

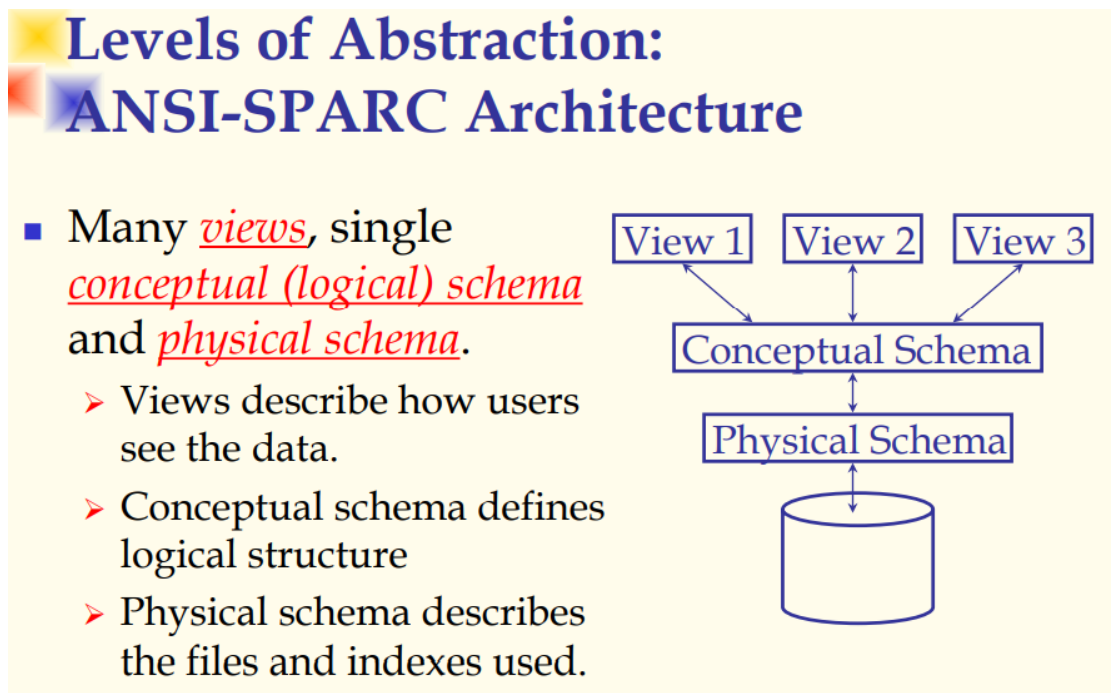
final exam paper

course name database principle examination time 95minutes

student number _____ student name _____

1. How does the modern database system realize the independence of data? In database design, is the higher the NF(Normal Form) that the data schema follows, the better? (12)

(1)



(2)

数据库设计中，数据模式遵循的范式不是越高越好，应取决于应用。数据库的范式主要目的是防止数据冗余，更新异常、插入异常和删除异常，而范式高会存在处理速度缓慢和处理逻辑复杂的问题，从而降低数据库性能，因此需要权衡考虑。

2. A hotel needs to establish a housing management system, and the demand analysis structure is as follows:
- A room has more than one bed, each room has room number (such as 201, 202, etc.), charge standard, number of beds and other information;
 - Guest information includes ID number, name, gender, address and other information;
 - For each stay of each guest, the check-in date, check-out date and prepayment information shall be recorded.

According to demand analysis, the designed relational data model is as follows:

Room (room number, charge standard, number of beds)

Guest (ID number, name, gender, date of birth, address)

Accommodation (room number, ID number, check-in date, check-out date, prepayment)

Assuming that the person who booked the room is the person who checked in, please give all primary keys and foreign keys that exist in the above three relational schemas. (12)

(1) Room

Primary keys: {room number}

foreign keys: null

(2) Guest

Primary keys: {ID number}

foreign keys: null

(3) Accommodation

Primary keys: {room number, ID number, check-in date}/{ room number, ID number, check-out date}

foreign keys: room number, ID number

3. For the three relational schemas in the above application scenario, write SQL statements that express the following query requirements (must be expressed in a single SQL statement):

(1) Use join query to find the name of the guest who booked room 210; (8)

(2) Query the name of the guest who booked all rooms; (8)

(3) Query the room number and guest name booked by only one person in January 2016; (8)

(4) For guests who have booked a room more than 60 times, query the maximum number of room bookings per guest in a single month and the year and month corresponding to the maximum number. (10)

Note: The functions for obtaining the year and month of Datetime data in SQL are getYear(...) and getMonth(...) respectively.

(1) Select G.name

From Guest G, Accommodation A

Where G.ID number=A.ID number And A.room number=210;

(2) Select G.name

From Guest G,

Where not exists

(Select R.room number

From Room R

Where not exists

(Select A.room number

From Accommodation A

Where A.room number=R.room number

```

        And A.ID number = G.ID number);
    )

```

问题：先算出客户预定房间总数（未去重），判断其是否等于房间总数 × 客

户可能预定一个房间多次

```

(3) Select A.room number, (Select G.name From Guest G Where G.ID
    number=A.ID number)
    From Accommodation A
    Where
        getYear(A.check-in date)=2016 And getMonth(A.check-in date)=January
    Group by A.room number
    Having count(G.ID number)=1;

```

问题：

a. 对住宿表根据 room number 进行分组后，count(*)=1× 不满足只被一个客户
预定

b. 对住宿表根据 room number, ID number 进行分组，count(*)=1× 结果是对房
间只预定了一次

```

(4) With
    GuestTemp(ID number) As
        (Select Distinct ID number From Accommodation
        Group by ID number, room number Having count(*)>60),
    AccommodationCount(ID number, year, month, count) As
        (Select ID number, getYear(check-in date),
        getMonth(check-in date),count(*)
        From Accommodation
        Group by ID number, getYear(check-in date), getMonth(check-in date)),
    AccommodationMaxCount(ID number, count) As
        (Select ID number, max(count)
        From AccommodationCount
        Group by ID number),
    Select AC.ID number, AC.count, AC.year, AC.month
    From GuestTemp GT, AccommodationCount AC,
        AccommodationMaxCount AMC
    Where AC.ID number=GT.ID number And AC.ID number=AMC.ID number
    And AC.count=AMC.count;

```

4. Compared with hierarchical and network database systems, query

optimization is more important for relational database systems. Is that right? Why? (12)

4. 正确. 因为相较于层次结构和网状结构, 本身结构较为简单, 很多都不包含属性, 是按树或者链的型式表示的. 在查询时只是逐链做简单的查询. 但关系型数据库系统的关系模型是基于关系(表)组织的. 在查询时会涉及很多拼接两个表的操作. 这个操作是很耗时的. 所以在查询时进行优化减小拼接表的工作量和时间非常重要.

是的, 层次和网状数据库的查询语言一般是过程语言, 而关系数据库的查询语言一般是非过程语言, 仅表达查询的要求, 而不说明查询的过程. 用户不关心查询语言的具体执行过程, 由DBMS确定合理、有效的执行策略. 查询优化是查询处理中重要和必要的一环, 对系统的性能影响颇大. 关系数据库比层次和网状数据库复杂, 因此对其来说查询优化更加重要.

5. Read the instructions below and answer questions (1) and (2). Assume that the transactions T1 and T2 corresponding to the two businesses are related to deposit relationship:

Transfer businesses -- T1(A,B, S) , transfers S dollars from account A to account B;

Interest calculation business -- T2, calculating interest for all current accounts (i.e., the original amount is X dollars, which is $X \times 1.2$ after interest calculation)

- (1) If the interest calculation business is designed to calculate interest on a single account separately, that is, T2(A) calculates interest on account A, and T2(B) calculates interest on account B. Is this scheme correct (3)? Why (6)?
- (2) Assume a concurrent scheduling of two transactions T1 and T2 as shown in the following table, introduce (S, U, X) lock (note: the compatibility matrix is as shown in the figure below), if the initial A=100, B= 60, S=20, what is the final result of A and B? (6) Is the concurrent scheduling correct? (3)

T ₁ (A,B,s)	T ₂
Read(A)	
A:=A-s	Read(A)
Write(A)	A:=A*1.2
	Write(A)

Read(B)	Read(B)
B:=B+s	B:=B*1.2
Write(B)	Write(B)

	Locks already owned by other transactions			
Lock request		S	U	X
	S	Y	Y	N
	U	Y	Y	Y
	X	N	N	N

5. c1) 不正确, 不符合一致性原则. 对于两个账户增加利息后的总资产值, 应在一个事务内计算. 否则如果分开计算, 在先计算SA的利息后, 如果A对B进行一次转账, 再计算B的利息, 会导致B的利息不准确, 出现类似读脏数据的问题.

(2) (S, U, X) 锁 中有三种锁.

S锁: 用于读访问

X锁: 用于写访问

U锁: 在事务更新操作开始前类似S锁, 开始后类似X锁.

当^{已经加}有S锁时, 仍然可以加U锁, ~~不能~~ S锁读取内容.

当加了U锁时, 仍可加S锁进行读. 根据相容矩阵, 仍可加.

U锁, X锁. (??) 根据课上内容, 此时不能再加U锁或X锁.

因为该U锁会升级成X锁.

当加X锁时, 不允许再加~~任何~~其他锁.

最终结果为 A = 120 B = 80

不是正确并发顺序

6. Try to define a trigger on the accommodation relationship table in question 2 to achieve the following function: For each Insert statement, if the inserted record does not have a check-out date, the operation will be rolled back. (12)

```
Create Trigger insertRollback
Before insert on Accommodation
Referencing New as A
For each statement
When A.check-out date=NULL
Rollback;
```