

2011~2012 学年第 1 学期期末考试试卷

《数据库原理》(A 卷 共 7 页)

(考试时间: 2012 年 1 月 6 日)

题号	一	二	三	四	五	六	成绩	核分人签字
得分								

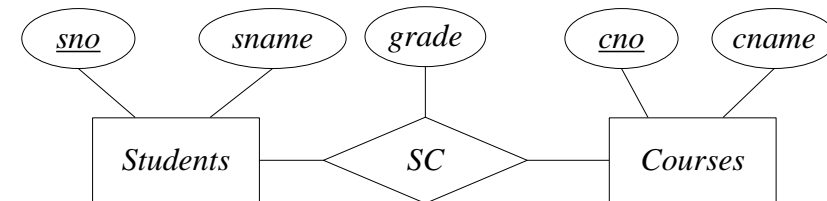
一、选择题 (20 分)

- What does the abbreviation DBMS stand for in the database technology? (A)
A. Database Management System B. Data Manipulation System
C. Database and Media System D. Data Base Manager System
- Which one of the following data models was proposed by Codd in his 1970's paper? (C)
A. hierarchical data model B. network data model
C. relational data model D. object-oriented data model
- What one of the following options is *not* a part of a data model? (B)
A. structure of data B. storage of data
C. operations on the data D. constraints on the data
- Assume that three relations $R(A, B)$, $S(A, B)$ and $T(A, B)$ have the same schema. Which one of the following relational algebra equations does *not* hold? (D)
A. $R \cap S = R - (R - S)$ B. $R \bowtie_{R.A=S.B} S = \sigma_{R.A=S.B}(R \times S)$
C. $(R \bowtie S) \bowtie T = R \bowtie (S \bowtie T)$ D. $\pi_A(R \cap S) = \pi_A(S \cap T)$
- Given the following two schemas
Movies (title, year, length, genre, studioName, producerC#)
MovieExec (name, address, cert#, netWorth)

Which of the following relational algebra expressions can express the constraints “the producerC# of each Movies tuple must also appear in the cert# of some MovieExec tuple” (A)

- A. $\pi_{\text{producerC\#}}(\text{Movies}) \subseteq \pi_{\text{cert\#}}(\text{MovieExec})$ B. $\pi_{\text{cert\#}}(\text{MovieExec}) \subseteq \pi_{\text{producerC\#}}(\text{Movies})$
C. $\pi_{\text{producerC\#}}(\text{Movies}) - \pi_{\text{cert\#}}(\text{MovieExec}) \neq \emptyset$ D. $\pi_{\text{cert\#}}(\text{MovieExec}) - \pi_{\text{producerC\#}}(\text{Movies}) \neq \emptyset$

- Given a relation $R(A, B, C)$, the functional dependency $AB \rightarrow C$ holds on R . If (a_1, b_1, c_1) , (a_2, b_2, c_2) , (a_1, b_1, c_3) , (a_1, b_2, c_4) are four tuples of R and we know $a_1 \neq a_2$ and $b_1 \neq b_2$, then which one of the following equations must hold? (B)
A. $c_1 = c_2$ B. $c_1 = c_3$ C. $c_1 = c_4$ D. $c_2 = c_4$
- Consider a relation $R(A_1, A_2, A_3)$ has the only key $\{A_1, A_2\}$. Which of the following statements is right? (D)
A. $\{A_1\}$ is also a key. B. $\{A_2\}$ is also a key.
C. $\{A_1, A_3\}$ is a superkey. D. $\{A_1, A_2, A_3\}$ is a superkey.
- Consider the following E/R diagram. Which is the correct relation converted from the relationship SC? (C)



- A. SC (sno, grade) B. SC (cno, grade)
C. SC (sno, cno, grade) D. SC (sno, cno)
- In the undo-logging rules, if transaction T modifies database elements X , then the log record of the form $\langle T, X, v \rangle$ must be written to disk (A)
A. before the new value of X is written to disk.
B. after the new value of X is written to disk.
C. after the $\langle \text{COMMIT } T \rangle$ log record is written to disk.
D. None of the above is correct.
 - Suppose T_1 and T_2 are two transactions, and A is a database element. Which of the following statements about database locking is *not* correct? (D)
A. If T_1 has already held an exclusive lock on A , then T_2 cannot obtain a shared lock on A .
B. If T_1 has already held a shared lock on A , then T_2 can obtain a shared lock on A .
C. If T_1 has already held an exclusive lock on A , then T_2 cannot obtain an exclusive lock on A .
D. If T_1 has already held a shared lock on A , then T_2 can obtain an exclusive lock on A .

二、关系代数（10 分）

11. 设“员工-项目”数据库的关系模式如下：

员工：E (eno, ename) 属性：eno 员工编号、ename 员工姓名

项目：P (pno, pname) 属性：pno 项目编号、pname 项目名称

员工参加项目：W (eno, pno) 属性：eno 员工编号、pno 项目编号

写出完成下列查询的关系代数表达式。

(1) 查询员工“李静”所参加的项目的名称。（5 分）

$$\pi_{pname}(\sigma_{ename='李静'}(E \bowtie W \bowtie P))$$

(2) 查询参加了两个或两个以上项目的员工的编号。（5 分）

$$\pi_{W_1.eno}(\rho_{W_1}(W)) \bowtie \rho_{W_2}(W)$$

$$W_1.eno=W_2.eno \text{ AND } W_1.pno \neq W_2.pno$$

三、SQL（30 分）

12. 设“学校教务管理”数据库中有如下 4 张表：

学生表 Student

sno	sname	ssex	sbirth	sdept
08001	张三	男	1988-02-19...	CS
08002	李四	女	1989-01-09...	CS
08003	王五	女	1990-12-08...	CE
08004	赵六	男	1989-08-30...	IS

教师表 Teacher

tno	tname	tdept
05001	张小明	CS
05002	王小华	IS
05003	李小强	CS
05004	赵小兰	CE

课程表 Course

cno	cname	ccredit	tno
1	高等数学	4	05003
2	数据库原理	5	05003
3	操作系统	3	05001
4	信息系统	4	05002

学生选课表 SC

sno	cno	score
08001	1	92
08001	2	85
08001	3	88
08002	2	90
08002	3	80

属性说明如下：

Student 表：sno 学生编号、sname 学生姓名、ssex 性别、sbirth 出生日期、sdept 学生所在系别；

Teacher 表：tno 教师编号、tname 教师姓名、tdept 教师所在系别；

Course 表：cno 课程编号、cname 课程名称、ccredit 课程学分、tno 任课教师编号；

SC 表：sno 学生编号、cno 课程编号、score 课程成绩。

编写 SQL 语句完成下列查询：

(1) 查询 CS 系的学生的姓名。(3 分)

该查询的结果应为:

sname
张三
李四

```
SELECT sname
FROM Student
WHERE sdept='CS';
```

(2) 列出每门课程的编号及选修该门课程的学生人数，并按课程编号的升序排序
(将学生人数列命名为 scnt)。(6 分)

```
SELECT cno, COUNT(sno) AS scnt
FROM SC
GROUP BY cno
ORDER BY cno
```

给出该查询的结果:

cno	scnt
1	1
2	2
3	1
4	1

(3) 查询选修“数据库原理”的学生的学号、姓名和系别。(6 分)

```
SELECT S.sno, S.sname, S.sdept
FROM Student S INNER JOIN SC ON S.sno = SC.sno
INNER JOIN Course C ON SC.cno = C.cno
WHERE C.cname='数据库原理';
```

给出该查询的结果:

sno	sname	sdept
08001	张三	CS
08002	李四	CS

(4) 查询选修了任课教师为“李小强”的课程的学生的学号、姓名、系别、选修课程的课号、课程名称、学分和所得成绩。(5 分)

该查询的结果应为:

sno	sname	sdept	cno	cname	ccredit	score
08001	张三	CS	1	高等数学	4	92
08001	张三	CS	2	数据库原理	5	85
08002	李四	CS	2	数据库原理	5	90

```
SELECT S.sno, S.sname, S.sdept, C.cno, C.cname, C.ccredit, SC.score
FROM Student S INNER JOIN SC ON S.sno = SC.sno
INNER JOIN Course C ON SC.cno = C.cno
INNER JOIN Teacher T ON C.tno = T.tno
```

WHERE T.tname='李小强';

(5) 完成下列更新操作:(5 分)

a) 插入一条学生选课记录 ('08001', 4, 90)

```
INSERT INTO SC VALUES ('08001', 4, 90)
```

b) 将学生“张三”所有课程的成绩都加 1 分

```
UPDATE SC
SET score = score + 1
WHERE sno IN ( SELECT sno FROM Student WHERE sname='张三');
```

c) 将学号为'08002'的学生选修 3 号课程的选课记录删除

```
DELETE FROM SC WHERE sno = '08002' AND cno = 3;
```

(6) 查询选修了全部课程的学生的学号和姓名。(5 分)

```
SELECT S.sno, S.sname
FROM Student AS S
WHERE NOT EXISTS
(
    (SELECT *
    FROM Course AS C
    WHERE NOT EXISTS
    (
        (SELECT *
        FROM SC
        WHERE SC.sno = S.sno AND SC.cno = C.cno)
    );
```

给出该查询的结果:(注意:要考虑(5)已执行完的更新操作)

sno	sname
08001	张三

四、规范化理论（10 分）

13. Suppose we have a relation $R(A, B, C, D, E, F)$ with a set of FD's $S = \{AB \rightarrow C, BC \rightarrow AD, D \rightarrow E, CF \rightarrow B\}$. Answer the following questions.

要求：写出解题过程。（5 分）

- a) What is $\{A, B\}^+$?
- 1) 将 $BC \rightarrow AD$ 分解为 $BC \rightarrow A$ 和 $BC \rightarrow D$
- 2) $X = \{A, B\}, AB \rightarrow C \Rightarrow X = \{A, B, C\}, BC \rightarrow A, BC \rightarrow D \Rightarrow X = \{A, B, C, D\}, D \rightarrow E \Rightarrow X = \{A, B, C, D, E\}. \therefore \{A, B\}^+ = \{A, B, C, D, E\}$
- b) Whether does $AB \rightarrow D$ follow from the set of FD's S or not?
- $\because D \in \{A, B\}^+, \therefore AB \rightarrow D$ does follow.
- c) Whether does $D \rightarrow A$ follow from the set of FD's S or not?
- 1) Compute $\{D\}^+ = \{D, E\}$
- $\because A \notin \{D\}^+, \therefore D \rightarrow A$ does not follow.

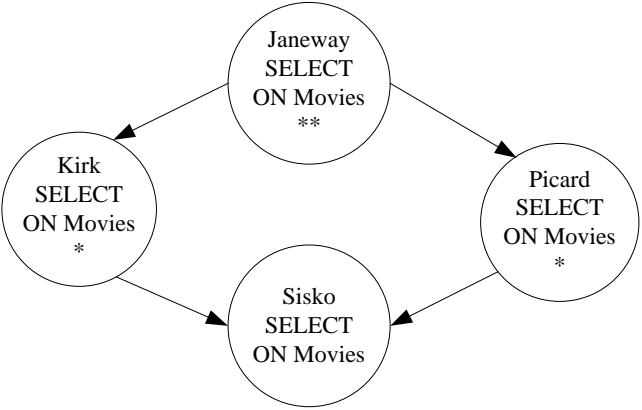
14. Consider a relation $R(A, B, C, D, E)$ with FD's $AB \rightarrow C, C \rightarrow B$, and $A \rightarrow D$. Decompose R into a set of relations that are all in 3NF using the synthesis algorithm for 3NF.

要求：写出解题过程。（5 分）

- 1) The given FD's are minimal basis.
- 2) 3NF synthesis: taking the attributes of each FD as a relation schema.
- That is, we get relations:
- $S_1(A, B, C), S_2(B, C), S_3(A, D)$
- $\because S_2$ is a proper subset of S_1
- \therefore drop S_2
- 3) R has two keys: $\{A, B, E\}$ and $\{A, C, E\}$
- Neither of these keys is a subset of the schemas chosen so far.
- Thus, we must add one of them.
- The final decomposition of R is: $S_1(A, B, C), S_3(A, D), S_4(A, B, E)$ or $S_1(A, B, C), S_3(A, D), S_4(A, C, E)$

五、问答题（15 分）

15. 在某数据库系统中，当前已形成如下授权图（grant diagram）：（5 分）



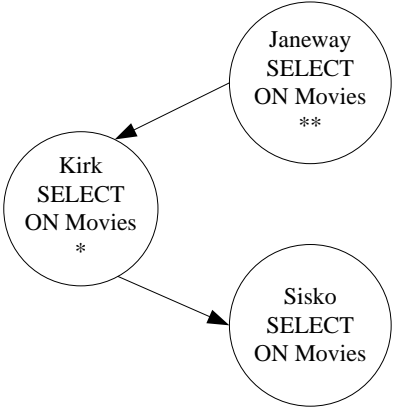
(1) 写出用户 Janeway 需执行的 GRANT 语句。

GRANT SELECT ON Movies TO Kirk, Picard
WITH GRANT OPTION;

(2) 写出用户 Picard 需执行的 GRANT 语句。

GRANT SELECT ON Movies TO Sisko;

(3) 此时，用户 Janeway 又执行了如下语句：
REVOKE SELECT ON Movies FROM picard CASCADE;
画出执行上面语句后的授权图。



16. 在 SQL Server 2008 的某数据库中建立表 T，其两个属性 a 和 b 均为 INT 类型，表 T 只有一行元组，其初始值为：

a	b
1	1

在表 T 上定义如下触发器：
CREATE TRIGGER TR
ON T FOR UPDATE
AS
DECLARE @a INT, @b INT
SELECT @a = a, @b = b FROM T
IF @b < 100 UPDATE T SET a = @b, b = @a+@b
数据库允许触发器递归调用，即触发器中的语句可以激发该触发器的再次执行。请问：

- (1) 执行 SQL 语句 UPDATE T SET a = 1 之后，触发器 TR 被调用了多少次？
- (2) 属性 a 和 b 的最终值是多少？

要求：写出分析与解题过程。（5 分）

答：
(1) 执行 UPDATE T SET a = 1 后，触发器 TR 执行，@a = 1, @b = 1
a = 1, b = 2 第 1 次 更新操作再次激发触发器 TR 执行，
a = 2, b = 3 第 2 次
a = 3, b = 5 第 3 次
a = 5, b = 8 第 4 次
a = 8, b = 13 第 5 次
a = 13, b = 21 第 6 次
a = 21, b = 34 第 7 次
a = 34, b = 55 第 8 次
a = 55, b = 89 第 9 次
a = 89, b = 144 第 10 次 第 11 次调用 TR 时，@b < 100 不满足
触发器 TR 共被调用 11 次。
(2) 属性 a 和 b 的最终值为：a = 89, b = 144

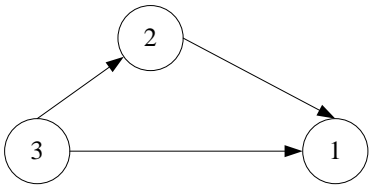
17. 事务 T_1 、 T_2 和 T_3 的一个调度（schedule），如下表所示：

T_1	T_2	T_3
$r_1(A)$		
		$r_3(B)$
		$w_3(B)$
	$r_2(B)$	
	$r_2(A)$	
	$w_2(B)$	
$r_1(B)$		
$w_1(A)$		

该调度是冲突可串行化（conflict-serializable）调度吗？为什么？如果是，写出通过交换相邻的两个操作的顺序将其转化为串行（serial）调度的过程；如果不是，说明为什么。

要求：写出分析与解题过程。（5 分）

答：



1) 是冲突可串行化。因为，优先图 中无回路。

2)

调度 $Sc1 = \underline{r_1(A)} \underline{r_3(B)} \underline{w_3(B)} R_2(B) R_2(A) W_2(B) R_1(B) W_1(A)$

交换 $r_1(A)$ 和 $r_3(B) w_3(B)$ ，得到

$r_3(B) w_3(B) \underline{r_1(A)} \underline{r_2(B)} \underline{r_2(A)} \underline{w_2(B)} r_1(B) w_1(A)$

交换 $r_1(A)$ 和 $r_2(B) r_2(A) w_2(B)$ ，得到

$Sc2 = r_3(B) w_3(B) r_2(B) r_2(A) w_2(B) r_1(A) r_1(B) w_1(A)$

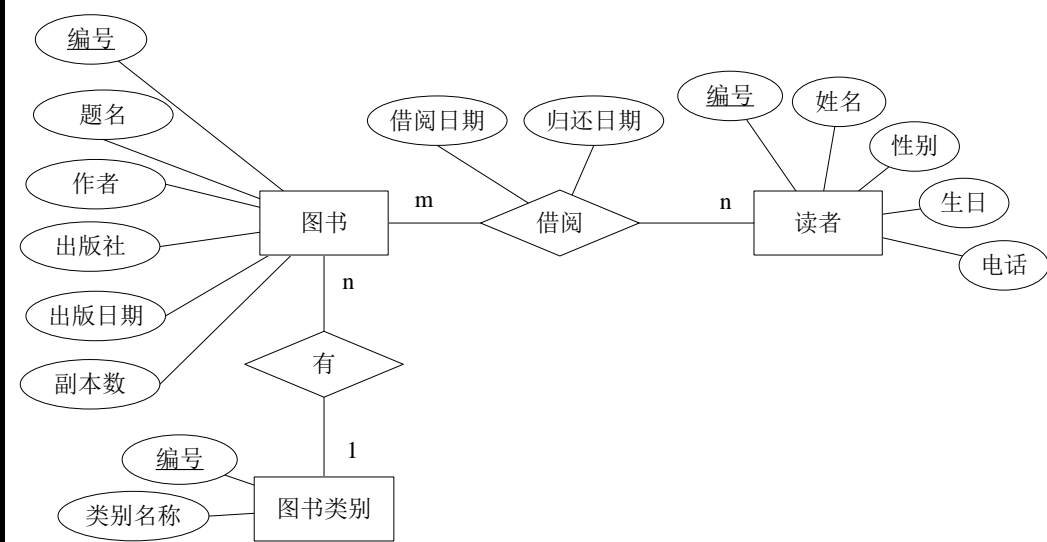
Sc2 是串行调度

六、数据库设计（15 分）

18. 请为某图书馆设计图书管理系统的数据库，需求如下：

- a) 图书馆的每种图书有属性：编号、题名、作者、出版社、出版日期、副本数；
- b) 图书分为若干类别，图书类别有属性：编号、类别名称；
- c) 图书馆的读者有属性：编号、姓名、性别、出生日期、电话号码；
- d) 每位读者可以借阅多种图书，但对于每种图书只能借阅一个副本；
- e) 每种图书可以被多位读者借阅，因为每种图书有多个副本；
- f) 读者借书时应记录该副本的借阅日期，还书时应记录该副本的归还日期。

(1) 根据需求画出该图书管理系统数据库的 E/R 图。（5 分）



(2) 将你画的 ER 图转换为 CREATE TABLE 语句并插入数据。

要求：

- a) CREATE TABLE 语句中要求施加表级的主键和外键约束；
- b) 约束性别属性必须取'男'或'女'两个值之一。
- c) 编写 INSERT INTO 语句将如下示例数据插入到创建的表中。

（5 分）

示例数据：

图书类别

编号	类别名称
C1	科技
C2	文学

图书

编号	题名	作者	出版社	出版日期	副本数	类别
B1	时间简史	霍金	甲	2008-10-01	5	C1
B2	三国演义	罗贯中	乙	2009-06-01	10	C2

读者

编号	姓名	性别	出生日期	电话号码
R1	李勇	男	1991-02-19	13811122233
R2	刘晨	女	1990-01-09	13911122233

借阅情况

读者	图书	借阅日期	归还日期
R1	B1	2012-1-6	
R1	B2	2012-1-6	
R2	B2	2012-1-6	

（注意：下一页还有题目）

```
-- =====
-- 创建数据表
-- =====

-- -----
-- 图书类别
-- -----
CREATE TABLE Category
(
    id          CHAR(2),
    name        VARCHAR(20)  NOT NULL,      -- 图书类别名称
    CONSTRAINT Category_PK PRIMARY KEY(id)
)

-- -----
-- 图书
-- -----
CREATE TABLE Book
(
    id          CHAR(2),
    title       VARCHAR(80)  NOT NULL,      -- 书名
    author      VARCHAR(80)  NOT NULL,      -- 作者
    publisher   VARCHAR(80)  NOT NULL,
    pubDate    DATETIME  NOT NULL,          -- 出版日期
    copyCount   INT    NOT NULL,          -- 副本数量
    categoryId  CHAR(2)NOT NULL,
    CONSTRAINT Book_PK PRIMARY KEY(id),
    CONSTRAINT Book_Category_FK FOREIGN KEY(categoryId) REFERENCES Category(id)
)

-- -----
-- 读者
-- -----
CREATE TABLE Reader
(
    id          CHAR(2),
    name        VARCHAR(20)  NOT NULL,      -- 读者姓名
    gender      CHAR(2)NOT NULL,          -- 性别
    birthday    DATETIME  NOT NULL,        -- 出生日期
    tel         VARCHAR(20)  NOT NULL,      -- 电话号码
    CONSTRAINT Reader_PK PRIMARY KEY(id)
)

-- -----
-- 借阅关系
-- -----
CREATE TABLE Borrow
(
    readerId    CHAR(2),
    bookId      CHAR(2),
    borrowDate  DATETIME  NOT NULL,        -- 借书日期
    returnDate   DATETIME,                -- 还书日期
    CONSTRAINT Borrow_PK PRIMARY KEY(bookId, readerId),
    CONSTRAINT Borrow_Book_FK FOREIGN KEY(bookId) REFERENCES Book(id),
    CONSTRAINT Borrow_Reader_FK FOREIGN KEY(readerId) REFERENCES Reader(id)
)
```

-- =====
-- 插入样本数据
-- =====

INSERT INTO Category VALUES ('C1', '科技')
INSERT INTO Category VALUES ('C2', '文学')

INSERT INTO Book VALUES ('B1', '时间简史', '霍金', '甲', '2008-10-01', 5, 'C1')
INSERT INTO Book VALUES ('B2', '三国演义', '罗贯中', '乙', '2009-06-01', 10, 'C2')

INSERT INTO Reader VALUES ('R1', '李勇', '男', '1991-02-19', '13811122233')
INSERT INTO Reader VALUES ('R2', '刘晨', '女', '1990-01-09', '13911122233')

INSERT INTO Borrow VALUES ('R1', 'B1', '2010-06-21', NULL)
INSERT INTO Borrow VALUES ('R1', 'B2', '2010-06-21', NULL)
INSERT INTO Borrow VALUES ('R2', 'B2', '2010-06-21', NULL)

(3) 编写 SQL 完成下列任务：（5 分）
a) 创建一个视图。该视图包括属性：读者编号和姓名，该读者当前借阅的图书编号、题名和借阅日期。

CREATE VIEW MyView
AS
SELECT Reader.id AS readerId, Reader.name, Book.id AS bookId, Book.title, Borrow.borrowDate
FROM Reader INNER JOIN Borrow ON Reader.id = Borrow.readerId
INNER JOIN Book ON Borrow.bookId = Book.id

b) 在图书的题名属性上创建一个索引。创建该索引的目的是什么？

CREATE INDEX MyIndex ON Book(title)
创建该索引的目的：提高通过题名查找图书的效率。

c) 创建一个存储过程。
要求：输入读者编号，输出该读者当前借阅的图书的本数。

CREATE PROCEDURE MyProc
@readId CHAR(2),
@num INT OUTPUT
AS
SELECT @num = COUNT(bookId)
FROM Borrow
WHERE readerId = @readId

