

## 第五章作业

姓名：袁昊男      学号：2018091618008

### 一、单选题

1. 以下哪一项工作通常不是 DBA 的职责？（B）  
A. 保障数据库系统正常运行      B. 编写数据库应用程序  
C. 进行数据库备份与恢复      D. 用户权限管理
2. 在事务程序中，执行下面哪个语句后，数据修改将被写回磁盘？（D）  
A. Begin      B. Update      C. Rollback      D. Commit
3. 在数据库中，并发控制的目的是什么？（B）  
A. 实现多事务并行执行      B. 一个事务执行不影响其他事务  
C. 减少事务执行等待时间      D. 提高 DBMS 执行事务的性能
4. 下面哪种级别的锁协议可以同时解决“脏读”“不可重复读”和“丢失更新”问题？（C）  
A. 一级加锁协议      B. 二级加锁协议      C. 三级加锁协议      D. 均不可以
5. 下面哪种备份文件是恢复数据库到故障点时刻状态必不可少的？（C）  
A. 数据库完整备份文件      B. 数据库差异备份文件  
C. 事务日志备份文件      D. 数据库文件备份

### 二、判断题

1. 数据库性能调优是 DBA 进行数据库管理的工作之一。（✓）
2. 只要是事务程序，它能够保证数据一致性。（✓）
3. 在数据库系统中，一旦用户登录数据库后，就可以访问该数据库。（X）
4. 若所有事务遵从两阶段加锁协议，则这些事务的任何并发调度都是可串行化调度。（✓）
5. 只要有数据库备份文件，就可以将数据库恢复到故障点状态。（X）

### 三、填空题

1. 事务 ACID 特性包括原子性、一致性、（隔离性）和（持续性）。
2. 在数据库中，用户能够锁定的最大粒度资源是（数据库）。
3. 能够解决各类数据不一致问题的事务隔离级别是（可串行化）。
4. 权限管理的基本操作包括授予权限、（收回权限）和（拒绝权限）。
5. 典型的数据库备份设备有磁盘阵列、（磁带库）和（光盘库）等。

### 四、简答题

1. DBMS 一般应具有哪些主要的数据库管理功能？

答：数据库定义（数据库结构创建、数据库结构修改、完整性约束定义、索引结构定义）、数据库运行管理（系统启停控制、并发控制、事务管理、安全性控制、完整性控制、日志管理、性能监控）、数据库组织与存储（缓冲区管理、数据组织、数据存取）、数据库维护（数据载入、数据转换、数据库备份、数据库恢复、数据库重构、性能优化）、数据库通信（数据同步、数据复制、数据传输）。

2. 在数据库系统中，事务程序主要解决什么问题？

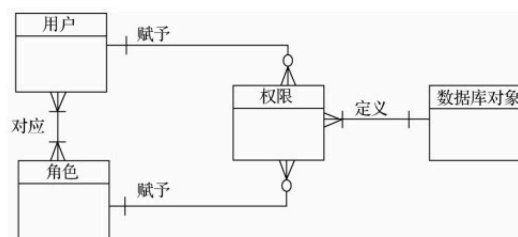
**答：**数据库事务管理是为了在多用户环境中事务程序共享访问数据库对象时，DBMS 确保数据库处于正常状态与数据一致性。即使在数据库事务程序运行中遇到异常或错误，数据库事务管理机制也应保证事务程序的正确执行，让数据库始终处于正常状态。

3. 在数据库系统中，如何预防事务死锁状况出现？

**答：**在数据库系统中，解决死锁问题主要有两类策略：①在并发事务执行时，预防死锁；②在死锁出现后，其中一个事务释放资源以解除死锁。在并发事务执行时，如果能使产生死锁的 4 个必要条件之一不成立，就可以最大限度地预防死锁。在并发事务执行时，一般采用超时法或事务等待图法检测系统是否出现死锁。如果出现了死锁，就需要将被死锁的事务进行资源解除处理。为了降低解除死锁带来的开销，通常选择一个处理死锁代价最小的事务进行撤销，释放该事务持有的所有锁定，使其他事务能够继续运行下去。

4. 数据库系统基本安全模型是什么？它如何实现数据安全访问？

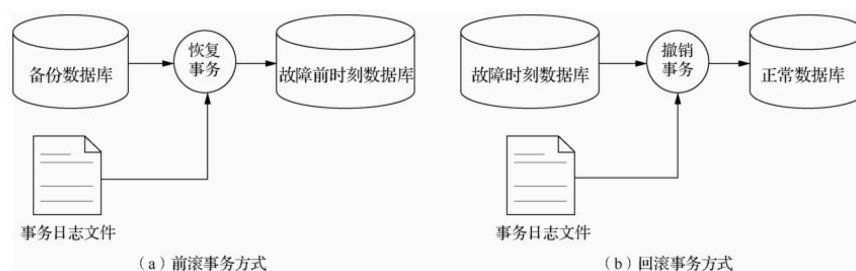
**答：**在数据库系统安全模型中，最基本的安全管理技术手段就是 DBMS 提供的用户授权与访问权限控制功能，该功能用来限制特定用户对特定对象进行授权操作，其数据库存取控制安全模型，如图所示。



**实现数据安全访问：**在数据库存取控制安全模型中，每个数据库对象被定义若干操作访问权限。每个用户可以对多个角色，每个角色也可对应多个用户。用户、角色均可以被赋予若干数据库对象的操作访问权限。一旦用户通过系统身份认证，DBMS 就限制该用户在权限许可的范围内针对特定数据库对象进行访问操作。

5. 实现数据库恢复的技术原理是什么？

**答：**数据库恢复技术是利用数据库备份文件和数据库事务日志文件来实现数据库恢复处理。应根据用户恢复要求，采用前滚事务方式或回滚事务方式恢复数据库。其实现原理如图所示。



**前滚：**这是在数据库备份版本基础上，通过系统执行事务日志文件中记录的操作命令来实现数据库恢复处理，即将系统记录的后像数据重新应用到数据库中，从而将数据库恢复到故障

发生前一刻的状态。

**后滚：**在故障后的数据库基础上，通过系统回滚事务操作来实现的。在恢复处理时，取消错误执行或部分完成的事务对数据库的修改，将系统记录的前像数据恢复到数据库中，从而将数据库恢复到故障发生前一刻的状态。

五、数据库的设计与安全实现

对汽车租赁管理系统进行需求分析，给出实体集、属性集、关系集、关系模式、关系表格（包括主键、外键、约束等），用 SQL 语句 create 这些表格。

1. 需求分析

随着我国经济的快速发展，汽车租赁行业呈现欣欣向荣之势。面对越来越庞大的市场，传统的人工管理方法已难以适应日渐繁多的业务信息。而将高科技融入租赁业，定能促进行业的平稳发展，也能保证行业发展的安全性和可持续性。因此必须使用信息技术手段开发出全新的管理数据库，以期待提高管理效率。

针对传统的汽车租赁管理系统中存在的问题以及一些可以继续完善的地方，利用数据库等高科技手段，建立起一套集人员管理、车辆管理、租赁管理于一体的租赁管理系统，使得管理高效化。此系统既能单独使用、又能配合现有的其他汽车租赁管理系统使用，补全其功能。同时，该系统能够配合前端网站，实现客户在线看、选、租车，租赁方根据客户要求为客户提供服务的一体化进程。

2. 实体属性集

- (1) 客户（客户编号，姓名，性别，身份证号，驾驶证编号，联系电话，居住地址）
- (2) 汽车（车辆编号，车主姓名，车主身份证号，车牌号，类型，颜色，状态）
- (3) 租赁价目（车辆编号，小时租赁价格）
- (4) 租赁登记（租赁单号，客户编号，车辆编号，租赁日期，租赁天数）
- (5) 租赁费用（租赁单号，客户编号，期内费用，超时费用，车辆损坏费用）

3. 关系模式

- (1) 客户与租赁登记：1:N，租赁登记非标识符依赖客户。
- (2) 客户与租赁费用：1:N，租赁登记非标识符依赖客户。
- (3) 汽车与租赁登记：1:1，租赁登记非标识符依赖汽车。
- (4) 租赁登记与租赁费用：1:1，租赁登记标识符租赁费用。
- (5) 汽车与租赁价格：1:1，租赁价格标识符依赖汽车

4. 关系表

表 1 客户表（CLIENT）

字段名称	字段编码	数据类型	字段大小	必填字段	备注
客户编号	ClientNo	Char	12	是	主键
姓名	Name	Varchar	10	是	
性别	Gender	Char	2	是	
身份证号	ClientID	Char	18	是	

驾驶证编号	LicenseID	Char	12	否	
联系电话	Tel	Char	11	是	
居住地址	Addr	Varchar	50	是	

表 2 汽车表 (CAR)

字段名称	字段编码	数据类型	字段大小	必填字段	备注
车辆编号	CarNo	Char	12	是	主键
车主姓名	HostName	Varchar	10	是	
车主身份证号	HostID	Char	18	是	
车牌号	CarID	Char	10	是	
类型	Type	Char	10	是	取值范围：轿车、越野车、SUV、货车、客车
颜色	Color	Char	10	是	
状态	State	Varchar	10	是	取值范围：已出租、未出租

表 3 租赁价目表 (RENT\_PRICE)

字段名称	字段编码	数据类型	字段大小	必填字段	备注
车辆编号	CarNo	Char	12	是	主键，外键
小时租赁价格	HourPrice	Money	/	是	

表 4 租赁登记表 (RENT\_REG)

字段名称	字段编码	数据类型	字段大小	必填字段	备注
租赁单号	RentNo	Char	12	是	主键
客户编号	ClientNo	Char	12	是	外键
车辆编号	CarNo	Char	12	是	外键
租赁日期	StartDate	Date	/	是	
租赁天数	Period	Numeric	(2, 0)	是	

表 5 租赁费用表 (RENT\_FEE)

字段名称	字段编码	数据类型	字段大小	必填字段	备注
租赁单号	RentNo	Char	12	是	主键，外键
客户编号	ClientNo	Char	12	是	外键
期内费用	FeeWithin	Money	/	是	
超时费用	FeeBeyond	Money	/	是	
车辆损坏费用	DamageFee	Money	/	是	

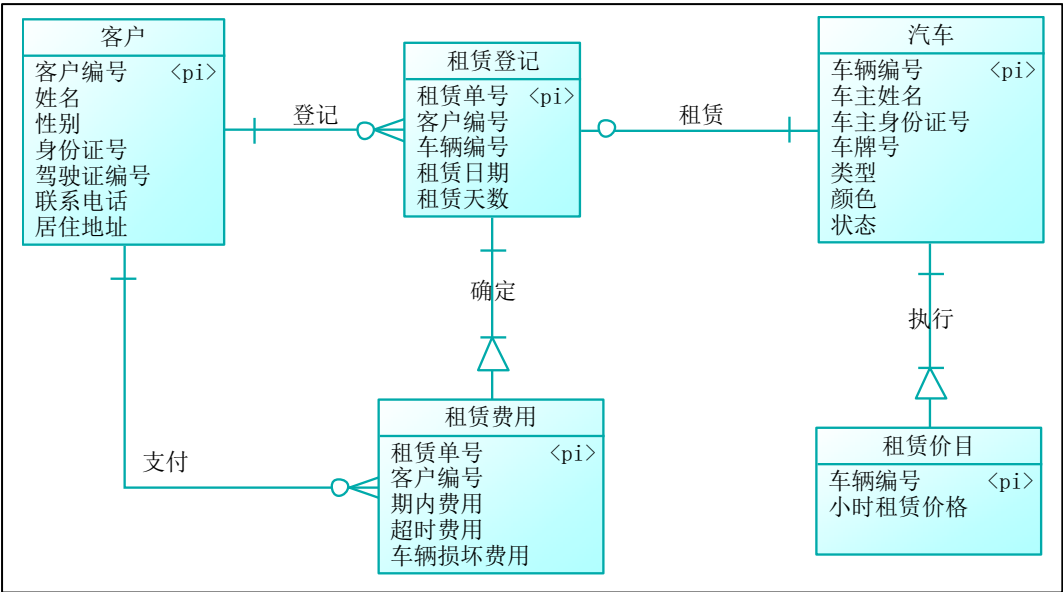
## 5. 角色与权限

表 6 角色与权限表

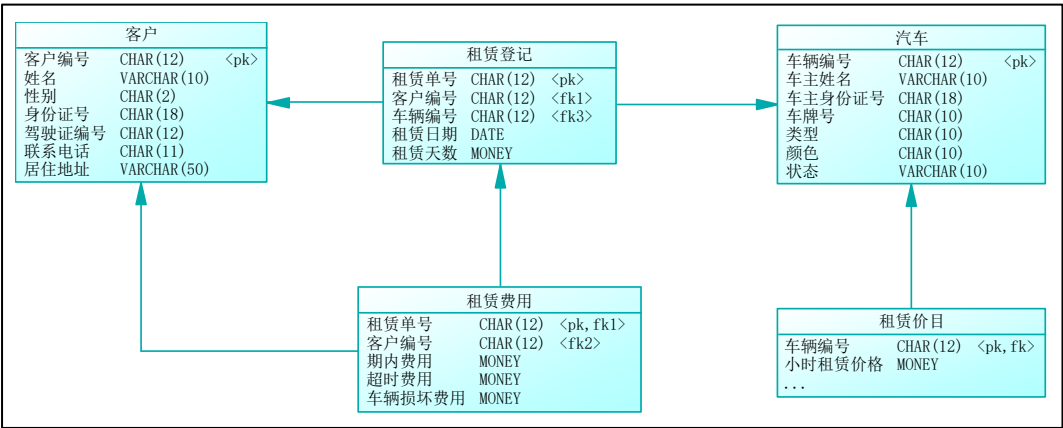
表	客户	业务员	经理	系统管理员
客户表	查询、插入 修改、删除	查询	查询	查询、插入 修改、删除
汽车表	查询	查询、插入 修改、删除	查询、插入 修改、删除	查询、插入 修改、删除
租赁价目表	查询	查询	查询、插入 修改、删除	查询、插入 修改、删除
租赁登记表	查询、插入	查询、插入、修改	查询、插入、修改	查询、插入 修改、删除
租赁费用表	查询	查询、插入、修改	查询、插入、修改	查询、插入

				修改、删除
--	--	--	--	-------

### 6. 逻辑数据模型



### 7. 物理数据模型



### 8. SQL 语句

```

1. CREATE TABLE public."CLIENT"
2. (
3.     "ClientNo" character(12) NOT NULL,
4.     "Name" character varying(10) NOT NULL,
5.     "Gender" character(2) NOT NULL,
6.     "ClientID" character(18) NOT NULL,
7.     "LicenseID" character(12) NOT NULL,
8.     "Tel" character(11) NOT NULL,
9.     "Addr" character varying(50) NOT NULL,
10.    PRIMARY KEY ("ClientNo")
11. )
12.
13.
14. CREATE TABLE public."CAR"
15. (
16.     "CarNo" character(12) NOT NULL,
17.     "HostName" character varying(10) NOT NULL,

```

```

18.     "HostID"      character(18)          NOT NULL,
19.     "CarID"       character(10)          NOT NULL,
20.     "Type"        character(10)          NOT NULL CHECK("Type" IN ('轿车', '
越野車', 'SUV', '貨車', '客車')),
21.     "Color"       character(10).          NOT NULL,
22.     "State"       character varying(10)  NOT NULL CHECK("State" IN ('已出租
', '未出租')),
23.     PRIMARY KEY ("CarNo")
24. )
25.
26.
27. CREATE TABLE public."RENT_PRICE"
28. (
29.     "CarNo"        character(12)          NOT NULL,
30.     "HourPrice"    money                  NOT NULL,
31.     PRIMARY KEY ("CarNo")
32. )
33.
34.
35. CREATE TABLE public."RENT_REG"
36. (
37.     "RentNo"       character(12)          NOT NULL,
38.     "ClientNo"     character(12)          NOT NULL,
39.     "CarNo"        character(12)          NOT NULL,
40.     "StartDate"    date                  NOT NULL,
41.     "Period"       numeric(2, 0)         NOT NULL,
42.     PRIMARY KEY ("RentNo"),
43.     CONSTRAINT "ClientNo_fk" FOREIGN KEY ("ClientNo")
44.         REFERENCES public."CLIENT" ("ClientNo") MATCH SIMPLE
45.         ON UPDATE NO ACTION
46.         ON DELETE NO ACTION,
47.     CONSTRAINT "CarNo_fk" FOREIGN KEY ("CarNo")
48.         REFERENCES public."CAR" ("CarNo") MATCH SIMPLE
49.         ON UPDATE NO ACTION
50.         ON DELETE NO ACTION
51. )
52.
53.
54. CREATE TABLE public."RENT_FEE"
55. (
56.     "RentNo"       character(12)          NOT NULL,
57.     "ClientNo"     character(12)          NOT NULL,
58.     "FeeWithin"    money                  NOT NULL,
59.     "FeeBeyond"    money                  NOT NULL,
60.     "DamageFee"    money                  NOT NULL,
61.     PRIMARY KEY ("RentNo"),
62.     CONSTRAINT "Rent_No_fk" FOREIGN KEY ("ClientNo")
63.         REFERENCES public."CLIENT" ("ClientNo") MATCH SIMPLE
64.         ON UPDATE NO ACTION
65.         ON DELETE NO ACTION
66. )

```