Question bank - DBMS NOTES

dbms (Anna University)

Scan to open on Studocu

AD3391 DATABASE MANAGEMENT SYSTEMS

QUESTION BANK

UNIT 1 - DATABASE FUNDAMENTALS

PART – A

1. Define database management system (remembering)

Database management system (DBMS) is a collection of interrelated data and a set of programs to access those data.

Examples: Microsoft Access, MySQL, Microsoft SQL Server, Oracle and FileMaker Pro are all examples of database management systems.

2. What is the purpose of DBMS? (remembering) (NOV/DEC 2014)

A database management system is a software tool that makes it possible to organize data in a database.

3. Write the characteristics that distinguish the database approach with the file-based approach. (creating) (APR/MAY 2015)

- Organized/Related
- Shared
- Permanent or Persistence
- Validity/integrity/Correctness
- Security
- Consistency
- Non-redundancy
- Easily Accessible
- Independence
- Recoverable
- Flexible to change

4. List out a few applications of DBMS. (analyzing)(APR/MAY 2019)

a) Banking

b) Airlines

c) Universities

d) Credit card transactions

e) Tele communication

5. What are the disadvantages of a file processing system? (understanding)(MAY/JUNE 2016)

The disadvantages of file processing systems are

a) Data redundancy and inconsistency

b) Difficulty in accessing data

c) Data isolation

d) Integrity problems

e) Atomicity problems

f) Concurrent access anomalies

6. What are the advantages of using a DBMS? (remember)

The advantages of using a DBMS are

a) Controlling redundancy

b) Restricting unauthorized access

c) Providing multiple user interfaces

d) Enforcing integrity constraints.

e) Providing backup and recovery

7) Define data model? (understanding) (APR/MAY 2011)

A data model is a collection of conceptual tools for describing data, data relationships, data semantics, and consistency constraints.

8) What is an entity relationship model? (evaluation)(MAY/JUNE 2016)

The entity-relationship model is a collection of basic objects called entities and relationships among those objects. An entity is a thing or object in the real world that is distinguishable from other objects.

9) What are attributes? Give examples. (remember)

An entity is represented by a set of attributes. Attributes are descriptive properties possessed by each member of an entity set. Example: possible attributes of customer entity are customer name, customer id, Customer Street, and customer city.

10) Define the terms i) Entity set ii) Relationship set (remembering)(APR/MAY 2019)

Entity set: The set of all entities of the same type is termed an entity set.

Relationship set: The set of all relationships of the same type is termed as a relationship set.

11) Define single-valued and multivalued attributes. (remembering)

Single valued attributes: attributes with a single value for a particular entity are called single-valued attributes. Multivalued attributes: Attributes with a set of values for a particular entity are called multivalued attributes.

12) What are stored and derived attributes? (understand) (NOV/DEC 2011)

Stored attributes: The attributes stored in a database are called stored attributes. Derived attributes: The attributes that are derived from the stored attributes are called derived attributes.

13) Define weak and strong entity sets? (remembering)

Weak entity set: entity sets that do not have key attributes of their own are called weak entity sets. Strong entity set: An entity set that has a primary key is termed a strong entity set.

14) What is a data model? (remembering) (NOV/DEC 2011)(APR/MAY 2019)

A Database model defines the logical design of data. The model describes the relationships between

different parts of the data. In the history of database design, three models have been in use.

- Hierarchical Model
- Network Model
- Relational Model

15) What is a storage manager? (understanding)

A storage manager is a program module that provides the interface between the low-level data stored in a database and the application programs and queries submitted to the system.

16) What are the two types of data independence?

1. Logical data independence

2. Physical data independence

17) Who is a DBA? What are the responsibilities of a DBA? (remembering)(APR/MAY 2011)

A database administrator (short form DBA) is a person responsible for the installation, configuration, upgrade, administration, monitoring, and maintenance of databases in an organization. The role includes the development and design of database strategies, system monitoring and improving database performance and capacity, and planning for future expansion requirements. They may also plan, coordinate and implement security measures to safeguard the database.

18) Define single-valued and multivalued attributes. (remembering)

Single valued attributes: attributes with a single value for a particular entity are called single-valued attributes. Multivalued attributes: Attributes with a set of values for a particular entity are called multivalued attributes.

19) What are Enhanced ER features?

The extended E-R features are specialization, generalization, attribute inheritance, and aggregation.

20) What are aggregate functions? And list the aggregate functions supported by SQL? (remembering)

Aggregate functions are functions that take a collection of values as input and return a single value. Aggregate functions supported by SQL are

Average: AVG

Minimum: MIN

Maximum: MAX

Total: SUM

Count: COUNT

Part – B

1. Explain all types of data models.
2. Write about the structure of database system architecture with a block diagram. (view the structure).
3. With help of a neat block diagram explain the architecture of a database management system.
4. Briefly explain about Views of the data.
5. What are the various components of database systems? Explain in detail.
6. Explain the purpose of database systems along with its applications.
7. (i) Brief the history of database systems.
   (ii) Briefly elucidate database users and administrators.
8. What is ER model? Explain the following:
   - Entities, Attributes and Entity sets
   - Relationships and relationship sets
9.
10. (i) What are the advantages and disadvantages of ER modeling.
    (ii) Explain the enhanced ER features.
11. Explain the conceptual design of ER model.
12. Explain Aggregation.

PART – C

1. Draw E-R diagram for the ―Restaurant Menu Ordering System‖, which will facilitate the food items ordering and services within a restaurant. The entire restaurant scenario is detailed as follows. The Customer is able to view the food item menu, call the waiter, place orders and obtain the final bill through the computer kept in their table. The waiters through their wireless tablet PC are able to initialize a table for customers, control the table functions to assist customers, orders send, orders to food preparation staff (chef) and finalize the customer‘s bill. The food preparation staffs (Chefs) with their touch display interfaces to the system, are able to view orders sent to the kitchen by waiters. During preparation, they are able to let the waiter know the status of each item and can send notifications when items are completed. The system should have full accountability and logging facilities and should support Supervisor actions to account for exceptional circumstances such as a meal being refunded or walked out on.

2. Demonstrate the features supported in Enhanced ER Model with your own database.

3. Draw an E-R diagram for a banking enterprise with almost all components and explain.

4. (i) Draw the E-R diagram for bank systems (Home Loan applications)
(ii) Illustrate specialization and generalization with your own example.

UNIT 2 - RELATIONAL DATABASE

Part - A

1. Define instance and schema? (remembering)

Instance: The collection of data stored in the database at a particular moment is called an Instance of the database.
Schema: The overall design of the database is called the database schema.

2. What is a candidate key? (remembering)

Minimal super keys are called candidate keys.

3. What are primary key constraints? (remembering)

A primary key is a constraint defined on a relational database table that prevents users from entering duplicate records into the table. i.e, UNIQUE & NOT NULL

4. What is a super key? (remembering)

A super key is a set of one or more attributes that collectively allows us to identify uniquely an entity in the entity set.

5. What is a SELECT operation? (remembering)

The select operation selects tuples that satisfy a given predicate. We use the lowercase letter ss to denote selection.

6. Define query language? (remembering)

A query is a statement requesting the retrieval of information. The portion of DML that involves information retrieval is called a query language.

7. What is a foreign key? (remembering)

A relation schema r1 derived from an ER schema may include among its attributes the primary key of another relation schema r2.this attribute is called a foreign key from r1 referencing r2.

8. What are the parts of SQL language? (remembering)

The SQL language has several parts:

Data - Definition language
Data Manipulation language
View definition
Transaction control
Embedded SQL
Integrity
Authorization

9. What are the three classes of SQL expression? (remembering)

SQL expression consists of three clauses: Select , From and Where.

10. What are aggregate functions? And list the aggregate functions supported by SQL? (remembering)

Aggregate functions are functions that take a collection of values as input and return a single value.
Aggregate functions supported by SQL are
Average: AVG
Minimum: MIN
Maximum: MAX
Total: SUM
Count: COUNT

11. What is the use of group by clause? (remembering)

Group by clause is used to apply aggregate functions to a set of tuples. The attributes given in the group by clause are used to form groups. Tuples with the same value on all attributes in the group by clause are placed in one group.

12. Describe a circumstance in which you would choose to use embedded SQL rather than using SQL alone. (analyzing)

Writing queries in SQL is typically much easier than coding the same queries in a general-purpose programming language. However not all kinds of queries can be written in SQL.Also no declarative actions such as printing a report, interacting with a user, or sending the results of a query to a graphical user interface cannot be done from within SQL. Under circumstances in which we want the best of both worlds, we can choose embedded SQL or dynamic SQL, rather than using SQL alone or using only a general-purpose programming language. Embedded SQL has the advantage of programs being less complicated since it avoids the clutter of the ODBC or JDBC function calls, but requires a specialized preprocessor.

13. What is embedded SQL? What are its advantages?

Embedded SQL is a method of combining the computing power of a programming language and the database manipulation capabilities of SQL. Embedded SQL statements are SQL statements written inline with the program source code of the host language. The embedded SQL statements are parsed by an embedded SQL preprocessor and replaced by host-language calls to a code library. The output from the preprocessor is then compiled by the host compiler. This allows programmers to embed SQL statements in programs written in any number of languages such as: C/C++, COBOL and FORTRAN.

14. Differentiate Static SQL and Dynamic SQL.

STATIC SQL	DYNAMIC SQL
In static SQL how database will be accessed is predetermined in the embedded SQL statement.	In dynamic SQL, how database will be accessed is determined at run time.
SQL statements are run at compile time	SQL statements are run at execution time
Parsing, validation, optimization, and generation of application plan are done at compile time.	Parsing, validation, optimization, and generation of application plan are done at run time.

15. Define tuple variable? (remembering)

Tuple variables are used for comparing two tuples in the same relation. The tuple Variables are defined in the from clause by way of the as clause.

16. What is relational model?

Relational Model (RM) represents the database as a collection of relations. A relation is nothing but a table of values. Every row in the table represents a collection of related data values. These rows in the table denote a real-world entity or relationship.

17. What are the advantages of Relational Database Model?

Simplicity: A Relational data model in DBMS is simpler than the hierarchical and network model.
Structural Independence: The relational database is only concerned with data and not with a structure. This can improve the performance of the model.
Easy to use: The Relational model in DBMS is easy as tables consisting of rows and columns are quite natural and simple to understand

Query capability: It makes possible for a high-level query language like SQL to avoid complex database navigation.
Data independence: The Structure of Relational database can be changed without having to change any application.
Scalable: Regarding a number of records, or rows, and the number of fields, a database should be enlarged to enhance its usability.

18. **What are relational integrity constraints?**

Relational Integrity constraints in DBMS are referred to conditions which must be present for a valid relation. These Relational constraints in DBMS are derived from the rules in the mini-world that the database represents.
There are many types of Integrity Constraints in DBMS. Constraints on the Relational database management system are mostly divided into three main categories are:

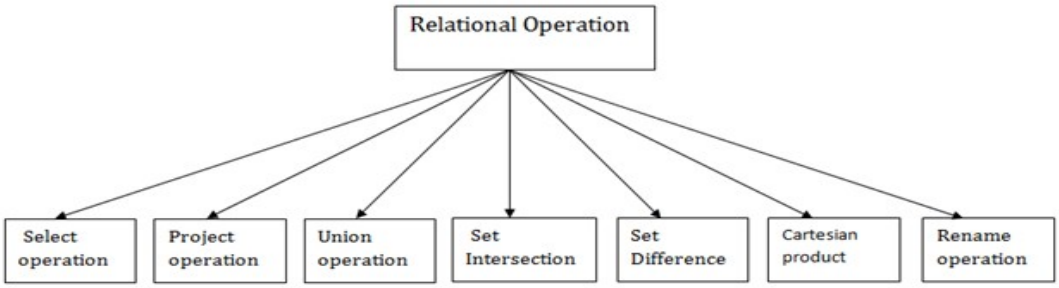
1. Domain Constraints
2. Key Constraints
3. Referential Integrity Constraints

19. Why we need a Key?

Here are some reasons for using sql key in the DBMS system.

- Keys help you to identify any row of data in a table. In a real-world application, a table could contain thousands of records. Moreover, the records could be duplicated. Keys in RDBMS ensure that you can uniquely identify a table record despite these challenges.
- Allows you to establish a relationship between and identify the relation between tables
- Help you to enforce identity and integrity in the relationship.

20. Types of relational operation?



Part B:

1. Define relational algebra. With suitable example.
2. Differentiate between foreign key constraints and referential integrity constraints with suitable example.
3. Describe about embedded SQL.

4. Justify the need of embedded SQL. Consider the relation student (studentno, name, mark and grade). Write embedded dynamic SQL statements in C language to retrieve all the students' records whose mark is more than 90.
5. List the operations of relational algebra and purpose of each with example.

6. Assume the following table. (Understand)
   Degree(degcode,name,subject)
   Candidate(seat no,degcode,name,semester,month,year,result)
   Marks(seatno, degcode,name,semester,month,year,papcode,result)
   Degcode-degcode,Name-name of the degree(Msc.MCOM)
   Subject – subject of the course Eg.Phy,Pap code –paper code eg.A1.
   Solve the following queries using SQL
   (1)Write a SELECT statement to display all the degree codes which are there in the candidate table
   but not present in degree table in the order of degcode. (4)
   (2)Write a SELECT statement to display the name of all the candidates who have got less than 40
   marks in exactly 2 subjects. (4)
   (3)Write SELECT statement to display the name, subject and number of candidates for all degrees
   in which there are less than 5 candidates.(4)
   (4)Write a SELECT statement to display the name of all the candidates who have got highest total
   marks in Ms.,(Maths).

7. (i) Differentiate Schema and instance.

   (ii) Briefly explain keys in DBMS.

8. Explain select,project and Cartesian product operations in relational algebra with an example.

9. Differentiate between foreign key constraints and referential integrity constrainsts with suitable example.

10. Explain the aggregate functions in SQL with an example

Part – C

1. Consider the following relational schema: Employee(empno,name,office,age) Books(isbn,title,authors,publisher) Loan(empno,isbn,date) Write the following queries in relational algebra.
   (i) Find the names of employees who have borrowed a book Published by XYZ Ltd.,
   (ii) Find the names of employees who have borrowed all books Published by XYZ Ltd.,
   (iii) Find the names of employees who have borrowed more than five different BOOKS Published by XYZ Ltd.,
   (iv) For each Publisher, find the names of employees who have borrowed more than five books of that Publisher.

2. Consider the following relational database

Employee(Employee-Name,street,city)
Works(Employee-Name,Company-Name,Salary)
Company(Company-Name,City)
Manager(Employee-Name,Manager-Name)
Give an SQL DDL definition of this database,Identify referential integrity constraints that should hold,and include them in the DDL definition.

3. Consider the following schema:
   Supplier(sid:integer,sname:string,address:string)
   Parts(pid: integer,pname: string,color: string)
   Catalog(sid: integer,pid:integer,cost:red)
   The key fields are underlined and the domain of each field is listed after the field name.Thereforesid is the key for Suppliers,pid is the key for Parts and sid and pid together form the key for Catalog. The Catalog relation lists the prices charged for parts by suppliers.Write the following queries in relational algebra and SQL.
   (i) Find the sids of suppliers who supply some red or green part
   (ii) Find the sids of suppliers who supply every part.
   (iii) Find the sids of suppliers who supply every red part or supply every green part.
4. Consider the relational table given below and answer the following SQL queries.Employee(Empno, Name, Department, Salary).
   (i) List all the employees whose name starts with the letter 'L'.
   (ii) Find the maximum salary given to employees in each department.
   (iii) Find the number of employees working in the 'accounts' department.
   (iv) Find the second maximum salary from the table.
   (v) Find the employee who is getting the minimum Salary.

5. Consider the following relations for a company Database Application:
   Employee(Eno,Name,Sex,Dob,Doj,Designation,Basic_Pay,Deptno)
   Department(Dept_no,Name)
   Project(Proj_no,Name,Dept_no)
   Worksfor(Eno,Proj_no,Date,Hours)
   The attributes specified for each relation is self-explanatory. However, the business rules are stated as follows. A department can control any number of projects. But only one department can control a project. An employee can work on any number of projects in a day. However, an employee cannot work more than once on a project he or she worked on that day. The primary keys are underlined.
   (i) Identify the foreign keys. Develop DDL to implement the above schema.
   (ii) Develop an SQL query to list the department number and the number of employees in each department.
   (iii) Develop a view that will keep track of the department number, the number of employees in the department, and the total basis pay expenditure for each department.
   (iv) Develop an SQL query to list the details of employees who have marked in more than three projects on a day.

UNIT 3 - DATABASE DESIGN

PART -A

1. **What is meant by functional dependencies? (remembering)**

Consider a relation schema R and a C R and ß C R. The functional dependency a ß holds on relational schema R if in any legal relation r(R), for all pairs of tuples t1 and t2 in r such that t1 [a] =t1 [a], and also t1 [ß] =t2 [ß].

2. **What is meant by computing the closure of a set of functional dependencies? (remembering)**

+ The closure of F denoted b y F is the set of functional dependencies logically implied by F.

3. **What is meant by normalization of data? (remembering)**

It is a process of analyzing the given relation schemas based on their Functional Dependencies (FDs) and primary key to achieve the properties Minimizing redundancy, Minimizing insertion, deletion and updating anomalies.

4. **Define Boyce codd normal forms(evaluating)**

A relation schema R is in BCNF with respect to a set F of functional + dependencies if, for all functional dependencies in F of the form. a->ß.

5. **Define normalization. (remembering)**

Normalization of data is a process during which unsatisfactory relation schemas are decomposed by breaking up their attributes into smaller relation schemas that possess desirable properties.

6. **What is 1NF? (remembering)**

The domain of attribute must include only atomic (simple, indivisible) values.

7. **What is 2NF? (remembering)**

A relation schema R is in 2NF if it is in 1NF and every non-prime attribute An in R is fully functionally dependent on primary key.

8. **What is 3NF?**

A relation shema R is in 3NF if it is in 2NF and no nonprime attribute of R is transitively dependent on the primary key. A funtional dependency X → Y in arelation shema R is a transitive dependency if ther is a set of attributes Z that is not a subset of any key of R, and both X→Y and Z→Y hold.

9. Define multivalued dependency. (remembering)

Multivalued dependencies: Multivalued dependencies are a result of 1NF which disallowed an attribute in a tuple to have a set of values. If we have two or more multivalued independent attributes in the same relation schema, we get into the problem of having to repeat every value of one of the attributes with every value of the other attributes with every value of the other attribute to keep the relation instances consistent.

10. What is the need for normalization? (remembering)

To ensure that the update anomalies do not occur.

- Normal forms provide a formal frame work for analyzing relation schemas based on their keys and on the functional dependencies among their attributes.
- A series of tests that can be carried out on individual relation schemas so that the relation database can be normalized to any degree.
- When a test fails, the relation violating that test must be decomposed into relations that individually meet the normalization tests.

11. In what way BCNF is different from 3NF? (remembering)

A relation schema R is in BCNF if whenever a functional dependency X→Y holds in R,then X is a superkey of R.The only difference between BCNF and 3NF : the 3NF allows A to be a prime if X is not a superkey, is absent from BCNF.

12. Show that, if a relational database is in BCNF, then it is also in 3NF.

This is because in many cases, there exists no database schema that is both BCNF and dependence preserving. If one prefers to have a dependence preserving database schema, then one has to choose a normal form, such as 3NF, that is weaker than BCNF.

13. Why are certain functional dependencies called as trivial functional dependencies?

Because the right-hand side is a subset of the left-hand side. Therefore, it is obvious that the right-hand side is dependent on the left-hand side. More unclear book definition: An FD is trivial if it is satisfied by all instances of a relation.

14.Define Denormalization(remembering)

Denormalization is a strategy that database managers use to increase the performance of a database infrastructure. It involves adding redundant data to a normalized database to reduce certain types of problems with database queries that combine data from various tables into a single table. The definition of denormalization is dependent on the definition of normalization, which is defined as the process of organizing a database into tables correctly to promote a given use.

15. Why 4NF is more desirable than BCNF? (remembering) (NOV/DEC 2014)

4NF is more desirable than BCNF because it reduces the repetition of information. If we consider a BCNF schema not in 4NF, we observe that decomposition into 4NF does not lose information provided that a lossless join decomposition is used, yet redundancy is reduced.

16. Define: Functional dependency. (remembering) (APR/MAY 2015)

A functional dependency is a constraint between two sets of attributes in a relation from a database. In other words, functional dependency is a constraint that describes the relationship between attributes in a relation.

17. State the anomalies of 1NF. (clarity) (NOV/DEC 2015)

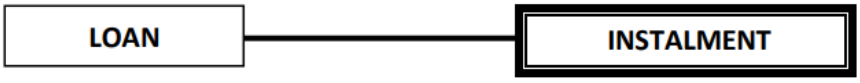
*INSERT anomalies
*UPDATE anomalies
*DELETE anomalies

18. Is it possible for several attributes to have the same domain? Illustrate your answer with suitable examples.(analyzing)

Yes , It is possible for several attributes to have the same domain. The attributes indicate different roles, for the domain. For example, in the STUDENT relation, the same domain Local_phone_numbers plays the role of Home_phone referring to the home phone of a student and the role of office phone, referring to the office phone of the student.

19. What is weak entity?Give example.(remembering)

Weak entity is an entity that depends on another entity. Weak entity doen't have key attribute of their own. Double rectangle represents weak entity.



20. What are the desirable properties of decomposition?

1. Lossless Decomposition.

2. Dependency Preservation- Dependency is an important constraint on the database.

3. Lack of Data Redundancy - Lack of Data Redundancy is also known as a Repetition of Information.

PART – B

1. What is normalization? Explain in detail about all Normal forms.
2. Briefly discuss about the functional dependency concepts.
3. Distinguish between lossless-join decomposition and dependency reserving decomposition.

4. What is Normalization? Explain First normal form. second normal form and third normal with an example,
5. i) Summarize the term anomalies. Explain BCNF in detail.(07)
   ii) Decide why BCNF is used and how it differs from 3 NF?(06)
6. (i) Analyze about lossless Decomposition.(07)
   (ii) Design your own database to illustrate 3NF.(06)
7. Explain about Functional Dependencies and its impact on the data base.(13)
8. Write short Notes on.
   (i) Non loss decomposition(6)
   (ii) Lossy decomposition(7)
9. Write short Notes on.
   (i) Join Dependencies(7)
   (ii) 5 th Normal Form(6)
10. (i) Explain the properties of Decomposition.
    (ii) Briefly elucidate Schema refinement.
11. Explain the following in brief:
    i. Dependency preservation.
    ii. Multivalued dependencies.
    iii. Join Dependencies.

PART-C

1.Look at this database, which is in 1NF, to see what you must do to put that database in DK/NF.

SALES

Customer_ID	Product	Price
1001	Laundry detergent	12
1007	Toothpaste	3
1010	Chlorine bleach	4
1024	Toothpaste	3

2. Consider the universal relation R={A,B,C,D,E,F,G,H,I} and the set of functional dependencies F={(A,B)->{C},{A}- >{D,E},{B}->{F},{F}->{G,H},{D}->{I,J}}.What is the key role for R? Decompose R into 2NF then 3NF relations.

3. Give an example of a relation that is in 3NF but not in BCNF. How will you convert that relation in to BCNF.(15)

4. State the need for Normalization of a database and explain the various Normal Forms(1st,,2nd,3rd,BCNF,4th,5th and Domain Key)with suitable examples.

UNIT 4: TRANSACTION MANAGEMENT

PART A

1. What is transaction? (remembering)

Collections of operations that form a single logical unit of work are called transactions.

2. What are the two statements regarding transaction? (remembering)

The two statements regarding transaction of the form:

o Begin transaction 2. End transaction

3. What are the properties of transaction? Define ACID properties.

The properties of transactions are:

Atomicity

Consistency

Isolation

Durability

4. What is recovery management component? (remembering)

Ensuring durability is the responsibility of a software component of the base system called the

recovery management component.

5. When is a transaction rolled back? (remembering)

Any changes that the aborted transaction made to the database must be undone. Once the changes caused by an aborted transaction have been undone, then the transaction has been rolled back.

6. What are the states of transaction? (remembering)(APR/MAY 2019)

The states of transaction are

Active
Partially committed
Failed
Aborted
Committed
Terminated

7. List out the statements associated with a database transaction? (analyzing)

Commit work
Rollback work

8. What is a shadow copy scheme? (remembering)

It is simple, but efficient, scheme called the shadow copy schemes. It is based on making copies of the database called shadow copies that one transaction is active at a time. The scheme also assumes that the database

9. Give the reasons for allowing concurrency? (understanding)

The reasons for allowing concurrency is if the transactions run serially, a short transaction may have to wait for a preceding long transaction to complete, which can lead to unpredictable delays in running a transaction. So concurrent execution reduces the unpredictable delays in running transactions.

10. What is average response time?

The average response time is that the average time for a transaction to be completed after it has been submitted.

11. What are the two types of serializability?

The two types of serializability is 1. Conflict serializability, 2. View serializability.

12. Define lock?

Lock is the most common used to implement the requirement is to allow a transaction to access a data item only if it is currently holding a lock on that item.

13. What are the different modes of lock? (remembering)

The modes of lock are:

1. Shared Lock
2. Exclusive Lock

14. Define deadlock? (remembering)

Neither of the transaction can ever proceed with its normal execution. This situation is called deadlock.

15. Define the phases of two phase locking protocol.

Growing phase: A transaction may obtain locks but not release any lock.
Shrinking phase: A transaction may release locks but may not obtain any new locks.

16. What are the two methods for dealing deadlock problem? (remembering)

The two methods for dealing deadlock problem is deadlock detection and deadlock recovery.

17. Differentiate strict two-phase locking protocol and rigorous two phase locking protocol.

In strict two phases locking protocol all exclusive mode locks taken by a transaction is held until that transaction commits. Rigorous two-phase locking protocol requires that all locks be held until the transaction commits.

18. What benefit does strict two-phase locking provide? What disadvantages result?

Because it produces only cascade less schedules, recovery is very easy. But the set of schedules obtainable is a subset of those obtainable from plain two-phase locking, thus concurrency is reduced.

19. List the two commonly used Concurrency Control techniques

1. Two Phase Locking

2. Serialization

20. List the SQL statements used for transaction control.

COMMIT

ROLLBACK

SAVEPOINT

SET TRANSACTION

SET CONSTRAINT

All transaction control statements, except certain forms of the COMMIT and ROLLBACK commands, are supported in PL/SQL.

21. **Write the ACID properties of the Transaction.**

    Atomicity
    Consistency
    Isolation
    Durability

22. What is meant by concurrency control?

Concurrency control is a database management systems (DBMS) concept that is used to address conflicts with simultaneous accesses.

23. Give an example of a Two-phase commit protocol.

A two-phase commit is a standard protocol in distributed transactions for achieving ACID properties. Each transaction has a coordinator who initiates and coordinates the transaction. For example, participants will block resource processes while waiting for a message from the coordinator. If for any reason this fails, the participant will continue to wait and may never resolve its transaction. Therefore, the resource could be blocked indefinitely. On the other hand, a coordinator will also block resources while waiting for replies from participants. In this case, a coordinator can also block indefinitely if no acknowledgement is received from the participant.

24.What is serializability?(NOV/DEC 2016) (remembering)

Serializability is the classical concurrency scheme. It ensures that a schedule for executing concurrent transactions is equivalent to one that executes the transactions serially in some order. It assumes that all accesses to the database are done using read and write operations.

25.List the four conditions for deadlock.(NOV/DEC 2016) (analyzing)

- Mutual Exclusion
- Hold and Wait
- No Preemption
- Circular Wait

26. Why DBMS needs a concurrency control? NOV 2017

In general, concurrency control is an essential part of Transaction management. It is a mechanism for correctness when two or more database transactions that access the same data or data set are executed concurrently with time overlap. According to Wikipedia.org, if multiple transactions are executed serially or sequentially, data is consistent in a database. However, if concurrent transactions with interleaving operations are executed, some unexpected data and inconsistent result may occur. Data interference is usually caused by a write operation among transactions on the same set of data in DBMS.

27.What is isolation Level?(Remember)

Isolation levels defines the degree to which a transaction must be isolated from the data modifications made by any other transaction in the database system. A transaction isolation level are defined by the following phenomena –

 Dirty Read

 Non-Repatable read

 Phantom Read

Based on these phenomena, The SQL standard defines four isolation levels :

 Read Uncommitted

 Read Committed

 Repeatable Read

 Serializable

28. What do you mean by phantom problem? (Remember)

If transactions operate at less than the maximum isolation level is the so-called phantom problem.

29.What is the need for save points? (Understand)

It might be possible for a transaction to establish intermediate save points while it is executing, and subsequently to roll back to a previously established save point, if required, instead of having to roll back all the way to the beginning.

30. What are the three problems that any concurrency control mechanism must address? (R)

The three problems are:

← The lost update problem

← The uncommitted dependency problem

← The inconsistent analysis problem

31.What is the last update problem? (Understand)

Transaction A retrieves some tuple t at time t1; transaction B retrieves that same tuple t at time t2; transaction A updates the tuple at time t3; and transaction B updates the same tuple at time t4; Transaction A's update is lost at time t4, because transaction B overwrites it with-out even looking at it.

32.What is the uncommitted dependency problem? (Understand)

The uncommitted dependency problem arises if one transaction is allowed to retrieve-or, worse, update-a tuple that has been updated by another transaction but not yet committed by that other transaction.

33. Define atomicity and consistency.(Remember)

Atomicity means that you can guarantee that all of a transaction happens, or none of it does; you can do complex operations as one single unit, all or nothing, and a crash, power failure, error, or anything else won't allow you to be in a state in which only some of the related changes have happened. Consistency means that you guarantee that your data will be consistent; none of the constraints you have on related data will ever be violated.

34. Define isolation and durability.(Remember)

Isolation means that one transaction cannot read data from another transaction that is not yet completed. If two transactions are executing concurrently, each one will see the world as if they were executing sequentially, and if one needs to read data that is written by another, it will have to wait until the other is finished. Durability means that once a transaction is complete, it is guaranteed that all of the changes have been recorded to a durable medium (such as a hard disk), and the fact that the transaction has been completed is likewise recorded.

35. What is serializable schedule?

A schedule is called serializable whenever executing the transactions sequentially, in some order, could have left the database in the same state as the actual schedule. Serializability is the commonly accepted criterion for correctness.

36.What type of locking needed for insert and delete operations?

There are two types of Locks

1. Shared lock

2. Exclusive lock

37.What is meant by log based Recovery?

1. The log is a sequence of records. Log of each transaction is maintained in some stable storage so that if any failure occurs, then it can be recovered from there.

2. If any operation is performed on the database, then it will be recorded in the log.

3. But the process of storing the logs should be done before the actual transaction is applied in the database. There are two approaches to modify the database:

 Deferred database modification

 Immediate database modification

PART – B

1. (a) How Transactions are possible in Distributed database? Explain briefly.
   b. What is Transaction state and its ACID properties?
   2. How can we achieve concurrency control achieved in DBMS through Serializability?
3. (a) What is deadlock prevention and dead lock detection method.
   (b) Explain the deadlock recovery technique.
4. Explain the following protocols for concurrency control.
i. Lock based protocols
ii. ii) Time stamp based protocols DATABASE MANAGEMENT SYSTEM.
5. Explain the concepts of serializability.
6. (i) Explain Two-phase locking protocol.
   (ii) Describe about the deadlock prevention schemes.
7. Write down in detail about Deadlock and Serializability.
8. Discuss in detail about transaction concepts and two phase commit protocol.
9. Write down in detail about intent locking and isolation levels.
10. Illustrate dead lock and conflict serializability with suitable example.
11. What is Concurrency? Explain it in terms of locking mechanism and two phase locking protocol.
12. What is deadlock? How does it occur? How transactions be written to
   (i) Avoid deadlock.
   (ii) Guarantee correct execution. Illustrate with suitable example.
13. Explain about Locking Protocols.
14. Briefly explain about Two phase commit.
15. Explain various recovery techniques during transaction in detail.

PART –C

1. Consider the following two transactions:(Nov/Dec 2016) **(analyzing)**

```
T1: read(A);
read(B);
if A = 0 then B := B + 1;
write(B).
T2: read(B);
read(A);
if B = 0 then A := A + 1;
write(A).
```

Add lock and unlock instructions to transactions T1 and T2, so that they on serve the two-phase locking protocol.Can the execution of these transactions result in a deadlock?

2.Discuss in detail about the ACID properties of a transaction.(Remember))(APR/MAY 2019)

3.What is concurrency control? How it is implemented in DBMS?Briefly elaborate with suitable diagrams and examples. **(Remember)** (Apr/May 2019)

4. Consider the following extension to the tree-locking protocol, which allows both shared and exclusive locks.

• A transaction can be either a read-only transaction, in which case it can request only shared locks, or an update transaction, in which case it can request only exclusive locks. (Remember)

• Each transaction must follow the rules of the tree protocol. Read-only transactions may lock any data item first, whereas update transactions must lock the root first.

Show that the protocol ensures serializability and deadlock freedom.

5. Explain why timestamp-based concurrency control allows schedules that are not recoverable. Describe how it can be modified through buffering to disallow such schedules.
6. What is meant by semantic query optimization? How does it differ from with example?
7. Discuss in detail about the testing of serializability.
8. Explain deferred and immediate modification versions of the log-based recovery scheme.

UNIT 5- IMPLEMENTATION TECHNIQUES AND NON-RELATIONAL MODEL

PART- A

1. **Define seek time?(Knowledge)**

   Seek time is the time to reposition the head and increases with the distance that the head must move. Seek times can range from 2 to 30 milliseconds. Average seek time is the average of all seek times and is normally one-third of the worst-case seek time.

2. **Define RAID and its types?**

   RAIDs are Redundant Arrays of Inexpensive Disks. There are six levels of organizing these disks:
   - 0 -- Non-redundant Striping
   - 1 -- Mirrored Disks
   - 2 -- Memory Style Error Correcting Codes
   - 3 -- Bit Interleaved Parity

 4 -- Block Interleaved Parity
 5 -- Block Interleaved Distributed Parity
 6 -- P + Q Redundancy

3. What is buffer manager? (Knowledge)

Programs in a DBMS make requests (that is, calls) on the buffer manager when they need a block from a disk. If the block is already in the buffer, the requester is passed the address of the block in main memory. If the block in not in the buffer, the buffer manager first allocates space in the buffer for the block, through out some other block, if required, to make space for the new block. If the block that is to be thrown out has been modified, it must first be written back to the disk. The internal actions of the buffer manager are transparent to the programs that issue disk-block requests.

4. Differentiate fixed length and variable length records?(Apply)

A file where all the records are of the same length is said to have fixed length records. Advantage: Access is fast because the computer knows where each record starts. Eg If each record is 120 bytes long then the 1st record starts at [Start of File] + 0 bytes the 2nd record starts at [Start of File] + 120 bytes the 3rd record starts at [Start of File] + 240 bytes etc..... Disadvantage: Using Fixed length records, the records are usually larger and therefore need more storage space and are slower to transfer (load or save).

One or more of the fields can be of differing lengths in each record, called variable length records Advantages: the records will be smaller and will need less storage space the records will load faster. Disadvantages: The computer will be unable determine where each record starts processing the records will be slower.

5. What is heap file and sequential file organization? (Remember)

Heap File Organization
Any record can be placed anywhere in the file. There is no ordering of records and there is a single file for each relation.
Sequential File Organization
Records are stored in sequential order based on the primary key.

6. **Define data dictionary? (Knowledge**)

A RDBMS needs to maintain data about the relations, such as the schema. This is stored in a data dictionary (sometimes called a system catalog):
 Names of the relations
 Names of the attributes of each relation
 Domains and lengths of attributes
 Names of views, defined on the database, and definitions of those views

 Integrity constraints
 Names of authorized users
 Accounting information about users
- Number of tuples in each relation
 Method of storage for each relation (clustered/non-clustered)
 Name of the index
 Name of the relation being indexed
 Attributes on which the index in defined
 Type of index formed

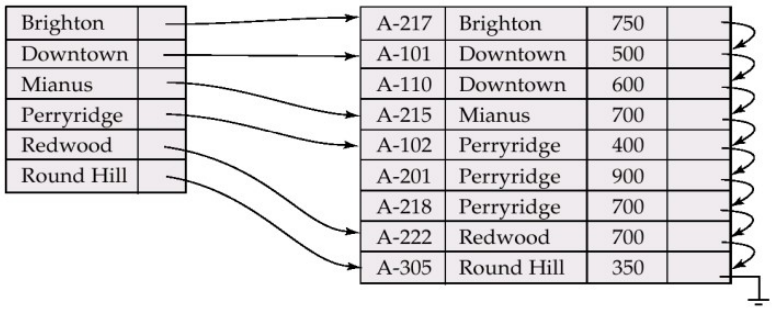
7. Explain indexing and hashing. (Understand)

Indexing mechanisms used to speed up access to desired data. E.g., author catalog in library Search Key - attribute to set of attributes used to look up records in a file. An index file consists of records (called index entries) of the form. Index files are typically much smaller than the original file.
Two basic kinds of indices:
Ordered indices: search keys are stored in sorted order
Hash indices: search keys are distributed uniformly across "buckets" using a "hash function".

8. What is sparse index? (Knowledge)

Sparse Index Files
Index records for some search-key values. To locate a record with search-key value K we:
Find index record with largest search-key value < K
Search file sequentially starting at the record to which the index record points Less space and less maintenance overhead for insertions and deletions. Generally slower than dense index for locating records.
Good tradeoff: sparse index with an index entry for every block in file, corresponding to least search-key value in the block.

9. What is dense index? (Remember)(APR/MAY 2019)
Dense index — Index record appears for every search-key value in the file.



10. What are the advantages and disadvantages of B+ tree?
Advantage of B+ -tree index files:
Automatic self-reorganization with small, local, changes, in the face of insertions and deletions. Reorganization of entire file is not required
Disadvantage of B+ -trees:
Extra insertion and deletion overhead, space overhead

11. What are the properties of B- Trees? (Remember)

B-tree properties: each node, in a B-tree of order n :

- key order
- at most n pointers
- at least n/2 pointers (except root)
- all leaves at the same level
- if number of pointers is k, then node has exactly k-1 keys

12. Comparison between static hashing and dynamic hashing?

Static hashing

Static hashing uses a h ash function in which the set of bucket adders is fixed. Such hash functions cannot easily accommodate databases that grow larger over time.

Dynamic hashing

Dynamic hashing allows us to modify the hash function dynamically. Dynamic hashing copes with changes in database size by splitting and coalescing buckets as the database grows and shrinks.

13. What is called query processing? (remembering)

Query processing refers to the range of activities involved in extracting data from a database.

14. What are the steps involved in query processing? (remembering)

The basic steps are:

Parsing and translation

Optimization and Evaluation

15. Define query optimization.

Query optimization refers to the process of finding g the lowest –cost method of evaluating a given query.

16. What are the disadvantages of B Tree over B+ Tree?

Disadvantages of B+ Tree

This method is less efficient for static tables.

Main disadvantage is that performance degrades as file size grows for lookups.

B+ Tree indexing maintain efficiency despite insertion and deletion of data.

Disadvantages of B Tree

Leaf and non-leaf nodes are of different size (complicates storage)

Deletion may occur in a non-leaf node (more complicated)

17. What is mirroring?(Remember)

The simplest approach to introducing redundancy is to duplicate every disk. This technique is called mirroring.

18. Mention all operations in files.

Operations on database files can be broadly classified into two categories
 Update -Update operations change the data values by insertion, deletion, or update.
 Retrieval - Retrieval operations, on the other hand, do not alter the data but retrieve them after optional conditional filtering. Creation and deletion of a file, there could be several operations, which can be done on files.
 Open
 Locate
 Read
 Write
 Close

19. Compare the Sequential access device versus Random access device with an example.

Function	Random Access (DISK)	Sequential Access (FILE)
Storage space allocation and tracking	Disk blocks	Volumes
Concurrent volume access	A volume can be accessed concurrently by different operations	A volume can be accessed concurrently by different operations
Client restore operations	One session per restore	Multiple concurrent sessions access different volumes simultaneously on both the server and the storage agent. Active versions of client backup data is collocated in active-data pools.

20. What is a query Execution plan?

The Query Execution Plans describe the steps and the order used to access or modify data in the database. Once you have this information you can identify what parts of the query are slow.

SQL Server can create execution plans in two ways:

- Actual Execution Plan - (CTRL + M) - is created after execution of the query and contains the steps that were performed

- Estimated Execution Plan - (CTRL + L) - is created without executing the query and contains an approximate execution plan

Execution plans can be presented in these three ways and each option offers benefits over the other.

- Text Plans
- Graphical Plans
- XML Plans

When beginning to work with execution plans, the graphical plan is usually the easiest place to start unless your plan is very complex, then the text plans are sometimes easier to read.

PART – B

1. Explain about RAID system. How does it improve performance and reliability. Discuss the level 3 and level 4 of RAID.
2. (i) Describe the index schemas used in databases. (07)
   (ii) Since indices speed query processing, why might they not be kept on several search keys? List as many reasons as possible. (06)
3. Describe the different types of file organization? Explain using a sketch of each of them with their advantages and disadvantages. (13)
4. Give a detailed description about Query Processing and Optimization. Explain the cost estimation of Query Optimization.
5. Discuss briefly about B+ tree index file with example. (07)
   How does a B-tree differ from a B+ - tree? why is a B+-tree usually preferred as an access structure to a data file?(06)
6. (i) Illustrate indexing techniques with suitable examples (07)
   (ii) Write notes on Hashing.(06)
7. What is meant by semantic query optimization? How does it differ from other query optimization technique? Give example. (13)
8. Describe about B tree index file with example.(13)
9. Explain the distinction between static and dynamic hashing. Discuss the relative merits of each technique in database applications
10. Develop a B+ tree to insert the following key elements(order of the tree 3)5,3,4,9,7,15,14,21,22,23. (13).
11. Explain the MongoDB data types and CRUD Operations.
12. Briefly explain the MongoDB architecture and mention the advantages of MongoDB.

PART -C

1. Construct B tree and B+ tree to insert the following key values(the order of the tree is three) 32,11,15,13,7,22,15,44,67,4.
2. The following key values are organized in an extendable hashing technique.
   1 3 5 8 9 12 17 28 Show the extendable hash structure for this file if the hash function is h(x)=x rod 8 and buckets can hold three records.
   Show how the extendable hash structure changes as the result of each of the following steps:
   INSERT 2
   INSERT 24
   DELETE 5
   DELETE 12.

3. What is query optimization? Explain the steps in query optimization.
4. With suitable diagrams,Explain in detail about the RAID levels(level 0,level 1,level 0+1,level 3,level 4,level.
5. Construct B+ tree to insert the following (order of the tree is 3) 26, 27, 28, 3, 4, 7, 9, 46, 48, 51, 2, 6.
6. Illustrate indexing and hashing techniques with suitable examples.