

模拟试卷四参考答案

一、选择题

1. C

2. B, D

解析

因为 B 中视图 V2 的一个字段是来自集函数 AVG, 所以不能更新; D 中视图 V4 含有内层嵌套, 且涉及的表是导出该视图的基本表, 所以也不能更新。

3. C

二、简答题

1.

关系模式 $R < U, F >$ 1NF。若 $X \rightarrow Y$ 且 $Y \not\rightarrow X$ 时 X 必含有码, 则 $R < U, F >$ BCNF。

满足 BCNF 关系的特性有:

所有非主属性对每一个码都是完全函数依赖;

所有的主属性对每一个不包含它的码, 也是完全函数依赖;

没有任何属性完全函数依赖于非码的任何一组属性。

2

因为计算机系统中硬件的故障、软件的错误、操作员的失误以及恶意的破坏是不可避免的, 这些故障轻则造成运行事务非正常中断, 影响数据库中数据的正确性, 重则破坏数据库, 使数据库中全部或部分数据丢失, 为了防止出现此类事件带来的灾难性后果, 必须首先调试运行 DBMS 的恢复功能。即把数据库从错误状态恢复到某一已知的正确状态(亦称为一致状态或完整状态)的功能。

DBMS 一般都使用数据转储和登录日志文件实现数据库系统恢复功能。针对不同的故障, 使用不同的恢复策略和方法。例如, 对于事务故障的恢复是由 DBMS 自动完成的, 对用户是透明的。

对于系统故障, 也是由 DBMS 完成恢复操作, 包括撤销(UNDO)故障发生时未完成的事务, 重做(REDO)已完成的事务。DBA 的任务是重新启动系统, 系统启动后恢复操作就由 DBMS 来完成了。

对于介质故障, 则恢复方法是由 DBA 重装最新的数据库后备副本和转储结束时刻的日志文件副本, 然后 DBA 启动系统恢复命令, 由 DBMS 完成恢复功能, 即重做已完成的事务。

3

视图是从一个或几个基本表导出的表。视图本身不独立存储在数据库中,是一个虚表。即数据库中只存放视图的定义而不存放视图对应的数据,这些数据仍存放在导出视图的基本表中。视图在概念上与基本表等同,用户可以如同基本表那样使用视图,可以在视图上再定义视图。

引进 VIEW 的优点有:

- (1) 视图能够简化用户的操作。
- (2) 视图使用户能以多种角度看待同一数据。
- (3) 视图对重构数据库提供了一定程度的逻辑独立性。
- (4) 视图能够对机密数据提供安全保护。

4

数据模型应该反映和规定本数据模型必须遵守的基本的通用的完整性约束条件。数据模型还应该提供定义完整性约束条件的机制,以反映具体应用所涉及的数据必须遵守的特定的语义约束条件。

在关系模型中,任何关系必须满足实体完整性和参照完整性两个条件。这是关系数据模型必须遵守基本的通用的完整性约束条件。

三、求解题

(1)

```
CREATE VIEW V - SSC(SNO,SN,CNO,CN,G)
AS SELECT S.SNO,S.SN,CNO,SC.CN,SC.G
FROM S,SC
WHERE S.SNO = SC.SNO
ORDER BY CNO;
```

(2)

```
SELECT SN, CN, G
FROM V - SSC
GROUP BY SNO
HAVING AVG(G) > 90;
```

四、求解题

(1)

```
SELECT DISTINCT SPJ.SNO
FROM SPJ,P
WHERE P.PNO = SPJ.PNO AND SPJ.JNO = J1 AND P.COLOR = 红 ;
```

(2)

```
SELECT J.JN
```

```
FROM J,SPJ
WHERE J.JNO = SPJ.JNO AND SPJ.SNO = S1 ;
```

(3)

```
SELECT DISTINCT SPJ.PNO
FROM S,J,SPJ
WHERE S.SNO = SPJ.SNO AND J.JNO = SPJ.JNO AND S.CITY = J.CITY;
```

(4)

```
SELECT DISTINCT SPJ.JNO
FROM S,J,SPJ
WHERE S.SNO = SPJ.SNO AND J.JNO = SPJ.JNO AND S.CITY < > J.CITY;
```

五、问答题

如果按照题中的顺序执行甲乙两个事务,则最后的 x 为 800,而不是正确的 500。为此,采用封锁的方法,将甲事务修改为:

```
WHILE( x 上已有排他锁)
{
    等待
}
对 x 加上排他锁
读 x
更新 x = x - 300
释放排他锁
将乙事务修改为:
```

```
WHILE( x 已有排他锁)
{
    等待
}
对 x 加上排他锁
读 x
更新 x = x - 200
释放排他锁
```

可以说明如下:

甲事务	时间	乙事务
XLOCK x	t1	
获得		
	t2	XLOCK x

		等待
更新 $x = x - 300$	t3	等待
$x = 700$		
Commit	t4	等待
UNLOCK x	t5	等待
	t6	获得 XLOCK x
	t7	更新 $x = x - 200$
		$x = 500$
	t8	Commit
	t9	UNLOCK x