

CS1301-DATABASE MANAGEMENT SYSTEMS

TWO MARKS:

UNIT: 1

INTRODUCTION AND CONCEPTUAL MODELLING

1. Define database management system?

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

2. List any eight applications of DBMS.

- a) Banking
- b) Airlines
- c) Universities
- d) Credit card transactions
- e) Tele communication
- f) Finance
- g) Sales
- h) Manufacturing
- i) Human resources

3. What are the disadvantages of file processing system?

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

4. What are the advantages of using a DBMS?

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 back up and recovery

5. Give the levels of data abstraction?

- a) Physical level
- b) logical level
- c) view level

6. Define instance and schema?

Instance: Collection of data stored in the data base at a particular moment is called an Instance of the database.

Schema: The overall design of the data base is called the data base schema.

7. Define the terms 1) physical schema 2) logical schema.

Physical schema: The physical schema describes the database design at the physical level, which is the lowest level of abstraction describing how the data are actually stored.

Logical schema: The logical schema describes the database design at the logical level, which describes what data are stored in the database and what relationship exists among the data.

8. What is conceptual schema?

The schemas at the view level are called subschemas that describe different views of the database.

9. Define data model?

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

10. What is storage manager?

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.

11. What are the components of storage manager?

The storage manager components include

- a) Authorization and integrity manager
- b) Transaction manager
- c) File manager
- d) Buffer manager

12. What is the purpose of storage manager?

The storage manager is responsible for the following

- a) Interaction with the file manager
- b) Translation of DML commands into low level file system commands
- c) Storing, retrieving and updating data in the database

13. List the data structures implemented by the storage manager.

The storage manager implements the following data structures

- a) Data files
- b) Data dictionary
- c) indices

14. What is a data dictionary?

A data dictionary is a data structure which stores meta data about the structure of the database i.e. the schema of the database.

15. What is an entity relationship model?

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

16. What are attributes? Give examples.

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, customer city.

17. What is relationship? Give examples

A relationship is an association among several entities.

Example: A depositor relationship associates a customer with each account that he/she has.

18. Define the terms

- i) **Entity set**
- ii) **Relationship set**

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

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

19. Define single valued and multivalued attributes.

Single valued attributes: attributes with a single value for a particular entity are called single valued attributes.

Multivalued attributes: Attributes with a set of value for a particular entity are called multivalued attributes.

20. What are stored and derived attributes?

Stored attributes: The attributes stored in a data base are called stored attributes.

Derived attributes: The attributes that are derived from the stored attributes are called derived attributes.

21. What are composite attributes?

Composite attributes can be divided in to sub parts.

22. Define null values.

In some cases a particular entity may not have an applicable value for an attribute or if we do not know the value of an attribute for a particular entity. In these cases null value is used.

23. Define the terms

- i) **Entity type**
- ii) **Entity set**

Entity type: An entity type defines a collection of entities that have the same attributes.

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

24. What is meant by the degree of relationship set?

The degree of relationship type is the number of participating entity types.

25. Define the terms

i) **Key attribute**

ii) **Value set**

Key attribute: An entity type usually has an attribute whose values are distinct from each individual entity in the collection. Such an attribute is called a key attribute.

Value set: Each simple attribute of an entity type is associated with a value set that specifies the set of values that may be assigned to that attribute for each individual entity.

26. Define weak and strong entity sets?

Weak entity set: entity set that do not have key attribute of their own are called weak entity sets.

Strong entity set: Entity set that has a primary key is termed a strong entity set.

27. What does the cardinality ratio specify?

Mapping cardinalities or cardinality ratios express the number of entities to which another entity can be associated. Mapping cardinalities must be one of the following:

- One to one
- One to many
- Many to one
- Many to many

28. Explain the two types of participation constraint.

- **Total:** The participation of an entity set E in a relationship set R is said to be **total** if every entity in E participates in at least one relationship in R.
- **Partial:** if only some entities in E participate in relationships in R, the participation of entity set E in relationship R is said to be **partial**.

29. Define the terms

i) **DDL**

ii) DML

DDL: Data base schema is specified by a set of definitions expressed by a special language called a data definition language.

DML: A data manipulation language is a language that enables users to access or manipulate data as organized by the appropriate data model.

30. Write short notes on relational model

The relational model uses a collection of tables to represent both data and the relationships among those data. The relational model is an example of a record based model.

31. Define tuple and attribute

- **Attributes:** column headers
- **Tuple:** Row

32. Define the term relation.

Relation is a subset of a Cartesian product of list domains.

33. Define tuple variable

Tuple variable is a variable whose domain is the set of all tuples.

34. Define the term Domain.

For each attribute there is a set of permitted values called the *domain* of that attribute.

35. What is a candidate key?

Minimal super keys are called *candidate keys*.

36. What is a primary key?

Primary key is chosen by the database designer as the principal means of identifying an entity in the entity set.

37. What is a super key?

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

38. Define- relational algebra.

The relational algebra is a procedural query language. It consists of a set of operations that take one or two relation as input and produce a new relation as output.

39. What is a SELECT operation?

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

40. What is a PROJECT operation?

The project operation is a unary operation that returns its argument relation with certain attributes left out. Projection is denoted by pie (π).

41. Write short notes on tuple relational calculus.

The tuple relational calculation is a non-procedural query language. It describes the desired information without giving a specific procedure for obtaining that information.

A query or expression can be expressed in tuple relational calculus as

$$\{t \mid P(t)\}$$

which means the set of all tuples 't' such that predicate P is true for 't'.

Notations used:

- $t[A] \rightarrow$ the value of tuple 't' on attribute, A
- $t \in r \rightarrow$ tuple 't' is in relation 'r'
- $\exists \rightarrow$ there exists

Definition for 'there exists' (\exists):

$$\exists t \in r(Q(t))$$

which means there exists a tuple 't' in relation 'r' such that predicate Q(t) is true.

- $\forall \rightarrow$ for all

Definition for 'for all' (\forall):

$$\forall t \in r(Q(t))$$

which means Q(t) is true for all tuples 't' in relation 'r'.

- $\Rightarrow \rightarrow$ Implication

Definition for Implication (\Rightarrow):

$P \Rightarrow Q$ means if P is true then Q must be true.

42. Write short notes on domain relational calculus

The domain relational calculus uses domain variables that take on values from an attribute domain rather than values for entire tuple.

43. Define query language?

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

44. Write short notes on Schema diagram.

A database schema along with primary key and foreign key dependencies can be depicted pictorially by schema diagram. Each relation appears as a box with attributes listed inside it and the relation name above it.

45. What is foreign key?

A relation schema r_1 derived from an ER schema may include among its attributes the primary key of another relation schema r_2 . This attribute is called a *foreign key* from r_1 referencing r_2 .

UNIT: 2

RELATIONAL MODEL

1. What are the parts of SQL language?

The SQL language has several parts:

- data - definition language
- Data manipulation language
- View definition
- Transaction control
- Embedded SQL
- Integrity
- Authorization

2. What are the categories of SQL command?

SQL commands are divided in to the following categories:

1. data - definition language
2. data manipulation language
3. Data Query language
4. data control language
5. data administration statements
6. transaction control statements

3. What are the three classes of SQL expression?

SQL expression consists of three clauses:

- Select
- From
- where

4. Give the general form of SQL query?

Select A₁, A₂....., A_n

From R₁, R₂....., R_m

Where P

5. What is the use of rename operation?

Rename operation is used to rename both relations and a attributes.

It uses the as clause, taking the form:

Old-name **as** new-name

6. Define tuple variable?

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.

7. List the string operations supported by SQL?

- 1) Pattern matching Operation
- 2) Concatenation
- 3) Extracting character strings
- 4) Converting between uppercase and lower case letters.

8. List the set operations of SQL?

- 1) Union
- 2) Intersect operation
- 3) The except operation

9. What is the use of Union and intersection operation?

Union: The result of this operation includes all tuples that are either in r1 or in r2 or in both r1 and r2. Duplicate tuples are automatically eliminated.

Intersection: The result of this relation includes all tuples that are in both r1 and r2.

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

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?

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. What is the use of sub queries?

A sub query is a select-from-where expression that is nested within another query. A common use of sub queries is to perform tests for set membership, make set comparisons, and determine set cardinality.

13. What is view in SQL? How is it defined?

Any relation that is not part of the logical model, but is made visible to a user as a virtual relation is called a view.

We define view in SQL by using the **create view** command. The form of the **create view** command is

Create view *v* as <query expression>

14. What is the use of with clause in SQL?

The **with** clause provides a way of defining a temporary view whose definition is available only to the query in which the **with** clause occurs.

15. List the table modification commands in SQL?

- Deletion
- Insertion
- Updates
- Update of a view

16. List out the statements associated with a database transaction?

- Commit work
- Rollback work

17. What is transaction?

Transaction is a unit of program execution that accesses and possibly updates various data items.

18. List the SQL domain Types?

SQL supports the following domain types.

- 1) Char(n)
- 2) varchar(n)
- 3) int
- 4) numeric(p,d)
- 5) float(n)
- 6) date.

19. What is the use of integrity constraints?

Integrity constraints ensure that changes made to the database by authorized users do not result in a loss of data consistency. Thus integrity constraints guard against accidental damage to the database.

20. Mention the 2 forms of integrity constraints in ER model?

- Key declarations
- Form of a relationship

21. What is trigger?

Triggers are statements that are executed automatically by the system as the side effect of a modification to the database.

22. What are domain constraints?

A domain is a set of values that may be assigned to an attribute .all values that appear in a column of a relation must be taken from the same domain.

23. What are referential integrity constraints?

A value that appears in one relation for a given set of attributes also appears for a certain set of attributes in another relation.

24. What is assertion? Mention the forms available.

An assertion is a predicate expressing a condition that we wish the database always to satisfy.

- Domain integrity constraints.
- Referential integrity constraints

25. Give the syntax of assertion?

Create assertion <assertion name>**check**<predicate>

26. What is the need for triggers?

Triggers are useful mechanisms for alerting humans or for starting certain tasks automatically when certain conditions are met.

27. List the requirements needed to design a trigger.

The requirements are

- Specifying when a trigger is to be executed.
- Specify the actions to be taken when the trigger executes.

28. Give the forms of triggers?

- The triggering event can be insert or delete.
- For updated the trigger can specify columns.
- The referencing old row as clause
- The referencing new row as clause
- The triggers can be initiated before the event or after the event.

29. What does database security refer to?

Database security refers to the protection from unauthorized access and malicious destruction or alteration.

30. List some security violations (or) name any forms of malicious access.

- Unauthorized reading of data
- Unauthorized modification of data
- Unauthorized destruction of data.

31. List the types of authorization.

- Read authorization
- Write authorization
- Update authorization
- Drop authorization

32. What is authorization graph?

Passing of authorization from one user to another can be represented by an authorization graph.

33. List out various user authorization to modify the database schema.

- Index authorization
- Resource authorization
- Alteration authorization
- Drop authorization

34. What are audit trails?

An audit trail is a log of all changes to the database along with information such as which user performed the change and when the change was performed.

35. Mention the various levels in security measures.

- Database system
- Operating system
- Network
- Physical
- human

36. Name the various privileges in SQL?

- Delete
- Select
- Insert
- update

37. Mention the various user privileges.

- All privileges directly granted to the user or role.
- All privileges granted to roles that have been granted to the user or role.

38. Give the limitations of SQL authorization.

- The code for checking authorization becomes intermixed with the rest of the application code.
- Implementing authorization through application code rather than specifying it declaratively in SQL makes it hard to ensure the absence of loopholes.

39. Give some encryption techniques?

- DES
- AES
- Public key encryption

40. What does authentication refer?

Authentication refers to the task of verifying the identity of a person.

41. List some authentication techniques.

- Challenge response scheme
- Digital signatures
- Nonrepudiation

42. Define Boyce codd normal form

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 $\alpha \rightarrow \beta$, where α

43. List the disadvantages of relational database system

- Repetition of data
- Inability to represent certain information.

44. What is first normal form?

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

45. What is meant by functional dependencies?

Consider a relation schema R and $\alpha \subseteq R$ and $\beta \subseteq R$. The functional dependency $\alpha \rightarrow \beta$ holds on relational schema R if in any legal relation $r(R)$, for all pairs of tuples t_1 and t_2 in r such that $t_1[\alpha] = t_2[\alpha]$, and also $t_1[\beta] = t_2[\beta]$.

46. What are the uses of functional dependencies?

- To test relations to see whether they are legal under a given set of functional dependencies.
- To specify constraints on the set of legal relations.

47. Explain trivial dependency?

Functional dependency of the form $\alpha \rightarrow \beta$ is trivial if $\beta \subseteq \alpha$. Trivial functional dependencies are satisfied by all the relations.

48. What are axioms?

Axioms or rules of inference provide a simpler technique for reasoning about functional dependencies.

49. What is meant by computing the closure of a set of functional dependency?

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

50. What is meant by normalization of data?

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.

51. Define canonical cover?

A canonical cover F_c for F is a set of dependencies such that F logically implies all dependencies in F_c and F_c logically implies all dependencies in F . F_c must have the following properties.

52. List the properties of canonical cover.

F_c must have the following properties.

- No functional dependency in F_c contains an extraneous attribute.
- Each left side of a functional dependency in F_c is unique.

53. Explain the desirable properties of decomposition.

- Lossless-join decomposition
- Dependency preservation
- Repetition of information

54. What is 2NF?

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

UNIT: 3

DATA STORAGE AND QUERY PROCESSING

1. What is an index?

An index is a structure that helps to locate desired records of a relation quickly, without examining all records

2. Define query optimization.

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

3. What are called jukebox systems?

Jukebox systems contain a few drives and numerous disks that can be loaded into one of the drives automatically.

4. What are the types of storage devices?

- ❖ Primary storage
- ❖ Secondary storage
- ❖ Tertiary storage
- ❖ Volatile storage
- ❖ Nonvolatile storage

5. What is called remapping of bad sectors?

If the controller detects that a sector is damaged when the disk is initially formatted, or when an attempt is made to write the sector, it can logically map the sector to a different physical location.

6. Define access time.

Access time is the time from when a read or write request is issued to when data transfer begins.

7. Define seek time.

The time for repositioning the arm is called the seek time and it increases with the distance that the arm is called the seek time.

8. Define average seek time.

The average seek time is the average of the seek times, measured over a sequence of random requests.

9. Define rotational latency time.

The time spent waiting for the sector to be accessed to appear under the head is called the rotational latency time.

10. Define average latency time.

The average latency time of the disk is one-half the time for a full rotation of the disk.

11. What is meant by data-transfer rate?

The data-transfer rate is the rate at which data can be retrieved from or stored to the disk.

12. What is meant by mean time to failure?

The mean time to failure is the amount of time that the system could run continuously without failure.

13. What is a block and a block number?

A block is a contiguous sequence of sectors from a single track of one platter. Each request specifies the address on the disk to be referenced. That address is in the form of a block number.

14. What are called journaling file systems?

File systems that support log disks are called journaling file systems.

15. What is the use of RAID?

A variety of disk-organization techniques, collectively called redundant arrays of independent disks are used to improve the performance and reliability.

16. What is called mirroring?

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

17. What is called mean time to repair?

The mean time to failure is the time it takes to replace a failed disk and to restore the data on it.

18. What is called bit-level striping?

Data striping consists of splitting the bits of each byte across multiple disks. This is called bit-level striping.

19. What is called block-level striping?

Block level striping stripes blocks across multiple disks. It treats the array of disks as a large disk, and gives blocks logical numbers.

20. What are the two main goals of parallelism?

- Load –balance multiple small accesses, so that the throughput of such accesses increases.

- Parallelize large accesses so that the response time of large accesses is reduced

21. What are the factors to be taken into account when choosing a RAID level?

- Monetary cost of extra disk storage requirements.
- Performance requirements in terms of number of I/O operations
- Performance when a disk has failed.
- Performances during rebuild.

22. What is meant by software and hardware RAID systems?

RAID can be implemented with no change at the hardware level, using only software modification. Such RAID implementations are called software RAID systems and the systems with special hardware support are called hardware RAID systems.

23. Define hot swapping?

Hot swapping permits the removal of faulty disks and replaces it by new ones without turning power off. Hot swapping reduces the mean time to repair.

24. What are the ways in which the variable-length records arise in database systems?

- ✓ Storage of multiple record types in a file.
- ✓ Record types that allow variable lengths for one or more fields.
- ✓ Record types that allow repeating fields.

25. What is the use of a slotted-page structure and what is the information present in the header?

The slotted-page structure is used for organizing records within a single block. The header contains the following information.

- The number of record entries in the header.
- The end of free space
- An array whose entries contain the location and size of each record.

26. What are the two types of blocks in the fixed –length representation? Define them.

- Anchor block: Contains the first record of a chain.
- Overflow block: Contains the records other than those that are the first record of a chain.

27. What is known as heap file organization?

In the heap file organization, any record can be placed anywhere in the file where there is space for the record. There is no ordering of records. There is a single file for each relation.

28. What is known as sequential file organization?

In the sequential file organization, the records are stored in sequential order, according to the value of a “search key” of each record.

29. What is hashing file organization?

In the hashing file organization, a hash function is computed on some attribute of each record. The result of the hash function specifies in which block of the file the record should be placed.

30. What is known as clustering file organization?

In the clustering file organization, records of several different relations are stored in the same file.

31. What are the types of indices?

- Ordered indices
- Hash indices

32. What are the techniques to be evaluated for both ordered indexing and hashing?

- Access types
- Access time
- Insertion time
- Deletion time
- Space overhead

33. What is known as a search key?

An attribute or set of attributes used to look up records in a file is called a search key.

34. What is a primary index?

A primary index is an index whose search key also defines the sequential order of the file.

35. What are called index-sequential files?

The files that are ordered sequentially with a primary index on the search key, are called index-sequential files.

36. What are the two types of indices?

- ✓ Dense index
- ✓ Sparse index

37. What are called multilevel indices?

Indices with two or more levels are called multilevel indices.

38. What is B-Tree?

A B-tree eliminates the redundant storage of search-key values .It allows search key values to appear only once.

39. What is a B+-Tree index?

A B+-Tree index takes the form of a balanced tree in which every path from the root of the tree to a leaf of the tree is of the same length.

40. What is a hash index?

A hash index organizes the search keys, with their associated pointers, into a hash file structure.

41. What is called query processing?

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

42. What are the steps involved in query processing?

The basic steps are:

- ❖ parsing and translation
- ❖ optimization
- ❖ evaluation

43. What is called an evaluation primitive?

A relational algebra operation annotated with instructions on how to evaluate is called an evaluation primitive.

44. What is called a query evaluation plan?

A sequence of primitive operations that can be used to evaluate a query is a query evaluation plan or a query execution plan.

45. What is called a query –execution engine?

The query execution engine takes a query evaluation plan, executes that plan, and returns the answers to the query.

46. What are called as index scans?

Search algorithms that use an index are referred to as index scans.

47. What is called as external sorting?

Sorting of relations that do not fit into memory is called as external sorting.

48. What is called as recursive partitioning?

The system repeats the splitting of the input until each partition of the build input fits in the memory. Such partitioning is called recursive partitioning.

49. What is called as an N-way merge?

The merge operation is a generalization of the two-way merge used by the standard in-memory sort-merge algorithm. It merges N runs, so it is called an N-way merge.

50. What is known as fudge factor?

The number of partitions is increased by a small value called the fudge factor, which is usually 20 percent of the number of hash partitions computed.

UNIT: 4

TRANSACTION PROCESSING

1. What is transaction?

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

2. What are the two statements regarding transaction?

The two statements regarding transaction of the form:

- ✓ Begin transaction
- ✓ End transaction

3. What are the properties of transaction?

The properties of transactions are:

- ✓ Atomicity
- ✓ Consistency
- ✓ Isolation
- ✓ Durability

4. What is recovery management component?

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?

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?

The states of transaction are

- ✓ Active
- ✓ Partially committed
- ✓ Failed
- ✓ Aborted
- ✓ Committed

- ✓ Terminated

7. What is a shadow copy scheme?

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 is simply a file on disk.

8. Give the reasons for allowing concurrency?

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.

9. 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.

10. What are the two types of serializability?

The two types of serializability is

- ✓ Conflict serializability
- ✓ View serializability

11. 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.

12. What are the different modes of lock?

The modes of lock are:

- ✓ Shared
- ✓ Exclusive

13. Define deadlock?

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

14. 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.

15. Define upgrade and downgrade?

It provides a mechanism for conversion from shared lock to exclusive lock is known as upgrade.

It provides a mechanism for conversion from exclusive lock to shared lock is known as downgrade.

16. What is a database graph?

The partial ordering implies that the set D may now be viewed as a directed acyclic graph, called a database graph.

17. What are the two methods for dealing deadlock problem?

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

18. What is a recovery scheme?

An integral part of a database system is a recovery scheme that can restore the database to the consistent state that existed before the failure.

19. What are the two types of errors?

The two types of errors are:

- ✓ Logical error
- ✓ System error

20. What are the storage types?

The storage types are:

- ✓ Volatile storage
- ✓ Nonvolatile storage

21. Define blocks?

The database system resides permanently on nonvolatile storage, and is partitioned into fixed-length storage units called blocks.

22. What is meant by Physical blocks?

The input and output operations are done in block units. The blocks residing on the disk are referred to as physical blocks.

23. What is meant by buffer blocks?

The blocks residing temporarily in main memory are referred to as buffer blocks.

24. What is meant by disk buffer?

The area of memory where blocks reside temporarily is called the disk buffer.

25. What is meant by log-based recovery?

The most widely used structures for recording database modifications is the log. The log is a sequence of log records, recording all the update activities in the database. There are several types of log records.

26. What are uncommitted modifications?

The immediate-modification technique allows database modifications to be output to the database while the transaction is still in the active state. Data modifications written by active transactions are called uncommitted modifications.

27. Define shadow paging.

An alternative to log-based crash recovery technique is shadow paging. This technique needs fewer disk accesses than do the log-based methods.

28. Define page.

The database is partitioned into some number of fixed-length blocks, which are referred to as pages.

29. Explain current page table and shadow page table.

The key idea behind the shadow paging technique is to maintain two page tables during the life of the transaction: the current page table and the shadow page table. Both the page tables are identical when the transaction starts. The current page table may be changed when a transaction performs a write operation.

30. What are the drawbacks of shadow-paging technique?

- Commit Overhead
- Data fragmentation
- Garbage collection

30. Define garbage collection.

Garbage may be created also as a side effect of crashes. Periodically, it is necessary to find all the garbage pages and to add them to the list of free pages. This process is called garbage collection.

32. Differentiate strict two phase locking protocol and rigorous two phase locking protocol.

In **strict two phase 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.

33. How the time stamps are implemented

- Use the value of the system clock as the time stamp. That is a transaction's time stamp is equal to the value of the clock when the transaction enters the system.
- Use a logical counter that is incremented after a new timestamp has been assigned; that is the time stamp is equal to the value of the counter.

34. What are the time stamps associated with each data item?

- W-timestamp (Q) denotes the largest time stamp if any transaction that executed WRITE (Q) successfully.
- R-timestamp (Q) denotes the largest time stamp if any transaction that executed READ (Q) successfully.

UNIT: 5

CURRENT TRENDS

1. What is meant by object-oriented data model?

The object-oriented paradigm is based on encapsulation of data and code related to an object in to a single unit, whose contents are not visible to the outside world.

2. What is the major advantage of object-oriented programming paradigm?

The ability to modify the definition of an object without affecting the rest of the system is the major advantage of object-oriented programming paradigm.

3. What are the methods used in object-oriented programming paradigm?

*read-only

*update

4. What is the main difference between read-only and update methods?

A read-only method does not affect the values of a variable in an object, whereas an update method may change the values of the variables.

5. What is the use of keyword ISA?

The use of keyword ISA is to indicate that a class is a specialization of another class.

6. Differentiate sub-class and super-class?

The specialization of a class is called subclasses.eg: employee is a subclass of person and teller is a subclass of employee. Conversely, employee is a super class of teller, and person is a super class of employee.

7. What is substitutability?

Any method of a class-say A can equally well be invoked with any object belonging to any subclasses B of A. This characteristic leads to code reuse, since the messages, methods, and functions do not have to be written again for objects of class B.

8. What is multiple inheritance?

Multiple inheritance permits a class to inherit variables and methods from multiple super classes.

9. What is DAG?

The class-subclass relationship is represented by a directed acyclic graph.eg: employees can be temporary or permanent. we may create subclasses temporary and permanent, of the class employee.

10. What is disadvantage of multiple inheritance?

There is potential ambiguity if the same variable or method can be inherited from more than one superclass.eg: student class may have a variable dept identifying a student's department, and the teacher class may correspondingly have a variable dept identifying a teacher's department.

11. What is object identity?

An object retains its identity even if some or all the values of variables or definitions of methods change overtime.

12. What are the several forms of identity?

*Value

*Name

*Built-in

13. What is a value?

A data value is used for identity. This form of identity is used in relational systems.eg: The primary key value of a tuple identifies the tuple.

14. What is a Name?

A user-supplied name is used for identity. This form of identity is used for files in file systems. The user gives each file a name that uniquely identifies it, regardless of its contents.

15 What is a Built-in

A notation of identity is built-into the data model or programming language and no user-supplied identifier is required. This form of identity is used in object-oriented systems.

16 What is meant by object identifiers?

Object-oriented systems use an object identifier to identify objects. Object identifiers are unique: that is each object has a single identifier, and no two

objects have the same identifier.

17. What are composite objects?

Objects that contain other objects are called complex objects or composite objects.

18. What is object containment?

References between objects can be used to model different real-world concepts.

19. Why containment is important in oosystems?

Containment is an important concept in oosystems because it allows different users to view data at different granularities.

20. Define object-relational systems?

Systems that provide object-oriented extensions to relational systems are called object-relational systems.

21. How persistent programming languages differ from traditional programming languages?

Database languages differ from traditional programming languages in that they directly manipulate data that are persistent—that is, data that continue to exist even after the program terminated. Relation in a database and tuples in a relation are examples of persistent data. In contrast, the only persistent data that traditional programming languages directly manipulate are files.

22. Define atomic domains?

A domain is atomic if elements of the domain are considered to be indivisible units.

23. Define 1NF?

First normal form is one which requires that all attributes have atomic domains.

24. What is nested relational model?

The nested relational model is an extension of relational model in which domains may be either atomic or relation valued.

25. List some instances of collection types?

*sets

*arrays

*multisets

26. How to create values of structured type?

Constructor functions are used to create values of structured types. A function with the same name as a structured type is a constructor function for the structured type.

27. Write a query to define tables students and teachers as sub tables of people?

Create table students of student under people

Create table teachers of teacher under people

28. What is a homogeneous distributed database?

In homogeneous distributed databases, all sites have identical database management system software, are aware of one another, and agree to cooperate in processing user's requests.

29. What is a heterogeneous distributed database?

In a heterogeneous distributed database, different sites may use different schemas, and different dbms s/w. The sites may not be aware of one another, and they may provide only limited facilities for cooperation in transaction processing.

30. What are the two approaches to store relations in distributed database?

*Replication

*Fragmentation

31. What are the two different schemes for fragmenting a relation?

*horizontal

*vertical

32. What is horizontal fragmentation?

Horizontal fragmentation splits the relation by assuming each tuple of r to one or more fragments.

33. What is vertical fragmentation?

Vertical fragmentation splits the relation by decomposing the scheme R of relation r.

34. What are the various forms of data transparency?

*fragmentation transparency

*replication transparency

*location transparency

35. Define decision tree classifiers?

As the name suggests decision tree classifiers use a tree: each leaf node has an associated class, and each internal node has a predicate associated with it.

16 MARK QUESTIONS

UNIT: 1

1. EXPLAIN ABOUT DATABASE SYSTEM STRUCTURE?

Storage manager

- Authorization and integrity manager
- Transaction manager
- File manager
- Buffer manager

Storage manager implements several data structure as a part of physical system implementation

- Data function
- Data dictionary
- Indices

The query processor

- DDL interpreter
- DML
- Query evaluation engine

2. DESCRIBE RELATIONAL MODEL?

Structure of relational data base

- ❖ Basic structure
- ❖ Database schema
- ❖ Keys
- ❖ Schema diagram
- ❖ Query languages

3. BRIEFLY EXPLAIN RELATIONAL ALGEBRA?

Fundamental operations

- Unary operations
- Binary operations

Select operations

$\sigma_{\text{branchname}='perryridge'}(\text{loan})$

The project operation

$\Pi_{\text{loannumber,amount}}(\text{loan})$

Composition of relational operations

- Relational algebra expressions

Union operations

$r \cup s$

r and s must be a same arity.

They must have the same no of attributes.

The set difference operations

r-s produce a relation containing those tuples in r but not in s.

The Cartesian product operations

The rename operations

4. WHAT IS DATA MODELS? EXPLAIN IT DETAIL?

Entity relationship model

- Rectangles
- Ellipse
- Diamonds
- Lines

Relational model

Relational model use a collection of tables to represent both data and the relationships among those data. Each table has a multiple columns and each columns has unique name

Other data models

- Object oriented data model
- Object relational data model
- Network data model
- Hierarchical data model

5. BRIEFLY DESCRIBE RELATIONAL CALCULUS?WITH SOME EXAMPLES?

The tuple relational calculus

A query in a tuple relational calculus is expressed as

$\{t \mid P(t)\}$

- Example Queries
- Formal definition
- Safety of expressions
- Expressive power of languages

The domain relational calculus

- Example Queries
- Formal definition
- Safety of expressions
- Expressive power of languages

UNIT-2

1. DESCRIBE INTEGRITY AND SECURITY?

- **Domain constraint**

Referential integrity

A value that appears in one relation for a given set of attributes also appear for a certain set of attributes in another relation. This condition is called referential integrity.

- **Referential integrity and E-R models**
- **Database modification**
- **Referential integrity in SQL**

2. WHAT IS AGGREGATE FUNCTION? BRIEFLY DESCRIBE IT?

Aggregate functions are functions that take a collection of values as input and return a single value. SQL offers 5 built-in aggregate functions:

- Average: **avg**
- Minimum: **min**
- Maximum: **max**
- Total: **sum**
- Count: **count**

Average: **avg**

Select avg (balance)

From account

Where branch-name='perryridge'

Count :**count**

select branch-name,**count**(distinct customer-name)

from depositor,account

where depositor.account-number=account.account-number

groupby branch-name

3. WHAT IS DATA DEFINITION LANGUAGE?EXPLAIN IT IN DETAIL?

The SQL DDL allows specification of not only a set of relations, but also information after each relation, including

- The schema for each relation
- The domain of values associated with each attribute
- The integrity constraints
- The set of indices to be maintained for each relation
- The security and authorization information for each relation
- The physical storage structure of each relation on disk

Domain Types in SQL

Char(n),
varchar(n), int, small int, numeric(p,d), real, double, precision,
float(n), date, time, timestamp.

Schema Definition in SQL

- Primary key
- Check

4. EXPLAIN MECHANISM OF NESTED QUERIES?

SQL provides a mechanism for nesting subqueries. A subquery is a select from where expression that is nested within another query. A common use of sub queries is to perform tests for set membership, make set comparisons, and determine set cardinality.

- **Set membership**
(select customer-name
from depositor)
- **Set comparison**
select distinct T.branch-name
from branch as T,branch as S
where T.assets > S.assets and S.branch-city='Brooklyn'
- **Test for Empty Relations**
Select customer-name
from borrower
where exists (select *
from depositor
where depositor.customer-
name=borrower.customer-name)
- **Test for the Absence of Duplicate Tuples**

5. WRITE SHORT NOTES ON MODIFICATION OF THE DATA BASE?

- **Definition**

delete from r

where P

- **Insertion**
insert into account
values ('A-9732','perryridge',1200)
- **Updates**
update account
set balance=balance*1.05
- **Update of a view**
- **Transaction**

UNIT-3

1. DESCRIBE FILE ORGANISATION?

A file is organized logically as a sequence of records. These records are mapped onto disk blocks.

- **Fixed-Length Records**
type deposit=record
Accountnumber:char(10);
branch name:char(22);
balance: real;
end
- **Variable length records**
 - *storage of multiple record types in a file
 - * Record types that allow variable lengths for one or more fields
 - *Record types that allow repeating fields
 - Byte string Representation
 - Fixed length representation
 - Reserved space
 - List representation

2. DEFINE RAID? BRIEFLY EXPLAIN IT?

A variety of disk organization techniques, collectively called redundant arrays of independent disks (RAID)

- Improvement of reliability via redundancy.
- Improvement in performance via parallelism
 1. Bit level striping
 2. Block level striping
- RAID levels
 - RAID level 0
 - RAID level 1
 - RAID level 2(memory style error correcting code)
 - RAID level 3 (Bit interleaved parity organization)

- RAID level 4 (Block interleaved parity organization)
- RAID level 5 (Block interleaved distributed parity)
- RAID level 6 (P+Q redundancy)

3WRITE SHORT NOTES ON INDEX STRUCTURE OF FILES?

There are two basic kinds of indices

- **Ordered indices**
- **Hash indices**

Each technique must be evaluated on the basis of these factors:

- **Access types**
- **Access time**
- **Insertion time**
- **Deletion time**
- **Space overhead**
- **Ordered indices**
- **Primary index**
- **Dense and sparse indices**
- **Multilevel index**
- **Index update**
- **Secondary indices**

B⁺-Tree index files

B⁺-Tree index structure is the most widely used of several index structures that maintain their efficiency despite insertion and deletion of data.

- **Structure of B⁺-Tree**
- **Queries on B⁺-Tree**
- **Update on B⁺-Tree**
- **B⁺-Tree file organization**
- **B-Tree index files**

4. EXPLAIN HASH FILE ORGANIZATION?

Hash functions

- **The distribution is uniform**

Hash functions assign each bucket the same number of search-key values from the set of all possible search-key values

- **The distributed in random**

In the average case each bucket will have nearly same no of values assigned to it, regardless of the actual distribution of search-key values

Handling of bucket overflows

- **Insufficient buckets**
- **Skew**

Open hashing

Under an alternative approach called open hashing

Close hashing

The form of hash structure that we have just described is something referred to as close hashing.

Hash indices

5. WHAT IS MAGNETIC DISKS?EXPLAIN IT?

Magnetic disk provides the bulk of secondary storage of modern computer system. The disk capacity is growing at over 50% per year. But the storage requirements of large applications has also been growing very fast and in some case every faster than the growth rate of disk capacities. A large data base may require 100 of disks.

- Physical characteristics of disk

Physical disks are relatively simple. Each disc platter has a flat circular shape

We can call magnetic disk as

- **hard disk**
- **Floppy disk**

The read write head store information on a sector magnetically as reversals of the direction of magnetization of the magnetic material. There may be hundreds of concentric tracks on a disc surface, containing thousands of sectors.

Unit 4

1. DESCRIBE LOG BASED RECOVERY

The most usable structure for recording data base modification is the LOG the log is a sequence of log records recording all the update activities in the data base. There are several types of log records. An update log records describes a single data base write it has these fields

- **Transaction identifier**
- **Data item identifier**
- **Old value**
- **New value**

The various types log records as.

- $\langle T_i \text{ start} \rangle$. Transaction T_i has started
- $\langle T_i, T_x v_1, v_2 \rangle$. Transaction T_i has performed a right on data item
- $\langle T_i \text{ commit} \rangle$ Transaction T_i has committed
- $\langle T_i \text{ about} \rangle$ Transaction T_i has aborted

- **Deferred data base modification**
- **Immediate data base modification**
- **Check point**
- **Shadow paging**

2. WHAT IS SERIALIZABILITY? EXPLAIN ITS TYPES?

The data base system must control concurrent execution of transactions, to ensure that the data base state remains consistent. There are different forms of schedule equivalence they lead to the notions of

- **Conflict serializability**
- **View serializability**

Conflict serializability

We say that i_i and i_j conflict if they are operations by different transaction on the same data item and at least one of these instruction is a write operations

View serializability

The concept of view equivalence leads to the concept of View serializability we say that a schedules S is view serializable if it is view equivalent to a serial scheduler

3. WRITE SHORT NOTES ON TRANSACTION STATE?

A transaction may not always complete its execution successfully such a transaction is termed aborted

A transaction must be in one of the following states

- Active
- Partially committed
- Failed
- Aborted
- Committed

4. BRIEFLY DESCRIBE CONCURRENCY EXECUTION?

- Lock – based protocols
- Locks

There are various modes in which a data item may be locked in this section we restrict our attention to two modes

- Shared
- Exclusive

```
T1 : lock – x(B);  
      read(B);  
      B:=B-50;  
      write(B);  
      unlock(B);  
      Lock-x(A);  
      read(A);  
      A:=A+50;  
      write(A);  
      unlock(A).
```

5. EXPLAIN CONCURRENCY CONTROL?

Concurrency control

Oracles multiversion concurrency control differs from the concurrency mechanism used by some other data base vendors. Read only queries are given a read –consistent snapshot which is view if the data base as it existed at the specific point in time, containing all update that we were committed by that point in time and not containing any updates that were not committed at any point in time thus read

clock are not used in read only queries don't interfere with other data base activity in term of locking.

Managed stand by data base

To ensure high availability oracle provide a managed stand by data base future A stand by data base is a copy of the regular data base ie in solved on the separate system. If a catastrophic failure occur on the primary system, the stand by system is activate and take over there by minimizing effect on failure on a availability. Oracle keeps the stand by data base up to date by constantly applying archived redo logs that are shipped from the primary data base the back up data base can be brought online in read-only mode and used for reporting and decision support queries

UNIT 5

1. WRITE SHORT NOTES ON DATA WARE HOUSING?

Data ware housing applications requires the transformation of data from many sources into a cohesive consistent step set of data configured appropriately for use in data ware house operation.

- **Distributed Transformation services**

Data ware housing is an approach to manage data in which heterogeneous data sources are migrated to a separate homogeneous data base

- **Online Analytical processing services**

OLAP services provide server and client capabilities to create and manage multidimensional OLAP data .

2. EXPLAIN NESTED RELATIONS?

Nested relations

The assumption of INF is a natural one in the bank examples we have considered. However, not all applications are best modeled by INF relations.

The nested relational model is an extension of the relational model in which domains may be either atomic or relation valued.

We illustrate nested relations by an example from a library. Suppose we store for each book the following in formations

- Book title
- Set of authors
- Publishers
- Set of keywords

We can see that if we define a relation for the preceding information, several domains will be monatomic

- **Authors**
- **Keywords**
- **Publishers**

Complex types

Collection and large object types

```
Create table books(  
  ...  
  Keyword-set setoff(varchar(20))  
  ...  
)
```

Structure types

Creation of values of complex types

3. WHAT IS INHERITANCE? DESCRIBE IT IN DETAIL?

Inheritance

Inheritance can be at the levels of types, or at the level of tables. We first consider inheritance of types, then inheritance at the level of labels.

Type inheritance

Suppose that we have the following type definition for people

```
create type person  
  (name varchar(20)  
  address varchar(20))
```

Table inheritance

Create table people of person

The consistency requirements for sub tables are

1. Each tuple of the sub table can correspond to at most one tuple in each of its immediate sub tables.
2. SQL:1999 has an additional constraint that all the tuples corresponding to each other must be derived from one tuple .

Overlapping sub tables

4 WHAT ARE THE TYPES OF REFERENCE? EXPLAIN IT WITH SUITABLE EXAMPLES?

Object oriented language provided the ability to refer the object attribute of the type can be referred to the specified type. We can define the type dept with a field name and a field head which is reference to the type person and a table dept of the type dept as followed

```
Create type dept(  
  Name varchar(20),  
  Head ref(person)scope people  
)  
Create table dept of dept
```

The table definition must specify that the reference is derived and must still specify a self referential attribute name. When interesting a tuple for dept we can then use

Insert into dept
Values('CS','john')

5. DESCRIBE QUERIES WITH COMPLEX TYPES?

The present extension of the SQL query language deal with the complex type
Let us start with the simple example:

Find the title and the name of the publisher of each book this query carries out the task:

Select title, publisher.name
From books

- **path expression**

The reference are dereference in 1999 by the → simple

Select head-> name, head->address

From dept

An expression such as "head->name" is called the path expression.

- **Collection valued attributes**

- **Nesting and unnesting**

The transformation of the nested relation in to a form with fewer (or no) the relation –valued attribute value is called unnesting

The reverse process of transformation a INF relation into a nested relation is called nesting.