9/10/23

ASSIGNMENT 2

PAGE NO.: 27

Unit: 3 & 6 The ER Model, Relational - Database Design & Transaction Management

Q.1 What is mean by lossless and lossy decomposition? Explain with example.

→ Lossless Decomposition.

It ensures that no data is lost during the compression and decompression process. The decomposition of relation R into R1 and R2 is lossless when the join of R1 and R2 produces the same relation as in R. Also referred as a non-additive decomposition.

→ Lossy Decomposition.

It sacrifices some of the data to achieve higher compression ratios. The decomposition of relation R into R1 and R2 is lossy when the join of R1 and R2 does not yield the same relation as in R. Also referred as lossy-join decomposition. From practical point of view, decomposition should not be lossy.

eg:

R

Ano	Balance	Bname
A01	5000	Rajkot
A02	5000	Surat

R1

Ano	Balance
A01	5000
A02	5000

R2

Balance	Bname
5000	Rajkot
5000	Surat

Same

Not Same

Ano	Balance	Bname
A01	5000	Rajkot
A02	5000	Surat

Lossless Decomposition

Ano	Balance	Bname
A01	5000	Rajkot
~~A01~~	~~5000~~	~~Surat~~
A02	5000	Rajkot
A02	5000	Surat

Lossy Decomposition

Q.2 Consider the relation scheme R(E,F,G,H,I,J,K,L,M,N) and the set of functional dependencies- {E,F} → {G}, {F} → {I,J}, {E,H} → {K,L}, {K} → {M}, {L} → {N}. What is the key for R?

—↓ Here, the attributes are (E,F), F, (E,H), K, L.
So if we find the closure of single function dependencies, it cannot form a key.
eg: closure of $(EH)^+$ = EHKLMN ≠ EFGHIJKLMN

Now because the left side attributes of any functional dependencies alone cannot form a key. Let us try with the combination of two functional dependencies.

∴ $(EFH)^+$ = {E,F,G,H,I,J,K,L,M,N}

∴ EFH is the key for R.

Q.3 What is Serializability? Explain conflict and view serializebility.

—↓ Serializability is a property that ensures the execution of concurrent transactions produces results equivalent to some serial execution of those transactions. There are two main concepts related to serializability:

i) Conflict Serializability.
It is a common method used to determine if a schedule of transactions is serializable. It is based on the idea of analyzing the

conflicts between transactions. A conflict between two transactions occurs when:

- One transaction reads a data item that another transaction has modified
- One transaction writes a data item that another transaction has also written.

eg: If Transaction A reads a value written by Transaction B, the schedule is conflict-serializable if the read by A follows the write by B.

ii) View Serializability.

It is a broader concept that takes into account the "view" of the database seen by a transaction. It doesn't just look at conflicts but also at how each transaction "views" the database, ensuring that their views are consistent. In other words, it looks at set of transactions and their read & write operations and checks if the final state of the database as seen by each transaction is equivalent to the database's state.

Q.4 What is locking? Define each type of locking.

→ locking is a mechanism used to control concurrent access to data and ensure data consistency in multi-user environments. When multiple users or transactions attempt to access and modify the same data anomalies can occur then locking prevents these issues by

providing a controlled and serialized way of accessing the data. There are two types of locking mode:

i) Shared Lock (S-Lock).

It is used when a transaction wants to read data without the intention of modifying it. It allows multiple transactions to read the data concurrently while preventing any transactions from acquiring an exclusive lock on the same data.

ii) Exclusive Lock (X-Lock)

It is used when a transaction intends to modify the data. It ensures that no other transaction can acquire a shared or exclusive lock on the same data until the exclusive lock is released.

Q.5 What is concurrency? Describe the problems occurs due to the concurrency.

→ Concurrency refers to the ability of a system to handle the multiple tasks or processes simultaneously. It allows multiple users or transactions to access and manipulate data at the same time.

→ While concurrency can improve system efficiency and responsiveness, it also includes several challenges and problems, which are:

i) Lost Update Problem

This problem occurs when two or more transactions attempt to update the same data simultaneously. If one transaction's update is overwritten by another, the changes made by the first transaction are lost.

ii) Dirty Read

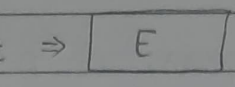
This occurs when one transaction reads data that has been modified by another transaction but not yet committed. This can lead to incorrect or inconsistent data being retrieved.

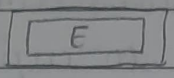
iii) Non-Repeatable Read.

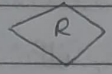
This occurs when a transaction reads the same data ~~when a transaction~~ reads the same data multiple times and the data changes between reads due to other transaction committing their updates. This can lead to problems in data analysis and decision-making.

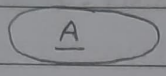
Q.6 Describe ER notations with example. and illustrate the design issues in database design.

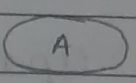
→ ER notation is a widely used modeling technique for designing database. It helps represent the structure and relationships between various entities in a clear and ~~visual~~ visual manner.

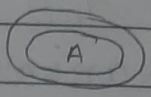
① Entity Set ⇒ 

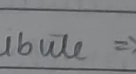
② Weak Entity Set ⇒ 

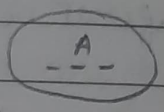
③ Relationship set ⇒ 

④ Primary key ⇒ 

⑤ Attribute ⇒ 

⑥ Multivalued attribute ⇒ 

⑦ Derived Attribute ⇒ 

⑧ Discriminating Attribute of Weak Entity ⇒ 

→ Design issues which occurs during database design :

i) Data Integrity : Ensuring that data in the database is accurate and consistent which include defining appropriate constraints, validating data inputs.

ii) Normalization : Structuring the database to eliminate data redundancy and prevent update anomalies which involves breaking down tables into smaller, related tables.

iii) Security : Protecting the database from unauthorized access and ensuring that sensitive data is kept secure. Design issues include access control, encryption and auditing.

iv) Backup and Recovery : Planning for data backup, disaster recovery and business continuity. This involves regular backups, recovery procedures and testing

v) User Interface : Designing user-friendly forms and interfaces for data entry and retrieval. This involves user experience and user interface design.

Q.7 What are the features of good relational design?

→ A good relational database design is essential for creating a database that is efficient, scalable and easy to maintain. That includes :

i) Data Integrity
ii) Normalization
iii) Scalability
iv) Consistency
v) Efficient Data Retrieval
vi) Testing and Validation
vii) Performance Tuning.

Q.8 Describe ACID properties along with an example

→ ACID is an acronym that stands for a set of properties that ensure the reliability and consistency of database transactions.

i) Atomicity

It ensures that a Transaction is treated as a single, indivisible unit of work. It means that either all the changes made by a Transaction are applied, or none of them are. If any part of a Transaction fails, the entire

transaction is rolled back and the database returns to its previous state.

eg: Consider a transaction to transfer Rs 50 from account A to account B. Here, if Rs. 50 is deducted from account A then it must be added to account B.

ii) Consistency

It ensures that a transaction brings the database from one consistent state to another consistent state. It means that a transaction must adhere to integrity constraints, business rules and other criteria specified for the database.

eg: Consider a database of airline reservations, a consistency rule could be that the total number of seats booked in a flight cannot exceed the total number of available seats. If it exceed this limit, it will be rolled back.

iii) Isolation

It ensures that the intermediate states of a transaction are not visible to other transaction. It prevents concurrent transactions from interfering with each other by ensuring that they run in isolation until they are complete.

eg: Once our transaction starts from one step its result should not be access by any other transaction until last step is completed.

iv) Durability

It guarantees that once a transaction is committed, its effects are permanent and will survive any system failures, such as power outages or crashes.

eg: After a successful online purchase, the transaction records that you've paid for a product and the stock is updated to reflect the reduced inventory.

Q.9 Explain different transaction states with diagram.

→ Different transaction states are :-

i) Active : This is the initial state where the transaction stays while it is executing. It transitions from the 'initial' state to the 'Active' state.

ii) Partially Committed : This is when a transactions executes its final operations successfully and is ready to make its changes permanent in the database.

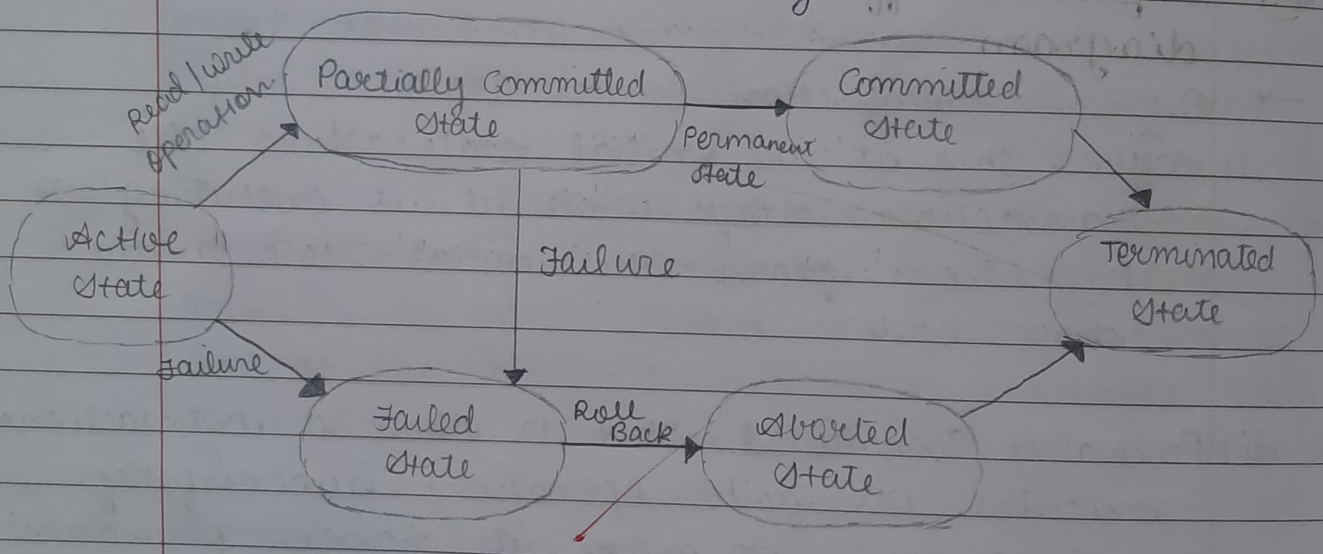
iii) Committed : Once a transaction has successfully completed and all its changes have been made permanent. They are stored in the database and will persist even in the case of a system failure.

iv) Failed : When the transaction, due to system failur

or other error, is unable to proceed to completion. It was unsuccessful due to reasons beyond its control which contradicts "Abort".

(v) Aborted: When a transaction encounters an error or if a rollback is explicitly requested, it enters this state. Any changes made by the transaction are undone and the database is restored to the state it was in before it began.

-> Transaction state Diagram :-

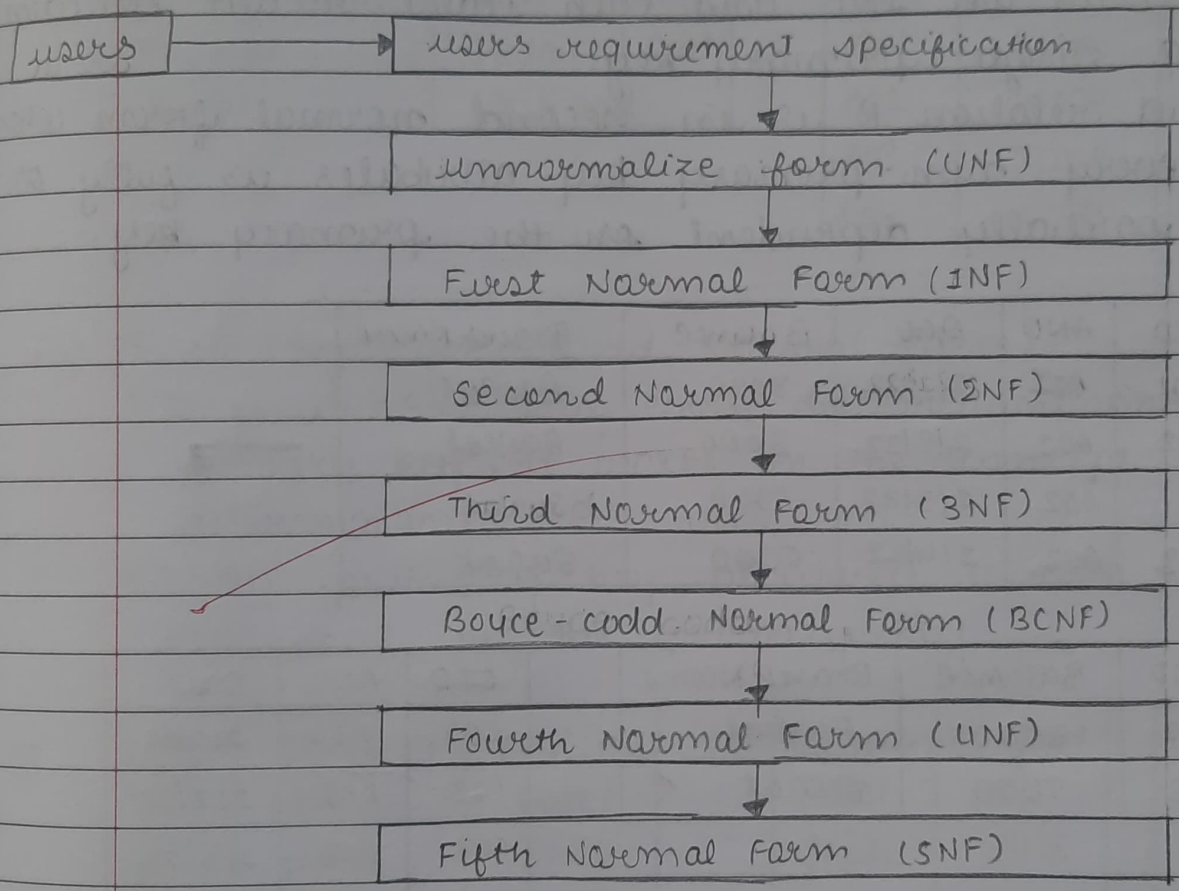


Q.10 What is Normalization? Describe the process of Normalization.

-> Normalization is a process in database design that helps organize and structure a relational database in such a way that it reduces data redundancy and anomalies, making the database more efficient, maintainable and consistent. The process of normalization involves breaking down a large table with redundant data

into smaller, related tables each with a specific purpose. These tables are then connected through relationships, primarily by defining foreign keys.

→ Process of Normalization :-



Q.11 Explain all Normal Forms with Example.

→ i) 1NF (First Normal Form)

> Each cells of a table should contain a single values.

> A Relation R is in first normal form if it does not contain any composite attributes or multi-valued attributes.

eg:

CID	Name	Address
C01	Ayu	Surat, Gujarat
C02	Mitesh	Jaipur, Rajasthan
C03	Jay	Mumbai, Maharashtra

→

CID	Name	City	State
C01	Ayu	Surat	Gujarat
C02	Mitesh	Jaipur	Rajasthan
C03	Jay	Mumbai	Maharashtra

ii) 2NF (Second Normal Form).

- It is in 1NF and each table should contain a single primary key.
- A relation R is in second normal form when every non-primary key attributes is fully or partially dependent on the primary key.

eg:

CID	ANO	Date	Balance	BranchName
C01	A01	1/1/22	3000	Rajkot
C02	A01	1/2/22	3000	Rajkot
C01	A02	1/3/22	5000	Surat
C03	A02	1/4/22	5000	Surat

→

ANO	Balance	BranchName
A01	3000	Rajkot
A02	5000	Surat

CID	ANO	Date
C01	A01	1/1/22
C02	A01	1/2/22
C01	A02	1/3/22
C03	A02	1/4/22

iii) 3NF (Third Normal Form)

- It is in 2NF and there is no transitive dependency
- Every non-key attribute is non-transitively dependent on the primary key.

eg:

ANO	Balance	BranchName	BrandAddress
A01	50000	Rajkot	Kalawad Road
A02	40000	Rajkot	Kalawad Road
A03	35000	Surat	CG Road
A04	25000	Surat	CG Road.

Branch Name	Branch Address
Rajkot	Kalawad Road.
Surat	CG Road.

ANO	Balance	BranchName
A01	50000	Rajkot
A02	40000	Rajkot
A03	35000	Surat
A04	25000	Surat

iv) BCNF (~~By~~ Boyce - Codd Normal Form)

- It is in 3NF and every determinant should be ~~primary key~~. It works on the concept of determinant.
- Every primary key attribute is non-transitively dependent on the primary key.

eg:

RNO	Subject	Faculty
101	DS	Patel
102	DBMS	Shah
103	DS	Jadeja
104	DBMS	Dave
105	DBMS	Shah
102	DS	Patel
101	DBMS	Dave
105	DS	Jadeja

→

Faculty	Subject
Patel	DS
Shah	DBMS
Jadeja	DS
Dave	DBMS

RNO	Subject
101	DS
102	DBMS
103	DS
104	DBMS
105	DBMS
102	DS
101	DBMS
105	DS

v) 4NF (Fourth Normal Form)

> It is in BCNF and has no multi-valued dependencies.

> If it table's attribute has multivalued dependency than it is not in 4NF.

eg: A table can have both functional as well as multivalued dependency together.

• RNO → Address, RNO →→ Subject, RNO →→ Faculty

vi) 5NF (Fifth Normal Form)

> It is in 4NF and it cannot have a lossless decomposition in to any number of smaller tables.

> None relations can further decomposed into sub-relation.

eg:

RID	RNO	Name	Subject	Result
1	101	Tony	DBMS	Pass
2	101	Tony	DS	Pass
3	102	Pepper	DBMS	Pass
4	102	Pepper	DS	Pass
5	103	Jarvis	DBMS	Fail

RNO	Name
101	Tony
102	Pepper
103	Jarvis

SID	Name
1	DBMS
2	DS

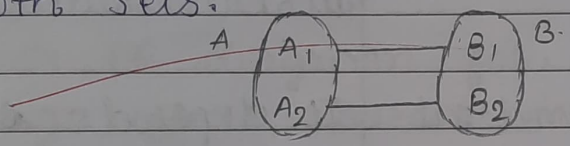
RID	RNO	SID	Result
1	101	1	Pass
2	101	2	Pass
3	102	1	Pass
4	102	2	Pass
5	103	1	Fail

Q.12 Describe different types of mapping cardinality in a binary relationship.

→ Mapping Cardinality, in a binary relationship defines the possible numbers of related instances in one entity set to the number of related instances in another entity set.

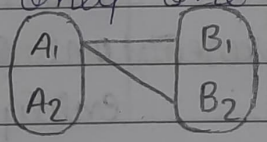
i) One-to-One

> Here, each entity in the source entity set 'A', is related to one and only one entity in the target entity set 'B' and vice-versa. This implies a unique pairing between entities in both sets.



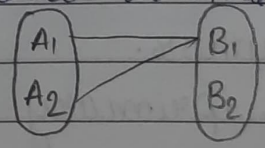
ii) One-to-Many

> Here, each entity in the source entity set 'A' is related to one or more entities in the target entity set 'B', but each entity in 'B' is related to only one entity in 'A'.



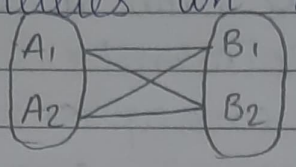
iii) Many-to-One

> Here, each entity in 'A' is related to one entity in 'B', but each entity in 'B' can be related to one or more entities in 'A'.



iii) Many-to-Many

> Here, each entity in 'A' can be related to multiple entities in 'B' and vice-versa.



Q.13 Differentiate strong entity set and weak entity set. Demonstrate the concept of both using real-time example using ER diagram.

Ans.

Strong Entity	Weak Entity
> It always has a primary key	It has a partial discrimination key.
> It is not dependent on any other entity	It depends on strong entity
> Represented by a single rectangle.	Represented by a double rectangle
> Two strong entity's relationship is represented by single diamond.	while the relation b/w one strong and one weak entity by double diamond.

• Example: (Library Database)

i) Strong Entity Set: "Book"

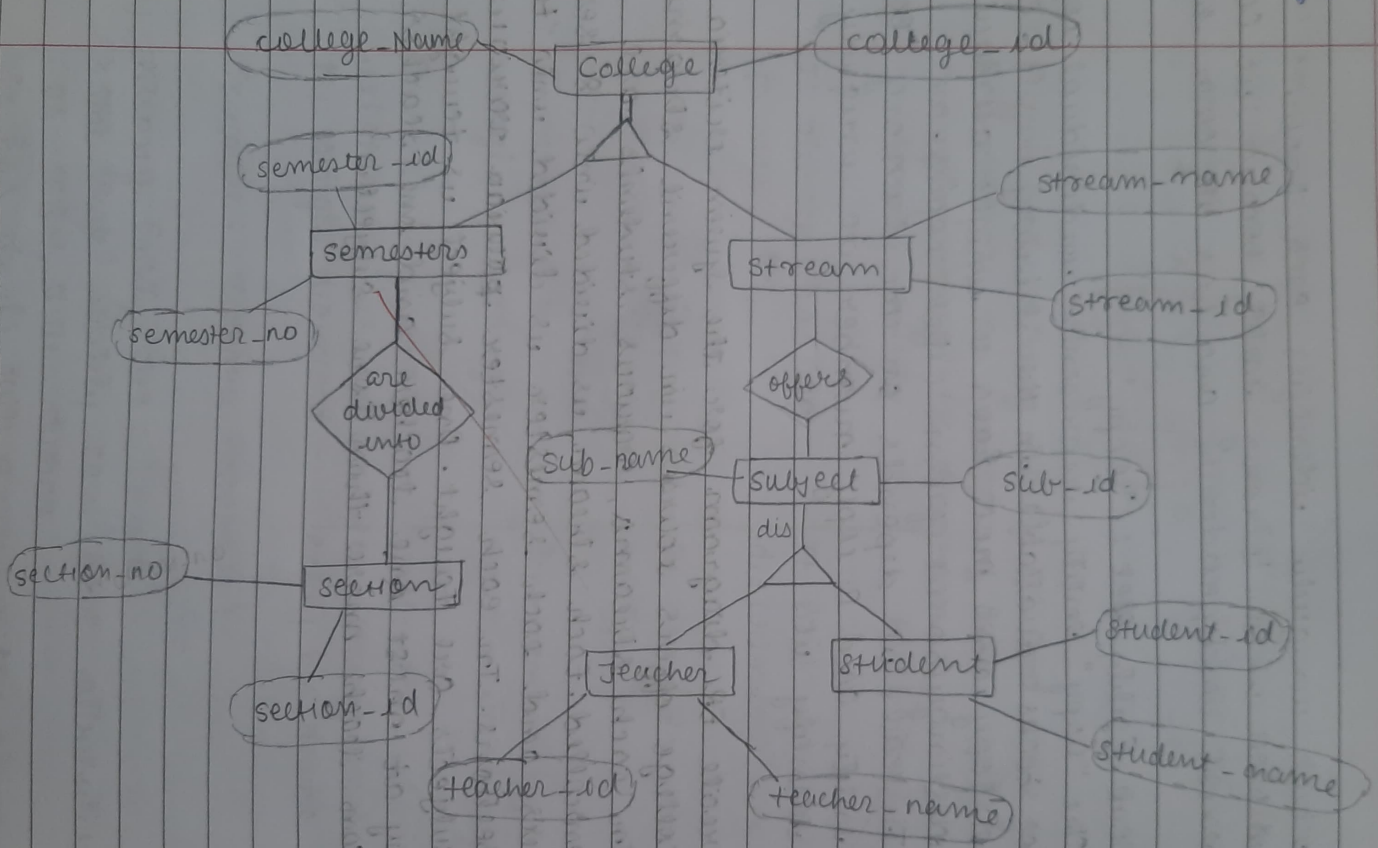
> The "Book" entity can have attributes such as ISBN, Title, Author.

> It has its own primary key attribute

which uniquely identifies each book.
> A book is a strong entity because it can exist independently and has its own library

ii) Weak Entity Set : "Copy".
> The "Copy" entity represents individual copies of books available in the library.
> A copy does not have a unique attribute to distinguish it from other copies.
> Instead, it depends on the owning "Book" entity and a copy number to uniquely identify it.

Q.14 Create ER diagram for the given criteria - College offers courses in different streams. In each streams, various students are enrolled. Each stream is divided into 8 semesters and each semester is divided into two sections. In each semester various subjects are taught. One subject is taught by at most one teacher and one teacher can teach more than one subject.


college_Name
College
college_id
semester_id
semesters
stream_name
stream
stream_id
semester_no
are divided into
offers
sub_name
subject
sub_id
dis
section_no
section
section_id
teacher
student
student_id
student_name
teacher_id
teacher_name

PAGE NO. 47

Q.15 Explain BCNF. How does it differ from 3NF?

-> Same as in Q.11.

	BCNF	3NF
>	The functional dependencies are already in INF, 2NF & 3NF	The functional dependencies are already in INF & 2NF.
>	It is comparatively more stronger than 3NF	It is less stronger than BCNF.
>	The redundancy is low	The redundancy is high.
>	Lossless Decomposition is hard to achieve.	Lossless Decomposition can be achieved.

Q.16 Explain two phase commit protocol with its advantages.

-> The two-phase Commit Protocol is a distributed transaction protocol used in distributed database systems to ensure the atomicity and consistency of transactions involving multiple participants commit or none of them do, avoiding inconsistencies that could arise in distributed environment.

° The two-phases are:

i) The Prepare Phase: The coordinator sends a "prepare to commit" request to all participating nodes. If all nodes are ready to commit, they reply with "Yes".

ii) The Commit Phase : If the coordinator receives "Yes" from all participants, it sends a "commit" message to all participants. If any participant responded with "NO" during the prepare phase, the coordinator sends an "abort" message to all the participants

—4 Advantages :-

> Fault Tolerance : The protocol is designed to handle network failures or crashes of participants.

> Blocking : The protocol prevents blocking by participants

> Scalability : It can be used in a distributed environment involving multiple participants, making it suitable for complex distributed systems

> Consistency : It enforces consistency across distributed systems by ensuring that all nodes reach an agreement on whether to commit or abort a transaction

Q.17 Explain validation - based protocol with an example.

—4 Validation - Based Protocol is used to manage concurrent transactions and ensure data consistency. Transactions do not need to wait for locks, which can improve the system's concurrency. It is also called optimistic concurrency control technique. They rely on timestamps to determine whether a transactions, should be committed or rolled back

eg: Two transactions read an item A with a value of 10. Then T1 updates its local copy of A to 12. T2 updates its local copy of A to 15. T1 reaches the ~~validation~~ validated successfully because no other transaction has written to A since T1 read it.

Q.18 Describe timestamp based locking protocol.

→ Timestamp-based locking protocols are used to ensure that transactions are executed in a consistent and serializable manner. It uses system time to determine the timestamp of the transaction. The older transaction is always given priority in this method. There are two main types of timestamp-based locking protocols: Timestamp Ordering and Thomas' Write Rule.

eg: Suppose, there are transactions T1, T2, T3.
T1 has entered at 0010
T2 has entered at 0020
T3 has entered at 0030
Priority: T1 → T2 → T3.

Q.19 What is mean by Serializability? Explain types of Serializability.

→ Same as Q. 3

Q.20 What is log based recovery? Explain immediate database modification technique for database recovery.

—> Log-Based Recovery is used to ensure the integrity and recoverability of a database in the event of a failure. such as a system crash or an unexpected error. A log is kept on stable storage.

- log contains:
  - > Start of transactions
  - > Transaction-id.
  - > Record-id.
  - > Type of Operation (insert, update, delete)
  - > Old value, new value.
  - > End Transaction that is committed or aborted.

—> The Immediate Database Modification Technique is also known as the "Immediate Update" technique where the database is updated directly during the execution of a transaction and the transaction log records these changes immediately before the changes are applied to the database. This approach provides high transaction throughput and low overhead compared to some other recovery techniques.

Q.21 What is deadlock? Explain three types of actions to be taken for recovery from deadlock.

—> Deadlock is a ~~mutation~~ situation in a multi-process or multi-threaded system where two or

more processes or threads are unable to proceed because each is waiting for the other to release a resource or resources.

→ Three Types of actions for recovery:

i) Detection & Resolution

› This involve periodically checking the system's state for the presence of a deadlock. If a deadlock is detected, the system takes appropriate actions to recover.

ii) Prevention

› This aims to avoid the occurrence of deadlocks entirely. It ensures that the necessary conditions for a deadlock to occur are never met.

iii) Avoidance

› It aims to allow processes to request resources while ensuring that the system remains in a safe state i.e. a state where deadlock cannot occur.

Q.22 Given the relation $R = \{A, B, C, D, E, F, G, H, I, J\}$ and the set of functional dependencies $F = \{\{A, B\} \rightarrow C, A \rightarrow \{D, E\}, B \rightarrow F, F \rightarrow \{G, H\}, D \rightarrow \{I, J\}\}$. What is the key for R? Find out relation presently in which NF?

→ Here, $R = \{A, B, C, D, E, F, G, H, I, J\}$

Functional Dependencies:

$F \Rightarrow AB \rightarrow C,\ A \rightarrow DE,\ B \rightarrow E,\ F \rightarrow GH,\ D \rightarrow IJ$

Now,

$A \to DE$ (given) $\Rightarrow A \to D$ & $A \to E$

Since, $A \to D$ and $D \to IJ$ (given) $\Rightarrow A \to IJ$

using union rule,

$A \to ADEIJ$

Thus, $AB \to ABEIJ$ (augmentation) also

$AB \to C$

$\therefore AB \to ABCDEIJ$

Since, $B \to F$ (given) and $F \to GH$ (given)

$\therefore B \to GH$ (Transitivity)

$\therefore AB \to ABCDEFGHIJ$.

So, AB is the key for R. &

R is in 2NF.

~~~ x ~~~ x ~~~

Signed

25/10/23

V. good
~~~