

## 数据库概论（复习样卷 CS-2022.01）

### 一、（单项选择题）

1. 在关系数据库系统的三级模式中，用于描述数据库中数据的全局逻辑结构的是……（ ）  
[A] 外模式 [B] 概念模式 [C] 内模式 [D] 物理模式
2. 在数据库系统中，负责数据库的设计、建立、管理、维护等全过程的人员被称为……（ ）  
[A] Application programmer [B] Casual users  
[C] Database Administrator [D] Naive users
3. 在四个世界假设中，逻辑数据模型对应的是……（ ）  
[A] 信息世界 [B] 计算机世界 [C] 现实世界 [D] 概念世界
4. 在扩充实体联系模型（EER 模型）中，IS-A 联系用于描述……（ ）  
[A] 实体与属性之间的组成关系 [B] 组合属性与成员属性之间的组成关系  
[C] 实体与实体之间的组成关系 [D] 超实体集与子实体集之间的继承关系
5. 假设在实体集 E 和实体集 F 之间存在一个二元联系 R，且  $\text{card}(E, R) = (1, 1)$ ， $\text{card}(F, R) = (1, n)$ 。那么，在联系 R 上从实体集 E 到实体集 F 的函数对应关系是……（ ）  
[A] 一对一 [B] 一对多 [C] 多对一 [D] 多对多
6. 在下列关于关系的陈述中，不正确的是……（ ）  
[A] 关系中的元组互不相同 [B] 关系中属性的值域互不相同  
[C] 关系中的元组是无序的 [D] 关系中的属性是无序的
7. 设有关系模式  $R(S, F)$ ，S 为关系 R 的属性集合，F 为关系 R 上的函数依赖集。如果  $R_1(S_1, F_1)$  和  $R_2(S_2, F_2)$  构成关系 R 的一个分解，该分解满足依赖保持性的含义是……（ ）  
[A]  $R = R_1 \bowtie R_2$  [B]  $S = S_1 \cup S_2$  [C]  $F = F_1 \cup F_2$  [D]  $F^+ = (F_1 \cup F_2)^+$
8. 在视图（view）定义命令中，在视图对应的子查询中不能使用……（ ）  
[A] distinct 保留字 [B] group by 子句 [C] having 子句 [D] order by 子句
9. 在 SQL 语言中，从一个用户那里回收访问权限的命令（动词）是……（ ）  
[A] DELETE [B] DROP [C] REVOKE [D] ROLLBACK
10. 在基表创建命令中，如果要定义约束“在被引用表中修改主关键字值时，需要在引用表中对外关键字值做同步修改”，则在外键定义子句中需要使用如下的约束定义成分……（ ）  
[A] ON DELETE CASCADE [B] ON DELETE RESTRICT  
[C] ON UPDATE CASCADE [D] ON UPDATE RESTRICT
11. 在使用游标(cursor)实现数据交换的过程中，获取下一条结果元组的命令是……（ ）  
[A] DECLARE [B] FETCH [C] OPEN [D] CLOSE
12. 在下述的调度中，不属于冲突可串行化调度的是（其中  $A \neq B$ ）……（ ）

- [A]  $R_2(A); R_1(B); W_2(A); W_1(B);$       [B]  $R_2(A); R_1(A); W_2(A); W_1(B);$   
 [C]  $R_2(A); W_1(B); W_2(A); R_1(A);$       [D]  $R_2(B); R_1(A); W_2(A); W_1(B);$
13. 在下述有关数据库事务的并发控制技术的描述中, 错误的是 ..... (      )
- [A] 如果一组并发事务之间的某个调度的执行结果等价于它们之间的某种串行执行的结果, 则该调度被称为‘可串行化调度’
- [B] ‘冲突可串行化调度’是‘可串行化调度’的充分而非必要条件
- [C] 如果所有事务都采用两阶段封锁协议, 由此产生的任意合法调度都是冲突可串行化调度
- [D] 在多粒度封锁协议中, 锁的申请采用‘由底向上’的顺序, 锁的释放采用‘自顶向下’的顺序
14. 在数据库管理系统中, 采用基于封锁的并发控制技术的目的是为了实现事务执行的 (      )
- [A] 原子性 Atomicity      [B] 一致性 Consistency  
 [C] 隔离性 Isolation      [D] 持久性 Durability
15. 数据库设计生命周期被划分为若干个阶段, 规范化设计属于其中的 ..... (      )
- [A] 需求分析      [B] 概念设计      [C] 逻辑设计      [D] 物理设计

## 二、(多项选择题)

1. 在下列关系代数的二元运算中, 不属于基本运算的是 ..... (      )
- [A] 并      [B] 交      [C] 差      [D] 笛卡尔积      [E] 自然联接
2. 采用数据库技术来实现数据管理, 其基本特点是 ..... (      )
- [A] 数据集成化      [B] 数据一致性      [C] 数据独立性      [D] 并发控制      [E] 故障恢复
3. 在数据库系统中, 与数据库故障恢复有关的功能是 ..... (      )
- [A] 并发控制      [B] 审计      [C] 日志      [D] 转储      [E] 数据库镜像
4. 在多粒度封锁协议中, 与共享意向排他锁(SIX)相容的锁类型有 ..... (      )
- [A] S 锁      [B] IS 锁      [C] IX 锁      [D] SIX 锁      [E] X 锁
5. 根据关系数据库规范化设计理论, 在下列有关范式的描述中, 正确的是 ..... (      )
- [A] 如果关系 R 能够满足 3NF, 那么该关系一定满足 BCNF
- [B] 如果关系 R 能够满足 BCNF, 那么该关系一定满足 3NF
- [C] 如果关系 R 中的非主属性集为空, 那么该关系一定满足 BCNF
- [D] 如果关系 R 中的非主属性集为空, 那么该关系一定满足 3NF
- [E] 如果关系 R 上的最小函数依赖集为空, 那么该关系一定满足 BCNF

### 三、（关系代数与关系演算）

设有一个商品零售数据库，其关系模式如下：

关系	属性	关系模式
客户	客户编号,姓名,所在城市,优惠折扣	C (cid, cname, city, discnt)
供应商	供应商编号,名称,所在城市,销售提成比例	A (aid, aname, city, percent)
商品	商品编号,名称,存放城市,库存数量,单价	P (pid, pname, city, quantity, price)
订单	订单编号,客户编号,供应商编号,商品编号,销售数量,销售金额	O (ordno, cid, aid, pid, qty, dols)

1、请用关系代数表示下列查询：

- 1) 查询折扣大于 10 的客户的编号和姓名
- 2) 查询折扣最高的客户的编号和姓名
- 3) 查询满足下述条件的供应商的编号：在所有有客户的城市中都销售过商品
- 4) 查询满足下述条件的供应商的编号：只向自己所在城市中的客户销售过商品
- 5) 查询满足下述条件的供应商的编号：向自己所在城市中的所有客户都销售过商品

2、请用关系演算表示下列查询：

- 6) 查询购买过商品的客户的编号和姓名
- 7) 查询满足下述条件的供应商的编号：只向自己所在城市中的客户销售过商品
- 8) 查询满足下述条件的供应商的编号：向自己所在城市中的所有客户都销售过商品

### 四、（SQL 语言应用）

设有一个课程成绩管理数据库，其关系模式如下：

关系	属性	关系模式
学生	学号, 姓名, 就读院系	Student ( sno, sname, dept )
教师	工号, 姓名, 工作院系	Teacher ( tno, tname, dept )
课程	课程号, 课程名, 开课院系, 学分, 课程类型	Course ( cno, cname, dept, credit, optional )
选课	学号, 课程号, 授课教师工号, 选课年份, 选课学期, 成绩	Scorelist ( sno, cno, tno, year, semester, grade )

其中：同一门课同一个学生只能有一条选课记录；课程类型分‘核心’和‘其他’两种；成绩全部采用百分制，成绩大于或等于 60 分才能通过课程选修并获得该门课程的学分。

1. 请用 SQL 语言定义一个‘学生学习统计’视图，用于统计每一位同学的累计选修课程门数，选修通过的课程数量、获得总学分数及平均成绩（不考虑还没有开始选课的同学）
2. 请用标准 SQL 语言表示下述的查询操作。
  - 1) 查询满足下述条件的课程的课程名及开课院系：课程名中含有‘数据库’；
  - 2) 查询满足下述条件的教师的工号和姓名：只讲授过自己所在院系开设的课程；
  - 3) 查询满足下述条件的学生的学号和姓名：选修通过了‘计算机’系开设的所有‘核心’课；
  - 4) 查询每一位教师授课的起止年份，结果返回教师工号、第一次上课年份和最后一次上课年份；（不考虑没有上过课的教师）
  - 5) 查询满足下述条件的学生的学号和姓名：选修通过了自己就读院系开设的所有‘核心’课。

## 五、（关系规范化）

给定关系  $R(A, B, C, D, E, F, G)$  及其上的函数依赖集：

$$M = \{ ADG \rightarrow E, D \rightarrow B, E \rightarrow BDG, AG \rightarrow CD \}$$

请回答下面的三个问题。（直接给出结果，不需要写出计算过程）

1. 请写出与  $M$  相等价的最小函数依赖集；
2. 请直接给出关系  $R$  的所有候选关键字；
3. 请使用 3NF 模式分解算法直接对关系  $R$  进行模式分解，并满足无损联接性和依赖保持性。

## 六、（关系规范化）

设有一个项目管理关系  $S$ ，其属性包括项目编号  $no$ ，项目名称  $A$ ，项目申请单位  $B$ ，申请单位的项目负责人  $C$ ，项目参与单位  $D$ ，参与单位的项目负责人  $E$ 。假设有如下的约束：项目编号具有唯一性；每个项目有唯一的一个申请单位和若干个参与单位，每个单位设一个项目负责人；每个单位可以申请多个项目，也可以参与多个项目；每个人只能负责一个项目（作为申请单位或参与单位的项目负责人）。

1. 请写出关系  $S$  上的最小函数依赖集。
2. 关系  $S$  最高能够满足到第几范式？请简单说明理由。
3. 请将关系  $S$  规范化到 BCNF，并说明分解过程是否满足无损联接性和依赖保持性。

## 七、（数据库设计）

设有一个乒乓球混双比赛信息管理系统，需要管理的信息有：每个代表团的编号和名称，每个运动员的编号、姓名和性别和，每对混双选手的编号和本次比赛的最终名次。其中：（1）代表团编号、运动员编号、混双选手编号分别是代表团、运动员、混双选手的标识属性；（2）每个代表团可以有若干名男运动员和若干名女运动员组成，每个运动员只能隶属于一个代表团；（3）每对混双选手由一名男运动员和一名女运动员组成，允许来自不同代表团的男女运动员混合组队；（4）比赛采用单败淘汰制，直至最后决出冠亚军；每一轮比赛失败的选手，后面不再比赛，其最终名次相互并列（例如：半决赛失败的两对选手并列第 3 名，在 8 进 4 比赛中失败的四对选手并列第 5 名）；（5）系统需要记录每一对选手在参加的每一场比赛中的比赛情况，包括比赛轮次、比赛对手、比赛结果（胜或败）。

1. 请画出该数据库系统的 EER 模型图，并标出实体集与联系之间的参与方式。
2. 请将上述 EER 图转换成对应的关系模式，并写出每个关系上的函数依赖和候选关键字。

## 八、（事务处理）

1. 请根据事务隔离级别和可能产生的并发错误之间的关系完成下表的填写。（可能发生用  $\checkmark$  表示，不可能发生用  $\times$  表示）

隔离级 \ 并发错误	dirty-read	non-repeatable read	phantom read	lost update
Read-Uncommitted				$\times$
Read-Committed				$\times$
Repeatable-Read				$\times$
Serializable				$\times$

2. 证明：由两阶段封锁事务构成的任意合法调度都是冲突可串行化的。

## 数据库概论（复习样卷）

（参考答案，CS-2022.01）

一、（单项选择题）每小题都只有一个正确选项。

题号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
答案	B	C	A	D	C	B	D	D	C	C	B	D	D	C	C

二、（多项选择题）每小题都有 1 个或多个正确选项。

题号	1	2	3	4	5
答案	BE	ABCDE	ACDE	B	BDE

三、（关系代数与关系演算）

1、请用关系代数表示下列查询：

- 1)  $\pi_{cid}(\sigma_{discnt > 10}(C))$
- 2) 令  $M := C$ ,  $\pi_{cid, cname}(C) - \pi_{C.cid, C.cname}(\sigma_{C.discnt < M.discnt}(C \times M))$
- 3)  $\pi_{O.aid, C.city}(C \bowtie O) \div \pi_{city}(C)$
- 4)  $\pi_{aid}(O) - \pi_{A.aid}(\sigma_{O.cid=C.cid \wedge O.aid=A.aid \wedge C.city \neq A.city}(C \times O \times A))$
- 5)  $\pi_{aid}(O) - \pi_{A.aid}(\pi_{A.aid, C.cid}(A \bowtie C) - \pi_{aid, cid}(O))$

2、请用关系演算表示下列查询：

6)

$$\exists city, discnt, ordno, aid, pid, qty, dols( \\ C(cid, cname, city, discnt) \wedge O(ordno, cid, aid, pid, qty, dols))$$

7)

$$\exists aname, acity, percent, ordno1, cid1, pid1, qty1, dols1( \\ A(aid, aname, acity, percent) \wedge O(ordno1, cid1, aid, pid1, qty1, dols1) \wedge \\ \neg(\exists cid, cname, ccity, discnt, ordno2, pid2, qty2, dols2( \\ C(cid, cname, ccity, discnt) \wedge O(ordno2, cid, aid, pid2, qty2, dols2) \wedge ccity \neq acity)))$$

8)

$$\begin{aligned} & \exists aname, city, percent (A(aid, aname, city, percent) \wedge \\ & \forall cid (\exists cname, discnt (C(cid, cname, city, discnt)) \rightarrow \\ & \exists ordno, pid, qty, dols (O(ordno, cid, aid, pid, qty, dols)))) \end{aligned}$$

#### 四、(SQL 语言应用)

1. create view stud\_view as

```
select v1.sno, v1.all_count, v2.c_count, v2.c_sum, v2.c_avg
from (select sno, count(*) as all_count from scorelist group by sno) v1,
     (select s.sno, count(*) as c_count, sum(c.credit) as c_sum, avg(s.grade) as c_avg
      from scorelist s, course c
      where s.cno = c.cno and s.grade >= 60
      group by s.sno) v2
where v1.sno = v2.sno;
```

2. 请用标准 SQL 语言表示下述的查询操作

1) *select cname, dept from Course where cname like '%数据库%';*

2) *select t.tno, t.tname from Teacher t, Scorelist s*  
*where t.tno = s.tno and not exists(*  
*select \* from Scorelist s1, Course c*  
*where s1.tno = t.tno and s1.cno = c.cno and t.dept <> c.dept);*

3) *select sno, sname from Student s*  
*where not exists(*  
*select \* from Course c*  
*where c.dept = '计算机' and c.optional = '核心' and not exists(*  
*select \* from Scorelist x*  
*where x.sno = s.sno and x.cno = c.cno and x.grade >= 60));*

4) *select tno, min(year) as first\_year, max(year) as last\_year*  
*from Scorelist*  
*group by tno;*

5) *select sno, sname from Student s*  
*where not exists(*  
*select \* from Course c*  
*where c.dept = s.dept and c.optional = '核心' and not exists(*  
*select \* from Scorelist x*  
*where x.sno = s.sno and x.cno = c.cno and x.grade >= 60));*

## 五、（规范化设计）

1.  $\{D \rightarrow B, E \rightarrow DG, AG \rightarrow CE\}$
2. AFG 和 AEF
3.  $R1(A, C, E, G), R2(D, E, G), R3(B, D), R4(A, F, G)$  或  $R4(A, E, F)$

## 六、（规范化设计）

1.  $F = \{no \rightarrow (A, B, C), C \rightarrow no, (no, D) \rightarrow E, E \rightarrow no\}$
2. **1NF**。理由：该关系的候选关键字是  $(no, D), (E, D), (C, D)$ ，存在非主属性 **A** 对于关键字的部分函数依赖， $S \notin 2NF$ ，因此只能满足 **1NF**。
3. 到 **BCNF** 的分解过程如下：

第一步：分解到 **3NF**，且满足无损连接性和依赖保持性。

- 1) **R1**(项目编号 **no**，项目名称 **A**，项目申请单位 **B**，申请单位的项目负责人 **C**)

函数依赖集  $F1 = \{no \rightarrow (A, B, C), C \rightarrow no\}$

- 2) **R2**(项目编号 **no**，项目参与单位 **D**，参与单位的项目负责人 **E**)

函数依赖集  $F2 = \{(no, D) \rightarrow E, E \rightarrow no\}$

第二步：分析第一步的分解结果， $R1 \in BCNF, R2 \notin BCNF$ ，因此需要对 **R2** 继续进行分解

- 3) **R21**(项目编号 **no**，参与单位的项目负责人 **E**)

函数依赖集  $F21 = \{E \rightarrow no\}$

- 4) **R22**(项目参与单位 **D**，参与单位的项目负责人 **E**)

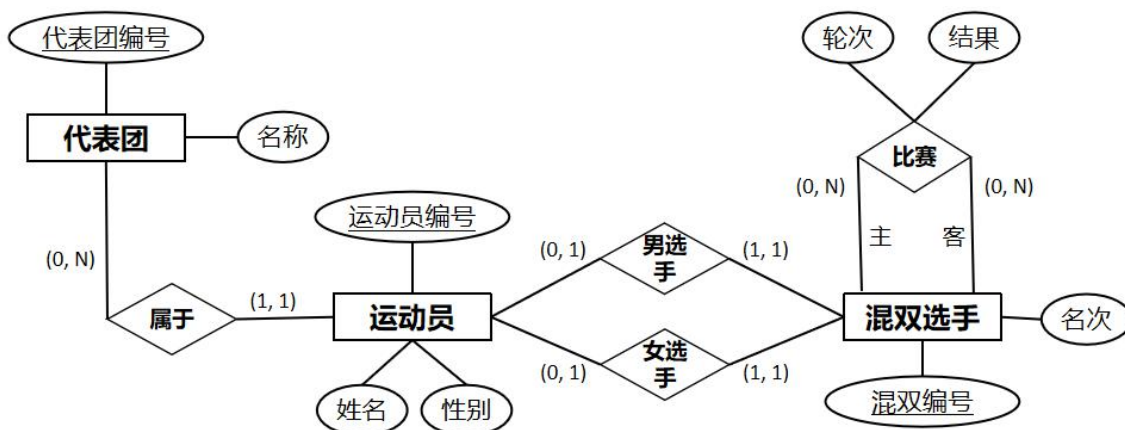
函数依赖集  $F22 = \{\}$

第二步的分解过程满足无损连接性，但不满足依赖保持性。因此，到 **BCNF** 的最终分解结果是  $\{R1, R21, R22\}$ ，且分解过程满足无损连接性，但不满足依赖保持性。



## 七、（数据库设计）

1. 答：（参考答案不唯一）



2. 答：

① 代表团(代表团编号, 名称)

函数依赖集：{ 代表团编号→名称 }

关键字：代表团编号

② 运动员(运动员编号, 姓名, 性别, 代表团编号)

函数依赖集：{ 运动员编号→(姓名, 性别, 代表团编号) }

关键字：运动员编号

③ 混双选手(混双编号, 名次, 男运动员编号, 女运动员编号)

函数依赖集：{ 混双编号→(名次, 男运动员编号),

男运动员编号→女运动员编号,

女运动员编号→混双编号 }

关键字（共有 3 个关键字）：混双编号, 男运动员编号, 女运动员编号

④ 比赛(主队混双编号, 客队混双编号, 轮次, 比赛结果)

函数依赖集：{ (主队混双编号, 客队混双编号)→(轮次, 比赛结果),

(轮次, 主队混双编号)→客队混双编号,

(轮次, 客队混双编号)→主队混双编号 }

关键字（共有 3 个关键字）：

(主队混双编号, 客队混双编号), (轮次, 主队混双编号), (轮次, 客队混双编号)

注：在本设计方案中，每一场比赛都会产生两条‘比赛’记录，参赛的双方轮流作为主队和客队来记录比赛结果。

## 八、（事务处理）

1. 请根据事务隔离级别和可能产生的并发错误之间的关系完成下表的填写。（可能发生用  $\checkmark$  表示，不可能发生用  $\times$  表示）

隔离级 \ 并发错误	dirty-read	non-repeatable read	phantom read	lost update
Read-Uncommitted	$\checkmark$	$\checkmark$	$\checkmark$	$\times$
Read-Committed	$\times$	$\checkmark$	$\checkmark$	$\times$
Repeatable-Read	$\times$	$\times$	$\checkmark$	$\times$
Serializable	$\times$	$\times$	$\times$	$\times$

2. 证明：由两阶段封锁事务构成的任意合法调度都是冲突可串行化的。

证：（归纳法）

- (1) 当调度 S 仅由一个事务组成时，调度 S 是冲突可串行化的
- (2) 假设：由 (n-1) 个 2PL 事务所构成的任意一个合法调度都是冲突可串行化的
- (3) 当调度 S 涉及 n 个 2PL 事务：T<sub>1</sub>, T<sub>2</sub>, …, T<sub>n</sub>，并且 T<sub>i</sub> 是调度 S 中第一个有解锁动作的事务，则我们可以得到以下结论：
  - 可以将 T<sub>i</sub> 的所有动作不经过任何冲突而移动到调度 S 的最前面；
  - 因此，可以将调度 S 转换为另一个冲突等价的调度 S'：
  - （ T<sub>i</sub> 的所有动作 ） （ 其它 (n-1) 个 2PL 事务的动作 ）
  - 根据步骤(2)的归纳假设的前提可得：调度 S' 的后半部分是其它 (n-1) 个 2PL 事务的一个‘冲突可串行化’的调度，即冲突等价于这 (n-1) 个 2PL 事务的某个串行调度。
  - 所以，调度 S' 冲突等价于这 n 个 2PL 事务的某个串行调度
  - 所以，调度 S' 是冲突可串行化的，即：调度 S 也是冲突可串行化的。