JMS

INDEX

Name: Sudhanshu Ranjan Adarsh Std.: _____ Sec.: _____

Roll No.: 23- - - - - -80 Sub.: DBMS

DCE

S. No.	Date	Title	Page No	Teacher's Sign / Remarks
1.	2024	Introduction to DBMS	1 - 10	Lokesh
2.	2024	Database System Concept and Architecture	11 - 26	Lokesh
3.	2024	Data Modelling Using E.R. Model	27 - 37	Lokesh
4.	2024	Relational Model	~~45~~ - 52	Lokesh
5.	2024	Normalization	38 - 44	Lokesh
6.	2024	Database Access and Security	53 - 62	Lokesh
7.	2024	MySQL and SQL	63 - 86	Lokesh

A DBMS is an application software that controls the data in a database that include storage, retrieval and mangemet.

EF Codd - Father of DBMS

JMS

Unit - I 1. Introduction to DBMS

i) Data :- The representation of raw facts in formalized manner suitable for communication, or processing by human or machine.

ii) Data Field (Data Item) :- A set of characters which are used together to represent a specific data element, is called data field.

iii) Record :- It is defined as the collection of related data items like class record.

iv) File :- It is defined as the collection of related records like student file consits of students information.

v) Information :- The data that has converted into some useful form, is called information.

vi) Data Processing :- The process to make raw data useful for future, is called data processing.

vii) Database :- A well-organized collection of structured and inter-related data, is called database.

viii) Database System :- The system that helps to keep data security and maintains it with user permission to retriere and manipulate data, is called database System.

Types of Database ⊙

A database is a collection of related objects including tables, form etc are created, organised and managed by DBMS.

There are four types of ~~ab~~ database:-

i> Knowledge database
ii> Decision making database
iii> Graphics Oriented dbs.
iv> Bibliographic database

i> Knowledge database:- It is used in AI application.
ii> Decision Making database:- It is used in administrative and management tasks.
iii> Graphic Oriented dbs:- It is used in computer oriented (graphic) design and animation.
iv> Bibliographic database:- It is used in all business organisation

Purpose of Database

The major purpose of database is to provide users with abstract view of data. Following are the some other purpose of database:-

i> Data Storage:- It provides a structured way of to store data efficiently.
ii> Data Retrieval:- It allows user to quickly retrieve specific data.
iii> Data Security:- It offers security features like authentication.

iv> Data Sharing :- It facilitate data sharing among multiple user.

v> Data backup :- It includes mechanism for regular data backup and recovery.

Database Language

To provide the various facilities to different types of user, A DBMS provide one or more specilized progamming language, called Database Language.

Different DBMS provide different database languages :-

i> SQL :- Structured Query Language.

ii> DDL :- Data Description Language

iii> DML :- Data Manipulation Language

Features of Database

Various features of database as follow :-

- i> Control of data redundancy
- ii> Data Consistency and Data Sharing
- iii> More information from same amount of data
- iv> Privacy and Security
- v> Improved data integrity
- vi> Physical data organization of database
- vii> Improved backups and recovery

Characteristics of Database ⊙

Following are the main characteristics of database as follows:-

i) Centralized System:- It is centralized and integrated data file which allow the sharing of data in multi-user environment.

ii) Reduction of Redundancy:- It organizes in such manner that it minimize the duplication of data.

iii) Data independency:- It takes the data definition as a part of database so it makes the data independent of application program.

iv) Data Consistency:- In a centralized system, there is no chance of multiple copies and hence data remains consistent.

v) Easy Access:- It provides a direct access to all data by all application with equal ease.

vi) Logical Relationship:- It provides a logical relationship between its record and data.

Advantages of Database Systems ⊙

Following are some advantages of centralized database system:-

i> Redundancy Control:- The information in a database system is integrated and it controls the redundancy of data.

ii> Sharing of data:- The data remains in a centralized database and a security may provide to avoid accendental change

iii> Data Consistency:- In a centralized system, there is no change of multiple copies and so data remains consistent

iv> Scale Economy:- The data requirements are over in single database so it saves time and cost.

v> Backup and Recovery:- The database has a better backup and recovery system.

Disadvantages:- ⊙

i> High Cost:- The DBMS is very large so it is expensive to purchase.

ii> High Hardware Cost:- It is a large program and need additional memory and processing power results to cost.

iii> High Programming Cost:- Hiring experienced programmer for programing result to cost

iv> Slow Processing:- The size of database increase, access time also increase so slow processing

v> Difficult Recovery:- If the system failure, it becomes difficult to recover.

Classification of DBMS Users

The person who use the proposed database for queries or accessing the data, called DBMS User.

The DBMS User classifies into three broad categories :-

i> Actors on the Scene
ii> Controllers
iii> DBMS worker behind Scene

i> **Actors on the Scene** :- The main aim of database is to provide an environment for storing and accessing data.
It can classify into two category :-
i> Application Programmer
ii> End User

i> Application Programmer :- The computer professional who interact with system by writing program in host languages, called application programer
(Pascal, Fortran, C, C++, COBOL)

ii> End Users :- A person for whom the software program and system is designed, is called end user. They access the database from terminal end.
Three types of end user :-
i> Sophisticated User
ii> Specilized User
iii> Naive User

Sophisticated - शिक्षित

JMS

i> Sophisticated User!- The user who interact the system without writing application program, is called sophisticated user.

ii> Specilized User!- They are special type of sophisticated user who write specilized database program which are not present in tredifional data processing program.

iii> Naive User!- They are totally unsophisticated user who never know writing program

ii> Controllers!- This is second group of person who actually are DBMS controller include database administrator, database designer and system analyst.

The database designer creates the backbone of database and it is created and manipulated by database administrator. The initial job is done by system analyst because they do overall ground work.

DBMS controllers as follows!-

i> System Analyst
ii> Database Designer
iii> Database Administrator

i> System Analysts:- Those user who investigate and analyse the whole system and check for feasibility of new system, so that DBMS can adopted, called system analysts.

Functions:- i> To analyse the condition and find if they favourable or not for DBMS.
ii> To analyse user requirement.
iii> To propose data item with size, type.

ii> Database Designer:- The person who consider the preliminary design purposed by system analyst to design conceptual schema of database, is called database designer. — Re

Functions:- i> To design conceptual schema for database.
ii> To define views of user classes
iii> To design physical file structure

iii> Data Base Administrator (DBA):- It is a person who provide necessary technical support for implementing strategic decision so, DBA is responsible for overall control at technical level. It has central control of data & program.

- full access, single person
- add user, delete user, who can view database

Functions:- i> Schema Definition:- DBA is responsible for creating original database schema by executing a set of data definition statement in DDL.

ii> Physical Schema definition:- The DBA must decide how data is to be represented in storage database. This process is called physical schema design.

iii> Maintainance of Database:- DBA is responsible for organising the system in such a way to get best for enterprise and makes appropriate adjustments.

iv> Granting of authorization for data access:
- Revoking -

iii> Workers behind the Scene:- Those people who actually performs the job of implementation of DBMS, is workers behind scene.

This group of people include :-
i> DBMS system designer
ii> DBMS system implementers
iii> Tool developers
iv> Operators
v> Maintainance personnel

i> DBMS system designer:- It provides all the physical facilities designs for the system to be developed.

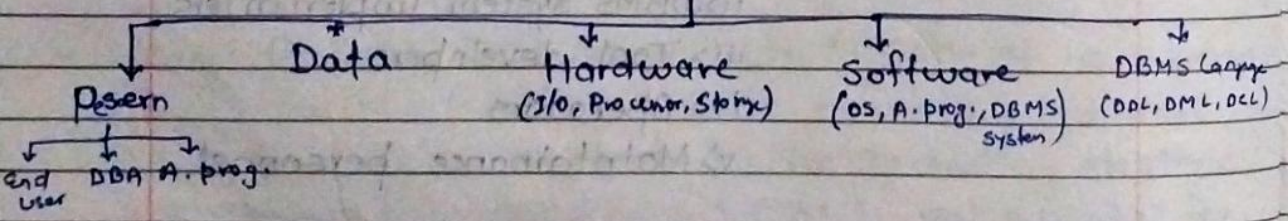
ii> DBMS system implementers:- Those people who help for hardware implementation of entire system.

iii> Tool developers:- Those people who help in developing the software tools for required DBMS.

iv> Operators:- Those people who know running and operation of hardware part of complete system. (may be many people)

v> Maintainance Personnel:- The technical people who is responsible for operation, maintainance and upgradation of systems.

Components of database



Data Model

- **Conceptual Data Model**
  - E-R Model
  - Object Oriented Mo.
  - Semantic Model
  - Functional Mo.
- **Representational Data Model**
  - Relational Mo.
  - Network Mo.
  - Hierarchical Mo.

  (Traditional D. Mo.)
- **Physical Data Model**
  - Unifying mod
  - frame

Unit-I Database System Concept and Architecture

The method used inside the DBMS to organise the structure of database, is known as data model. It is a collection of conceptual tool that describes data, data relationship, data semantics and data constraints.

On the types of concept, it can classify into three types:-

i> Conceptual data model
ii> Representational data model
iii> Physical data model

i> **Conceptual Data Model**:- It is a high-level representation (object base) of data and the relationship among them within a system.

Key Characteristics:

i> Entity:- It represents a ~~real-world~~ fundamental concept like a student or place.

ii> Attributes:- It represents some characteristics of entity.

iii> Relationships:- It defines how the entities are connected to each other.

It is called object base data model because it uses entity, attributes and relationship to show concept of database, how is it organise in DBMS. (object like)

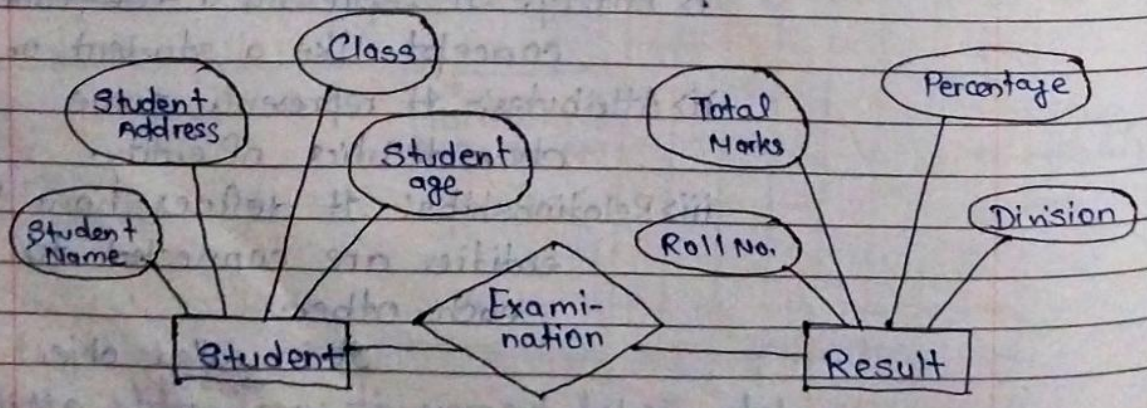
There are four types of conceptual data model :-

i> E-R (Entity - Relationship) Model
ii> Object Oriented Model
iii> Semantic data Model
iv> Functional data Model

i> E-R Model :- It is a conceptual framework used in database design to represent the relationship between different entities with a system.

E-R diagram consists of four components :-
i> Rectangle, ii> Ellipse, iii> Diamonds, iv> Lines
i> Rectangle :- It represents entity set.
ii> Ellipse :- It represents attributes.
iii> Diamond :- It represents relationship among entity set
iv> Line :- It links attributes to entity set and entity set to relationships.

Diagram :-



ii> Object Oriented Model:- The model which is based on a clear object oriented approach, is called Object oriented model.

iii> Semantic Data Model:- The model which gives more emphasis on semantics while operating on data log, is called semantic data model.

It defines meaning of data within it interrelationship

semantic checks (grammatical constraint) every time on data.

iv> Functional Data Model:- The data model which uses the data operating according to pre-defined module, is called functional data model.

query processor, storage manager are two modules.

ii> Representational Data Model:- It describes the data (Record base) in much detail as possible without regard how they will be physically implimented in database. Normalization is occur at this level.

There are three types of representational data model:-

i> Relational Model
ii> Net Work Model
iii> Hierarchical Model

} Traditional data model

i> Relational Model:- It uses collection of tables to represent data and relationship among them. Each table has multiple column with unique name. One table connected with other by using key & relationship.

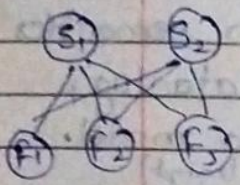
predicate calculus is used

Entity → rows
attribute → columns

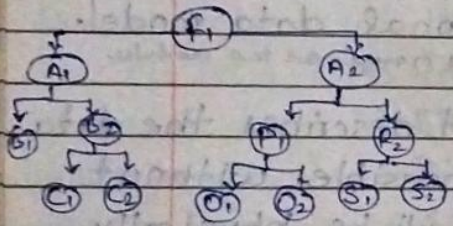
Two-dimensional array is used to implement

Features:- i> Data is organised in terms of rows and columns.
ii> The intersection of a cell must give a single value.

ii> Network Model:- In this model, the data is represented by collection of records and relationship among them by links. It uses data structure diagram for its representation and links and pointers for its implimentation. It is extension of hierarchical model. Artificial record is used to implind



iii> Hierarchical Model:- It contains collection of records connected in form of tree. It is similar to a network model in structure except that one child can have only one parent. It allows one to many relationships while a network model allow many to one.



Network M.	Hierarchical M.
i> A child may have many parents node.	A child may have only one parent node.
ii> It represents data in many-to-one relationship.	It represent data in one-to-many relationship.
iii> It is easier and faster to access the data.	It is not easier and faster to access the data.
iv> Organise data in a graph structure.	Arrange data in tree like structure.

iii> Physical Data Model:- It is a database specification representation (Low level) of how data is organized and accessed in database. It represents information like record formats, record ordering and access paths. It shows all table structures like key, column name, type etc.

Type-
* Unifying model.
* Frame model.

Schema

description of database

collection of metadata

The logical and visual blueprint of a database that defines its structure, organization and relationship, is called schema. It describes the objects that are represented in database and relationship among them. It is described using SQL.

Student
| Name | Age | Roll |

There are three types of schema in database system:-

i> Physical schema (Internal)
ii> Logical schema (conceptual)
iii> Subschema schema (external)

i> Physical Schema:- The structural description of database at internal level, called physical schema. It defines the complete structure of database. It defines structure in terms of size, type etc.

Same diagram

ii> Logical Schema:- The structural description provided at intermediate level, called logical schema.

iii> Subschema:- The structural description provided at higher level, called subschema. There can be more subschema, and only one logical & physical schema for one database.

Database State

The data stored in a database at a particular moment of time, is called database state. It is also called current set of instances. It is an extension of schema.

The distinction between database schema and database state is that when we define a new database, we specify its database schema only. At this point, the corresponding database state is empty state with no data.

Various types of database states are:-

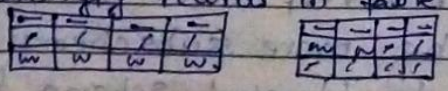
i) Empty State
ii) Initial State
iii) Online state
iv) Offline state
v) Restoring State
vi) Recovering state
vii) Recovery Pending state
viii) Suspect State
ix) Emergency State

Instances of Database

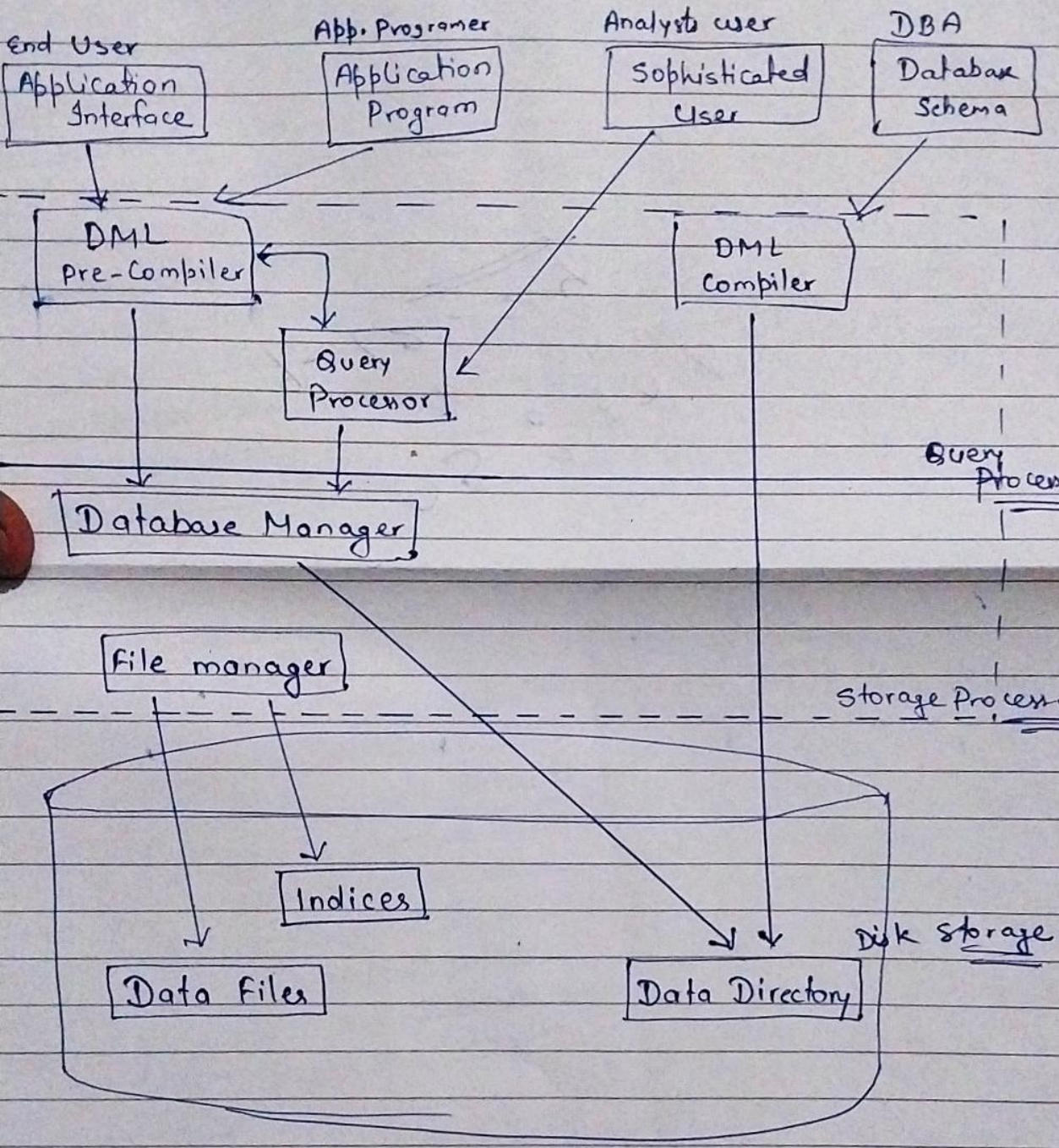
The collection of data stored in the database at a particular moment is called instance of a database. It allows to change or modify the information already contained in it. A database having many instances shows that database has changed.

The instance represents the data actually stored in records of database.

changing records in table



DBMS Architecture/Structure/Component



DBMS Architecture/Abstraction ②

It hides complexity through some levels of abstraction.

The characteristics that allows program-data and program-operation independance, is called data abstraction.

The DBMS organise, process and access the queried data element from database. It manages large bodies of information. It provides an abstract view of data. The three levels of architecture are three different view of data:-

i> Physical ~~View~~ Level
ii> Logical ~~View~~ Level
iii> View level

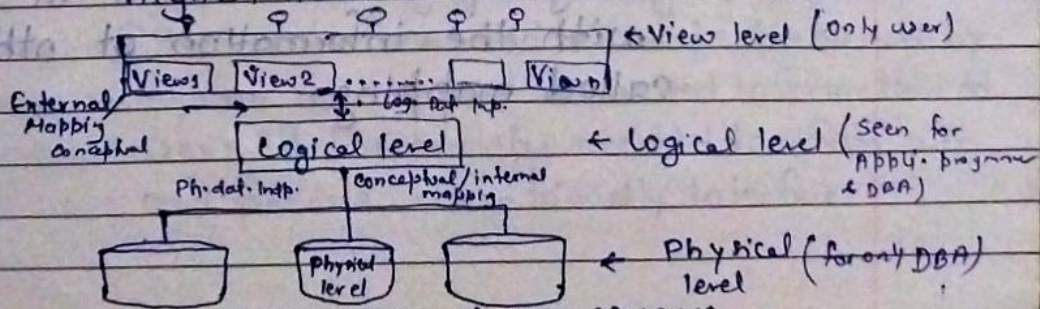
i> Physical Level (Internal Schema):- It describes how the data is actually stored on physical storage medium. It describes complex low level data in detail of storage. It may or may not be changed.

aspects:
i) Storage allocation
ii) Access path

ii> Logical Level (Conceptual Schema):- It defines what data are stored in a database and what relationships exists between those data. It hides details of physical storage structure & describe data type entity etc. It can't be changed.

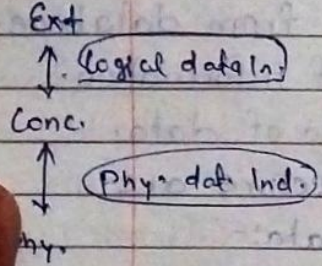
iii> View Level (External Schema):- It provides a security mechanism for preventing users from accessing part of database. It is the highest level of abstraction and visible to user.

Ex. Diag.
i. Map.
ii. D. Indp.



Data Independance

The ability to modify the definition of schema at any level without affecting the schema definition in the next higher level, called data independance.



The two levels of data independance are:-

i) Physical Data Independance
ii) Logical Data Independence

i) Physical Data Independence:- Making changes to the physical schema without changing the logical schema, called physical data indep. It is used to separate conceptual level from internal level.

ii) Logical Data Independance:- Changing the logical schema without changing the external schema, is called logical data Independence. It is used to separate external schema from ~~logical~~ conceptual schema.

Mapping ⊙

The methodology to link the group of information of one level with the information at other level, is called mapping.

There are two types of mapping between three different views:-

i> External/Conceptual Mapping

ii> Conceptual/Internal Mapping

i> External/Conceptual Mapping:- Each external schema is related to conceptual schema by external/conceptual mapping. A mapping between the external and conceptual view gives correspondance among records and relationships of external and conceptual view.

ii> Conceptual/Internal Mapping:- Conceptual schema is related to internal schema by conceptual-internal mapping. It enables the DBMS to find actual record in physical storage. Mapping between conceptual and internal levels specifies the method of deriving conceptual record from physical database.

DBMS Interface

A user interface that allows for the ability to input queries to a database without using query language, called DBMS interface. Various steps performed by system to provided requested information to user. All these steps are hidden from user and provide user-friendly interface.

Various DBMS user interface are:-

i> GUI
ii> Menu Based Interface
iii> Form Based Interface
iv> Interface for DBA
v> Interface for parametric user
vi> Command Oriented Interface
vii> Report Writter

i> GUI:- The Graphical User Interface (GUI) presents the graphical form to wer. The wer can communicate with computer, wing presented diagrammatic form.

ii> Menu Based Interface:- It user list of options, called menu. The user can select the option from list of option for request. There can be multiple menu to a main menu.

iii> Form Based Interface:- In this, the wer can interact through a form with system. A form is displayed to each wer and wer then fills the required data in form.

iv> Interface for DBA:- A special interface that allows special commands for database system and work between database system and wer, is called DBA Interface. These commands are creating and changing database schema, reorganizes

C-29 (50)
96 (14)

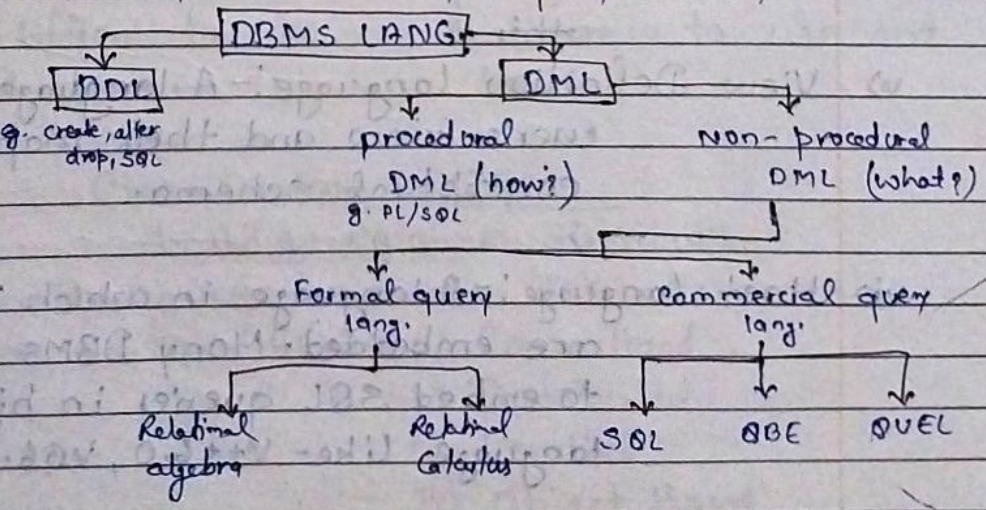
v> Interface for Parametric User:- It is developed by programmer to have a small set of operation that user must perform repeatedly.

vi> Command Oriented Interface:- It is also called line Oriented Interface. In this, the user type a query expression in some relational query language.

vii> Report Writters:- A type of interface in which a report is generated for required output. The answer of every query on these interface is always a report.

Database Languages

It provides an appropriate languages for each categories of user to express database queries and updates. These languages are used to create and maintain database on computer like- dBase, MS-ACCESS, Foxpro, Oracle, Paradox etc.



DBMS provide many languages to facilitates to users:-

i> Data Description language:- It is a language that allows user to define data and their relationships to other types of data. It is used to create file, database, tables etc but not used to manipulate data in database

ii> Data Manipulation language:- A language that provides a set of operation to support the basic data manipulation operation on data held in database. It allows user to insert, update, delete and retrieve data from database.

iii> Data Sub language:- It's combination of both DML and DDL.

iv> Storage Definition language:- A language used to specify internal schema in database. ~~In this~~

v> View Definition language:- A language used to specify user's view and their mapping to conceptual schema.

vi> Host language:- A language in which DML commands are embedded. Many DBMS have a facility to embed SQL queries in high level programming language like- V++6.0, VB6.0 etc.

vii> Fourth Generation language:- It is compact, efficient and procedural programming language wed to improve the productivity of DBMS.
Ex - SQL, code generator etc

viii> Data Control language:- DCL statement control access to data and database wing statements like GRANT and REVOKE.

Difference Between Procedural DML and Non-procedural DML

	Procedural DML	Non-Procedural DML
i>	It specify how output of DML statement be obtained.	It specify what output is to be obtained.
ii>	It treats record individually	It operates on set of records.
iii>	It's embedded in high-level language.	It uses directly queries to DBMS to find certain data.
iv>	Network and hierarchical DMLs are normally procedural.	Relational DBMS use SQL, QBE for data retrieval.
v>	It is difficult to use and learn.	It is easy to use and learn.

Classification of DBMS based on database distribution

There are four main distribution system for DBMS:- i> Centralized
ii> Distributed
iii> Parallel
iv> Object Based

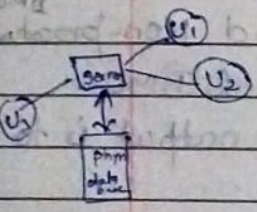
DMS

Date:
Page No.: 24

i> Centralize DBMS:- It consists of a single processor together with its associated data storage device. The management of the system and its data are controlled centrally.

Advantages:

i> Improved Data Consistency:- It stores all the data in a single location.



ii> Enhanced Security:- It allows for centralize control over data access and makes it easier to implement

iii> Simplified Data Mangement:- With data stored in one location, management tasks are more straightforward.

iv> Cost Efficiency:- It require less hardware leading to lower cost.

v> Improved Data Integrity:- It intigrates the data ensuring that data remain accurate

Disadvantages:

i> Single Point Failure:- If the central system crashes, all systems can go down.

ii> Scalability Issue:- It can struggle to handle load leads to slow response

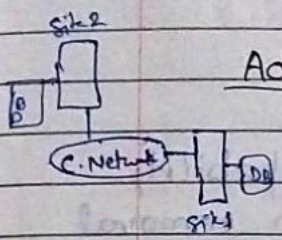
iii> Network Dependency:- All system are dependent to network for demand.

iv> Limited Flexibility:- These are less flexible in adopting to need of distributed demands.

Homogenow & Hetrogenow are type of DDBMS

JMS

ii> Distributed DBMS!- In this, actual database and DBMS software are distributed from various sites. It is a software that manages distributed database to make this distribution transparent to user.



Advant!
- i> Modular Development
- ii> Better Response
- iii> More Relaible
- iv> Lower Cost

Disadvt!
- i> Need for expensive software
- ii> Large processing overhead
- iii> Data integrity problem
- iv> Improper data distribution

iii> Parallel DBMS!- It is wed to improve performance through parallelization of various operation like loading data and building indexes. In this, multiple CPU connected to a computer.

Advant!
- i> Better speed of processing
- ii> Better cost
- iii> Reliablity and improve data capacity

Disadvt!
- i> Difficult Programming
- ii> Problem in result communication
- iii> Huge power consumption
- iv> Need of better cooling technology

iv> Object Based DBMS:- A DBMS in which information is represented in form of objects, is called object base DBMS. It supports the creation and modelling of data as object.

Advant:

i> Improved Performance
ii> Enriched modelling capability
iii> Independence mismatch removal
iv> More expressive query language
v> Multilevel complexity

Disadvt:

i> Lack of universal data modelling
ii> Lack of experience
iii> Lack of standards
iv> Complexity and competition
v> lack of support for views & security.

Lokesh

ISAM - Indexed Sequential Access Method

JMS

Unit-II 3. Data Modelling using E.R Model

The process of creating data model for information by applying formal technique, called data modelling.
It is a method to document a software using diagram & symbols.
There are three types of data modelling :-

i) Primitive Model
ii) Traditional data Mod.
iii) Semantic "

i) Primitive Model :- These were developed with the concept of conventional file system. These are more associated with storage. It is also called file based system. It use record to store entities in a file.

ii) Traditional Model :- The data model which are used to describe data at logical and view level, called traditional model. These are used as record base model.
- Relational Mod.
- Network Mod.
- Hierarchical Mod.

iii) Semantic Model :- The data model which consider semantics at time of operation on data, called semantic model. Data is featured in rules and checked each time.

An unique objects that exist in real world & has its own identity. — Entity

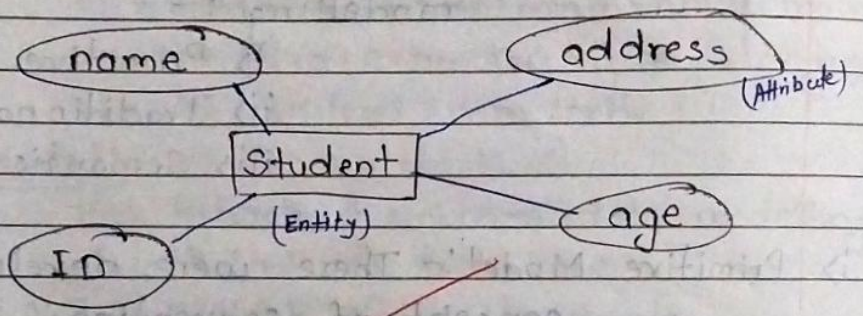
JMS

E-R Model Concept

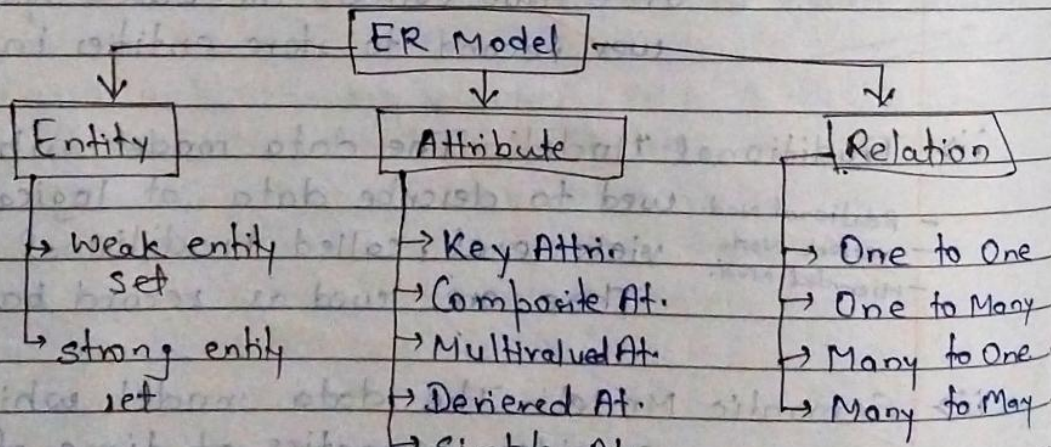
PP Chen 1976

The Entity-Relationship model was introduce by PP Chen in 1976. It is conceptual data model that views the real world as entities & relationship.

School Database:-



Components of ER Diagram



Entity

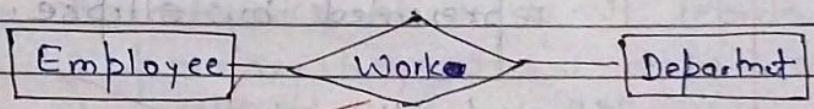
It is a thing in real world that distinguishable from all object. Anything about which we store information, called entity.

Types
i> Dependent
ii> Independent

Entity Set

It is a set of entity of some type that share some property. Like-

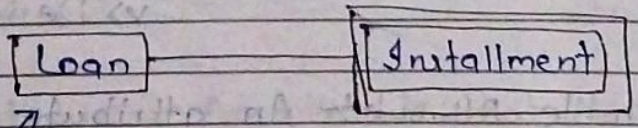
There can be many entity type in one relational database



There are two types of entity set:-
i> Weak Entity set
ii> Strong Entity set

i> Weak Entity Set:- An entity that depends on another entity, called weak entity set. It doesn't contain any key attribute and it is represented by double rectangle. It is also called child or subordinate entity.

It has no Primary key



Strong Et. set

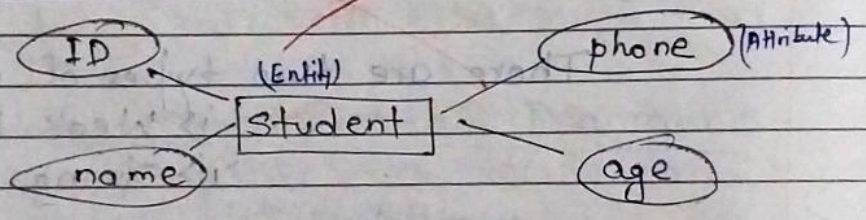
Weak Et. set

ii> Strong Entity Set :- An entity set that contain sufficient attribute to uniquelly identify all entity, called strong entity set. It is represented by single rectangle. It is called dominant entity.
(It has Pri. K)

Attributes

The attribute is used to describe the property of entity. It contain any no. of attribute. It is represented by ellipse.

Student Attribute :-

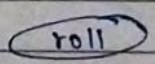


There are five types of attribute :-

i> Simple Attri. (atomic att.)
ii> Composite Attri.
iii> Key Attri
iv> Multivalued Attri.
v> Derived Attri.

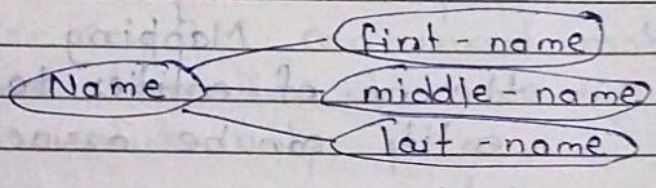
i> Simple Attribute :- An attribute that cannot be further subdivide into components, is called simple attribute. It is represented by ellipse.

Ex-

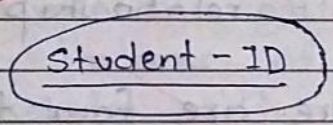


Single Value Attri - Single value (age)
Multi-vale Attri - Multible (Hobby)

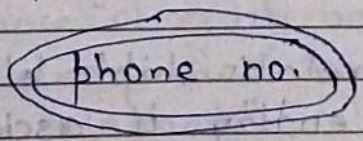
ii> Composite Attribute:- An attribute that can be split into component, is called composite attribute. It is represented by ellipse and connect with them.



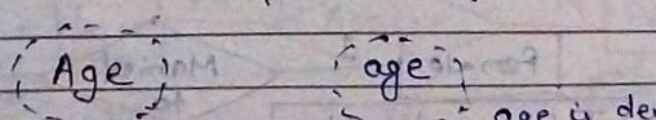
iii> Key Attribute:- It is used to represent main characteristics of entity. It represents a primary key. It is represented by ellipse with underline.



iv> Multi-valued Attribute:- The attribute which have more than one value, is called multi-valued attribute. It is represented by double ecllipse.



v> Deriered Attribute:- An attribute that is derived from other attribute, called derived atti. It is represeted by dashed ecllipse.



The number of occurance of one entity for a single occurance of related entity. → Cardinality

JMS

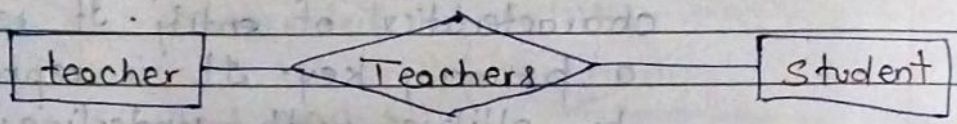
Relationship/Mapping Constraint
(Cardinality, Degree)

No. of entities associated with relationship is called degree.
- Unary/Recursive
- Binary relati.
- Ternary rela.

Ex- n=1 degree=1

A relationship is used to describe relation between entity. It is represented by diamond or rhombus.

Mapping constraint express the no. of entities to which another entity can be associated via relationship set.



relationship.

No. of relationship is called Cardinality ratio.

There are four types of mapping constraint or relationship :-

Binary relati.
- deg=2
∴ two entity

i) One to One relat.
ii) One to Many relat.
(i) Many to One relat.
iv) Many to Many relat.

i) One to One relationship :- When only one instance (two table) of entity is asciated with relationship is called one to one relationship.

Ex- Female can marry to one male and male can marry to one female.

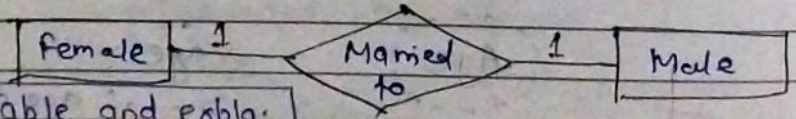


Table and expla. of diag. & Table ← all relationship

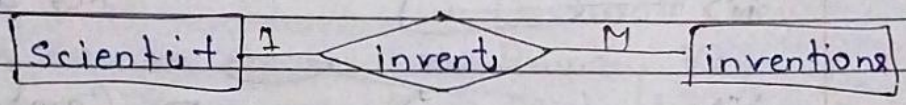
Teacher HOD Depart.

i. Total Participation - double line

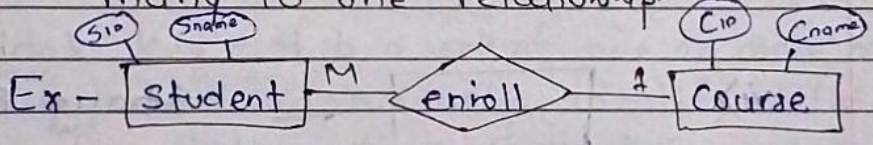
ii. Partial Part-

iii> One to many relationship:- When only one instance (~~two table~~) of entity on left and more instance of entity on right associates with relationship, is called one to many relati.

for ex- Scientist can invent many invention, but invention is done by specific scientist only.

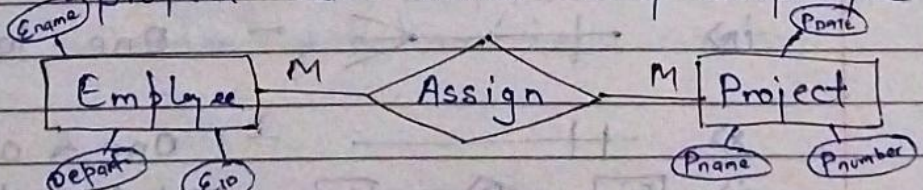


iii> Many to One ~~relationship~~:- When more than one instance of entity on left and only one instance of entity on right associates with relationship, is called many to one relationship.



iv> Many to Many relationship:-(When more than one (~~more Table~~) instance of entity on left and more than one instance of entity on right associates with relationship, called many to many relat.

Ex- Employee can assign many project and project can have many employee

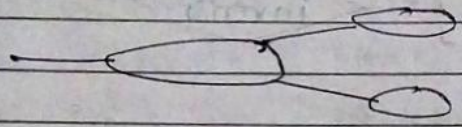


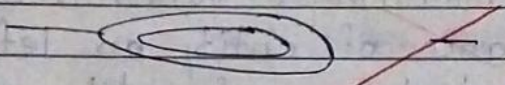
Notations of E-R Diagram

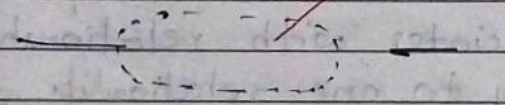
Database can be represented using notations. In E-R diagram, these are used to express cardinality.

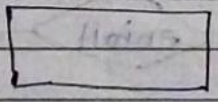
i)  – Attribute

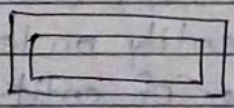
ii)  – Key attribute

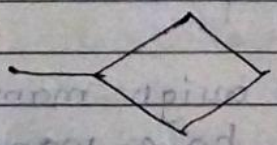
iii)  – Composite attribute

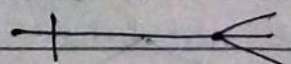
iv)  – Multivalue Attri.

v)  – Derived Attri.

vi)  – Strong Entity set

vii)  – Weak Entity Set

viii)  – Relationship

ix)  – One to many

x)  – One & only one

xi) 

– Total Participation

Key Attribute in DBMS

Key is a value which can always used to uniquely identify an object instance, records and established relationship between table.

There are five type of keys:-

i> Super Key
ii> Candidate key
iii> Primary Key
iv> Composite key
v> Foreign Key

i> Super Key:- It is a set of one or more attribute that taken collectively allow us to identify entity in entity set.

ii> Candidate Key:- It is a ~~set~~ of one or more attribute (unique, may primary key) that can uniquely identify within relation. ~~It must contain unique value.~~ (Roll no, Pan no)

iii> Primary Key:- All attribute combination inside a relation that serve as candidate key. (unique, not null) (Adhaar, Pan, voter ID etc)

iv> Composite Key:- When primary key consist of more than one attribute, called composite key. ~~It combine two or more relations in a table.~~

v> Foreign Key:- It is a column whose value are same as primary key of another table. It combines two or more relations at a time. It acts as cross-refrence between table.
(This may be primary key of other table)

Domain of Attribute

A domain is something like a data type. Each attribute is associated with set of value called domain of that attribute.

Collection of all values that an attribute have called domain of attribute

So, domain is a limit of data value from which relational attribute is limited. Atomic and non-atomic are two types.

Database Design with E-R Model

To model E-R database, we have to design its schema first. Following steps are given.

i> Identify Entities:- Identify the role, event or concept about the end-user want to store data.

ii> Find Relationship:- Find natural association between pair of entities.

iii> Draw Rough ERD:- Put entity in rectangle and relationship on connect entity.
line Segment

iv> Fill in Cardinality:- Determine no. of occurance of one entity for single occurance.

v> Define Primary Key:- Identify data that identify only one occurance.

vi> Draw Key-Based ERD:- Eliminate Many to Many relation and include primary and foreign key in each entity.

vii> Identify Attribute:- Name information details which need to system.

viii> Map Attribute:- For each attribute, match it with one entity.

ix> Draw fully attributed ERD:-

x> Check Result:- Check the final E-R diagram accurately depict the system.

OLTP – Online Transaction Processing
OLAP – Online Analytical Processing

JMS

$x \rightarrow y$ (x determined, y determinant)

Unit-III ~~Relational Model~~

5. Normalization

Normalisation is a process of organising the data in database to avoid data redundancy, insertion anomaly, update anomaly and deletion anomaly. The purpose of this is to produce a stable set of relations that is a faithfull model of operation.

Adv:
- Easy to use
- Flexibility
- Security
- Easy to implement
- Clarity
- Reliable

Functional Dependency

(association)

It is a constraint between two set of attribute in relation from a database. It describes a relationship between attributes in a relation.

FD: $x \rightarrow y$

Adv:
i) Data Normalization
ii) Query Optimization
iii) Data Consistency
iv) Data Quality

There are three types of functional dependencies :-

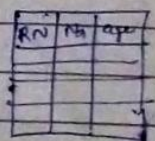
If $t_1(a) == t_2(a)$
$\therefore t_1(b) = t_2(b)$

i) Trival Dep.
ii) Non-trival Dep.
iii) Transitive Dep.

i) Trival Dependency :- A functional dependency of an attribute on subset of itself.

$x \rightarrow y$ (y subset of x)

Ex- {student ID, Address} → {Address} is trival



ii> Non-trival Dependency:- If a functional dependency $x \to y$ hold, where y is not subset of x, then it is called non-trival dependency.

iii> Transitive Dependency:- A functional dependency which caused by axiom of transitivity, is called transitive dependency. $a \to b \& b \to c$ $\therefore a \to c$

It has more than three attribute.

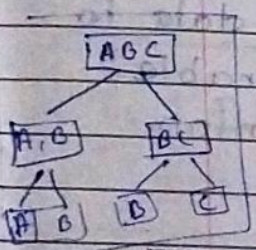
~~It is made by two dependency.~~

axiom of transitivity — Rule

DECOMPOSITION

A process of breaking down the functions of organistion into progressively greater level of detail, is called functional decomposition.

It breaks table into many table.



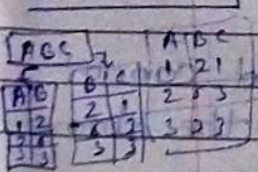
Adv:
- No redonding
- Loss less dependency preserving

There are two types of decomposition:-
- i> Lossy Deco.
- ii> Non-loss Deco.

i> Lossy Decomposition:- When a relation is break into multiple relational schema in a way that results in information loss when trying to retrieve original relation is called lossy decomposition.

Ex- A table <student> is broken into <stud dept> and <sdept> and if we try to join it, we won't be able to get original.

join operation



∴

A	B	C
1	2	1
2	5	3
2	3	3
3	5	3
3	2	3

ii> Non-Loss Decomposition:- The decomposition in which the join, results into same original relation, is called non-loss decomposition. It is used to remove redundant data.

join operation

Ex-:
If a table <employee> is breaks down into <empdetail> and <dept detail>, and if we try to join it, we get original.

Modification Anomalies

Changing the data in some relation can have undesirable consequence, is called modification anomalies (faults).

Adv:-
1> Integrity
2> Scalability
3> Flexibility
4> Security

Dis:
1> Performance
2> Complexity
3> Redundancy

There are three types of it:-
i> Deletion Anom.
ii> Insertion Anom.
iii> Updation Anom.

i> Deletion Anomaly:- It occurs when deleting a record also removes other data that shouldn't be deleted.

ii> Insertion Anomaly:- When a new record cannot be added to a table due to the absence of other related data, it occurs.

iii> Updation Anomaly:- It occurs when data is updated in some rows but not all, resulting in inconsistent data.

Normal Forms

A defined standard structure for relational database in which a relation may not be nested, is called normal form.

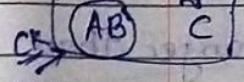
The stage at which a table is organised is known as its normal form.

Types of Normal form

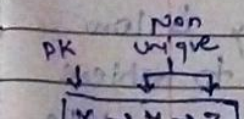
i> First Normal Form :- A relation will be in 1NF if it contains an atomic value, is called 1NF. It states that an attribute of a table cannot hold multiple values. It must hold only single-valued attribute.

ii> Second Normal Form :- In the 2NF, relation must be in 1NF. No attribute of the table should be functionally dependent on only-one part of primary key. In this, all non-key attributes are fully functional.

Full dep. is necessary



iii> Third Normal Form :- A relation is in 3NF, if it is in 2NF and no non-prime attribute is functionally dependent on other non-prime attribute and also no attribute of table should be transitively functionally dependent on primary key. It is used to reduce data duplication and achieve data integrity.



i) $x \to y$ ⇒ x determine y
ii) $x \to\to y$ ⇒ x multi-determine y

JMS

iv) Boyce Code Normal Form:- BCNF is the advance version of 3NF. It is stricter than 3NF. A table is in ~~3NF~~ BCNF, if every functional dependency $x \to y$, 'x' is super key of table. and candidate key

v) Fourth Normal Form:- A relation is in 4NF, if it is in Boyce Code Normal Form (BCNF) and has no multi-valued dependency. It occurs when two or more independent multi-valued fact about the same attribute occurs within same relation.

$X \to\to Y$
not it is

vi) Fifth Normal Form:- The 5NF is also known as project-join NF. A relation is in 5NF, if it is in 4NF and ~~doesn't~~ have loseless decomporition into smaller tables. It is satisfied when all the tables are broken into many tables as possible order.

3NF	BCNF
i) It stands for third normal form.	It stands for Boyce Code Normal Form
ii) It is less stronger than BCNF	It is more stronger than 3NF
iii) Functional dependency is in 1NF and 2NF	Functional dependency is in 1NF, 2NF and 3NF
iv) The redundancy is high	The redundancy is low
v) It is easier to achieve	It is difficult to achieve
vi) It can be obtained without sacrifying all dependencies.	Dependencies may not be preserved.

i) 1NF

Table_Product

ProductID	Color	Price
1.	red, green	15.99
2.	Yellow	23.99
3.	Green	17.50
4.	yellow, blue	9.99
5.	red	29.99

Not in 1NF ∵ in tuple 1 & 4 there are multiple values for attribute 'Color'.

So,

Table_Product_Price

ProductID	Price
1.	15.99
2.	23.99
3.	17.50
4.	9.99
5.	29.99

Table_Product_Colour

ProductID	Color
1.	red
1.	green
2.	Yellow
3.	Green
4.	Yellow
4.	blue
5.	red

Now, above both the table contains distinct/atomic values so these tables follows the constraints of 1NF.

11). 2NF

CustomerID	StoreID	PurchaseLocation
1.	1.	Los Angeles
1.	3.	San Francisco
2.	1.	Los Angeles
3.	2	New York
4.	3	San Francisco

Composite PK (CustomerID + StoreID) in above table.

Table-Purchase

CustomerID	StoreID
1	1
1	3
2	1
3	2
4	3

Table-store

StoreID	PurchaseLoc
1	Los Angeles
2	New York
3	San Francisco

Now, In table Table-Store, Purchase Location is fully dependent on primary key StoreID.

iii) 3NF

Table_Book_Detail

BookID	GeneralID	Genral Type	Price
1	1	Gardening	25.99
2	2	Sports	14.99
3	1	Gardening	10.00
4	3	Travel	12.99
5	2	Sports	17.99

There is Transitive Dep.

BookID → GeneralID GenID → GenalType

So,

Table_Book

BookID	General ID	Price
1	1	25.99
1	3	14.99
2	1	10.00
3	2	12.99
4	3	17.99

Table_Genral

GeneralID	GenType
1	Gardenerig
2	Sports
3	Travel

vi). 5NF

Student	Project	Supplier
Alice	P1	Sup1
Alice	P2	Sup2
Bob	P1	Sup1
Bob	P2	Sup

So,

Student_Project

Student	Project
Alice	P1
Bob	P2

Project_Supplier

Project	Suplier
P1	Sup1
P2	Sup 2

Student_Supplier

Student	Supplier
Alice	Sup1
Bob	Sup 2

iv). BCNF

$R(A, B, C, D)$

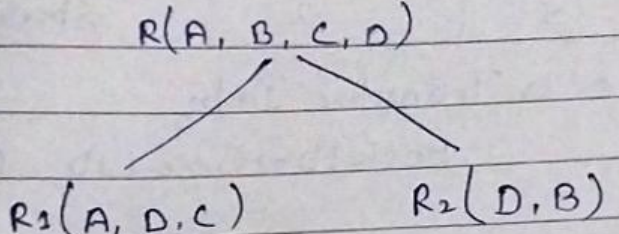
SP → $A \rightarrow BCD$

CK → $BC \rightarrow AD$

$D \rightarrow B$

Here, $D \rightarrow B$, D is not key

So,



v). 4NF

Student	Course	Hobby
Alice	Math	Painting
Alice	Math	Singing
Alice	Science	Painting
Alice	Science	Singing

In the above relation, Course and Hobby attribute are dependent on student, This creates redundancy

So,

Student	Course
Alice	Math
Alice	Science

Student	Hobby
Alice	Painting
Alice	Singing

Here, each relation represent single value & removes redundancy

π → Projection
⋈ → Natural join

Advts. of Normal Form:

i> It helps to data redundancy.
ii> Greater overall database organisation
iii> Data consistency within database
iv> Much more flexible database design.
v> It increase storage effiencicy.

Disadvts. of Normal Form:

i> Database cannot be built before knowing the wer needs.
ii> It requires much-more CPU, memory and I/O so, it reduce performance.
iii> It require more joins to get desired result.
iv> It is much time-consuming
v> Difficult to normalise relations of higher degree

DENORMALIZATION

It is a technique used on a previously normalized database to increase performance. It is used to improve read performance of database with loosing some write performance. In this, we add redundant data to one or more tables, this helps to avoid costly joins of relational database.

Used to combine Tables

Advts: i> Retrieving of data is faster
ii> Simpler queries to retrieve.
iii> Reduced Complexity
iv> Ideal for real-time data access

Disadvt. of Denor.:-

i> The updation and insertion of data is more expensive.
ii> It makes the code harder to write.
iii> Data redundancy requires more storage.
iv> Data may be inconsistent.

Normalization	Denormalization
i> Non-redundancy and consistency data are stored in set schema.	Data are combined to execute the query quickly.
vii> Used to divide tables.	Used to combine tables
ii> Data redundancy and inconsistency is reduced.	Redundancy is added for quick execution of queries
iii> Data integrity is maintained.	Data integrity is not maintained
iv> Redundancy is reduced.	Redundancy is added instead of reduction.
v> Number of tables is increased.	Number of tables is decreased
vi> It optimized the use of disk space.	It doesn't optimize the disk space.

Unit-II 4. Relational Model

E.F Codd 1970

Relational model was a collection of tables to represent both data and relationship among those data. It was introduced by E.F Codd in 1970. Each table has multiple columns and each column has unique name. Each table contains record of a particular type. Each record type defines a fixed number of fields, attributes.

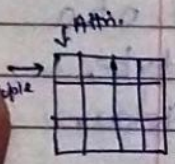
A Relational Database Management System (RDBMS) is a program that lets you to create, update and administer a relational database. Ex- Oracle, SQL Postgre SQL, MySQL etc.

Features:
i) Accessibility:- Queries can be automatically compiled, executed without restoring to programing.
ii) Correctness:- Its semantics are complete.
iii) Flexiblity:- It separates logical from the physical model and increse fleriblity
iv) Multiple Views:- It directs to present different user groups.
v) Concurrency Control
vi) Transaction Management
vii) Security
viii) Backup & Recovery

Important Terminology
(Components)

i> Attribute (Column) :- The property that defines an entity. Ex- Roll No, Name etc.

ii> Relation Schema :- It defines the structure of relation and represents the name of relation with attribute. Ex- student (Name, Roll No, age)



iii> Touple (Row) :- Each row in the relation is called tuple.

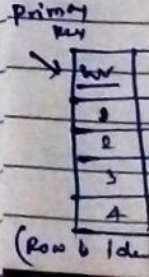
iv> Relation Instance :- The set of tuples of a relation at a particular instance called relation instance.

v> Degree :- The number of attributes in the relation, is called degree of that relation. Unary, binary, ternary ... nary type.

vi> Cardinality :- The number of touples in a relation, is called cardinality.

vii> Column :- It represent the set of values for a particular attribute.

viii> Domain - A set of atomic values that may be assigned for individual attribute.



Relation Key

There are the keys that are used to identify the row uniquely and table.

A set of one or more columns whose combined values are unique among all occurances in table.

On the basis of uniqueness, the key is divided into various type:-

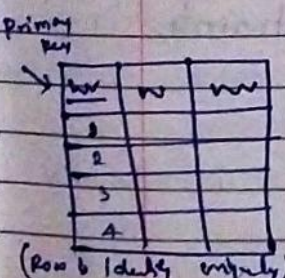
i> Primary Key
ii> Candidate Key
iii> Super Key
iv> Foreign Key
v> Alternate Key
vi> Composite Key

i> Primary Key:- It is defined as the candidate key (Unique + Not Null) that is selected to identify tuples uniquely within relation. It is an attribute that uniquely identifies a specific instances of entity.
It is non redundant mean no duplicat value.

sign-@

ii> Candidate Key:- When more than one attribute serves (unique + Irreducability property) as unique identifier, they are called candidate key. These attributes posses the property of uniqueness. A table can have more candidate key.

CK = SK - PK



iii> Super Key :- It is defined as an attribute that uniquely identifies tuple in a relation. It has the property of uniqueness but not necessarily the irreducibility property.
SK = PK + CK

iv> Foreign Key :- It is a column in one table that refrences the primary key of another table, establishing a link between table.
it is derived from Primary Key.

v> Alternate Key :- Alternate key of any table are those candidate key which are not currently selected as primary key.

AK = CK - PK

vi> Composite Key :- A primary key that made of more than one attribute, called composite key.

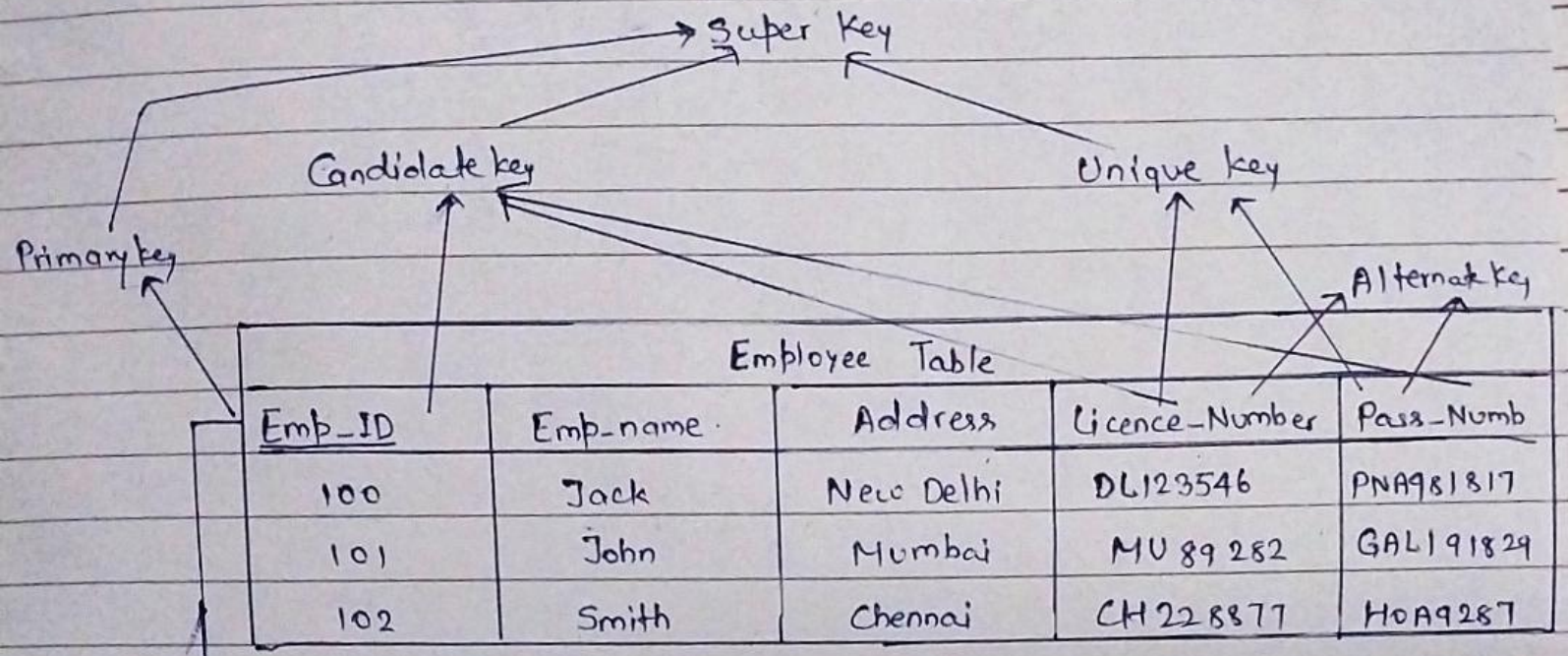
Relational Constraints

Relational Integrity constraint ensures that the changes made to database by users do not result in loss of data consistency.

The major constraints that are defined under constraints are as follows :-

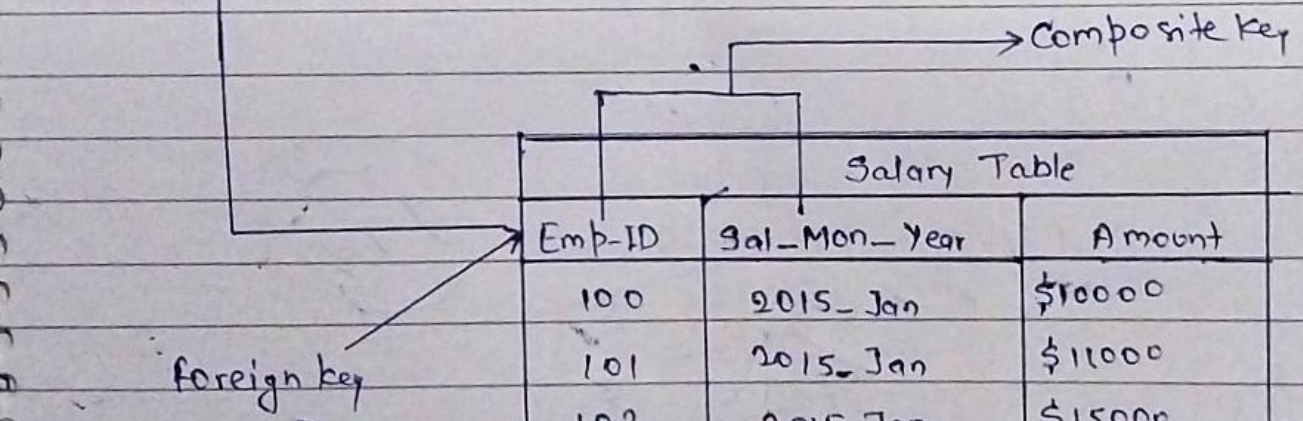
i> Key Constraints
ii> Refrential Constraints
iii> Domain Constraints

Keys in SQL



Employee Table

Emp_ID	Emp_name	Address	Licence_Number	Pass_Numb
100	Jack	New Delhi	DL123546	PNA981817
101	John	Mumbai	MU89282	GAL191829
102	Smith	Chennai	CH228877	HOA9287



Salary Table

Emp-ID	Sal_Mon_Year	Amount
100	2015_Jan	$10000
101	2015_Jan	$11000
102	2015_Jan	$15000
100	2015_feb	$10000
101	2015_feb	$11000
102	2015_feb	$15000

An assertion is a predicate which expresses a condition that we wish the database always to satisfy. Refrential & Domain are form

JMS

i> Key Constraints :- The restriction imposed because of existance of candidate key, called key constraints. It is the uniqueness constraint specified by definition of key. It is also called entity integrity. It cannot be Null.

integrity Rule-1

ii> Refrential Constraints :- These constraints are imposed due to existance of foreign key. These specification defines the restrictions for foreign key. It is also called refrential integrity.

integrity Rule-2

iii> Domain Constraints :- The integrity constant which defines a range of values to be checked for an attribute, is called domain constraint

iv> Null :- Table can hold null values

Properties of Relation

A relation has following properties :-

i> It has a name that distinct from other relations.
ii> Each cell of the relation contains one value.
iii> Each attribute has a distinct name.
iv> Each tuple is distinct; there are no duplicate tuples.
v> The order of tuples has no significance
vi> The order of attributes has no significance.

Triggers

A trigger is a statement that a system executes as side effect of modification to the database. It is always defined on table.

Types
ow Trig.
tatement Trig.
efore & After Trig.

To design a trigger, we must :-
i> Specify the condition under which trigger is to be executed.
ii> Specify the actions to be taken when the trigger executes.

Characteristics of Trigger

i> It can only be associated with tables.
ii> It cannot return data to user.
iii> It prevents incorrect changes in data.
iv> It is fired automatically by SQL server.
v> It can be nested upto 16 levels.

Relational Database

A relational database is a set of formally described tables from which data can be accessed in many ways without having to reorganized the database tables. It uses SQL.

Advts: i> Easy to Use:- It is easy to use because it contains information as table.
ii> Flexibility :- The information can be easily manipulated by operators.

iii> Precision:- The usage of relational data in manipulation with great precision.

iv> Security:- Security control can also be implemented by moving attribute to table

v> Data Independance:- It is achieved more easily with normalisation structure.

Disadvt:

i> Performance:- It experience performance issues as database grows.

ii> Scalablity:- It has low scabblity.

iii> Cost:- It requires software hence it is costly.

iv> Complexity:- It is complex to design & maintain

DBMS	RDBMS
i> It stands for Database Man. Sys.	It stands for Relational Da. Ma. Sy.
ii> It is not based on relationship	It is based on relationship.
iii> The speed of operation is less.	The speed of operation is high.
iv> It offers limited facilities.	It offers many facilities.
v> The platform is normally a DOS.	The platform used is DOS, UNIX
vi> It use the concept of file.	It uses concept of Tables.
vii> It uses 3GL	It uses 4GL.
viii> dbase, Foxbase are example of DBMS	Oracle, focus, Ingres are example of RDBMS
ix> It doesn't provide.	It provide set-oriented database language.

Difference between E-R Model and Relational Model

ER Model	Relational
i> It represents the collection of entities and relation between those entities.	It represents the collection of tables and relation between those tables.
ii> It is easier to understand the relation between entities.	It is less easier to understand the relation between tables.
iii> It describe data as entity set, relationship & attribute.	It describes data in a table; as domain, tuple.
iv> It describes mapping cardinality.	It doesn't describe mapping cardinality.
v> It is entity specific.	It is a table specific.

Unit - IV 6. Database Access and Security

The data security is a term that helps to analyze data risk, shields sensetive data from external and internal actors.

Data Access Strategy

The manner in which applications store, retrieve and manage data, is called data access strategy. A proper architectural model for this is decided and factors like performance, deployment are taken care during development.

Various designs used for data access strategy include:-

i) Move processing to data
ii) Hold database resource for minimum time.
iii) Pass all data back to client

The most effective and fastest data access strategies are based on proper categorization with indexes & views.

INDEXES

An indexe is a copy of selected columns of data from a table, that can be searched very efficiently in table. It is basically a data structure that stores the value for a specific column. It is used to search queries faster.

The index is used by DBMS to find a row in a table quickly. Database allows more than one index in a table.

Various types of index are:-

i) Primary Index
ii) Secondry Index
iii) Clustering Index

i) Primary Index:- An index on a set of fields that include the unique primary key for field, is called primary index. It is fastest and most predictable index. It requires the rows in data block.

ii) Secondry Index:- A data structure that allows user to access and search data in database using information other than primary key, is called secondry index. It improves query performance for efficient searching.

iii) Clustering Index :- A special type of index that records the way that records (data) are physically stored in a table, is called clustering index. A table can have only one clustered index.

but for column

Creating an Index

An index is also a table so it has a data structure. It can be created on single column or combination of columns.

Creating an index in SQL CREATE INDEX command is used.

Syntax:

Create [unique] Index <index_name> [column----]

↑ name of Index

Creating Unique Index

The unique index don't allow duplicate values for indexed column.

To create a unique index on ENAME column of EMP table,

Syntax:

SQL> CREATE UNIQUE INDEX ENAM_UN_INDEX ON EMP (ENAME); COMPOSITE INDEX

Droping Existing Index

When index is no longer require for queries, it might be removed by DROP INDEX command

Syntax:

DROP INDEX <index_name>

SQL> DROP INDEX COM_UN_INDEX ← (query)

USER_INDEXES used for detail index

Guidelines for using Indexe

The following point should be taken while using indexes:-

i) The no. of indexes not be too many.
ii) Always use primary key and unique key constraints.
iii) Do not index small tables with few rows.
iv) An index can be created for only tables
v) While creating a composite index, not more than 32 columns be used

VIEWS

Views are kind of virtual table. It has also rows and columns but it doesn't store data on disk. It does not physically exist. It is created by a query joining one or more tables.

Creating a Views

We can create view using CREATE VIEW statement.

Syntax:

```
CREATE VIEW view_name
  SELECT column1,...
  FROM table-name
  WHERE condition;
```

Join Views

A view which has more than one table that does not we GROUP BY, ROWNUM and set operation clause, is called join view.

Changing View Definition

To change the definition of view, it must replace view using:-

i> View can be dropped and then recreated

ii> View definition can be changed by Create or Replace command.

Inline Views

A view which is created by placing a subquery in ~~FORM~~ FROM clause which define data source, is called Inline view.

Uses:- i> Restrict data access:- It provide a security by restricting access to predefine tables.

ii> Hiding Data Complexity:- It hides the complexity that exist in multiple joins.

iii> Store Complex Query:- It is used to store complex query.

iv> Rename Columns:- View can be used to rename the column without affecting the base tables.

Database Security

A set of tools that protect a database and data storage unit is called database security. (from intentional or accidental threats)

There are two types of database security :-

i> Data level security
ii> System level security

i> Data level security :- A set of measures that prevents data from corruption and unauthorized access throughout data lifecycle, is called data level security.

ii> System level security :- A set of measures that allow users to assign user privilages and specify credential (supplied) for system to access data, is called system level security.

Security Controls

A combination of tools that makes more difficult to access and use data, is called security controls.

Flow Control
Inference Control
Access Control

Two External Machenism
- Administrative Control
- physical Control

There are two modes of security control :-

i> Authentication
ii> Authorization

i> Authentication :- The process of confirming that a user logs in only if he has the right to perform activities, is called authentication. Ex- biometrics for finger print.

ii> Authorization :- The process of obtaining information about authentic user then provide access to user after confirmation, is called authorization. Diffent ways of permission are primary permission, secondry permision public permission etc.

Requirement of Database Security

The database sequrity provides basic requirements :-

ii> Database integrity
i> Protection from improper access
iii> Protection from inference
iv> User Authentication.
v> Multi level Security
vi> Accountability and Auditing

Stept to Deploy Database Security

To provide full-fledge database, one must practive of database :-

i> Ensure physical database security
ii> Use web application
iii> Use database firewall
iv> Use data encryption
v> Minimiz value of database.

Access Control

Access control is a security technique that regulates who can view or use what resource in database. It can be called as access approval where the system decides to grant or reject access request based on authentication and authorization of user.

There are three types of access control :-

i> Discretionary Acc. Con.
ii> Mandatory Acc. Con.
iii> Role Based Acc. Con.

i> Discretionary Access Control (DAC) : It is a type of security access control that restricts or grants the access of object as per access policy.

ii> Mandatory Access Control (MAC) :- It is a type of security access control by which OS constraints the ability of subject to perform some operation.

iii> Role Based Access Control (RBAC) :- It is a type of access control that restricts system access to authorized user. It can implement both DAC and MAC.

Database Protection

Providing mechanism in the system to safeguard (protect) important information, files and software from corruption, loss, is called database protection. It involves data management and data availability.

The first and foremost step in protecting data is hardware redundancy. Hardware and software firewalls, antivirus software can be installed to protect data.

One of the most critide part of data protection is Data backup strategy. There are many software solutions for data backup and protection include Norton Ghost backup software, Backup MyPc etc.

Grant and Revoke

The Grant and Revoke commands are used to enforce database security in multi-user environment. Both these commands are called Data Control Language. These are used to provide or removes privilege on database.

1) Grant :- It is used to provide access or privilege on the database objects to users.

Syntax:

```
GRANT <privilege name>
ON    <object name>
TO    <user ID>
```

Example:

```
GRANT SELECT
  ON   STUDENTS
  TO   USER12;
```

This command grants a select permission on table named students to user 12.

ii> REVOKE:- It is used to revoke some or all previlege which have been granted to user in past.

Syntax:

```
REVOKE  <privilege name>
  ON    <object name>
  TO    <user ID>
  From  <user name>
```

Example:

```
REVOKE SELECT
  ON   STUDENTS
  FROM PUBLIC;
```

Lokesh

Grant	Revoke
i> DCL command grant permission to user	DCL command removes permission to wer
ii> It assigns access rights to user	It revoks access right to user
iii> It needs to specify permission for user.	It doesn't need to specify permission for user