DBMS - CS310

A. SAI CHANDRA KOUSHIK.
19BCS006

1.

- No its not possible to all operators in DBMS
- It's possible if we have ~~both~~ clustered index and one non-clustered index while using both empname and empid.
- It isn't possible if we have either both are clustered index or non-clustered index while using empname and empid.

2. (i) Representing Information
(ii) External Schemas
(iii) Logical Schemas
(iv) Access data
(v) Update data
(vi) Representing data.

3. A DBMS is typically shared among many users. Transaction from these users can be interleaved to improve the execution time of user's qurieres. By interleaving qurieres, users don't have to wait for other user's transactions to complete fully before their own transaction begins. Without interleaving, if user A begins a transaction that will take 1osec to complete and user B wants to begin a transaction, user B would have to wait an additional 10seconds for user A transaction to complete before the database would begin processing user B's request. So finally the given quote is "TRUE".

4. (a) A user must gurantee that his/her transaction doesn't corrupt data or insert nonsense in the database. The xyz data base a user must gurantee that a cash withdraw transaction accurately models the amount a person removes from his or her account.

(b) A DBMS must guarantee that transactions are executed fully and independently of orther transcations. Transcations will either complete fully, or will be aborted and return to intial state.

19BCS006

5. Yes, We can determine the Primary key of the relation with the help of instance.

Ex In a one to many relation we can consider the Column / attribute with unique vaules as a primary key.

6. (a) Create clustered index ix_empname_index ON Student table (StudentName DESC)

" select email from student table '

(b)

| Student ID | Student Name | Email | Age |
|---|---|---|---|
| 1005 | krishna | kris@xyz | 22 |
| 1030 | John | Null | 23 |
| 1020 | John | Jh@xyz | 22. |

7.

e (R1, catalog)  e (R2, catalog)

$\Pi_{R_1.pid}\ \sigma_{R_1:Pid} = R2.pid \wedge R_1.Sid != R_2.sid (R_1 \times R_2)$

| SID | PID | cost |
|---|---|---|
| 1 | 1 | ₹10.00 |
| 2 | 1 | ₹ 9.00 |
| 2 | 3 | ₹ 34.00 |
| 3 | 1 | ₹ 11.00 |

$R_1 \times R_2$ gives us:

| SID | PID | cost | SID | PID | cost |
|---|---|---|---|---|---|
| 1 | 1 | 10 | 1 | 1 | 10 |
| 1 | 1 | 10 | 2 | 1 | 9 |
| 1 | 1 | 10 | 2 | 3 | 34 |
| 1 | 1 | 10 | 3 | 1 | 11 |
| 2 | 1 | 9 | 1 | 1 | 10 |
| 2 | 1 | 9 | 2 | 1 | 9 |
| 2 | 1 | 9 | 2 | 3 | 34 |
| 2 | 1 | 9 | 3 | 1 | 11 |
| 2 | 3 | 34 | 1 | 1 | 10 |
| 2 | 3 | 34 | 2 | 1 | 9 |
| 2 | 3 | 34 | 2 | 3 | 34 |
| 2 | 3 | 34 | 3 | 1 | 11 |
| 3 | 1 | 11 | 1 | 1 | 10 |
| 3 | 1 | 11 | 2 | 1 | 9 |
| 3 | 1 | 11 | 2 | 3 | 34 |
| 3 | 1 | 11 | 3 | 1 | 11 |

$\sigma_{R_1.Pid = R_2.Pid}$ gives us:

| SID | PID | Cost | SID | PID | cost |
|---|---|---|---|---|---|
| 1 | 1 | 10 | 1 | 1 | 10 |
| 1 | 1 | 10 | 2 | 1 | 9 |
| 1 | 1 | 10 | 3 | 1 | 11 |
| 2 | 1 | 9 | 1 | 1 | 10 |
| 2 | 1 | 9 | 2 | 1 | 9 |
| 2 | 1 | 9 | 3 | 1 | 11 |
| 2 | 3 | 34 | 2 | 3 | 34 |
| 3 | 1 | 11 | 1 | 1 | 10 |
| 3 | 1 | 11 | 2 | 1 | 9 |
| 3 | 1 | 11 | 3 | 1 | 11 |

$\sigma_{R_1.pid = R_2.pid \wedge R_1.sid != R_2.sid}$ gives

| SID | PID | Cost | SID | PID | Cost |
|---|---|---|---|---|---|
| 1 | 1 | 10 | 2 | 1 | 9 |
| 1 | 1 | 10 | 3 | 1 | 11 |
| 2 | 1 | 9 | 1 | 1 | 10 |
| 2 | 1 | 9 | 3 | 1 | 11 |
| 3 | 1 | 11 | 1 | 1 | 10 |
| 3 | 1 | 11 | 2 | 1 | 9. |

SQL :

```
Select  C.sid
From    Catalog  C
Where   exists  Select  C1.sid
                From    Catalog  C1
                where   C1.Pid  = C.pid  AND
                        C1.sid ≠ C.sid
```

1 a BCS00

8) $\Pi_{S.name}(\Pi_{sid}((\sigma_{color='red'} Parts) \bowtie (\sigma_{cost<100}(catalog)) \bowtie Suppliers))$

"Invaild querg"

9.) The view of Emp can be updated automatically by updating Emp

```
CREATE VIEW SeniorEMP (Eid name age salary)
AS SELECT E.eid, E.ename, E.age, E.salary
   FROM EMP E
   WHERE E.age > 50.
```