><td>Y</td><td>Z</td><td>W</td></tr><tr><td>a</td><td>b</td><td>c</td><td>d</td></tr><tr><td>a</td><td>b</td><td>e</td><td>f</td></tr><tr><td>c</td><td>a</td><td>c</td><td>d</td></tr></table>	X	Y	Z	W	a	b	c	d	a	b	e	f	c	a	c	d	<table><tr><td>Z</td><td>W</td></tr><tr><td>e</td><td>f</td></tr><tr><td>c</td><td>d</td></tr></table>	Z	W	e	f	c	d
X	Y																																																	
a	d																																																	
b	a																																																	
c	c																																																	
X	Y																																																	
d	a																																																	
b	a																																																	
d	c																																																	
Y	Z																																																	
b	b																																																	
b	e																																																	
c	d																																																	
X	Y	Z	W																																															
a	b	c	d																																															
a	b	e	f																																															
c	a	c	d																																															
Z	W																																																	
e	f																																																	
c	d																																																	

请写出下列运算的结果：

- 1) $R \cup S$
- 2) $R \cap S$
- 3) $R \times S$
- 4) $U \div V$
- 5) R 和 T 外部并
- 6) U 与 T 的外连接、左外连接、右外连接

3. 已知学生表 S、课程表 C 和学生选课表 SC 的关系模式如下：

S (Sno, Sname, Sex, age), 即 (学号、姓名、性别、年龄)

C (Cno, Cname, Teacher), 即 (课程号、课程名, 教师)

SC (Sno, Cno, Grade), 即 (学号、课程号、成绩)

试用关系代数表示下列查询：

- 1) 查询“王欣”老师所授课程号和课程名；
- 2) 查询选修了课程名为“数据库”或者“数据结构”的学生学号；
- 3) 查询“李琳”同学所选修的课程号及课程名；
- 4) 查询至少选修了两门课程的学生学号；
- 5) 查询全部学生都选修的课程号和课程名；
- 6) 查询至少选修了“王欣”老师所授全部课程的学生姓名。

4. 利用关系代数式的等价规则, 判断下面等式是否成立(指出使用了哪些规则):

$$1) \delta_{R.A=S.A \wedge R.B=S.B} (R \times S) = R \bowtie_{R.A=S.A \wedge R.B=S.B} S$$

$$\begin{aligned} 2) & \Pi_{sno, sname} (\delta_{cname='db'} (\Pi_{sno, sname, cname} (\delta_{S.sno=SC.sno \wedge SC.cno=C.cno} (S \times SC \times C)))) \\ &= \Pi_{sno, sname} (S \bowtie SC \bowtie \delta_{cname='db'} (C)) \end{aligned}$$

第三章 关系数据库标准查询语言 SQL

1. 简述 SQL 语言的特点。
2. 简述视图的作用。
3. 已知学生表 student、课程表 course 和学生选课表 s_c 的关系模式如下：

student (sno, sname, sex, age, dept, place)

即 (学号、姓名、性别、年龄, 所属院系, 籍贯),

course (cno, cname, credit, pcno)

即 (课程号、课程名, 学分, 预修课程号),

s_c (sno, cno, grade)

即 (学号、课程号、成绩),

- 1) 查询考试成绩不及格的学生的学号及成绩；

- 2) 查询年龄在 19~25 岁 (包含) 之间的学生的姓名、院系和年龄, 并按年龄降序排列;
 - 3) 查询姓名中含有“浩”字的学生信息;
 - 4) 按院系查询学生总人数;
 - 5) 计算选修了 008 号课程的学生的平均成绩、最高分及最低分;
 - 6) 求平均成绩在 85 分以上 (含) 的各门课程及平均成绩。
4. 已知学生表 S、课程表 C 和学生选课表 SC 的关系模式如下:
- S (Sno, Sname, Sex, age), 即 (学号、姓名、性别、年龄)
- C (Cno, Cname, Teacher), 即 (课程号、课程名, 教师)
- SC (Sno, Cno, Grade), 即 (学号、课程号、成绩)
- 试用 SQL 语句实现下列查询:
- 1) 查询“张琳”老师所授课程号和课程名;
 - 2) 查询选修课程名为“C 语言”或者“数据库”的学生学号;
 - 3) 查询“陈浩”同学所选修课程的课程号及课程名;
 - 4) 查询至少选修课程号为 C1 和 C2 的学生姓名;
 - 5) 用 EXISTS 查询学习课程号为 C5 的学生姓名和年龄;
 - 6) 查询不选修 C3 课程的学生姓名和性别。
5. 设有以下关系:
- 职工: E(职工号, 姓名, 性别, 职务, 家庭住址, 部门号)
- 部门: D(部门号, 部门名称, 地址, 电话)
- 保健: B(保健号, 职工号, 检查日期, 健康状况)
- 用关系代数 (1-4 题), SQL 语言完成全部功能:
- 1) 查询所有女科长的姓名和家庭地址;
 - 2) 查询部门名称为‘办公室’的科长姓名和家庭地址;
 - 3) 查询部门名称为‘财务科’、健康状况为‘良好’的职工姓名和家庭住址;
 - 4) 删除职工关系中职工号为‘1006’的记录;
 - 5) 将职工号为 1006 的职工健康状况改为‘一般’;
 - 6) 建立健康状况为‘差’的职工情况的视图。
6. 简述嵌入式 SQL 中, 主语言和 SQL 语言之间有哪几种通信方式。

第四章关系数据库设计理论

- 1、设关系模式 $R(ABCD)$, F 是 R 上成立的函数依赖集, $F = \{A \rightarrow C, C \rightarrow B\}$, 相对于 F 写出关系模式 R 的主关键字。
- 2、设关系模式 $R(ABC)$, F 是 R 上成立的函数依赖, $F = \{B \rightarrow C, C \rightarrow A\}$, 那么 $\rho = \{AB, AC\}$ 相对于 F 是否保持无损分解和函数依赖? 说明理由。
- 3、关系模式 $R(ABCD)$, F 是 R 上成立的函数依赖, $F = \{AB \rightarrow CD, A \rightarrow D\}$ 。
 - 1) 试说明 R 不是 2NF 模式的理由;
 - 2) 试把 R 分解成 2NF 模式集。
- 4、设关系模式 $R(ABC)$, F 是 R 上成立的函数依赖, $F = \{C \rightarrow B, B \rightarrow A\}$ 。
 - 1) 试说明 R 不是 3NF 模式集;
 - 2) 试把 R 分解为 3NF 模式集。
- 5、设有关系模式 R (职工名, 项目名, 工资, 部门号, 部门经理), 如果规定每个职工可以参加多个项目, 每个项目都可以各领一份工资; 每个项目只属于一个部门管理; 每个部门只有一个部门经理。要求:
 - 1) 写出关系模式 R 的函数依赖和主键;
 - 2) R 是 2NF 模式吗? 若不是请说明理由, 并把 R 分解到 2NF 模式集;
 - 3) 把 R 分解到 3NF 模式集。
- 6、现需要建立一个关于学生、班级、系和社团等信息的一个关系数据库系统, 一个系有若干专业, 每个专业每年只招一个班, 每个班有若干名学生, 一个系的学生住在同一宿舍区, 每个学生可以参加若干个社团, 每个社团有若干名学生。需要考虑记录以下信息:

学生属性有: 学号、姓名、出生年月、系名、班级号、宿舍区;

班级属性有: 班级号、专业名、系名、人数、入学年份;

系的属性有: 系名、系号、系办公地点、人数;

社团属性有: 社团名、成立年份、地点、人数、学生参加社团的年份。

请写出你设计的关系模式, 指出每个关系模式的候选键、外键, 写出每个关系模式的函数依赖集。

第五章 数据库的保护

- 1、简述数据库安全性控制有哪些措施？
- 2、简述数据库并发操作会带来哪些问题？
- 3、简述事务的特性。
- 4、简述数据库故障的几种类型及相应恢复方法。
- 5、已知两个关系模式：

职工(职工号，姓名，年龄，职务，工资，部门号)

部门(部门号，名称，经理名，电话)

用 SQL 语言定义这两个关系模式，并实现以下完整性约束条件：

- 1) 定义每个关系模式的主键；
 - 2) 定义关系中的参照完整性；
 - 3) 定义每个职工的年龄不能超过 60 岁。
- 6、已知学生表 student、课程表 course 和学生选课表 s_c 的关系模式如下：

student (sno, sname, sex, age, dept, place)

即 (学号、姓名、性别、年龄、所属院系、籍贯)，

course (cno, cname, credit, pcno)

即 (课程号、课程名、学分、预修课程号)，

s_c (sno, cno, grade)

即 (学号、课程号、成绩)，

试定义以下完整性约束：

- 1) student 中学生年龄在 15~25 之间；
- 2) 在 s_c 中插入元组时其 sno 和 cno 必须在关系 student 和 course 中出现；
- 3) 删除关系 course 中元组时，首先要把 s_c 中具有同样 cno 的元组删除；
- 4) 修改 student 中某个 sno 时，同时修改 s_c 中 sno 的值。

第六章 数据库的设计

- 1、简述数据库设计的基本步骤。
- 2、要建立一个公司数据库，公司有多个部门，每个部门有多个职员，一个职员仅属于一个部门，且一个员工在一个工程中工作，一个工程有多个职员参与，有多个供应商为各个工程供应不同的设备。

部门属性：部门名，电话

职员属性：职员号、姓名、性别

工程属性：工程名、地点

设备属性：设备号、设备名、产地

供应商属性：姓名、电话

请完成以下处理：

1) 手机满足上述要求的 E-R 图

2) 将 E-R 图转换为关系模式，用下划线给出其主键。

3、有运动队和运动会两个方面的实体：

运动队方面有：

运动队：队名、教练员名、队员姓名

队员：队名、队员姓名、性别、项目名

其中，一个运动队有多个队员，一个队员仅属于一个运动队，一个队仅一个教练。

运动会方面有：

运动队：队编号、队名、教练姓名

项目：项目名、参加运动队的编号、队员姓名、性别、比赛场地。

其中，一个项目有多个运动队参加，一个运动员可以参加多个项目，一个项目一个比赛场地。

1) 分别设计运动队和运动会两个局部 E-R 图；

2) 将它们合并为一个全局 E-R 图(注意合并时的冲突)。

数据库基础习题

助教：魏晓东 胡心亭

QQ群：724255380

第一章

1. 简述数据、数据库、数据库管理系统、数据库系统的含义及其联系？ ppt1.1.1
2. 信息模型与数据模型的有何区别，试述数据模型的三要素。
 - 信息模型是指按照用户的观点对信息建模
 - 数据模型是按照计算机系统的观点对数据建模
 - 数据模型的三要素：数据结构，数据操作，完整性约束

（结构）数据模型

常见错误：

数据模型的三个方面要求：

比较真实模拟真实世界

容易为人所理解

便于计算机实现

数据模型两个层次： ppt1.2

概念（数据）模型：也称信息模型

（结构）数据模型：又称为（基本）数据模型

3. 简述数据库 SPARC 模式结构的内容。

并列关系，不是外模式包含外视图...

	schema	= view	= level
External	外模式 子模式	用户视图 外视图	外部级 局部逻辑级
conceptual	模式 概念模式	概念视图 DBA视图	概念级 全局逻辑级
internal	内模式 存储模式	内部视图	内部级 物理级 存储级

数据库系统的模式结构：三模式两映像

4. 什么是数据独立性？数据独立性包含哪两种？

把数据的定义和描述从应用程序中分离出去

物理独立性：当存储结构改变时，可以通过修改概念模式/内模式的映像使概念级保持不便，这样外部级和应用程序也不会改变。

逻辑独立性：当概念级发生改变时，可以通过修改概念模式/外模式的映像使外部级尽量保持不变，应用也就不需改变。

5. 关系模型相对于层次和网状模型来说有哪些优缺点？

优点：

- 1) 建立在严格的数学概念基础上
- 2) 概念单一，实体、联系均用关系来表示
- 3) 存取路径对用户透明，数据独立性更高，保密性更好。
简化程序员工作和数据库开发建立工作。

缺点：

- 1) 存取路径对用户透明，查询效率不高
- 2) 因存取路径对用户透明，必须对用户查询进行优化，增加了开发 DBMS 的难度。

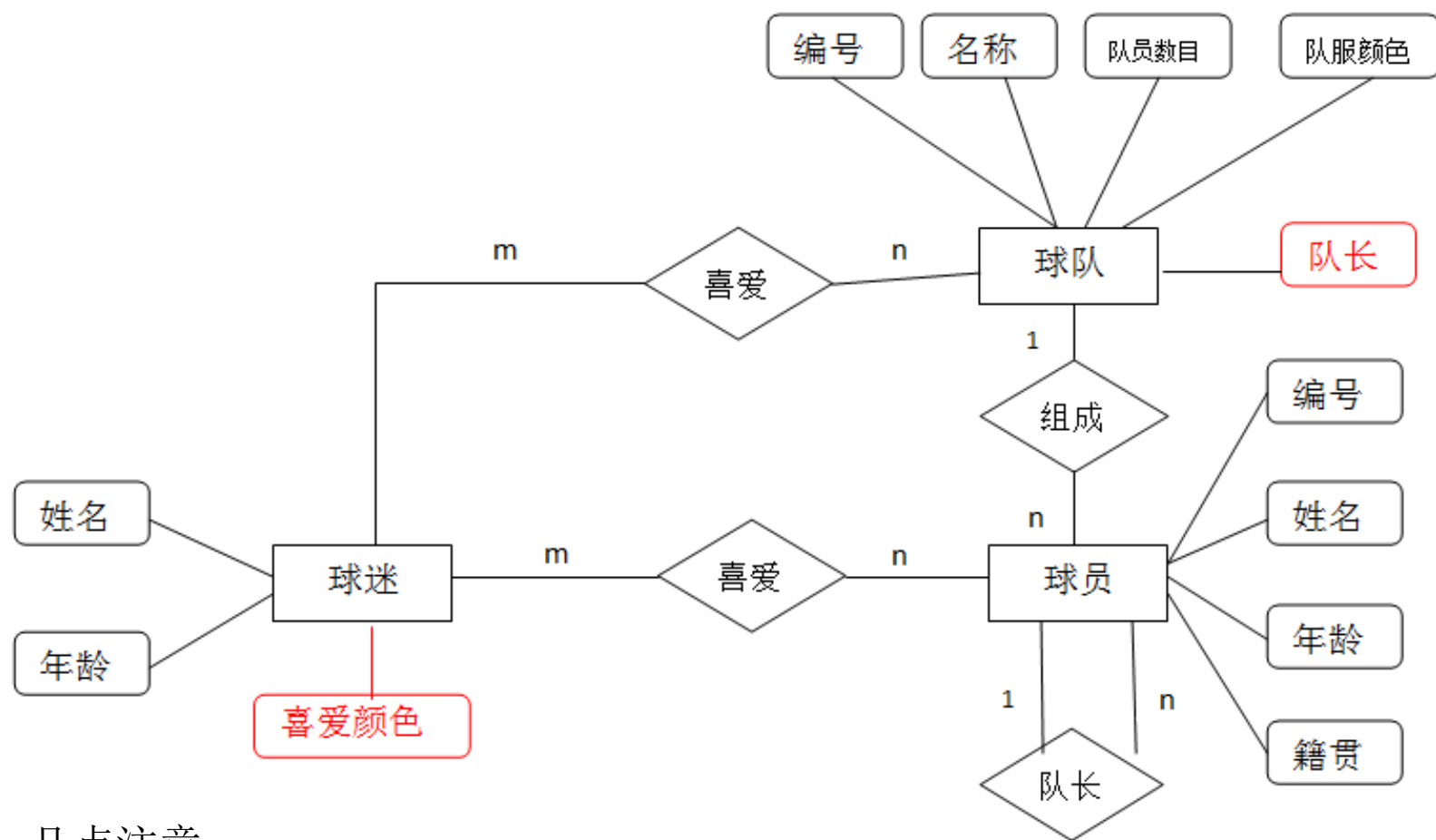
6、有一个记录球队，球员，球迷的数据库：

(1) 球队：编号，名称，队员数目，队长，队服颜色；

(2) 球员：姓名，编号，年龄，籍贯，球队；

(3) 球迷：姓名，年龄，喜爱球队，喜爱队员，喜爱颜色

用E-R图画该数据库信息模型：



几点注意：

- 队长对球队可以1:1联系或作为属性
- 不要重复关系和属性 如喜爱颜色和球迷-喜爱-颜色
- 线上要有标注
- 属性是圆框不是长方形

第二章

2. (3) $R \times S$

排列组合

R

X	Y
a	d
b	a
c	c

S

Y	Z
d	a
b	a
d	d

X	Y	Y	Z
a	d	d	a
a	d	b	a
a	d	d	d
b	a	d	a
b	a	b	a
b	a	d	d
c	c	d	a
c	c	b	a
c	c	d	d

2. (4) $U \div V$

U					V	
X	Y		Z	W	Z	W
a	b		c	d	e	f
a	b		e	f	c	d
c	a		c	d		

X	Y
a	b

2. (5) R 和 T 外部并

R

X	Y
a	d
b	a
c	c

T

Y	Z
b	b
b	e
c	d

➤外部并（Outer Union）

若关系R和S不同类，则新关系的属性由R和S的属性组成，公共属性只取一次，新关系的元组由属于R或S的元组构成，新增的属性上均填空（null）

外部并：

X	Y	Z
a	d	null
b	a	null
c	c	null
null	b	b
null	b	e
null	c	d

不需连接

2. (6) U与T的外连接，左外连接，右外连接

T		U			
Y	Z	X	Y	Z	W
b	b	a	b	c	d
b	e	a	b	e	f
c	d	c	a	c	d

外连接：

自然连接时，把该舍弃的元组也保存在新关系中，在新增加的属性上填空值（null）

X	Y	Z	W
a	b	e	f
a	b	c	d
c	a	c	d
null	b	b	null
null	c	d	null

T	
Y	Z
b	b
b	e
c	d

U			
X	Y	Z	W
a	b	c	d
a	b	e	f
c	a	c	d

- 左外连接: **U与T**
保留U中的项

X	Y	Z	W
a	b	c	d
a	b	e	f
c	a	c	d

- 右外连接:
保留T中的项

X	Y	Z	W
null	b	b	null
a	b	e	f
null	c	d	null

3.已知学生表 S、课程表 C 和学生选课表 SC 的关系模式如下：
S (Sno, Sname, Sex, age)，即（学号、姓名、性别、年龄）
C (Cno, Cname, Teacher)，即（课程号、课程名，教师）
SC (Sno, Cno, Grade)，即（学号、课程号、成绩）
试用关系代数表示下列查询：

(1)查询“王欣”老师所授课程号和课程名；

$$\pi_{cno, cname}(\sigma_{teacher="wangxin"}(C))$$

(2)查询选修了课程名为“数据库”或者“数据结构”的学生学号；

$$\pi_{sno}(SC \bowtie \pi_{cno}(\sigma_{cname="DB" \vee cname="DBMS"}(C)))$$

(3)查询“李琳”同学所选修的课程号及课程名；（考试可能会考优化）

$$\pi_{cno}(SC \bowtie \pi_{sno}(\sigma_{sname="lilin"}(S))) \bowtie \pi_{cno, cname}(C)$$

3. 已知学生表 S、课程表 C 和学生选课表 SC 的关系模式如下：
 S (Sno, Sname, Sex, age)，即 (学号、姓名、性别、年龄)
 C (Cno, Cname, Teacher)，即 (课程号、课程名，教师)
 SC (Sno, Cno, Grade)，即 (学号、课程号、成绩)

(4) 查询至少选修了两门课程的学生学号；

$$\pi_{\text{sno}}(\delta_{1=4 \wedge 2 \neq 5}(\text{SC} \times \text{SC}))$$

(5) 查询全部学生都选修的课程号和课程名；

$$(\pi_{\text{cno}, \text{sno}}(\text{SC}) \div \pi_{\text{sno}}(S)) \bowtie \pi_{\text{cno}, \text{cname}}(C)$$

(6) 查询至少选修了“王欣”老师所授全部课程的学生姓名

$$\pi_{\text{sname}}((\pi_{\text{cno}, \text{sno}}(\text{SC}) \div \pi_{\text{cno}}(\delta_{\text{teacher}=\text{'wangxin'}}(C))) \bowtie S)$$

常见错误：

$$(6) \quad \pi_{\text{sname}}(\pi_{\text{sno}, \text{sname}}(S) \bowtie \pi_{\text{sno}, \text{sco}}(SC) \bowtie \pi_{\text{cno}}(\sigma_{\text{Teacher}=\text{'wangxin'}}(C)))$$

这样是选出了所有选了王欣课的老师

没有注意表和关系 $\sigma_{\text{cname}=\text{'DB'}}(SC)$

4. 利用关系代数式的等价规则,判断下面等式是否成立(指出使用了哪些规则):

$$1) \sigma_{R.A=S.A \wedge R.B=S.B} (R \times S) = R \bowtie_{R.A=S.A \wedge R.B=S.B} S$$

$$2) \Pi_{sno, sname} (\sigma_{cname='db'} (\Pi_{sno, sname, cname} (\sigma_{S.sno=SC.sno \wedge SC.cno=C.cno} (S \times SC \times C)))) \\ = \Pi_{sno, sname} (S \bowtie SC \bowtie \sigma_{cname='db'} (C))$$

1)成立, 选择与连接操作的结合率

2)成立, 投影的串接定律

选择与连接操作的结合率

选择对自然连接的分配率

第三章

1. 简述 SQL 语言的特点。（课件3.1.2）

- 综合统一：集DDL、DML、DCL于一体，语言风格统一
- 面向集合的操作方式：操作对象、查询结果都可以是元组的集合
- 高度非过程化：存取路进透明。
- 以统一的语法结构提供两种使用方式：自含式、嵌入式
- 语言简洁，易学易用

2. 简述视图的作用。（课件3.5.4）

- 视图能简化用户的操作
- 视图可以使用户多角度看待同一数据
- 视图对重构数据库提供了一定的逻辑独立性
- 视图能对数据提供安全保护

3. 已知学生表student、课程表course和学生选课表s_c的关系模式如下：
student (sno, sname, sex, age, dept, place) 即（学号、姓名、性别、年龄, 所属院系, 籍贯），
course (cno, cname, credit, pcno) 即（课程号、课程名, 学分, 预修课程号），
s_c (sno, cno, grade) 即（学号、课程号、成绩），

考试的时候注意表名不要写错

1) 查询考试成绩不及格的学生的学号及成绩;

```
select sno, grade from s_c where grade < 60;
```

2) 查询年龄在19~25岁（包含）之间的学生的姓名、院系和年龄，并按年龄降序排列；

```
select sname, dept, age
```

```
from student
```

```
where age between 19 and 25
```

```
order by age desc; (不能写作19<age<25)
```

3) 查询姓名中含有“浩”字的学生信息；

```
select * from student where sname like '%浩%';
```

4) 按院系查询学生总人数；

```
select dept, count(distinct sno) from student group by dept;
```

5) 计算选修了008号课程的学生们的平均成绩、最高分及最低分;

```
select avg(grade), max(grade), min(grade)
from s_c
where cno = '008'; (尽量用单引号)
```

给它们赋个名字: `select avg(grade) avg, ...`

6) 求平均成绩在85分以上(含)的各门课程及平均成绩。

```
select cno, avg(grade) from s_c group by cno
having avg(grade)>=85;
```

```
select cname, avg(grade) from course, s_c where
course.cno = s_c.cno group by cname having avg(grade)>=85;
```

```
select course.cno, course.cname, avg(grade) from course, s_c
where course.cno = s_c.cno group by course.cno, course.cname
having avg(grade)>=85; 两两对应
```


4. 已知学生表S、课程表C和学生选课表SC的关系模式如下：

S (Sno, Sname, Sex, age)，即（学号、姓名、性别、年龄）

C (Cno, Cname, Teacher)，即（课程号、课程名，教师）

SC (Sno, Cno, Grade)，即（学号、课程号、成绩）

试用SQL语句实现下列查询：

1) 查询“张琳”老师所授课程号和课程名；

select Cno,Cname, from C where Teacher='张琳';

2) 查询选修课程名为“C语言”或者“数据库”的学生学号；

select Sno from C , SC where SC.Cno = C.Cno AND C.Cname IN ('C语言','数据库');

**select distinct Sno from SC where Cno in (select Cno from C
where Cname = 'C语言' or cname = '数据库')**

不可以写成 **Cname = 'C语言'or '数据库'**

3) 查询“陈浩”同学所选修课程的课程号及课程名;

**select Cno, Cname from S, C, SC where S.Sno = SC.Sno and
SC.Cno = C.Cno and Sname = '陈浩';**

(4) 查询至少选修课程号为 C1 和 C2 的学生姓名;

Exists: select sname from s where exists
(select * from sc where s.sno=sc.sno and sc.cno='C1')
and exists
(select * from sc where s.sno=sc.sno and sc.cno='C2')

In: select sname from s where sno in
(select sno from sc where sc.cno='C1')
and sno in
(select sno from sc where sc.cno='C2')

更多写法:

```
select s.sname
```

```
from sc,s
```

```
where sc.sno=s.sno and cno='C1' and
```

```
s.sno in (select sno from sc where cno='C2')
```

```
select s.sname
```

```
from s, sc X, sc Y
```

不可以写成sc.sno

```
where s.sno = X.sno and X.sno = Y.sno and X.cno = 'C1' and Y.cno =  
'C2'
```

```
select s.sname from s
```

```
where sno in (select X.sno 或where exists (select *
```

```
from sc X, sc Y where X.sno = Y.sno and X.cno = 'C1' and Y.cno = 'C2')
```

5) 用EXISTS查询学习课程号为C5的学生姓名和年龄;

select Sname, age from S where EXISTS

(select * from [s,]SC where S.Sno=SC.Sno and Cno='C5');

这里**from**后面不要加s,sc 不然会有很多结果 **distinct**

(6) 查询不选修 C3 课程的学生姓名和性别

```
select sname, sex from s
```

```
where NOT EXISTS (select * from sc where sno = s.sno AND cno  
= 'C3')
```

```
select sname, sex from s
```

```
where sno NOT IN (select sno from sc where sno = s.sno AND  
cno = 'C3')
```

常见错误写法:

```
select sname, sex from S, SC
```

```
where S.Sno = SC.Sno and SC.Cno <>'C3'
```

Sno	Cno
001	C1
001	C3

5. 设有以下关系:

职工: E(职工号, 姓名, 性别, 职务, 家庭住址, 部门号)

部门: D(部门号, 部门名称, 地址, 电话)

保健: B(保健号, 职工号, 检查日期, 健康状况)

用关系代数 (1-4题), SQL语言完成全部功能:

1) 查询所有女科长的姓名和家庭地址;

select 姓名, 家庭住址 from E where
性别 = '女' and 职务 = '科长';

$\pi_{\text{name, adress}}(\sigma_{(\text{sex}='women', \text{job}='kezhang')} E)$

2) 查询部门名称为‘办公室’的科长姓名和家庭地址;

所有参与连接的表都要写上

select 姓名, 家庭住址 from E,D where E.部门号 = D.部门号
AND 部门名称 = '办公室' and 职务 = '科长';

$$\pi_{\text{name, adress}}(\delta_{(\text{job}=\text{'kezhang'}, \text{dept_id}=\text{'office'})} E \bowtie D)$$

3) 查询部门名称为‘财务科’、健康状况为‘良好’的职工姓名和家庭住址;

select 姓名,家庭住址 from E,D,B

where E.部门号=D.部门号 and E.职工号=B.职工号 and 部门名称='财务科' and 健康状况='良好';

$$\pi_{\text{name, adress}}(\delta_{(\text{health}=\text{'lianghao'}, \text{dept_id}=\text{'caiwu'})} E \bowtie D \bowtie B)$$

4) 删除职工关系中职工号为‘1006’的记录;

delete from E where 职工号='1006';

$$E - \delta_{(\text{id}=\text{'1006'})} E)$$

5) 将职工号为1006的职工健康状况改为‘一般’;
update B SET 健康状况='一般' where 职工号='1006';

6) 建立健康状况为‘差’的职工情况的视图。
create view bad_health as
select * from E,D,B where E.职工号=B.职工号
AND E.部门号=D.部门号 AND 健康状况='差';

6.简述嵌入式 SQL 中，主语言和 SQL 语言之间有哪几种通信方式.

课本113页

(1) SQL 通信区（SQLCA）

(2) 主变量

(3) 游标

数据库基础习题课 (4-6章)

助教：魏晓东 胡心亭

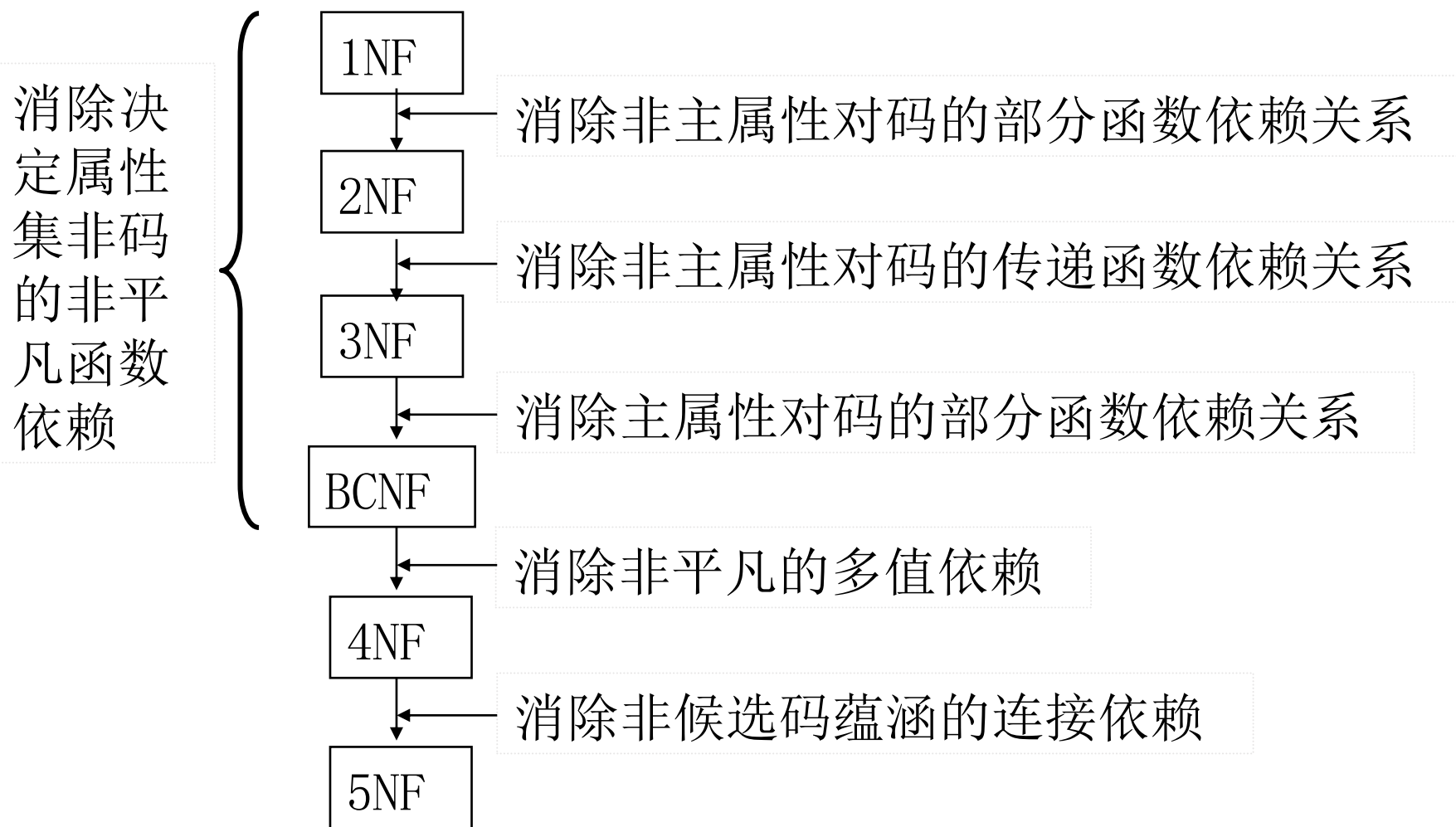
2018年5月21日

第四章 关系数据库设计理论

- 全部内容：数据依赖，范式，关系模式的规范化
 - 关系模式： $R(U, D, DOM, F)$ U 属性组， F 依赖关系
 - 数据依赖($X \rightarrow Y$)：函数依赖（完全、部分、传递），多值依赖，连接依赖
 - 多值依赖： $X \twoheadrightarrow Y$ $Z = U - X - Y$ Y 依赖于 X 但与 Z 无关
 - 连接依赖：课件15页
- 范式：数据依赖满足某种条件级别的关系模式的集合
 - $5NF \in 4NF \in BCNF \in 3NF \in 2NF \in 1NF$

范式规范化(课件16页):

码：关系模式 $R(U, F)$ ， K 为属性组合，若 $K \xrightarrow{f} U$ ，则 K 是一个候选码
主属性和非主属性：包含在任意候选键中的属性叫主属性。



关系模式分解：

- 分解前后关系模式等价：
无损连接性，保持函数依赖
- 无损连接：最简单的情况有共同的主键做自然连接
- 函数依赖：根据已有的函数依赖分解

第四章

1. 设关系模式 $R(ABCD)$, F 是 R 上成立的函数依赖集, $F=\{A \rightarrow C, C \rightarrow B\}$, 相对于 F 写出关系模式 R 的主关键字。

主关键字 (A,D)

2. 设关系模式 $R(ABC)$, F 是 R 上成立的函数依赖, $F=\{B \rightarrow C, C \rightarrow A\}$, 那么 $\rho=\{AB, AC\}$ 相对于 F 是否保持无损分解和函数依赖? 说明理由。

没有保持无损分解和函数依赖 丢失了 $B \rightarrow C$ 的函数依赖

Sno	Sdept	Sloc
95001	CS	A
95002	IS	B
95003	MA	C
95004	IS	B
95005	PH	B

关系模式SL (Sloc, Sno, Sdept)
Sdept—>Sloc, Sno—>Sdept,
所以 Sno—>Sloc

分解后的关系

NL		DL	
Sno	Sloc	Sdept	Sloc
95001	A	CS	A
95002	B	IS	B
95003	C	MA	C
95004	B	PH	B
95005	B		

NL和DL的自然连接结果

Sno	Sdept	Sloc
95001	CS	A
95002	IS	B
95002	PH	B
95003	MA	C
95004	IS	B
95004	PH	B
95005	IS	B
95005	PH	B

3、关系模式 $R(ABCD)$ ， F 是 R 上成立的函数依赖， $F=\{AB \rightarrow CD, A \rightarrow D\}$ 。

- 1) 试说明 R 不是 2NF 模式的理由；
- 2) 试把 R 分解成 2NF 模式集。

1) 2NF定义: 如果一个关系模式 $R \in 1NF$ ，并且每一非主属性都完全依赖于 R 的码，则 $R \in 2NF$ 。

码是 (A,B) ， D 部分依赖于码

2) $\{(ABC), (AD)\}$

4、设关系模式 $R(ABC)$ ， F 是 R 上成立的函数依赖， $F=\{C \rightarrow B, B \rightarrow A\}$ 。

1) 试说明 R 不是 3NF 模式集；

2) 试把 R 分解为 3NF 模式集。

3NF定义

如果一个关系模式 R 中不存在非主属性对码的传递依赖，则 $R \in 3NF$

1) 码是 C ， A 是非主属性 $C \xrightarrow{t} A$

2) $\{(C,B), (C,A)\}$ or $\{(C,B), (B,A)\}$

5、设有关系模式R（职工名，项目名，工资，部门号，部门经理），如果规定每个职工可以参加多个项目，每个项目都可以各领一份工资；每个项目只属于一个部门管理；每个部门只有一个部门经理。要求：

- 1) 写出关系模式R的函数依赖和主键；
- 2) R是2NF模式吗？若不是请说明理由，并把R分解到2NF模式集；
- 3) 把R分解到3NF模式集。

1) 主键：（职工号，项目名）

函数依赖：（职工号，项目名） \xrightarrow{f} 工资

项目 \xrightarrow{f} 部门号

部门号 \xrightarrow{f} 部门经理

（职工号，项目名） \xrightarrow{p} 部门号

项目 \xrightarrow{t} 部门经理

（职工号，项目名） \xrightarrow{p} 部门经理 ?

2) 不是2NF，因为存在非主属性对码的部分函数依赖 （职工号，项目名） \xrightarrow{p} 部门号

R1（职工号，项目名，工资） R2（项目名，部门号，部门经理）

3) R1（职工号，项目名，工资） R2（项目名，部门号） R3(部门号，部门经理)

6、现需要建立一个关于学生、班级、系和社团等信息的一个关系数据库系统，一个系有若干专业，每个专业每年只招一个班，每个班有若干名学生，一个系的学生住在同一宿舍区，每个学生可以参加若干个社团，每个社团有若干名学生。

需要考虑记录以下信息：

学生属性有：学号、姓名、出生年月、系名、班级号、宿舍区；

班级属性有：班级号、专业名、系名、人数、入学年份；

系的属性有：系名、系号、系办公地点、人数；

社团属性有：社团名、成立年份、地点、人数、学生参加社团的年份。

请写出你设计的关系模式，指出每个关系模式的候选键、外键，写出每个关系模式的函数依赖集。

e.g.

学生（学号、姓名、出生年月、班级号）

系-专业（专业名、系号）

系（系号、系名、系办公地点、人数、宿舍区）

班级（班级号、专业名、人数、入学年份）

社团（社团名、成立年份、地点、人数）

社团-学生（社团名、学号、学生参加社团的年份）

第五章 数据库保护

- 数据库安全性： 用户标识与鉴别， 存取权限控制， 使用视图， 审计， 数据加密
- 数据库的完整性：
 - 完整性约束条件分为(列级， 元组级， 关系级),(静态， 动态)
 - 静态关系级约束： 实体完整性约束， 参照完整性， 函数依赖， 统计
 - 完整性规则的形式化表示 (D, O, A, C, P)
 - 实体完整性： 定义主键： `CONSTRAINT pk_1 PRIMARY KEY (S#)`
 - 参照完整性： 定义外键：
`CONSTRAINT fk_1 FOREIGN KEY (C#) REFERENCES C(C#) {ON DELETE| UPDAE {RESTRICT | CASCADE| SET NULL}}`
外键： 关系模式R中的属性集是其他关系模式的主键， 那么该属性集对于关系模式R而言是外键
 - 自定义完整性： `SA int CONSTRAINT CHK1 CHECK (SA<100),`

- 数据库并发控制:

- 事物特性: 原子性, 一致性, 隔离性, 持久性
- 并发操作四大问题: 修改丢失, 污读, 不可重复读, 幻影读
- 衡量并发事务正确性的标准: 可串行性
- 数据库的封锁机制: 排他锁(独占锁):X(修改) ; 共享锁: S(查询) ; 修改锁

事务隔离级别		0 Read Uncommitted	1 Read Committed	2 Repeatable Read	3 Serializable
X锁	操作结束释放				
	事务结束释放	√	√	√	√
S锁	操作结束释放		√		
	事务结束释放			√	√
一致性 保证	防止丢失修改	√	√	√	√
	防止脏读		√	√	√
	可重复读			√	√
	防止幻影读				√

第五章

5、已知两个关系模式：

职工(职工号, 姓名, 年龄, 职务, 工资, 部门号)

部门(部门号, 名称, 经理名, 电话)

用 SQL 语言定义这两个关系模式，并实现以下完整性约束条件：

- 1) 定义每个关系模式的主键；
- 2) 定义关系中的参照完整性；
- 3) 定义每个职工的年龄不能超过 60 岁。

1) **constraint pk_1 primary key (职工号)**

(在**职工表**中定义)

constraint pk_2 primary key (部门号)

(在**部门表**中定义)

2) **constraint fk_1 foreign key (部门号) references 部门 (部门号)**

(在**职工表**中定义)

3) **constraint c1 check age between 0 and 60**

(在**职工表年龄**后面定义)

6、已知学生表 student、课程表 course 和学生选课表 s_c 的关系模式如下：

student (sno, sname, sex, age,dept,place) 即 (学号、姓名、性别、年龄,所属院系, 籍贯) ,

course (cno, cname, credit, pcno) 即 (课程号、课程名, 学分, 预修课程号) ,

s_c (sno, cno, grade) 即 (学号、课程号、成绩) ,

试定义以下完整性约束：

- 1) student 中学生年龄在 15~25 之间；
- 2) 在s_c 中插入元组时其sno 和cno 必须在关系student 和course 中出现；
- 3) 删除关系 course 中元组时，首先要把 s_c 中具有同样 cno 的元组删除；
- 4) 修改 student 中某个 sno 时，同时修改 s_c 中 sno 的值。

(1) constraint c1 check (age between 15 and 25)

(2) constraint f1 foreign key(sno) references student(sno)

constraint f2 foreign key(cno) references course(cno)

(3) constraint f3 foreign key(cno) references course(cno) on delete cascade;

(4) constraint f4 foreign key(sno) references student(sno) on update cascade;

第六章 数据库设计

1.简述数据库设计的基本步骤。

- 需求分析
- 概念结构设计： 设计局部视图 集成视图
- 逻辑结构设计： 设计逻辑结构 优化逻辑结构
- 数据库物理设计： 设计物理结构 评价物理结构
- 数据库实施： 数据库系统的物理实现 试验性运行
- 数据库运行维护

2、要建立一个公司数据库，公司有多个部门，每个部门有多个职员，一个职员仅隶属于一个部门，且一个员工在一个工程中工作，一个工程有多个职员参与，有多个供应商为各个工程供应不同的设备。

部门属性：部门名，电话

职员属性：职员号、姓名、性别

工程属性：工程名、地点

设备属性：设备号、设备名、产地

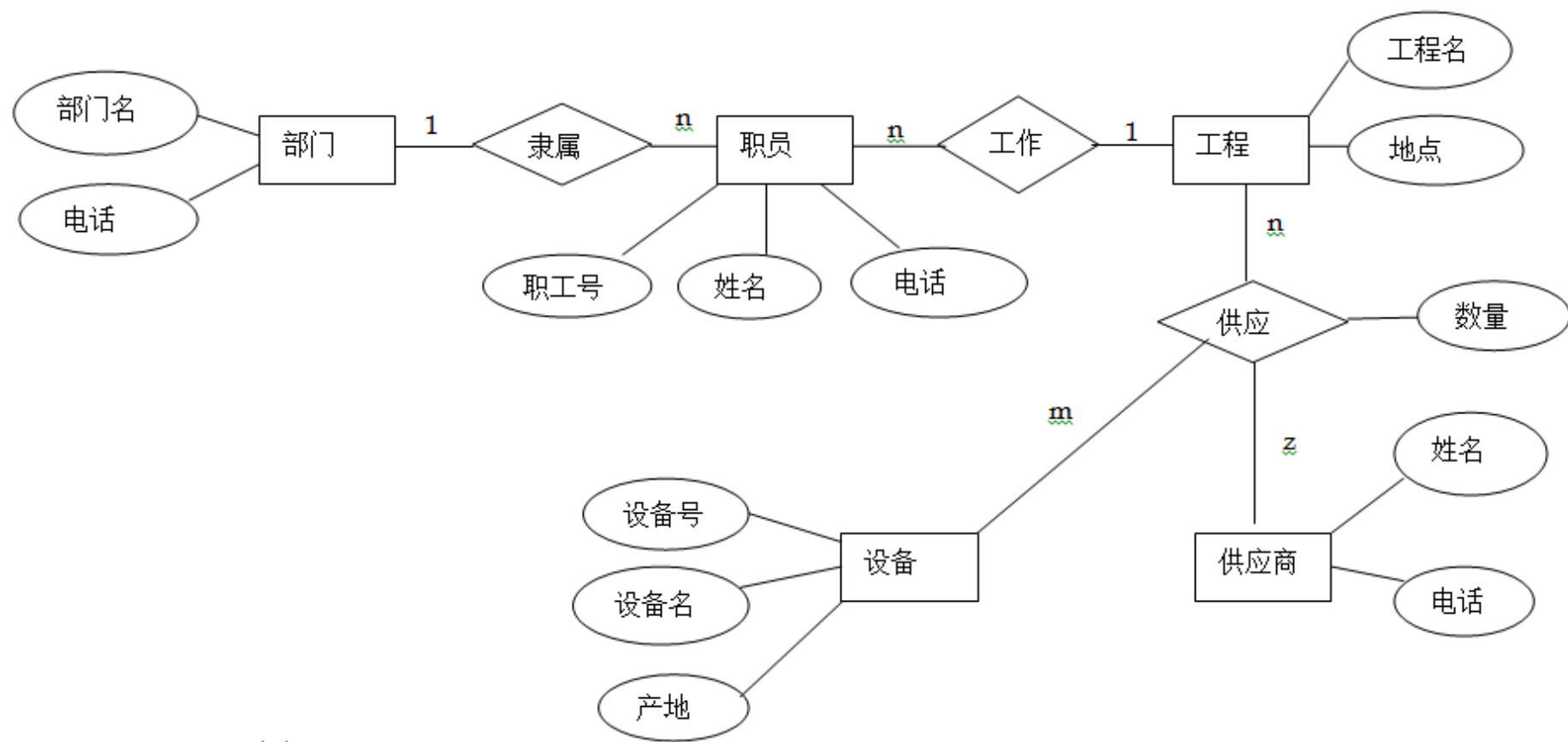
供应商属性：姓名、电话

请完成以下处理：

1) 设计满足上述要求的 E-R 图

2) 将 E-R 图转换为关系模式，用下划线给出其主键

(1)



(2)

部门 (部门名, 电话)

职员 (职员号, 姓名, 电话, 部门名, 工程名)

工程 (工程名, 地点)

供应商 (姓名, 电话)

设备 (设备号, 设备名, 产地)

供应 (供应商姓名, 工程名, 设备号, 数量)

3、有运动队和运动会两个方面的实体：

运动队方面有：

运动队：队名、教练员名、队员姓名

队员：队名、队员姓名、性别、项目名

其中，一个运动队有多个队员，一个队员仅属于一个运动队，一个队仅一个教练。

运动会方面有：

运动队：队编号、队名、教练姓名

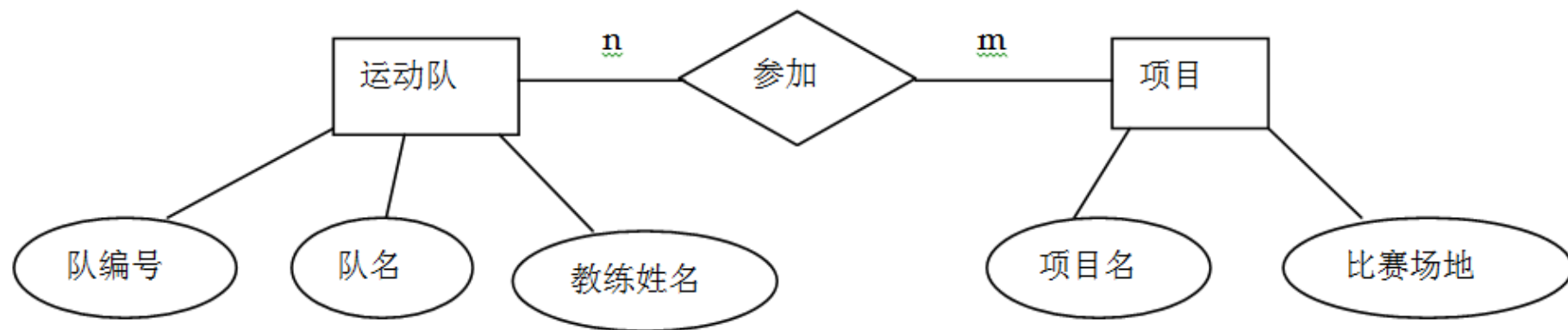
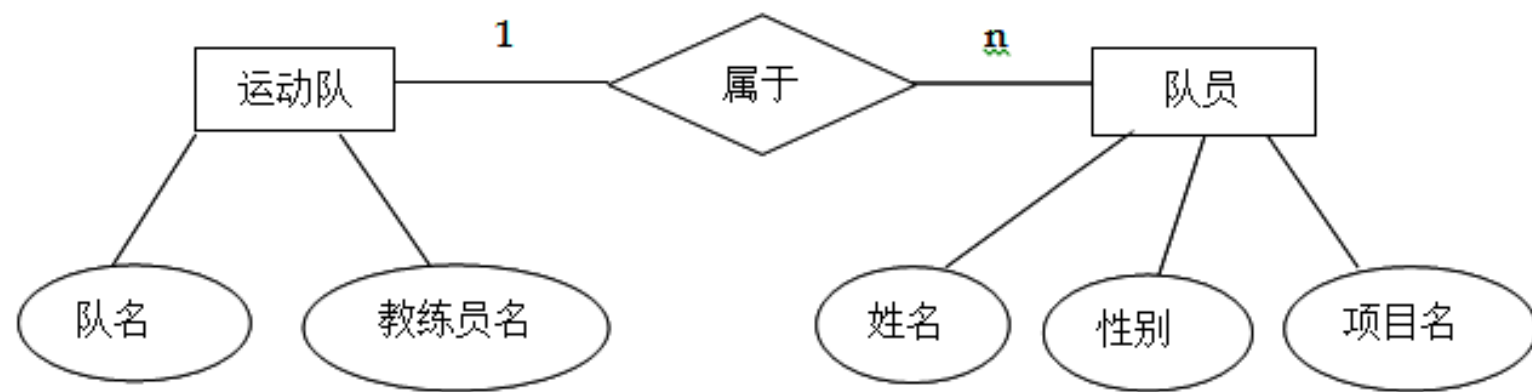
项目：项目名、参加运动队的编号、队员姓名、性别、比赛场地。

其中，一个项目有多个运动队参加，一个运动员可以参加多个项目，一个项目一个比赛场地。

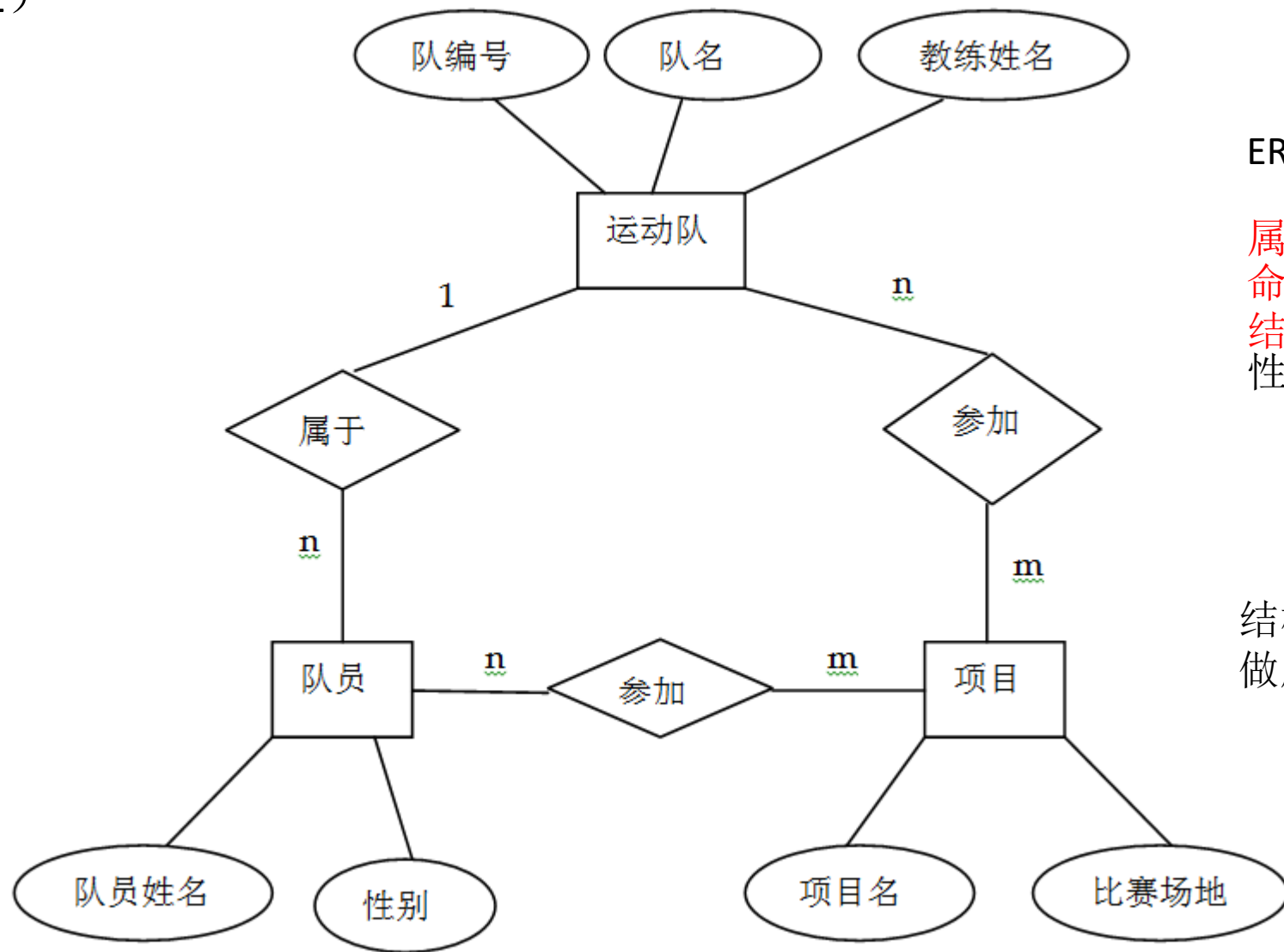
1) 分别设计运动队和运动会两个局部 E-R 图；

2) 将它们合并为一个全局 E-R 图(注意合并时的冲突)

(1)



(2)



ER图合并时可能存在的问题:

属性冲突: 属性域冲突、属性单位冲突

命名冲突: 同名异义, 异名同义

结构冲突: 同一对象抽象不同, 同一实体属性不同, 联系类型不同

结构冲突: 项目在两个局部E-R图中, 一个做属性, 一个做实体, 合并统一为实体。

祝大家考试顺利
521快乐~

