

---

## 2019 数据库期末复习内容

### 1. 选择题

- (1) (数据库) 是存储在计算机内结构化的数据的集合
- (2) 下述关于数据系统正确的描述是 (数据库系统减少了数据冗余)。
- (3) 数据库 (DB)、与数据库系统 (DBS) 和数据库管理系统 (DBMS) 之间的关系是 ( DBS 包含 DB 和 DBMS )。
- (4) 数据库管理系统的工作不包括 (为定义的确据库提供操作系统)。
- (5) 用于对现实世界进行描述的工具是 (数据模型)。
- (6) 在数据库的三级模式结构中, 描述数据库中全局逻辑结构和特征的是 (模式 )。
- (7) E-R 图用来建立数据库的 (概念模型 )。
- (8) 下列模型中不能直接描述实体间多对多的联系的是 (关系模型 )。
- (9) 数据模型的三个要素是 (数据结构、数据操作、完整性约束)
- (10) 数据库的设计人员和用户之间沟通信息的桥梁是 (实体联系图)。
- (11) 数据的独立性分为 (逻辑独立性和物理独立性)。
- (12) 数据库通常是根据 (数据模型) 来分类的。
- (13) 关系数据库系统能实现专门的关系运算, 包括 (选择、投影、连接、除)。
- (14) 在关系模型中, 下述关于关键字的说法正确的是 (可以由一个或多个其值能唯一标识该关系模式中任何元组的属性组成 )。
- (15) 被称为关系的两个不变性, 由关系数据库系统自动支持的是 (实体完整性和参照完整性)。
- (16) 在关系中不能出现相同元组的约束是通过 (主码) 来实现。
- (17) 下列数据模型中, 数据独立性最高的是 (关系模型)。
- (18) 规范化理论是关系数据库进行逻辑设计的理论依据。根据这个理论, 关系数据库中的关系必须满足: 每一属性都是 (不可分解的 )。
- (19) 关系数据库规范化是为了解决关系数据库中 (插入异常、删除异常和数据冗余) 问题而引入的。
- (20) 在关系数据库中, 任何二元关系模式的最高范式必定是 ( BCNF )
- (21) 若属性 X 函数依赖于属性 Y 时, 则属性 X 与属性 Y 之间的关系是 (多对一 )。
- (22) 设有关系模式: 部门 (部门代码, 部门名称, 部门成员, 部门经理) 由于其中 ( 部门成员 ) 属性存在, 使该关系不属于第一范式。

(23) 若关系模式 R 中的属性都是主属性, 则 R 最高属于 ( 3NF ) 。

(24) 若要彻底消除插入异常和删除异常, 关系模式最低要属于 (BCNF)。

(25) 关系模式分解的无损连接和保持函数依赖两个特性之间的联系是 (没有必然联系)。

(26) 关系模式中数据依赖的存在, 可能会导致成立蕴涵后者数据插入异常, 这是指 (该插入的数据未被插入)。

(27) 设要对关系模式 R (学号, 姓名, 课程代码, 课程名, 课程成绩) 进行分解, 将其规范化到 3NF, 下列 (B) 是正确的规范化结果。(学号, 姓名) R2 (课程代码, 课程名) R3 (学号, 课程代码, 课程成绩)

(28) 有如下关系 R (是 BCNF)。

零件号	单价
P1	25
P2	8
P3	25
P4	9

(29) 对用户访问数据库的权限加以限定是为了保护数据库的 (安全性) 。

(30) 下面 (可移植性) 不是数据库系统必须提供的数据库控制功能。

(31) 数据库系统中, 对存取权限定义为 (授权) 。

(32) 在数据库的安全性控制中, 为了保证用户只能存取他有权存取的数据, 在授权的定义中, 数据对象的 (范围越小), 授权子系统就越灵活。

(33) 数据库安全性遭到破坏的情况属于 (非法用户读取数据库中的数据) 。

(34) 数据库的 (完整性) 是指数据的正确性和相容性。

(35) 数据完整性保护中的约束条件是指 (值的约束和结构的约束)。

(36) 实体完整性和参照完整性属于 (静态关系约束) 。

(37) 数据类型、取值范围、空值约束属于 (静态列级约束)。

(38) “断言” 是数据库系统采用的 (完整性约束)。

(39) 脏数据的读出是 (并发性) 遭到破坏的情况。

(40) SQL 中 ROLLBACK 语句的主要作用是 (事务卷回)。

(41) SQL 中提供的 (GRANT) 语句用于实现数据存取的安全控制 。

(42) SQL 中 COMMIT 语句的主要作用是 (提交事务)。

(43) 解决并发操作带来的数据不一致性问题时普遍采用 (封锁)。

(44) 不允许其它事务对这个锁定目标再加任何类型的锁是 (排他锁)。

填空题:

1. 三级模式之间的两层映像保证了数据库系统中的数据能够具有较高的 逻辑独立性、物理独立性。
2. 关系的完整性约束条件包括三大类: 实体完整性、参照完整性、用户定义的完整性。
3. 关系代数运算中, 专门的关系运算有 投影、选择、连接、除。
4. 数据库是长期存储在计算机内、有组织的、可共享的数据集合。
5. 数据模型分成两个不同的层次。概念模型从用户的观点对现实世界的信息建模, 主要用于数据库设计。数据模型从计算机系统的观点, 对信息建模。
6. 常用的数据模型有层次模型、网状模型和关系模型三种。

## 2. 判断题

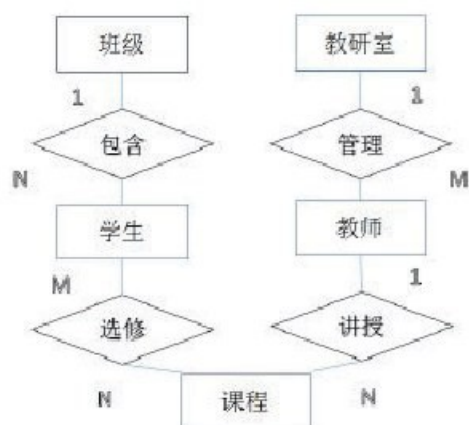
- (1) 在文件系统阶段, 数据的独立性差。 ✓
- (2) 对于数据库系统, 负责定义数据库内存, 决定存储结构和存储策略及安全授权等工作的是应用程序员。✗
- (1) 假设一个仓库可存放多种商品, 同种商品只能存放在一个仓库中, 则从仓库到商品的联系是多对一的联系。  
✗
- (2) 在数据库系统的模式结构中, 外模式有且仅有一个。✗
- (3) 数据独立性是指应用程序与数据之间相互独立, 不受影响。✓
- (1) 一个关系模型的逻辑结构是一张二维表, 表中的一行即为一个属性, 表中的一列即为一个元组。✗
- (2) 当数据库的存储结构改变了, 由数据库管理员对模式/内模式映像作相应改变, 可以使模式保持不变, 从而保证了程序与数据的物理独立性。( ✓ )
- (1) 相对于非关系模型, 关系模型的缺点之一是存取路径对用户透明, 需进行查询优化。 ✓
- (2) 关系模型的三个组成部分, 不包括完整性约束。✗
1. 数据库物理存储方式的描述称为内模式。 ✓
  2. 在数据库三级模式间引入二级映像的主要作用是提高数据与程序的安全性。✗
  3. 关系模型是数据模型, 而网状模型不是数据模型。✗
  4. 同一关系模型的任意两个元组值不能完全相同。✓
  5. 在关系中能唯一标识元组的属性集的是主码。✓
- (1) 属于BC 范式的关系模式中已经完全消除了插入异常和删除异常。✓
- (2) 在关系模式  $R(A, B, C, D)$  中, 有函数依赖集  $F=\{B \rightarrow C, C \rightarrow D, D \rightarrow A\}$ , 则  $R$  能达到 3NF ✗

## 应用题

设有如下教学环境: 一个班级有多名学生, 一个学生只属于一个班级。一个学生可以选修若干门课程, 一门课程



只有一个教师主讲。每位教师属于一个教研室，一个教研室有若干位教师。请画出 E-R 图。



## 关系代数练习

给定如下关系：

$S(S\#, SN, SA, SD)$ ， $S\#$ 表示学号， $SN$ 表示姓名， $SA$ 表示年龄， $SD$ 表示所在系。

$C(C\#, CN, PC\#)$ ， $C\#$ 表示课程号， $CN$ 表示课程名， $PC\#$ 表示先修课程号。

$SC(S\#, C\#, G)$ ， $S\#$ 、 $C\#$ 含义同上， $G$ 表示成绩。

请用关系代数完成如下查询操作：

- (1) 查询选修课程号为 C1、成绩为 A 的所有学生姓名。
- (2) 查询年龄为 23 岁的学生所选修的课程名。
- (3) 查询选修“数据库原理”课程的学生姓名。
- (4) 查询选修所有课程的同学姓名。
- (5) 查询选修了先修课程号 PC 为 6 的课程学生学号。

解：(1)  $\Pi_{SN} (S \bowtie \Pi_{S\#} (\sigma_{C\# = 'C1' \wedge G = 'A'} (SC)))$

(2)  $\Pi_{CN} (C \bowtie \Pi_{C\#} (SC \bowtie \Pi_{S\#} (\sigma_{SA=23} (S))))$

(3)  $\Pi_{SN} (S \bowtie \Pi_{S\#} (SC \bowtie \Pi_{C\#} (\sigma_{CN = '数据库原理'} (C))))$

(4)  $\Pi_{SN} (S \bowtie \Pi_{S\#} ((SC) \div (\Pi_{C\#}(C)))$

(5)  $\Pi_{S\#} (SC \bowtie \Pi_{C\#} (\sigma_{PC = '6'} (C)))$

=====

给定如下关系：

S (S#, SN, SA, SD), S#表示学号, SN 表示姓名, SA 表示年龄, SD 表示所在系。

C (C#, CN, PC#), C#表示课程号, CN 表示课程名, PC#表示先修课程号。

SC (S#, C#, G), S#、C#含义同上, G 表示成绩。

请用关系代数完成如下查询操作：

- (1) 查询所有小于 20 岁学生的学号、姓名。
- (2) 查询计算机系的学生所选修的课程名。
- (3) 查询选修 “数据库原理” 课程的学生姓名。
- (4) 查询选修所有课程的同学姓名。
- (5) 查询选修了先修课程号 PC 为 6 的课程学生学号。

(1)  $\Pi_{S\#, SN} (\sigma_{SA=20} (S))$

(2)  $\Pi_{CN} (C? \Pi_{C\#} (SC? \Pi_{S\#} (\sigma_{SD='计算机'} (S))))$

或者:  $\Pi_{CN} (\sigma_{SD='计算机'} (C? SC ? S))$

(3)  $\Pi_{SN} (S? \Pi_{S\#} (SC ? \Pi_{C\#} (\sigma_{CN='数据库原理'} (C))))$

或者:  $\Pi_{SN} (\sigma_{CN='数据库原理'} (C? SC ? S))$

(4)  $\Pi_{SN} (S? \Pi_{S\#} (SC \div (\Pi_{C\#?} (C))))$

或者:  $\Pi_{SN} (S? \Pi_{S\#} (SC \div C))$

(5)  $\Pi_{S\#} (SC ? \Pi_{C\#} (\sigma_{PC='6'} (C)))$

或者:  $\Pi_{S\#} (\sigma_{PC='6'} (C? SC))$

设职工社团数据库有三个基本表：

职工 (职工号, 姓名, 年龄, 性别, 工资, 住址);

社会团体 (编号, 名称, 负责人, 活动地点);

参加 (职工号, 编号, 参加日期)。

用关系代数表达式实现下列 1—5 小题：

1. 查询年龄大于 20 岁的职工姓名和性别。
2. 查询参加社会团体编号为 101 的所有职工的姓名和年龄。
3. 查询负责人为刘刚，活动地点在北二区小礼堂的社会团体的名称。
4. 查询在西三环居住、工资在 1000 元以上职工参加社团的编号。
5. 查询参加健美社团、活动时间在周六的职工的姓名和职工号。
  1.  $\Pi$  姓名, 性别 ( $\sigma$  年龄  $> 20$  (职工))
  2.  $\Pi$  姓名, 年龄 ( $\sigma$  编号 = 101 (职工  $\bowtie$  参加))
  3.  $\Pi$  名称 ( $\sigma$  负责人 = '刘刚'  $\wedge$  活动地点 = '北二区小礼堂' (社会团体))
  4.  $\Pi$  编号 ( $\sigma$  住址 = '西三环'  $\wedge$  工资  $> 1000$  (职工  $\bowtie$  参加))
  5.  $\Pi$  姓名, 职工号 ( $\sigma$  名称 = '健美'  $\wedge$  参加时间 = '周六' (职工  $\bowtie$  参加  $\bowtie$  社会团体))

用 SQL 语句实现下列 6—12 小题：

6. 定义职工表，职工号不能为空，其值唯一。

答案. `CREAT TABLE 职工 (职工号 CHAR (10) NOT NULL UNIQUE,  
                                姓名 CHAR (8) NOT NULL,  
                                年龄 SMALLINT,  
                                性别 CHAR (2));`

7. 建立视图：参加人情况（职工号，姓名，社团编号，社团名称，参加日期）

答案. `CREAT VIEW 参加人情况 (职工号, 姓名, 社团编号, 社团名称, 参加日期)  
AS SELECT 参加. 职工号, 姓名, 社会团体编号, 名称, 参加日期  
FROM 职工, 社会团体, 参加  
WHERE 职工. 职工号 = 参加. 职工号 AND 参加. 编号 = 社会团体. 编号;`

8. 查找参加唱歌队或篮球队的职工号和姓名。

答案. `SELECT 职工号, 姓名  
FROM 职工, 社会团体, 参加  
WHERE 职工. 职工号 = 参加. 职工号 AND 参加. 编号 = 社会团体. 编号  
AND 社会团体. 名称 IN ('唱歌队', '篮球队');`

9. 求每个社会团体的参加人数。

答案: `SELECT 编号, COUNT (职工号)  
FROM 参加  
GROUP BY 编号;  
或者:  
SELECT 社会团体. 名称, COUNT (职工号)`



---

```
FROM 参加, 社会团体
WHERE 社会团体. 编号=参加. 编号
GROUP BY 名称;
```

10. 查询名字中第二个字为“明”字的职工的姓名和职工号。

答案: SELECT 职工号, 姓名

```
FROM 职工
WHERE 姓名 LIKE ' _ 明% ' ;
```

11. 将职工号=6887 的年龄改为 45 岁。

答案: UPDATE 职工

```
SET 年龄=45
WHERE 职工号='6887 ' ;
```

12. 把对社会团体和参加两个表的数据查看、插入和删除数据的权力赋给用户李平, 并允许他再将此权力授予其他用户。

答案: . GRANT select,insert ,delete ON TABLE 社会团体, 参加 TO 李平

```
WITH GRANT OPTION ;
```

## 现有关系数据库如下

学生(学号, 姓名, 性别, 专业, 奖学金)

课程(课程号, 名称, 学分)

学习(学号, 课程号, 分数)

用关系代数表达式实现下列 1—2 小题:

1. 检索“英语”专业学生的学号、姓名、性别。

2. 检索课程号为“10031”, 课程成绩高于 90 分的所有学生的学号。

1.  $\Pi$ 学号, 姓名, 性别( $\sigma$ 专业='英语'(学生))

2.  $\Pi$ 学号 ( $\sigma$ 分数>90 $\wedge$ 课程号='10031'(学习))

用 SQL 语言实现下列 3—10 小题:

3. 检索姓王学生的姓名、专业。

答案: SELECT 姓名, 专业 FROM 学生 WHERE 姓名 LIKE “王%” ;

---

4. 检索奖学金在 200 元以下的学生姓名和专业。

答案: SELECT 姓名, 专业 FROM 学生 WHERE 奖学金<200;

5. 检索选修课程号为“C135”课程的学生信息, 包括学号, 姓名和专业

答案: SELECT 学号, 姓名, 专业 FROM 学生, 学习  
WHERE 课程号='C135' and 学生.学号=学习.学号;

6. 检索学习‘C1’号课程的学生最高分。

答案: SELECT MAX(分数) FROM 学习 WHERE 课程号='C1' ;

7. 向学生表中插入一个记录(‘980810’, ‘王东’, ‘男’, ‘计算机’, 150)

答案: .INSERT INTO 学生  
VALUES (‘980810’, ‘王东’, ‘男’, ‘计算机’, 150);

8. 定义“英语”专业学生所学课程的信息视图 AAA, 包括学号、姓名、课程号和分数

答案: CREATE VIEW AAA(学号, 姓名, 课程号, 分数)  
AS SELECT 学号, 姓名, 课程号, 分数  
FROM 学生, 学习  
WHERE 学生.学号 =学习.学号 AND 专业='英语' ;

9. 检索所有有成绩的学生学号和课程号。

答案: SELECT 学号, 课程号 FROM 学习 WHERE 成绩 IS NULL;

10. 检索选修 C56 课程的学生学号和成绩, 检索结果按分数的降序排列。

答案: .SELECT 学号, 成绩  
FROM 学习  
WHERE 课程号= ‘C56’  
ORDER BY 成绩 DESC;

## 有如下三个关系表

st(sno, sn, sa, ss): 学生表格, sno: 学号 sn: 姓名 sa: 年龄 ss: 性别



---

sc(sno,cno,score): 学生选择课程表格, sno: 学号 cno: 课程号 score: 成绩

c(cno,cn,ct): 课程表格, cno: 课程号 cn: 课程名 ct: 课程学时

1. 用关系代数写出下面查询的表达式 (每题 4 分)

$$\Pi_{SNO, SN, SA}(\sigma_{SS='女'}(ST))$$

(1) 查找所有女生的学号、姓名和年龄。

(2) 查找选修了“数据库技术” 的学生学号和成绩。

$$\Pi_{SNO, SCORE}(\sigma_{CN='数据库技术'}(SC \bowtie C))$$

(3) 查找选修了“软件工程” 的男生姓名和年龄。

$$\Pi_{SN, SA}(\sigma_{CN='软件工程' \wedge SS='男'}(ST \bowtie SC \bowtie C))$$

2. 请按以下题目的要求写出相应的 SQL 语句 (每题 4 分)

(1) 定义学生关系表, 要求学号属性非空, 取值唯一;

```
create table st(sno char(10) not null unique,  
sn char(20) ,  
sa int,  
ss char(2));
```

(2) 查询年龄大于 26 岁的男生学号和姓名;

```
select sno, sn from st where sa>26 and ss='男' ;
```

(3) 查询全体学生的姓名和出生年份;

```
select sn, 2018-sa from st;
```

(4) 查询年龄 18 到 30 岁之间的学生姓名和年龄;

```
select sn, sa from st where sa between 18 and 30;
```

(5) 查询所有姓孙的学生学号、姓名;

```
select sno,sn from st where sn like '孙%';
```

(6) 查询男女学生的平均年龄;

```
select ss , avg(sa) from st group by ss;
```

(7) 查询学习 5 号课程的最高分和最低分。

```
select max(score),min(score)
```

```
from sc;
```

### 综合题:

某个学校有若干个系, 每个系有若干个学生, 开设若干个课程, 每门课程有若干学生选修, 某一门课程可以为不同的系开设。用 E-R 图画该学校的概念模型。

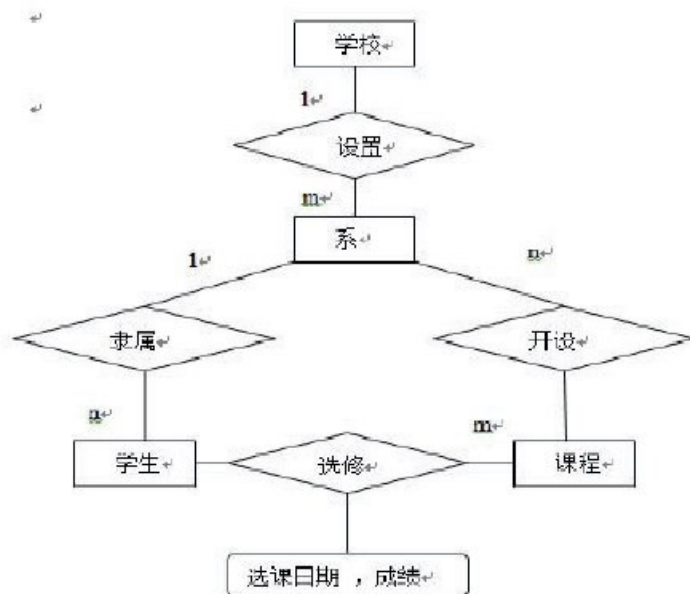
描述学生的属性有: 学号、姓名、出生日期、系号、班号;

描述系的属性有: 系名、系号、系办公室地点、人数;

描述学校的属性有: 学校名、成立年份、地点;

描述课程的属性有: 课程号、课程名、学分。

(1) 设计学校的 E-R 图。



(2) 将该 E-R 图转换为关系模式。

关系模型如下。

学生 (学号、姓名、出生日期、系号、班号)。

---

系（系名、系号、系办公室地点、人数）。

学校（学校名、成立年份、地点）

课程（课程号、课程名、学分）

（3）指出转换结果中每个关系模式的候选码。

学生关系的主码：用户名

系的主码：系号

学校主码：学校名

课程主码：课程号

## 有三个关系如下

（1）学生关系 Student，包括学号 Sno、姓名 Sname、年龄 Sage、性别 Ssex；

（2）课程关系 Course，包括课程号 Cno、课程名 Cname、任课教师 Cteacher；

（3）学生选课关系 SC，包括 Sno、Cno 和成绩 Grade。

1. 定义学生关系 Student，学号不能为空，其值唯一，姓名值也唯一，学号为主码；

```
CREATE TABLE Student
```

```
(Sno CHAR (10) NOT NULL UNIQUE PRIMARY KEY,
```

```
Sname CHAR(30) UNIQUE,
```

```
Sage INT ,
```

```
Ssex CHAR(2)) ;
```

2. 查询 1 号课程的平均成绩；

```
SELECT AVG (Grade) FROM SC WHERE Cno= '1' ;
```

3. 查询学号“95001”的学生选修全部课程号和成绩；



---

```
SELECT Cno, Grade FROM SC WHERE Sno= '95001' ;
```

**4. 查询年龄在19-20岁之间的男生和女生的数量;**

```
SELECT Ssex, COUNT (Sno)
FROM Student
WHERE Sage>=19 and Sage<=20
GROUP BY Ssex;
```

或者:WHERE Sage BETWEEN 19 and 20

**有三个关系如下:**

- (1) 学生关系 Student, 包括学号 Sno、姓名 Sname、年龄 Sage、性别 Ssex;
- (2) 课程关系 Course, 包括课程号 Cno、课程名 Cname、任课教师 Cteacher;
- (3) 学生选课关系 SC, 包括 Sno、Cno 和成绩 Grade。

**1. 查询 “软件工程” 课程的平均成绩;**

```
SELECT AVG(Grade)
FROM Course, SC
WHERE Cname= '软件工程'
AND =;
```

或者: SELECT AVG(Grade)

```
FROM SC WHERE Cno IN (SELECT Cno FROM Course WHERE Cname= '软件工程' )
```

**2. 查询名字中第2个字为"阳"字的学生的姓名和学号。**

```
SELECT Sname, Sno
FROM Student
WHERE Sname LIKE ' _阳%';
```

**3. 查询刘阳同学所选修的课程号和课程名**

```
SELECT , Cname FROM Course WHERE Cno IN
(
    (SELECT Cno FROM SC WHERE Sno IN
    (SELECT SnoFROM Student WHERE Sname= '刘阳' ));
```

或者: SELECT , Sname FROM Student, Course, SC

```
WHERE Sname= '刘阳' AND = AND =;
```

**4 查询学生数据库课的成绩, 输出学生姓名和成绩, 按成绩排序 (降序) 。**

```
SELECT Sname,Grade FROM Student,Course,SC WHERE = AND = AND ='数据库' ORDER BY Grade DESC;
```

**5. 检索没有成绩的学生姓名和课程号。**

---

```
Select Sname,Cno From Student,SC Where Grade is NULL AND =
```

## 有三个关系如下

- (1) 学生关系 Student, 包括学号 Sno、姓名 Sname、年龄 Sage、性别 Ssex;
- (2) 课程关系 Course, 包括课程号 Cno、课程名 Cname、任课教师 Cteacher;
- (3) 学生选课关系 SC, 包括 Sno、Cno 和成绩 Grade。

使用 SQL 语句实现:

1. 为学生表建立一个聚簇索引 STUNO, 按学号升序和年龄降序。

```
CREAT CLUSTER INDEX Stuon ON Student(SNO ASC,SAGE DESC);
```

2. 查询王寒同学没有选修的课程的课程号

```
SELECT Cno
FROM Course
WHERE Cno NOT IN
(
  SELECE Cno
  FROM SC, Student
  WHERE =
  AND Sname=" 王寒" );
```

3. 创建一个成绩高于 90 分的学生视图 S1 , 包括学生学号、出生年份、所选课程号、成绩。

```
CREAT VIEW S1 (Sno,Sbirth,Cno,Grade )
AS SELECT Sno,2018-Sage,Cno, Grade
FROM Student,SC WHERE = AND Grade>90
```

4. 向课程表中增加属性任课教师(Tname), 类型是字符型

```
ALTER TABLE Course Add Tname char(20)
```

5. 将课程号为 '003' 课程成绩增加 10 分

```
UPDATE SC SET Grade= Grade+10
WHERE Cno= '003'
```

有三个关系如下

- 
- (1) 学生关系 Student，包括学号 Sno、姓名 Sname、年龄 Sage、性别 Ssex;
  - (2) 课程关系 Course，包括课程号 Cno、课程名 Cname、任课教师 Cteacher;
  - (3) 学生选课关系 SC，包括 Sno、Cno 和成绩 Grade。

下面使用关系代数表达式实现：

1. 查询课程名为数据库原理，授课教师为刘洋的课程号。

$$\Pi_{cno} (\sigma_{cname='数据库原理' \wedge Cteacher='刘洋'} (Course))$$

2. 查询既选修了课程 1 又选修了课程 3 的学生学号,姓名。

$$\Pi_{Sno, Sname} (\sigma_{cno='1'} (Student \bowtie SC)) \cap \Pi_{Sno, Sname} (\sigma_{cno='2'} (Student \bowtie SC))$$

使用 SQL 语句实现：

3. 将数据库原理课程的成绩提高 10 分。

```
UPDATE SC SET GRADE=GRADE+10
where '数据库原理' = (select Cname from Course where =;
```

4. 查询至少 2 门课在 80 分以上的学生学号及课程数

```
SELECT Sno, COUNT(*)
FROM SC
WHERE Grade>=80
GROUP BY Sno
HAVING COUNT(*)>=2;
```

5. 查询“C 语言\_课程设计”的课程号

```
SELECT Cno
FROM Course
WHERE Cname Like "C 语言\_程序设计" ESCAPE '\';
```

有四个关系如下：



- 
- (1) S(sno, sname, city);
- (2) P(pno, pname, color, weight);
- (3) J(jno, jname, city) ;
- (4) SPJ(sno, pno, jno, qty) ; 其含义见教材 P64。

下面使用关系代数表达式实现:

1. 查询供应红色零件的供应商号。

$$\Pi_{Sno} \left( \sigma_{color = \text{“红”}} (S \bowtie SPJ \bowtie P) \right)$$

2. 查询北京的供应商供应工程 j1 零件的供应商号及数量。

$$\Pi_{sno, qty} \left( \sigma_{city = \text{“北京”} \wedge jno = \text{“j1”}} (S \bowtie SPJ \bowtie P) \right)$$

使用 SQL 语句实现:

- 3、建立上述的供应情况表 SPJ。

```

Creat table SPJ
(
sno char(5) not null unique,
pno char(5) not null unique,
jno char(5) not null unique,
qty int );

```

- 4、查询为上海的工程提供零件的供应商号。

```

SELECT  distict sno FROM    SPJ
where jno in (select  jno from J where city= “上海” );
或者:SELECT  distict sno
FROM SPJ, J
where city= “上海” and =;

```

5. 创建北京的工程号及名称视图 J-BJ 。

```

CREATE VIEW J-BJ
AS
SELECT Jno, Jname
FROM J
WHERE city= ‘北京’ ;

```

设有如下所示的关系 R。请回答下列问题

职工号	职工名	年龄	性别	单位号	单位名
E1	赵	20	F	D3	CCC
E2	钱	25	M	D1	AAA
E3	孙	38	M	D3	CCC
E4	李	26	F	D3	CCC

(1) R 中存在那些函数依赖 (2 分)

职工号→职工名, 职工号→单位号, 职工号→单位名, 单位号→单位名

(2) 关系模式 R 的候选码是什么 (1 分)

关系模式 R 的候选码是职工号和职工名。

(3) 试问关系模式 R 最高已经达到第几范式为什么 (1 分)

关系模式 R 是 2NF, 因为在关系模式中, 存在非主属性对候选码的传递函数依赖: 职工号→单位名。

(4) 如果 R 不属于 3NF, 请将 R 分解为 3NF。并指出分解后各关系的主码。(2 分)

规范的关系模式为 R1 和 R2。

R1 (职工号, 职工名, 年龄, 性别, 单位号), R1 的主码为职工号。

R2 (单位号, 单位名), R2 的主码是单位号。

1. 某大学科研处为管理教师发表论文情况建立数据库管理系统, 需要下述信息。

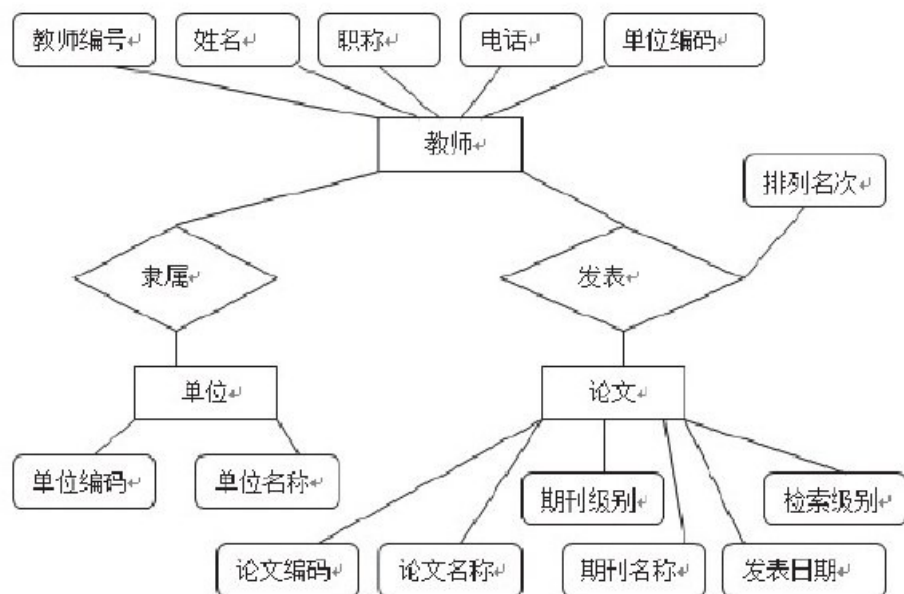
教师: 教师编号, 姓名, 职称, 电话, 单位编码。

单位: 单位编码, 单位名称。

论文: 论文编码, 论文名称, 期刊名称, 发表日期, 期刊级别, 检索类别。

其中, 一个单位有多名教师, 一个教师只属于一个单位。一个教师可以发表多篇论文, 一篇论文有多个作者。此外, 需要记录作者在论文中的名次。

(1) 设计该数据库管理系统的 E-R 图。



(2) 将该 E-R 图转换为关系模型结构。

教师 (教师编号, 姓名, 职称, 电话, 单位编码)

单位 (单位编码, 单位名称)

论文 (论文编码, 论文名称, 期刊名称, 发表日期, 期刊级别, 检索类别)

发表 (论文编号, 教师编号, 排列名次)

(3) 指出转换结果中每个关系模式的候选码。

教师关系的主码: 教师编号

单位关系的主码: 单位编码

论文关系的主码: 论文编号

发表关系的主码: 论文编号, 教师编号



R:				S:		
A	B	C		A	B	C
1	2	3		3	4	5
3	4	5		1	3	4
1	7	8		2	7	6

R	S	$RB<SB$	⊗		
1	2	3	3	4	5
1	2	3	1	3	4
1	2	3	2	7	6
	4	5	2	7	6

R-S				$R \cap S$				$R \cup S$				$\sigma_{B < '5'}(R)$				$\pi_{3, 2}(S)$	
A	B	C		A	B	C		A	B	C		A	B	C		C	B
1	2	3		3	4	5		1	2	3		1	2	3		5	4
1	7	8						3	4	5		3	4	5		4	3
								1	7	8						6	7

								1	3	4						
								2	7	6						

R ⋈ S		
A	B	C
3	4	5

R × S					
1	2	3	3	4	5
1	2	3	1	3	4
1	2	3	2	7	6
3	4	5	3	4	5
3	4	5	1	3	4
3	4	5	2	7	6
1	7	8	3	4	5
1	7	8	1	3	4
1	7	8	2	7	6

关系运算如下：

R		
A	B	C
1	2	3
4	5	6

7	8	9
---	---	---

S		
A	B	C
1	2	3
3	4	6
5	6	9

W	
D	E
7	5
4	8

R ∪ S		
A	B	C
1	2	3
4	5	6
7	8	9
3	4	6
5	6	9

R ∩ S		
A	B	C
1	2	3

R − S		
A	B	C
4	5	6
7	8	9

R × W				
A	B	C	D	E
1	2	3	7	5
1	2	3	4	8
4	5	6	7	5
4	5	6	4	8
7	8	9	7	5
7	8	9	4	8

Π <sub>C, A</sub> (R)	
C	A
3	1
6	3
9	5