

Kip Taylor

Database

Last hw

1. Answer below.

Let $R(ABCDE)$

A	B	C	D	E
a ₁	b ₁	c ₁	d ₁	e ₁
a ₁	b ₂	c ₃	d ₂	e ₄
a ₂	b ₃	c ₂	d ₂	e ₄
a ₂	b ₄	c ₂	d ₂	e ₄
a ₃	b ₂	c ₃	d ₂	e ₄
a ₄	b ₁	c ₁	d ₁	e ₁

$\{AB \rightarrow C, B \rightarrow C, C \rightarrow D, D \rightarrow E, E \rightarrow D\}$
Key = {AB}
Not in 2NF i.e. partial dependency $B \rightarrow C$
So Decompose

A	B	C	D	E
a ₁	b ₁	b ₁	c ₁	d ₁
a ₁	b ₂	b ₂	c ₃	d ₂
a ₂	b ₃	b ₃	c ₂	d ₂
a ₂	b ₄	b ₄	c ₂	d ₂
a ₃	b ₂	b ₂	c ₃	d ₂
a ₄	b ₁	b ₁	c ₁	d ₁

AB:CK $B \rightarrow C$ B:CK
 $C \rightarrow D$
 $D \rightarrow E$

Now R is in 2NF but not in 3NF
 as transitive dependency exist $C \rightarrow D, D \rightarrow E$
 So, decompose

R1(AB)	A	B	R2(BC)	B	C	R3(CD)	C	D	R4(DE)	D	E
a ₁	b ₁	b ₁	c ₁	c ₁	d ₁	d ₁	e ₁				
a ₁	b ₂	b ₂	c ₃	c ₃	d ₂	d ₂	e ₄				
a ₂	b ₃	b ₃	c ₂	c ₂	d ₂	d ₂	e ₄				
a ₂	b ₄	b ₄	c ₂	c ₂	d ₂	d ₂	e ₄				
a ₃	b ₂	b ₂	c ₃	c ₃	d ₂	d ₂	e ₄				
a ₄	b ₁	b ₁	c ₁	c ₁	d ₁	d ₁	e ₁				

AB:CK B:CK C:CK D:CK

Now Relation is in 3NF, lossless,
 dependency preserve, BCNF

Hope this covers what was asked. If you need any more let me know please.

2. Answer below.

- a. SQL Injection are common for below reasons
Prevalence of SQL Injection vulnerabilities
As database contains critical information

SQL injection attack is a attack that send malicious commands to database to retrieve sensitive data.

We can prevent this by:

1. By encrypting sensitive data
2. Giving accounts to users with least privileges necessary.
3. Ensure that data is valid.
4. Use stored procedures.
5. Use a firewall
6. Apply regular software patches.
7. Monitor SQL statements continuously.

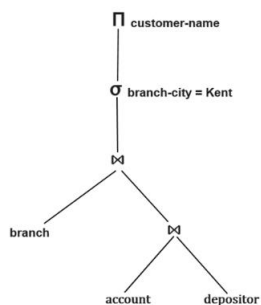
3. Answers below

It is not a good programming practice to create indices on every searchable attribute because;

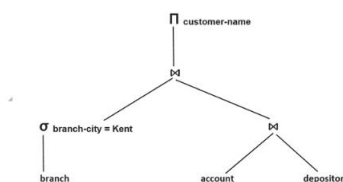
- Firstly, an index requires extra storage spaces in memory.
- An index involves additional overhead in terms of both CPU time and disk I/O at the time of insertion and deletion of records.
- As far as updates are concerned, they are not an issue for indices on primary key, but if the index is on a non-primary key, it needs to be changed on every update as well.
- In fact, for queries having conditions for many search keys, using indices on just a few of the keys is sufficient to provide better efficiency.
- Finally, it is sufficient to say that if too many indices exist in the database, then the performance improvement is not much.

4. Answer below.

Attempted to optimize the initial version we can modify a little bit. We can perform the join operation first and then select. the city as Kent. This method will help reduce redundancy in our relational algebra query



1. INITIAL EXPRESSION TREE



5. ACID stands for, Atomicity, Consistency, Isolation, Durability.
- a. Atomicity: It states that a database transaction must contain everything or nothing. It requires complex ways to be achieved. If a transaction has various steps, all steps must occur simultaneously.
 - b. Consistency: It states that data cannot be changed without following database rules. If a transaction occurs in violation, error occurs and roll back happens.
 - c. Isolation: It states condition of database management to ensure levels of maintaining data isolation from various simultaneous operations, so that lock condition doesn't occur, and deadlock doesn't happen.
 - d. Durability: Ensures permanent data saving so that during a database crash, data can be easily recovered during recovery. Storage happens on a secondary memory storage. It completes reliability of the ACID system.
 - e. A transaction is the lowest entity in a database which must follow ACID properties, carries out data retrieval or update.
 - f. A lock occurs when a process locks data it is working on in anticipation of another process accessing that data.
 - J. A deadlock occurs when a process 1 locks data and blocks process 2 and process 2 blocks data it is using and process 1 is made to wait. All this happens simultaneously.

That should cover all types with scenarios in simplification.