

Endsem

1) Using empname as a clustered index is possible only when every employee will have a unique name. if this is ensured, the tuples will have be organized according empname alphabetically.

Using empid as a clustered index is definetily possible considering everyone already has a unique id assigned to them. The tuples will be organized according to empid.

using both empname & empid as clustered indexes may not be possible but it is possible to have one clustered index and one non-clustered index.

2) • DDL is important in representing information because it is used to describe external and logical schemas.

• DML is used to access and update data. it is not important for representing the data.

3. True

→ A DBMS is typically shared among many users.

Transactions from these users can be interleaved to improve the execution time of users queries.

By interleaving queries, users do not 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 10 secs to complete, and user B wants to begin a transaction, user B would have to wait an additional 10 secs for user A's transaction to complete before the database would begin processing user B's request.

4)

a) A user must guarantee that his or her transaction does not corrupt data or insert nonsense in the database. For example, in a banking database, a user must guarantee that a cash withdraw transaction accurately models the amount a person removes from his or her account.

A database application would be worthless if a person removed 20\$ from an atm but the transaction set their balance to Zero.

b) A DBMS must guarantee that transactions are executed successfully and independently of other transactions. an essential property of a DBMS is that a transaction should execute automatically, or as if it is the only transaction running.

Also, transactions will either complete fully, or will be aborted and the database returned to its initial state.

this ensures that the database remains consistent.

Query in Relational Algebra

lalithaanjale.P
19BCS087

7) $P(R_1, \text{Catalog})$
 $P(R_2, \text{Catalog})$

$$\pi_{R_1.Pid \cap R_1.Pid = R_2.Pid \wedge R_1.Sid = R_2.Sid} (R_1 \times R_2)$$

SQL Query

Select distinct $R_1.Pid$ from Catalog R_1 , Catalog R_2

Where $R_1.Pid = R_2.Pid$ and $R_1.Sid \neq R_2.Sid$.

By using the following

SID	PID	Cost
-----	-----	------

1	1	\$ 10.00
---	---	----------

2	1	\$ 9.00
---	---	---------

2	3	\$ 34.00
---	---	----------

3	1	\$ 11.00
---	---	----------

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

SID	PID	Cost	SID	PID	Cost
1	1	\$ 10.00	1	1	\$ 10.00
1	1	\$ 10.00	2	1	\$ 9.00
1	1	\$ 10.00	3	1	\$ 11.00
2	1	\$ 9.00	1	1	\$ 10.00
2	1	\$ 9.00	2	1	\$ 9.00
2	1	\$ 9.00	3	1	\$ 11.00
2	3	\$ 34.00	2	3	\$ 34.00
3	1	\$ 11.00	1	1	\$ 10.00
3	1	\$ 11.00	2	1	\$ 9.00
3	1	\$ 11.00	3	1	\$ 11.00

$R_1 \times R_2$ gives us

SID	PID	Cost	SID	PID	Cost
1	1	\$ 10.00	1	1	\$ 10.00
1	1	\$ 10.00	2	1	\$ 9.00
1	1	\$ 10.00	2	3	\$ 34.00
1	1	\$ 10.00	3	1	\$ 11.00
2	1	\$ 9.00	1	1	\$ 10.00
2	1	\$ 9.00	2	1	\$ 9.00
2	1	\$ 9.00	2	3	\$ 34.00
2	1	\$ 9.00	3	1	\$ 11.00
2	3	\$ 34.00	1	1	\$ 10.00
2	3	\$ 34.00	2	1	\$ 9.00
2	3	\$ 34.00	2	3	\$ 34.00
2	3	\$ 34.00	3	1	\$ 11.00
3	1	\$ 11.00	1	1	\$ 10.00
3	1	\$ 11.00	2	1	\$ 9.00
3	1	\$ 11.00	2	3	\$ 34.00
3	1	\$ 11.00	3	1	\$ 11.00

$\sigma_{R_1.Pid = R_2.Pid \wedge R_1.sid \neq R_2.sid}$ gives us.

SID	PID	Cost	SID	PID	Cost
1	1	\$ 10.00	2	1	\$ 9.00
1	1	\$ 10.00	3	1	\$ 11.00
2	1	\$ 9.00	1	1	\$ 10.00
2	1	\$ 9.00	3	1	\$ 11.00
3	1	\$ 11.00	1	1	\$ 10.00
3	1	\$ 11.00	2	1	\$ 9.00

8) $\pi_{sname}(\pi_{sid}((\sigma_{color = Ored} O(Parts))) * ($

$(\sigma_{cost < 100} (catalog)) * Suppliers))$

Invalid query.

Explanation :- this relational algebra statement does not return anything because of the sequence of projection operators. Once the sid is projected, it is only field in the set. therefore projecting on same cannot return anything.

9) The following view on Emp can be updated automatically by updating Emp:

```
CREATE VIEW SeniorEmp (eid, name, age, Salary)
```

```
AS SELECT E.eid, E.name, E.age, E.salary FROM
```

```
Emp E WHERE E.age > 50
```