IARE '24 DISCORD TEAM

II-II

DBMS MODULE 1 SOLUTIONS

PRIYANANDINI • LALITHA • UJJWAL • IKRAM

CONCEPTUAL MODELING
INTRODUCTION

DBMS MODULE 1

PART-B

1) Compare and Contrast file Systems with database systems.

BASIS	FILE SYSTEM	DBMS
Structure	The file system is software that manages and organizes the files in a storage medium within a computer.	DBMS is software for managing the database.
Data Redundancy	Redundant data can be present in a file system.	It provides backup and recovery of data even if it is lost.
Backup and Recovery	It doesn't provide backup and recovery of data if it is lost.	It provides backup and recovery of data even if it is lost.
Query processing	There is no efficient query processing in the file system.	Efficient query processing is there in DBMS.
Consistency	There is less data consistency in the file system.	There is more data consistency because of the process of normalization.
Cost	It is less expensive than DBMS.	It has a comparatively higher cost than a file system.
User Access	Only one user can access data at a time.	Multiple users can access data at a time.
Security Constraints	File systems provide less security in comparison to DBMS.	DBMS has more security mechanisms as compared to file systems.

2) Define Data Abstraction and discuss levels of Abstraction.

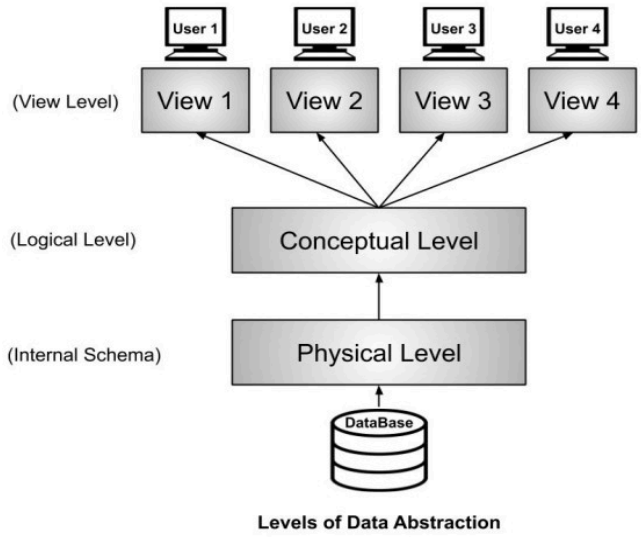
Data Abstraction is a process of hiding unwanted or irrelevant details from the end user. It provides a different view and helps in achieving data independence which is used to enhance the security of data.

The database systems consist of complicated data structures and relations. For users to access the data easily, these complications are kept hidden, and only the relevant part of the database is made accessible to the users through data abstraction.

Levels of abstraction for DBMS

Mainly there are three levels of abstraction for DBMS, which are as follows –

- Physical or Internal Level
- Logical or Conceptual Level
- View or External Level



Levels of Data Abstraction

Physical or Internal Level

It is the lowest level of abstraction for DBMS which defines how the data is actually stored, it defines data-structures to store data and access methods used by the database. Actually, it is decided by developers or database application programmers how to store the data in the database.

For example, customer information is stored in tables, and data is stored in the form of blocks of storage such as bytes, gigabytes, etc.

Logical or Conceptual Level

Logical level is the intermediate level or the next higher level. It describes what data is stored in the database and what relationship exists among those data. It tries to describe the entire or whole data because it describes what tables to be created and what are the links among those tables.

The logical level is used by developers or database administrators (DBA).

View or External Level

It is the highest level. At the view level, there are different levels of views and every view only defines a part of the entire data. It also simplifies interaction with the user and it provides many views or multiple views of the same database.

View level can be used by all users (all levels' users).

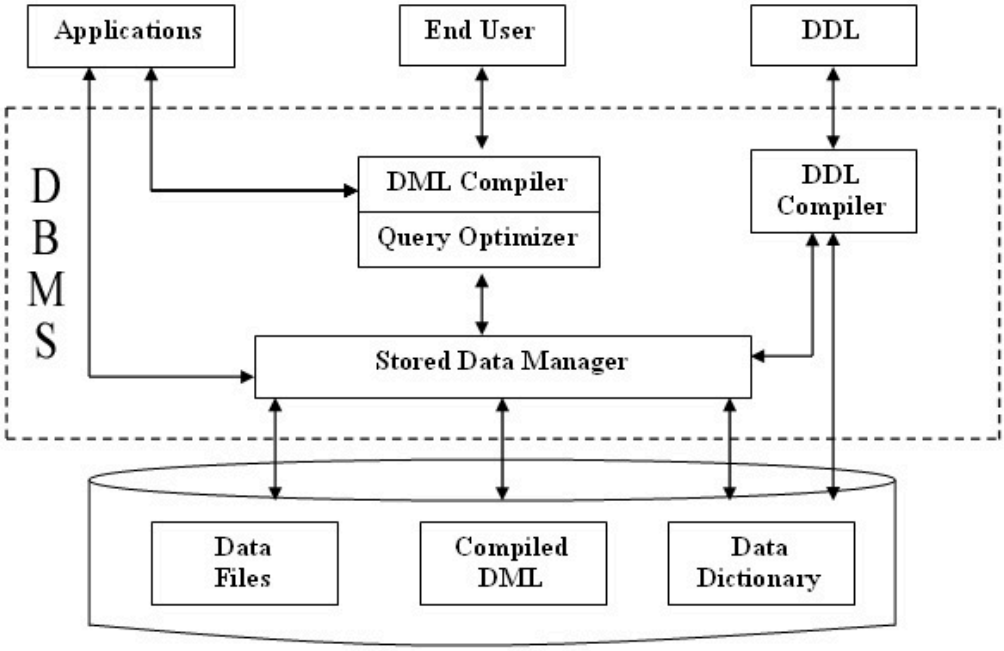
3) Discuss different types of Data models.

- Entity-Relationship Data Model: An ER model is the logical representation of data as objects and relationships among them. These objects are known as entities, and a relationship is an association among these entities. It was widely used in database design. A set of attributes describe the entities. For example, student_name, and student_id describe the 'student' entity. A set of the same type of entities is known as an 'Entity set', and the set of the same type of relationships is known as a 'relationship set'.
- Relational Data Model: This type of model designs the data in the form of rows and columns within a table. Thus, a relational model uses tables for representing data and in-between relationships. Tables are also called relations.
- Object-based Data Model: An extension of the ER model with notions of functions, encapsulation, and object identity, as well. This model supports a rich type system that includes structured and collection types. Thus, in the 1980s, various database systems following the

Object-Oriented approach were developed. Here, the objects are nothing but the data-carrying its properties.

- Semi-structured Data Model: The semi-structured data model permits the specification of data where individual data items of the same type may have different sets of attributes.

4) Describe the Structure of DBMS



The database system is divided into three components:

1. Query Processor
2. Storage Manager
3. Disk Storage

Query Processor:

It interprets the requests (queries) received from the end-user via an application program into instructions. It also executes the user request which is received from the DML compiler.

Query Processor contains the following components –

- DML Compiler – It processes the DML statements into low-level instruction (machine language), so that they can be executed.
- DDL Interpreter – It processes the DDL statements into a set of tables containing meta data (data about data).
- Embedded DML Pre-compiler – It processes DML statements embedded in an application program into procedural calls.
- Query Optimizer – It executes the instruction generated by DML Compiler.

Storage Manager:

Storage Manager is a program that provides an interface between the data stored in the database and the queries received. It is also known as Database Control System. It maintains the consistency and integrity of the database by applying the constraints and executing the DCL statements. It is responsible for updating, storing, deleting, and retrieving data in the database.

It contains the following components –

- Authorization Manager – It ensures role-based access control, i.e,. checks whether the particular person is privileged to perform the requested operation or not.
- Integrity Manager – It checks the integrity constraints when the database is modified
- Transaction Manager – It controls concurrent access by performing the operations in a scheduled way that it receives the transaction. Thus, it ensures that the database remains in a consistent state before and after the execution of a transaction.
- File Manager – It manages the file space and the data structure used to represent information in the database.
- Buffer Manager – It is responsible for cache memory and the transfer of data between the secondary storage and main memory.

Disk Storage:

It contains the following components –

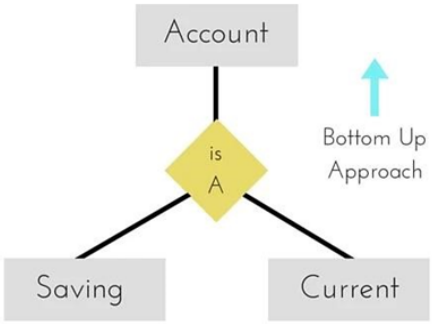
- Data Files – It stores the data.

- Data Dictionary – It contains information about the structure of any database object. It is the repository of information that governs the metadata.
- Indices – It provides faster retrieval of data items.

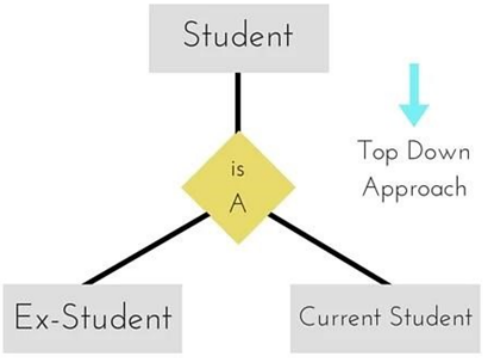
5) Discuss additional features of the ER-Models.

ER model stands for an Entity-Relationship model. ER Models use a defined set of symbols such as rectangles, diamonds, ovals and connecting lines to depict the interconnectedness of entities, relationships and their attributes. They mirror grammatical structure, with entities as nouns and relationships as verbs. additional features of the ER-Models:

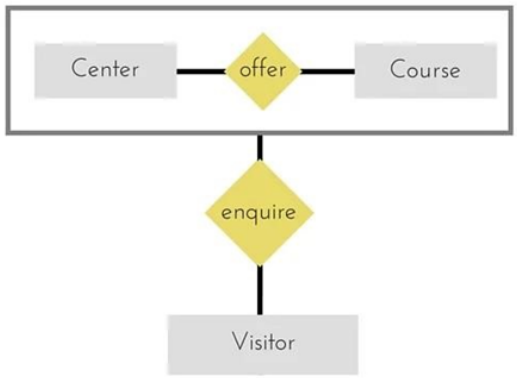
1. Generalisation
2. Specialisation
3. Aggregation



Generalisation is a bottom-up approach in which two lower level entities combine to form a higher level entity. In generalisation, the higher level entity can also combine with other lower level entities to make further higher level entities. It's more like a Superclass and Subclass system, but the only difference is the approach, which is bottom-up. Hence, entities are combined to form a more generalised entity, in other words, sub-classes are combined to form a super-class.



Specialization is opposite to Generalization. It is a top-down approach in which one higher level entity can be broken down into two lower level entities. In specialization, a higher level entity may not have any lower-level entity sets, it's possible.



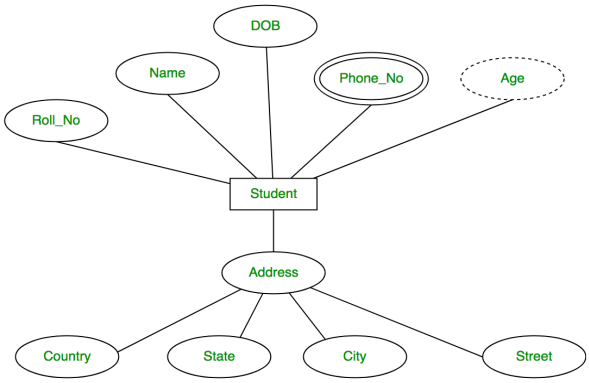
Aggregation is a process when relation between two entities is treated as a single entity. In the diagram above, the relationship between Center and Course together, is acting as an Entity, which is in relationship with another entity Visitor. Now in the real world, if a Visitor or a Student visits a Coaching Center, he/she will never enquire about the center only or just about the course, rather he/she will enquire about both.

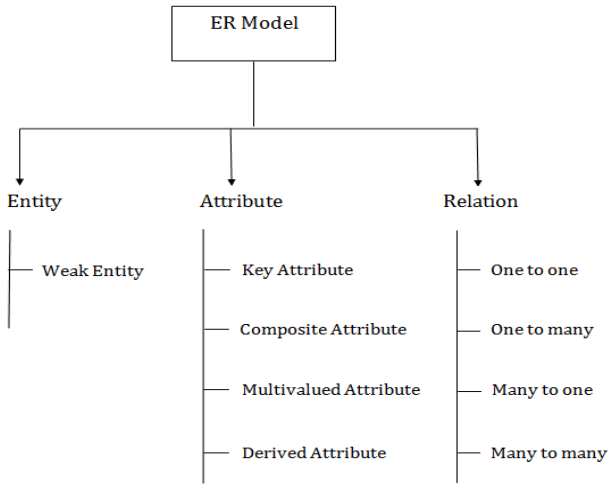
6) Discuss about the Concept Design with the ER Model.

Refer here

ER model-

- ER model stands for an Entity-Relationship model. It is a high-level data model. This model is used to define the data elements and relationship for a specified system.
- It develops a conceptual design for the database. It also develops a very simple and easy to design view of data.
- In ER modeling, the database structure is portrayed as a diagram called an entity-relationship diagram.
- For example, Suppose we design a school database. In this database, the student will be an entity with attributes like address, name, id, age, etc. The address can be another entity with attributes like city, street name, pin code, etc and there will be a relationship between them.





Include part B 15th answer!

7) Explain in detail Different types of Data Independence with examples.

Refer here

Data Independence

- Data independence can be explained using the three-schema architecture.
- Data independence refers characteristic of being able to modify the schema at one level of the database system without altering the schema at the next higher level

There are two types of data independence:

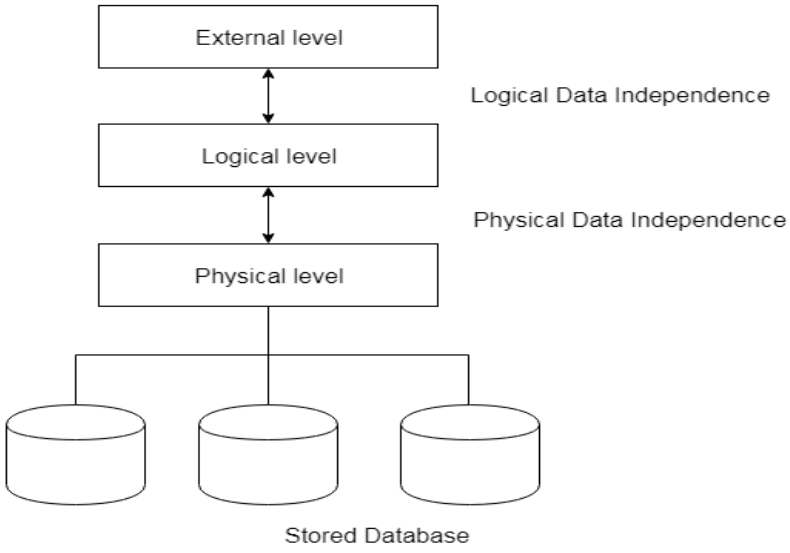
1. Logical Data Independence

- Logical data independence refers to the characteristic of being able to change the conceptual schema without having to change the external schema.
- Logical data independence is used to separate the external level from the conceptual view.
- If we do any changes in the conceptual view of the data, then the user view of the data would not be affected.
- Logical data independence occurs at the user interface level.

2. Physical Data Independence

- Physical data independence can be defined as the capacity to change the internal schema without having to change the conceptual schema.
- If we do any changes in the storage size of the database system server, then the Conceptual structure of the database will not be affected.
- Physical data independence is used to separate conceptual levels from the internal levels.

- Physical data independence occurs at the logical interface level.



8) Explain different types of database users and write the functions of DBA.

There are four different types of database system users. Different types of user interfaces have been designed for the different types of users.

- Naive users:Naive users are unsophisticated users who interact with the system by invoking one of the application programs that have been written previously. For example, a clerk in the university who needs to add a new instructor to Users is differentiated by the way they expect to interact with the system department A invokes a program called New - hire. This program asks the clerk for the name of the new instructor, her new ID, the name of the department (that is, A), and the salary
- Application programmers: Application programmers are computer professionals who write application programs. Application programmers can choose from many tools to develop user interfaces. Rapid application development (RAD) tools are tools that enable an application

programmer to construct forms and reports with minimal programming effort.

- Sophisticated users: Sophisticated users interact with the system without writing programs. In- stead, they form their requests either using a database query language or by using tools such as data analysis software. Analysts who submit queries to explore data in the database fall in this category
- Specialized users: Specialized users are sophisticated users who write specialized database applications that do not fit into the traditional data processing framework.

Functions of DBA:

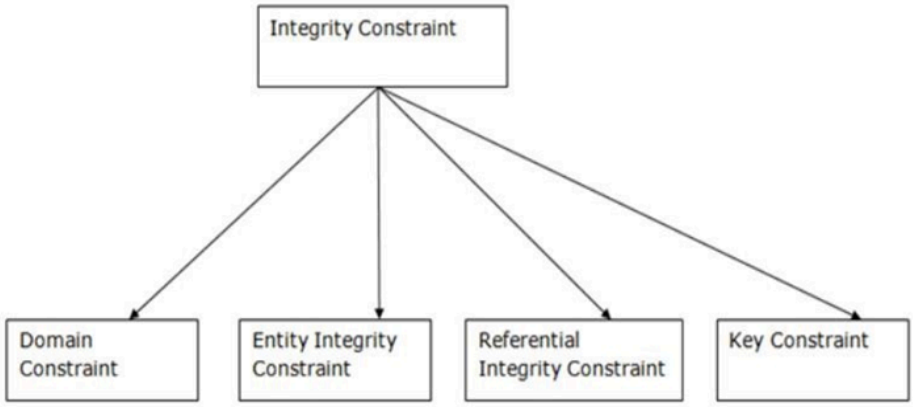
One of the main reasons for using DBMSs is to have central control of both the data and the programs that access those data. A person who has such central control over the system is called a database administrator (DBA). The functions of a DBA include:

- The DBA creates the original database schema by executing a set of data definition statements in the DDL.
- Storage structure and access-method definition.
- Schema and physical-organization modification.
- Routine maintenance.
- Periodically backing up the database.
- Ensuring that enough free disk space is available for normal operations, and upgrading disk space as required. – Monitoring jobs running on the Database.

9) List out different types of integrity constraints.

- Integrity constraints are a set of rules. It is used to maintain the quality of information.

- Integrity constraints ensure that the data insertion, updating, and other processes have to be performed in such a way that data integrity is not affected.
- Thus, integrity constraint is used to guard against accidental damage to the database.



1. Domain constraints:Domain constraints can be defined as the definition of a valid set of values for an attribute.
2. Entity integrity constraints:The entity integrity constraint states that primary key value can't be null.
3. Referential Integrity Constraints:A referential integrity constraint is specified between two tables.
4. Key constraints: Keys are the entity set that is used to identify an entity within its entity set uniquely.

10) Discuss about Different keys used in database design with examples.

KEYS in DBMS is an attribute or set of attributes which helps you to identify a row(tuple) in a relation(table). They allow you to find the relation between two tables. Keys help you uniquely identify a row in a table by a combination of one or more columns in that table.

Key is also helpful for finding unique records or rows from the table. Database key is also helpful for finding unique records or rows from the table.

Types of Keys in DBMS (Database Management System):

1. Primary Key 2. Candidate Key 3. Foreign Key 4.Super Key

1) Primary Key:

PRIMARY KEY in DBMS is a column or group of columns in a table that uniquely identifies every row in that table. The Primary Key can't be a duplicate meaning the same value can't appear more than once in the table. A table cannot have more than one primary key. Example:

In the following example, StudID is a Primary Key.

StudID	Roll No	First Name	LastName	Email
1	11	Tom	Price	abc@gmail.com
2	12	Nick	Wright	xyz@gmail.com
3	13	Dana	Natan	mno@yahoo.com

Candidate Key:

CANDIDATE KEY in SQL is a set of attributes that uniquely identify tuples in a table. Candidate Key is a super key with no repeated attributes. The Primary key should be selected from the candidate keys. Every table must have at least a single candidate key. A table can have multiple candidate keys but only a single primary key.

Candidate key Example:

In the given table Stud ID, Roll No, and email are candidate keys that help us to uniquely identify the student record in the table.

StudID	Roll No	First Name	LastName	Email
1	11	Tom	Price	abc@gmail.com
2	12	Nick	Wright	xyz@gmail.com

3	13	Dana	Natan	mno@yahoo.co m

Foreign key:

FOREIGN KEY is a column that creates a relationship between two tables. The purpose of Foreign keys is to maintain data integrity and allow navigation between two different instances of an entity.It acts as a cross-reference between two tables as it.

Example:

DeptCode	DeptName
001	Science
002	English
005	Computer

Teacher ID	Fname	Lname
B002	David	Warner
B017	Sara	Joseph
B009	Mike	Brunton

In this key DBMS example, we have two tables, a teacher and department in a school. However, there is no way to see which search works in which department.

In this table, adding the foreign key in Deptcode to the Teacher name, we can create a relationship between the two tables.

Teacher ID	DeptCode	Fname	Lname
B002	002	David	Warner

B017	002	Sara	Joseph
B009	001	Mike	Brunton

Super key:

A superkey is a group of single or multiple keys which identifies rows in a table.

A Super key may have additional attributes that are not needed for unique identification. Example:

EmpSSN	EmpNum	Empname
9812345098	AB05	Shown
9876512345	AB06	Roslyn
199937890	AB07	James

In the above-given example, EmpSSN and EmpNo names are superkeys.

11) Distinguish between a strong entity set and a weak entity set?

Strong Entity Set	Weak Entity Set
it has its own primary key.	It does not save sufficient attributes to form a primary Key on its own.
It is represented by a rectangle.	It is represented by a double rectangle.
It contains a primary key represented by an underline.	It contains a Partial Key or discriminator represented by a dashed underline.
The member of strong entity set is called as dominant entity set.	The member of weak entity set is called as subordinate entity set.
The Primary Key is one of its attributes which uniquely Identifies its member.	The Primary Key of weak entity set is a combination of partial Key and Primary Key of the strong entity set.
The relationship between two strong entity set is represent by a diamond symbol.	The relationship between one strong and a weak entity set is represented by a double diamond sign. It is known as identifying relationship.
The line connecting strong entity set with the relationship is single	The line connecting weak entity set with the identifying relationship is double.
Total participation in the relationship may or may not exist.	Total participation in the identifying relationship always exists.

12) Differentiate relation schema and relational instance?

1. A schema is the design representation of a database whereas an instance is the snapshot of a database at a particular moment.
2. Instance changes very frequently, whenever data is removed or added to the database. As against, the changes in schema occur rarely.
3. For example, schema and instance can be easily perceived by analogy to a program. At the time of writing a program in a programming language, the variables of that program are declared at first, this is analogous to the schema definition. Additionally, each variable in a program must have some values associated at a particular time; this is similar to an instance.

Definition of instance: The data stored in a database at a particular moment of time is called an instance of database. Database schema defines the variable declarations in tables that belong to a particular database; the value of these variables at a moment of time is called the instance of that database.

Definition of schema: Design of a database is called the schema. Schema is of three types:

- Physical schema
- logical schema
- view schema.

—----------doubt—----------

13) List and explain the design issues of entity relationship ER Design Issues.

Normally, users often mislead the concept of the elements and the design process of the ER diagram. Thus, it leads to a complex structure of the ER diagram and certain issues that do not meet the characteristics of the real-world enterprise model. The basic design issues of an ER database schema are:

1. Use of Entity Set vs Attributes :

   The use of an entity set or attribute depends on the structure of the real-world enterprise that is being modeled and the semantics associated with its attributes. It leads to a mistake when the user uses

the primary key of an entity set as an attribute of another entity set. Instead, he should use the relationship to do so. Also, the primary key attributes are implicit in the relationship set, but we designate it in the relationship sets.

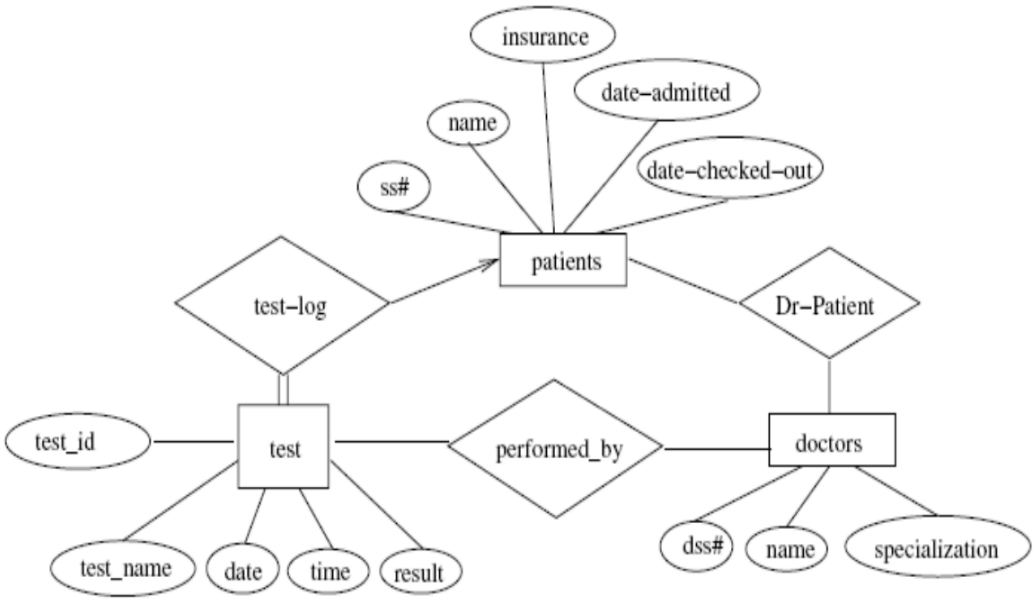
2. Use of Entity Set vs. Relationship Sets:
   It is difficult to examine if an object can be best expressed by an entity set or relationship set. To understand and determine the right use, the user needs to designate a relationship set for describing an action that occurs in-between the entities. If there is a requirement of representing the object as a relationship set, then it's better not to mix it with the entity set.
3. Use of Binary vs n-ary Relationship Sets:
   Generally, the relationships described in the databases are binary relationships. However, non-binary relationships can be represented by several binary relationships.
   For example, we can create and represent a ternary relationship 'parent' that may relate to a child, his father, as well as his mother. Such relationships can also be represented by two binary relationships i.e, mother and father, that may relate to their child.
   Thus, it is possible to represent a non-binary relationship by a set of distinct binary relationships.

4. Placing Relationship Attributes:
   The cardinality ratios can become an effective measure in the placement of the relationship attributes. So, it is better to associate the attributes of one-to-one or one-to-many relationship sets with any participating entity sets, instead of any relationship set. The decision of placing the specified attribute as a relationship or entity attribute should possess the characteristics of the real world enterprise that is being modeled.
   For example, if there is an entity which can be determined by the combination of participating entity sets, instead of determining it as a

separate entity. Such type of attribute must be associated with the many-to-many relationship sets.

Thus, it requires the overall knowledge of each part that is involved in designing and modeling an ER diagram. The basic requirement is to analyze the real-world enterprise and the connectivity of one entity or attribute with another.

14) Construct an ER-Diagram for a hospital with a set of patients and a set of medical doctors. Associated with each patient a log of the various tests and examinations conducted.



15) Describe about Basic Concepts of ER Model in DBMS.

The ER model defines the conceptual view of a database. It works around real-world entities and the associations among them. At view level, the ER model is considered a good option for designing databases.

Entity:

An entity can be a real-world object, either animate or inanimate, that can be easily identifiable. For example, in a school database, students, teachers, classes, and courses offered can be considered as entities. All these entities have some attributes or properties that give them their identity.

An entity set is a collection of similar types of entities. An entity set may contain entities with attributes sharing similar values.

For example, a Students set may contain all the students of a school; likewise a Teachers set may contain all the teachers of a school from all faculties. Entity sets need not be disjoint.

Attributes:

Entities are represented by means of their properties, called attributes. All attributes have values. For example, a student entity may have name, class, and age as attributes.

There exists a domain or range of values that can be assigned to attributes. For example, a student's name cannot be a numeric value. It has to be alphabetic. A student's age cannot be negative, etc.

Keys:

Key is an attribute or collection of attributes that uniquely identifies an entity among an entity set.

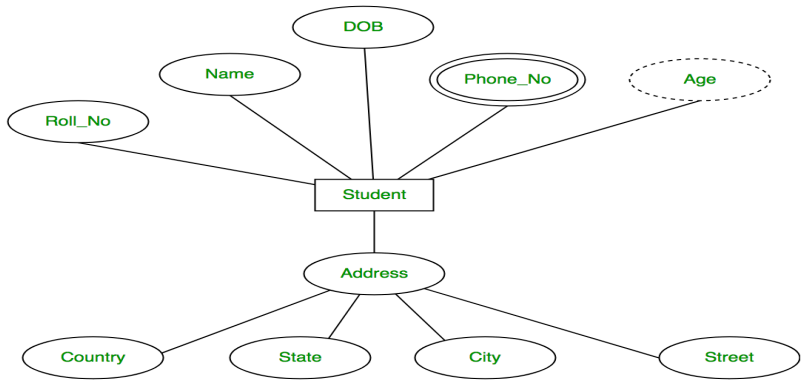
For example, the roll_number of a student makes him/her identifiable among students.

- Super Key – A set of attributes (one or more) that collectively identifies an entity in an entity set.
- Candidate Key – A minimal super key is called a candidate key. An entity set may have more than one candidate key.

- Primary Key – A primary key is one of the candidate keys chosen by the database designer to uniquely identify the entity set.
- Relationship:

  The association among entities is called a relationship. For example, an employee works_at a department, a student enrolls in a course. Here, Works_at and Enrolls are called relationships.
- Cardinalities:

  Cardinality defines the number of entities in one entity set, which can be associated with the number of entities of another set via the relationship set.

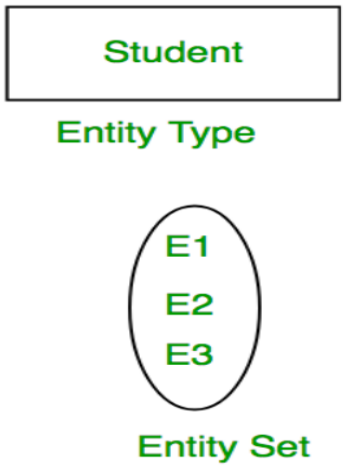


16) Explain ER Model, with its Entity and Entity Set?

ER Model is used to model the logical view of the system from data perspective which consists of these components:

An Entity may be an object with a physical existence – a particular person, car, house, or employee – or it may be an object with a conceptual existence – a company, a job, or a university course.

An Entity is an object of Entity Type and the set of all entities is called an entity set. e.g.; E1 is an entity having an Entity Type Student and the set of all students is called Entity Set. In ER diagram, Entity Type is represented as:



17)Discuss about ER Model and its Relationships?

ER model

- ER model stands for an Entity-Relationship model. It is a high-level data model. This model is used to define the data elements and relationship for a specified system.
- It develops a conceptual design for the database. It also develops a very simple and easy to design view of data.
- In ER modeling, the database structure is portrayed as a diagram called an entity-relationship diagram.
- Relationship:
  The association among entities is called a relationship. For example, an employee works_at a department, a student enrolls in a course. Here, Works_at and Enrolls are called relationships.
- Include diagram from 6Q

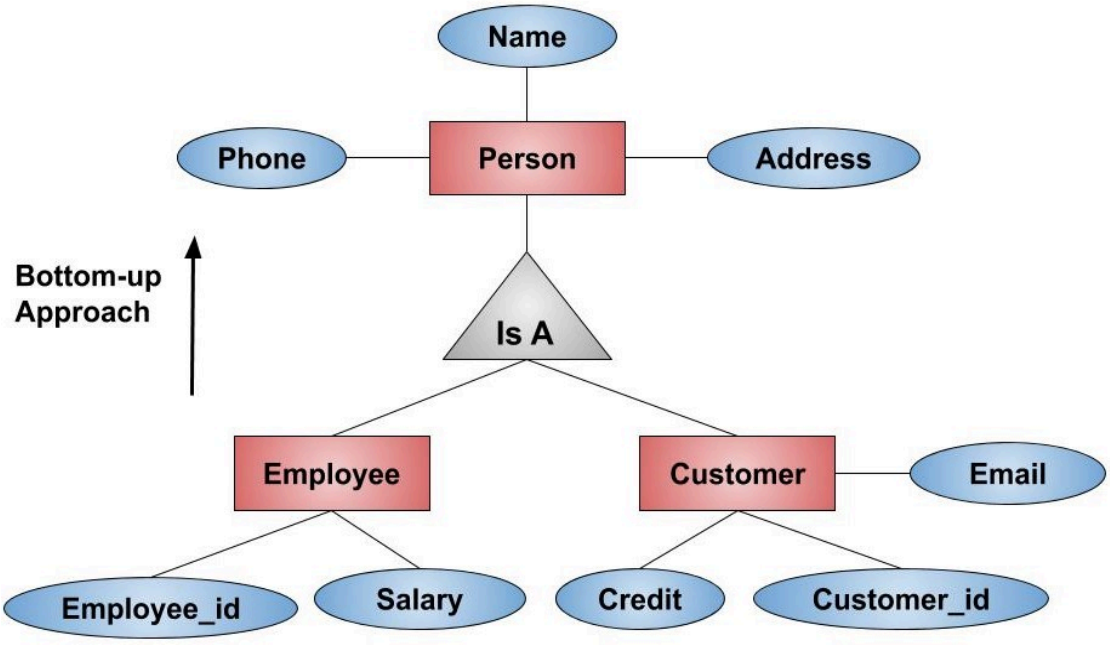
18)Discuss about generalization with a neat diagram?

Generalization is a bottom-up approach in which multiple lower-level entities are combined to form a single higher-level entity. Generalization is usually used

to find common attributes among entities to form a generalized entity. It can also be thought of as the opposite of specialization.

The following enhanced entity relationship diagram expresses entities in a hierarchical database to demonstrate generalization:

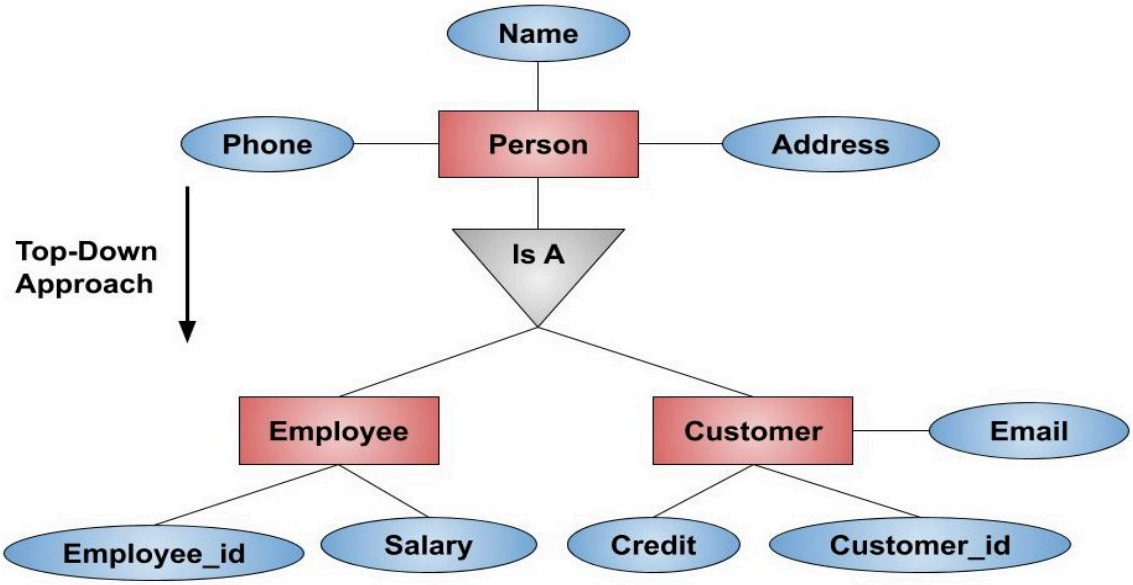
Explain in your own words



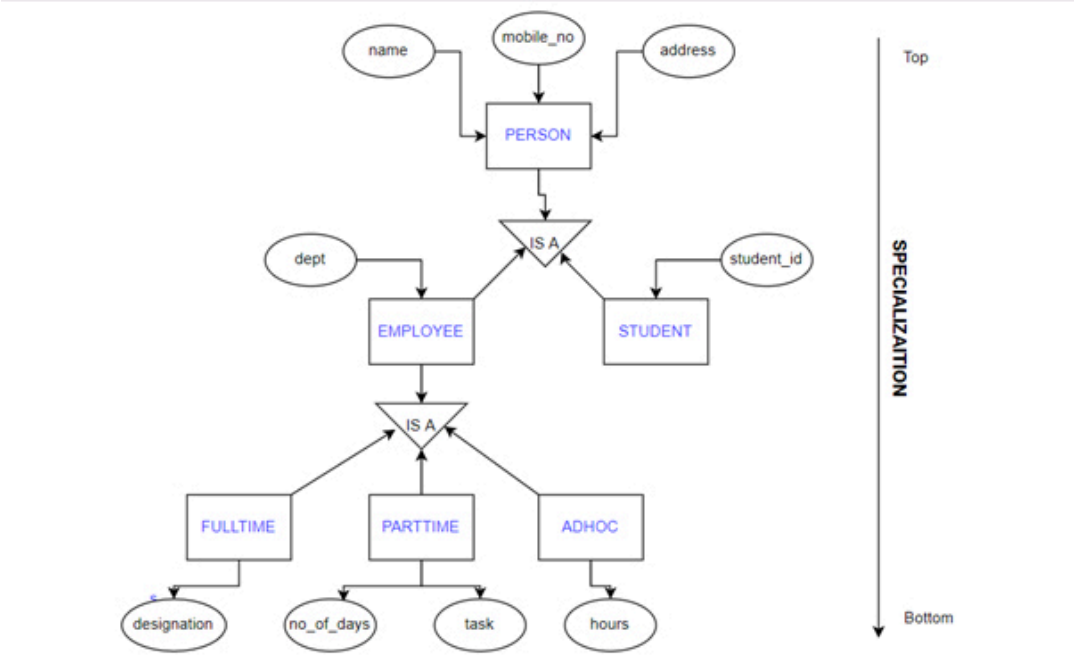
Generalization

19)Explain specialization with a neat diagram?

Specialization is a top-down approach in which a higher-level entity is divided into multiple specialized lower-level entities. In addition to sharing the attributes of the higher-level entity, these lower-level entities have specific attributes of their own. Specialization is usually used to find subsets of an entity that has a few different or additional attributes.
The following enhanced entity relationship diagram expresses the entities in a hierarchical database to demonstrate specialization:



Specialization

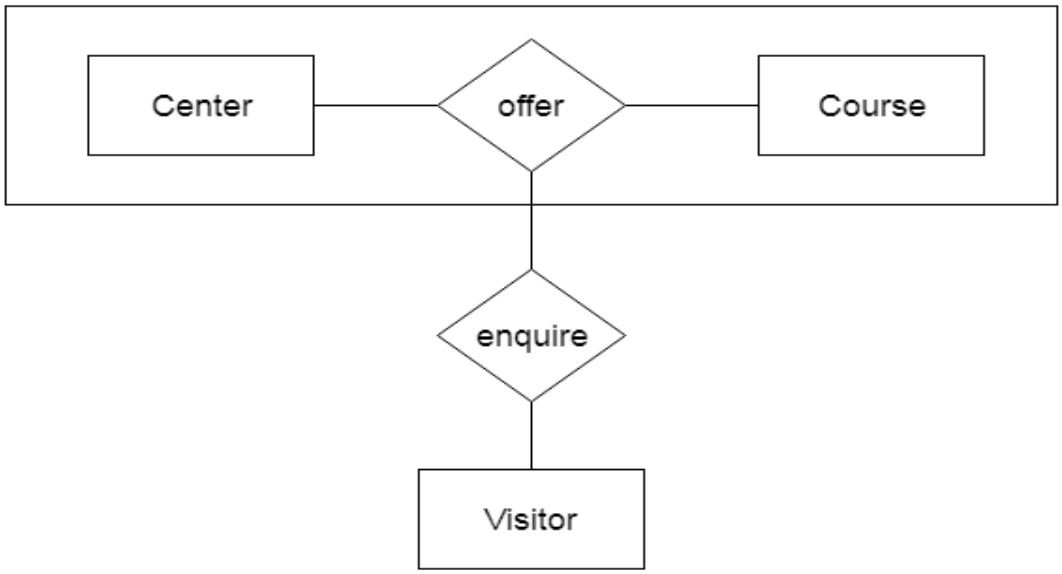


20)Describe aggregation with a neat diagram?

For 18,19,20 refer PART-B question no 5

Aggregation refers to the process by which entities are combined to form a single meaningful entity. The specific entities are combined because they do not make sense on their own. To establish a single entity, aggregation creates a relationship that combines these entities. The resulting entity makes sense because it enables the system to function well.

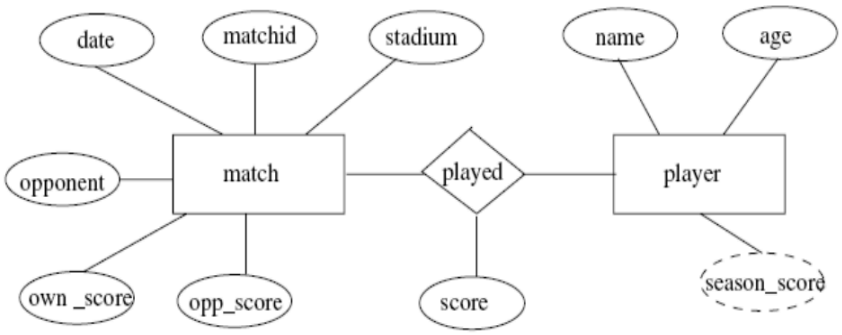
For example: Center entity offers the Course entity act as a single entity in the relationship which is in a relationship with another entity visitor. In the real world, if a visitor visits a coaching center then he will never enquiry about the Course only or just about the Center instead he will ask the enquiry about both.



PART-A

1 Construct an E-R diagram for keeping track of the exploits of your favorite sports team. You should store the matches played, the scores in each match,

the players in each match and individual player statistics for each match. Summary statistics should be modeled as derived attributes.



2) Let E1 and E2 be two entities in an E/R diagram with simple single-valued attributes. R1 and R2 are two relationships between E1 and E2, where R1 is one-to- many and R2 is many-to-many. R1 and R2 do not have any attributes of their own. Calculate the minimum number of tables required to represent this situation in the relational model.

We require 3 tables.

3) Analyze and find whether modifications made at conceptual level makes application programs written by users at view level to be modified in a database. Analyze your answer with illustration.

- Data independence refers characteristic of being able to modify the schema at one level of the database system without altering the schema at the next higher level
- If we do any changes in the conceptual view of the data, then the user view of the data would not be affected.

- If we do any changes in the storage size of the database system server, then the Conceptual structure of the database will not be affected.

Not complete!

4) We can convert any weak entity set to a strong entity set by simply adding appropriate attributes. Analyze why, then, do we have weak entity sets?

We want to avoid the data duplication and consequent possible inconsistencies caused by duplicating the key of the strong entity.

- Weak entities reflect the logical structure of an entity being dependent on
  another entity.
- Weak entities can be deleted automatically when their strong entity is deleted.
- Weak entities can be stored physically with their strong entities.

5) What are the responsibilities of a DBA? If we assume that the DBA is never interested in running his or her own queries; does the DBA still need to understand query optimization? Why?

The functions of a DBA include:

• Schema definition. The DBA creates the original database schema by executing a set of data
definition statements in the DDL.

• Storage structure and access-method definition.

• Schema and physical-organization modification.

1 Routine maintenance.

2 Periodically backing up the database.

3 Ensuring that enough free disk space is available for normal operations, and upgrading disk space as required.

– Monitoring jobs running on the Database.

6) Describe the structure of a DBMS. If your operating system is upgraded to support some new functions on OS files (e.g., the ability to force some sequence of bytes to disk), which layer(s) of the DBMS would you have to rewrite to take advantage of these new functions.

7) Why did relational models become more popular compared with other record based models?

In the early 1980s microcomputer-based implementations of database management systems, overwhelmingly based on the relational model, became available. Since that time, the relational model has gained wide acceptance and has continued to be the subject of research studies to extend its capabilities.

8) Describe the process to convert the ER model into relational schema.

ER diagrams can be mapped to relational schema, that is, it is possible to create relational schema using ER diagram. We cannot import all the ER constraints into a relational model, but an approximate schema can be generated. There are several processes and algorithms available to convert ER Diagrams into Relational Schema.
They are:

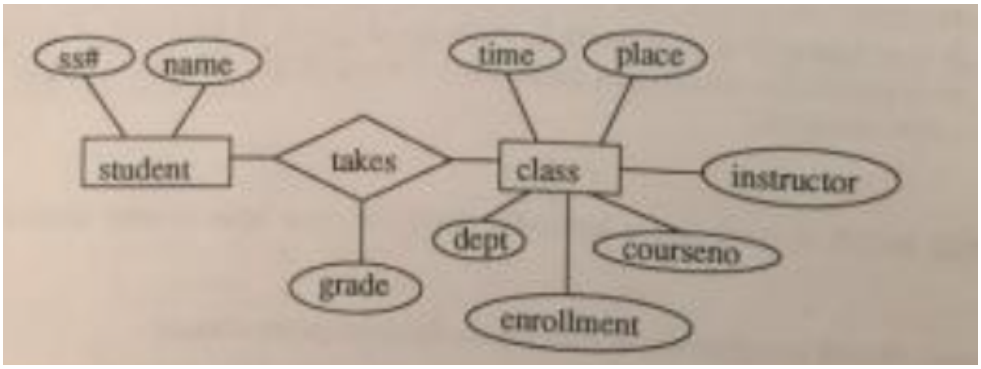
1. Mapping Entity
2. Mapping Relationship
3. Mapping Weak Entity Sets
4. Mapping Hierarchical Entities

{ ___DOUBT___ }

9) Discuss the disadvantages of the file processing system, and explain how these disadvantages are avoided in DBMS?

REFER PART B Q1

10) Design a relational database for a university registrar's office. The office maintains data about each class, including the instructor, the number of students enrolled, and time and place of the class Meetings. For each student - class pair, a grade is recorded.



PART C

1)List the advantages of DBMS.

Advantage of Database Management System (DBMS):

- Better Data Transferring
- Better Data Security
- Better data integration
- Minimized Data Inconsistency
- Faster data Access

- Better decision making
- Increased end-user productivity
- Simple.

2) List the database Applications.

- Accounting: payments, receipts, account balance, assets.
- Human Resources: employee records, salaries, tax deductions
- Manufacturing: production, inventory, orders, supply chain
- Online Retails: order tracking, customized recommendations
- Banking and Finance: all transactions
- Credit card Transaction: generation of monthly statements.
- Finance: storing information about holdings and sales, 3. Universities: registration, grades
- Airlines: reservations, schedules
- Telecommunications: keeping records of calls made, generating monthly bills.

3) Define instances and schemas of the database.

Definition of instance: The data stored in a database at a particular moment of time is called an instance of database. Database schema defines the variable declarations in tables that belong to a particular database; the value of these variables at a moment of time is called the instance of that database.
Definition of schema: Design of a database is called the schema. Schema is of three types:

- Physical schema
- logical schema
- view schema.

4) Discuss Data Independence.

Data Independence

- Data independence can be explained using the three-schema architecture.
- Data independence refers to the characteristic of being able to modify the schema at one level of the database system without altering the schema at the next higher level.

There are two types of data independence:

1. Logical Data Independence
   - Logical data independence refers to the characteristic of being able to change the conceptual schema without having to change the external schema.
2. Physical Data Independence
   - Physical data independence can be defined as the capacity to change the internal schema without having to change the conceptual schema.

5)How application programs access databases?

6) Define (i) Database (ii) DBMS.

• Database: Database is a collection of interrelated data.

• DBMS: database is a collection of interrelated data and a set of programs can access that system.

7) List out main components of Database storage structure?

There are four main components on which the working of a DBMS depends. This includes:

- Data: The main component is the data. The entire database is set based on the data and the information processed based on it. This data acts as

a bridge between the software and hardware components of DBMS. This can further be divided into three varieties:

- User Data – The actual data based on which the work is done
- Metadata – This is the data of the data, i.e., managing the data required to enter the information
- Application MetaData – This is the structure and format of the queries

8) What are the main responsibilities of the Transaction management component?

Transactions are a set of operations used to perform a logical set of work. A transaction usually means that the data in the database has changed. One of the major uses of DBMS is to protect the user's data from system failures.
It is used to solve Read/Write Conflict. It is used to implement Recoverability, Serializability, and Cascading. Transaction Management is also used for Concurrency Control Protocols and Locking of data. Transactions can be implemented using SQL queries and Server.

9) Outline main functions of Query Processor.

Query Processing is the activity performed in extracting data from the database. In query processing, it takes various steps for fetching the data from the database. The steps involved are:

1. Parsing and translation
2. Optimization
3. Evaluation

10)Define (i) Entity (ii) Attribute

Entity:An entity is an object that exists and is distinguishable from other objects . Example : specific person , company , event , plant.
Attributes:Entities are represented by means of their properties, called attributes. All attributes have values. For example, a student entity may have name, class, age as attributes. There exists a domain or range of values that can be assigned to attributes. For example, a student's name cannot be a numeric value. It has to be alphabetic. A student's age cannot be negative, etc.

11)Define Relationship and Relationship set.

Relationship:The association among entities is called a relationship. For example, employee entities have relation work.

Relationship Set: Relationships of similar type are called relationship sets. Like entities, a relationship too can have attributes. These attributes are called descriptive attributes. Degree of relationship The number of participating entities in an relationship defines the degree of the relationship

- Binary = degree 2
- Ternary = degree 3
- n-ary = degree

12)Discuss about Data Definition language.

- It is used to define database structure or pattern.
- Using the DDL statements, you can create the skeleton of the database.
- Data definition language is used to store the information of metadata like the number of tables and schemas, their names, indexes, columns in each table, constraints, etc.

Here are some tasks that come under DDL:

- Create: It is used to create objects in the database.
- Alter: It is used to alter the structure of the database.

- Drop: It is used to delete objects from the database.
- Truncate: It is used to remove all records from a table.
- Rename: It is used to rename an object.
- Comment: It is used to comment on the data dictionary.

13)Discuss about Data Manipulation language.

It is used for accessing and manipulating data in a database. It handles user requests.

- Select: It is used to retrieve data from a database.
- Insert: It is used to insert data into a table.
- Update: It is used to update existing data within a table.
- Delete: It is used to delete all records from a table.
- Merge: It performs UPSERT operation, i.e., insert or update operations.
- Call: It is used to call a structured query language or a Java subprogram.

14) List responsibilities of a DBA.

The functions of a DBA include:

•The DBA creates the original database schema by executing a set of data definition statements in the DDL.

• Storage structure and access-method definition.

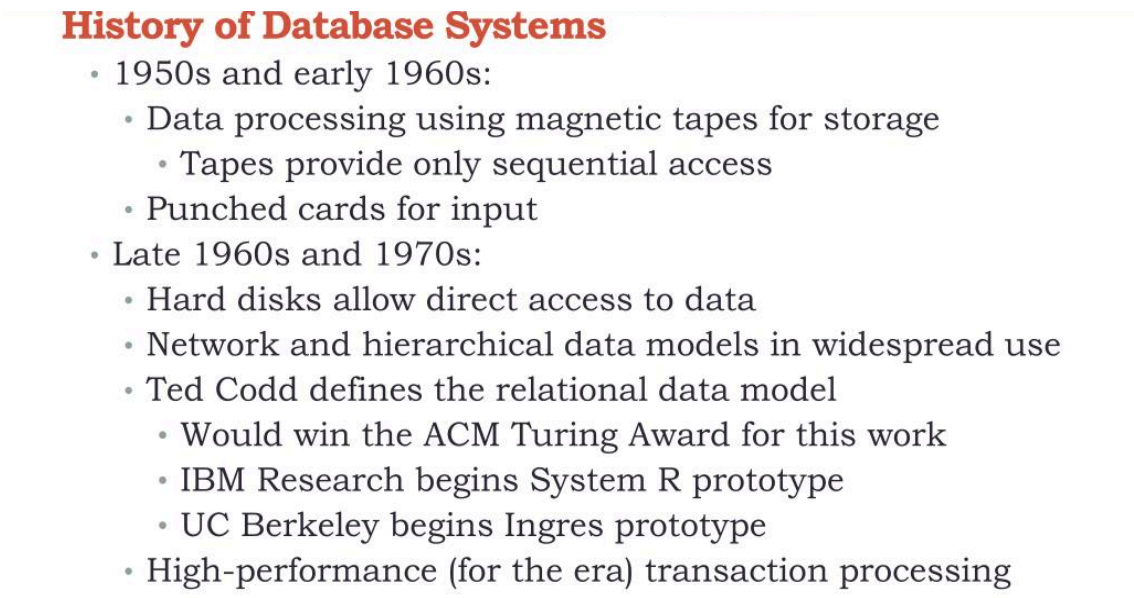
• Schema and physical-organization modification.

1. Routine maintenance.

2.Periodically backing up the database.

3.Ensuring that enough free disk space is available for normal operations, and upgrading disk space as required. – Monitoring jobs running on the Database.

15) Outline the History of Database Systems.



16) Discuss how you can change the data in the table.

The UPDATE statement changes existing data in one or more rows in a table.

17)List various types of attributes.

In ER diagram, attributes associated with an entity set may be of the following types-

1. Simple attributes
2. Composite attributes
3. Single valued attributes
4. Multi valued attributes
5. Derived attributes
6. Key attributes

18) Discuss How you can alter and destroy tables?

The SQL ALTER TABLE command is used to add, delete or modify columns in an existing table. You should also use the ALTER TABLE command to add and drop various constraints on an existing table.

Syntax

The basic syntax of an ALTER TABLE command to add a New Column in an existing table is as follows.

```
ALTER TABLE table_name ADD column_name datatype;
```

DROP TABLE:

The DROP TABLE command deletes a table in the database.
The following SQL deletes the table "Shippers":

Example:

```
DROP TABLE Shippers;
```

19) Define a data model? List the types of data model used.

REFER PART B Q 3

20)List the levels of data abstraction.

REFER PART B Q 2