

DBMS :- It is a software program which is used to allow the users to perform manipulations and manage how the data is stored and retrieved from the databases.

DB :- It is the collection of tables where the data is stored.

Tables :- Tables is in combination of rows and columns.

* DBMS software systems are 1) MySQL 2) Oracle

file system :- Earlier to store the data in secondary storage device i.e., harddisk it requires to follow file system mechanism. These are different drawbacks.

(i) The data isn't organised in the file which increases searching time and wastage of time.

(ii) It allows the duplication of data in files.

(iii) There is inconsistency in the data.

(iv) There is no security and backup.

DBMS :- To reduce the drawbacks of file system the DBMS was introduced in the year 1960 by Charles Bachman.

Advantages of DBMS :-

1) It uses efficient technique to access the data.

2) Data integrity constraints and security is enforced.

3) Concurrent access of data and crash recovery.

4) There is inconsistency of data by using the linkage between two tables i.e., foreign key.

5) There is data independence i.e., the changes over the data in one level will not effect over other level.

6) Data administration i.e., every user can access the data but a centralised administrator maintains the data.

7.) Reduce the usage of application development because,
the DBMS already have inbuild functionality to perform
manupulation over the data.

Data Models :- DBMS allows the users to define the data in
the database. There are different models on
which the data is defined.

1.) Relational Data Model

- * This model uses the tabular format to store the data (or) define the data
- * create table AIE&DS (rollno varchar(6), name varchar(10), branch varchar(5), Address varchar(20));
- * To know the structure of data base which is already created, we can use desc tablename ; /desc AIE&DS;

↓
description.

O/P	rollno	varchar(6)
	name	varchar(10)
	branch	varchar(5)
	Address	varchar(20)

Syntax for insertion :-

insert into tablename values (value1,value2,...);

insert into AIE&DS values ('1', 'x', 'AIE&DS', 'xyz');
(or)

insert into AIE&DS values ('&2', '&4', 'AIE&DS', 'xyz');

To display the content :-

select * from tablename;

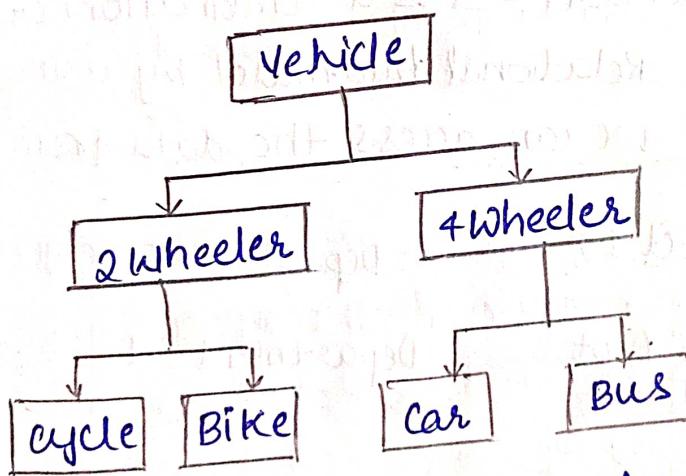
select * from AIE&DS;

rollno	name	branch	address
1	x	AIE&DS	XYZ
2	y	AIE&DS	YZX

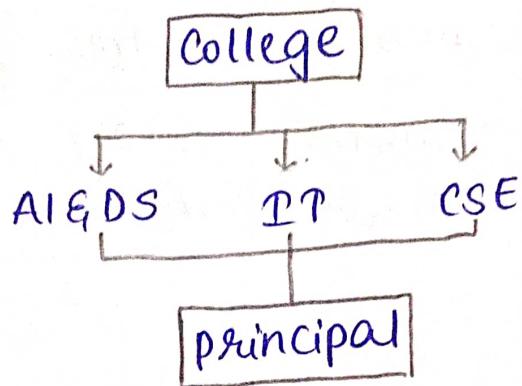
Terminology :-

1. Attributes or fields :- It refers to the column names
2. Tuples or rows :- It indicates the rows of entry in the table
3. Degree :- It refers to the total no. of attributes in the relational
4. Cardinality :- It refers to the no. of tuples in relational

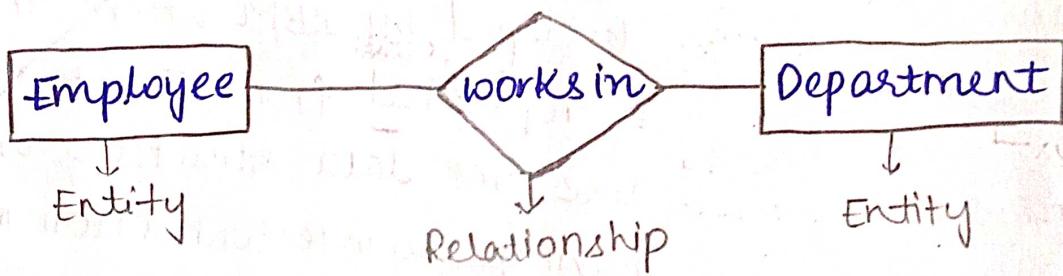
Hierarchical Data Model :- It is the oldest model which is developed by IBM where the data is represented from top to bottom just like tree data structure and without having any interconnection in between the data



Network Data Model :- In the case of network data model where the data is interconnected in between them in such case we can use this model



Entity-Relationship Data Model :- Entity refers to the real world object of independent type. In this model if uses diagrams to represent the data by using these diagrams the developer can understand how the data base is implemented.



Object Oriented Data Model :- It is a combination of OOP and relational data model by using objects we can access the data from the table

Employee (object 1)

Emp Id	Emp Name	Dept

Department (object 2)

Department Id	Dept Name

Object Relational Data Model :- It is a combination of OO Model and relational data model by using object for each and every column it requires to initialise objects to access the entire table.

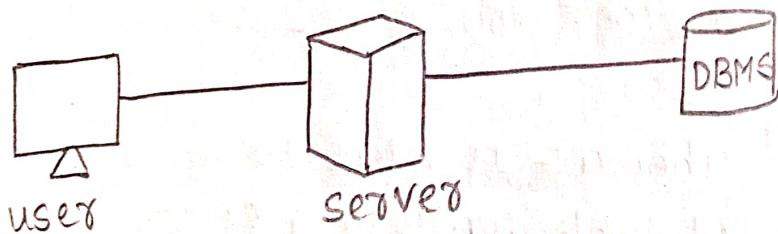
Employee (Object 1)

Emp Id Emp Name Department

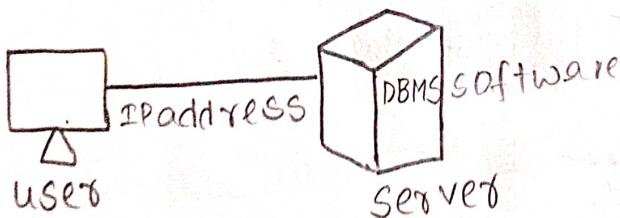
(2)

V2K problem is used to reduce this problem, the ORDM is used from these models relationship, ER, object oriented data model are widely used.

3-Tier level Architecture



2-Tier Level Architecture

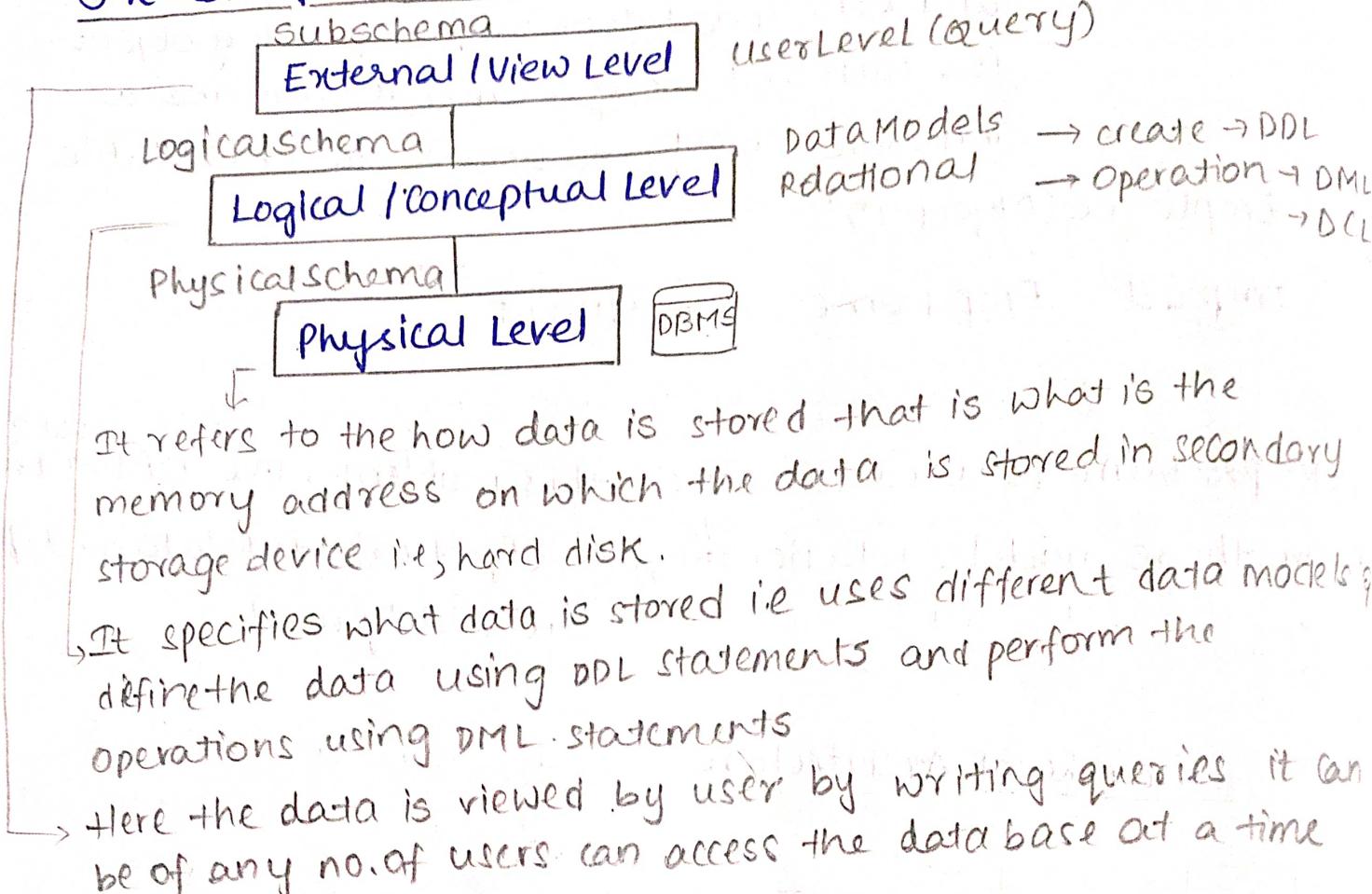


1-Tier level Architecture



DB directly available to the user

3 levels of Abstraction / Architecture



desc tablename;

 |
 schema

 |
structure of table

 |
column name
 Datatype

schema :- It refers to the structure of table consisting of column names and datatypes

(3)

Data Independence : Changing the memory address of data from one location to another in the physical level it will not effect the data in logical level

changing the data in logical level it will not affect that address location of data in physical level

By displaying part of the data it won't affect the complete data in database

Stored in Data Dictionary

- | | | | |
|---------|------------------------------|---|-------------------------------------|
| (1) DDL | - Data Definition language | → | create
Alter
Drop
Truncate |
| (2) DML | - Data Manipulation language | → | |
| (3) DCL | - Data Control language | → | |

Data Dictionary:- special type of data base which is managed by Database administrator contain MetaData

1.) DDL statements - Create, Alter, Drop, Buncate

creation of table
using integral
constraints (IC)

- (i) unique → Atleast one null value
- (ii) Not Null
- (iii) Primary Key
- (iv) Check → condition
- (v) Default

syntax : create table tablename

(columnname datatype ...);

create table student (rollno varchar(5) primary key,
name varchar(10), Branch varchar(5) default
'AEDS', Marks number(3) check (marks >= 65),
Email varchar(20), phone no number(10) 'UNIQUE',
address varchar(20))

Q1 :- Table created student

Alter :- It is used to modify the columns in the table, i.e.
we can add or delete columns.

To add the column

Syntax : alter table tablename add
columnname datatype(size);

Eg : alter table student add
age number(3);

To delete the column

alter table tablename drop column columnname;

Eg : alter table student drop column age;

Drop :- It is used to delete the data

Syntax :- Drop table tablename;

Eg :- Drop table students;

Select * from student

O/P :- Table doesn't exist.

Truncate

Syntax :- Truncate table tablename;

Eg :- Truncate table student;

Select * from student

O/P :- No rows

19/10/2022

Delete Comment

It is used to remove the data from the table

To Delete few rows from table by using condition

delete from tablename where condition;

Ex:- delete from student

Select * from student

O/P :- No rows from student

Delete the details of students whose marks is greater than 65.

DCL - Data Control Languages

It is used to control the users to access the data by using access stimulators

(i) Grant :- It is used to provide permission to access the data to the other users.

Grant insert/delete/update/select on tablename to username

Grant insert on student to ito2;

(ii) Revoke :- It is used to cancell the permissions to access the data from table

18/10/22

2) DML statements :- It is data manipulation Language

It refers to the modification of data in the table using insert, delete, update and select commands

There are two types of DML

→ Procedural

→ Non Procedural

Procedural : Refers to specify what data needed and how to get the data

[procedural :- sequence of data]

Example : NO SQL

Non-Procedural (or) Declarative :- Refers to specify what data is needed and doesn't request how to get the data

insert command : It is used to enter the data into the table

There are 3 methods to insert the data into table

syntax : `insert into tablename values (value1, value2, ...);`

Eg : `insert into student values ('21', 'Megha', 'AIDS', 75, mani1943@gmail.com, 9998887776);`

Method2 : `insert into tablename (col1, col2, ..., coln)`
`values (val1, val2, ..., valn);`

Method3 : `insert into tablename values`

Select command : used to display the data from table
there are 2 different ways to display the data.

To display all the data.

Syntax : select * from tablename;

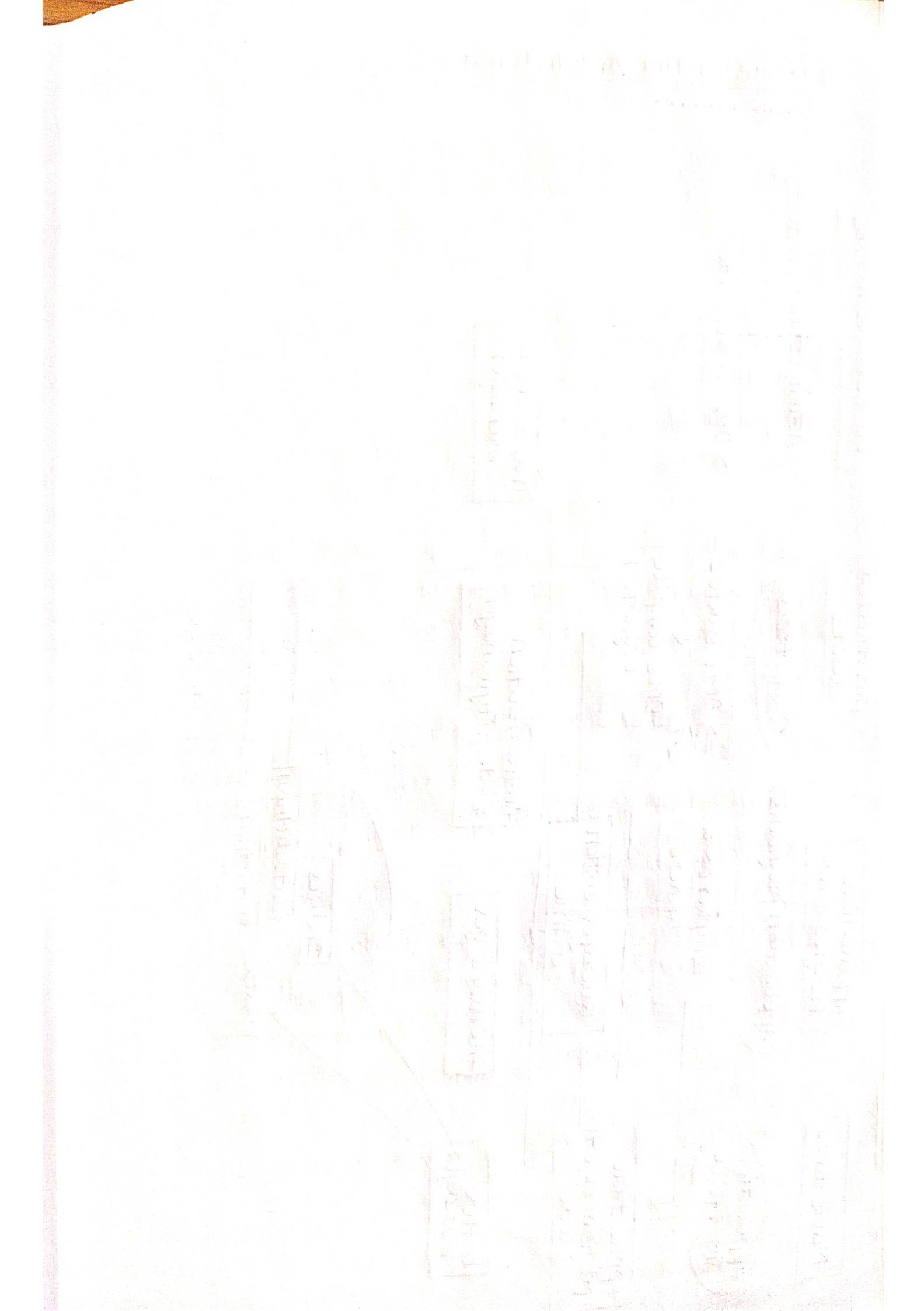
To display few data

Syntax : select col1, col2, ... from tablename;

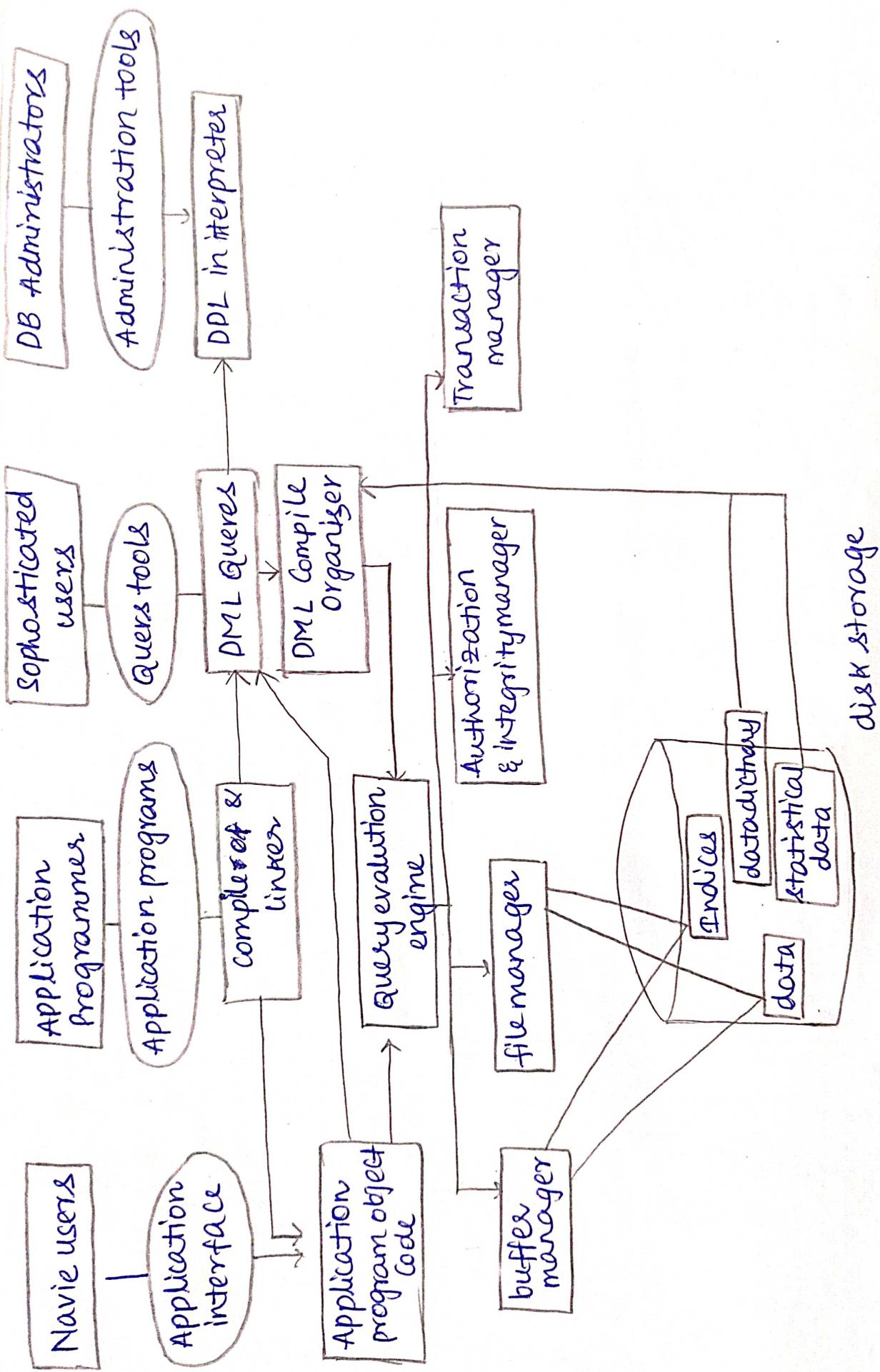
update : It is used to modify the data

Syntax : update tablename set columnname = value where condition;

To display the marks of rakesh



Database System Architecture :-



Transaction Management

- Transaction refers to the action that is accomplished to complete a task
 - The set of operations over the data base that are successful or failed as a group is called transaction
- Roll back :- It is used to retrieve the deleted or updated transaction once the transaction is failed.
- Commit :- Once the transaction is successful then we need to type commit in order to store the data permanently
- Every transaction has 4 properties. They are ACID Properties
- Atomicity :- It refers to the transaction either completely successful or none of the transaction is performed
- Isolation :- The operations over the transaction will not effect on other transaction. Every transaction is treated individually
- Durability :- In the case of hardware failure by using recovery mechanisms the transactions are roll backed or retains the data
- Consistency :- Once the transaction is completed the results are effected on the users bank account
- DBMS ensures the correctness of data.

[9]

Introduction to Database Design

There are 6 steps / Phases to design Data base.

1. Requirement Analysis
2. Conceptual DB Design
3. Logical

4. Schema Refinement

5. Physical DB Design

6. Application & Security DB Design

1. Requirement Analysis

To construct the data base, data base designer need to collect all the requirements from the user. for example to construct the bank database the designer need to take the input from bank owner

2. Conceptual DB Design

By taking the inputs from requirement analysis. The designer need to develop a description of data by using E-R Model which creates easy for users to understand the data

3. Logical DB Design

The DB Designer need to choose one data model that is relational data model by using this make the execution of DDL statements to generate logical structure of Database i.e called schema

4. Schema Refinement

In this phase the developer needs to check for the duplications and the requirements satisfied or not, then by using normalization techniques makes the refinement process

5. Physical DB Design

The DB Designer store the data in secondary storage devi using efficient data structure techniques like indexing

6. Application & Security DB Design

The DB Designer understand what type of users are accessing the DB & how to provide security to data for every user by using access rights

repeat the process until all the requirements of users are satisfied, it is called tuning process.

Entities

It is the real world object which is different from others it can be of physically existence or tangible objects like student, person, teacher etc & logically existence objects are intangible objects like table, chair, fan, light etc

Entity sets

M Tech Students
It is a set of entities having same type for example B Tech Studen

Attributes

An entity is described by using set of Attributes

All the entities in the entity set have same attributes.

Domain

It represents a type of data and the range of values for an attribute

Entity Type

Collection of Entities having same attributes which the feature of independence for example student table is a entity type

Types of Attributes

- 1.) Simple Attribute
- 2.) Composite Attribute
- 3.) Single valued & Multi-valued Attribute
- 4.) Derived Attribute
- 5.) Descriptive

1.) Simple Attribute :- the value of attribute cannot be divided into sub parts Eg:- RollNo, Empid, dept NO

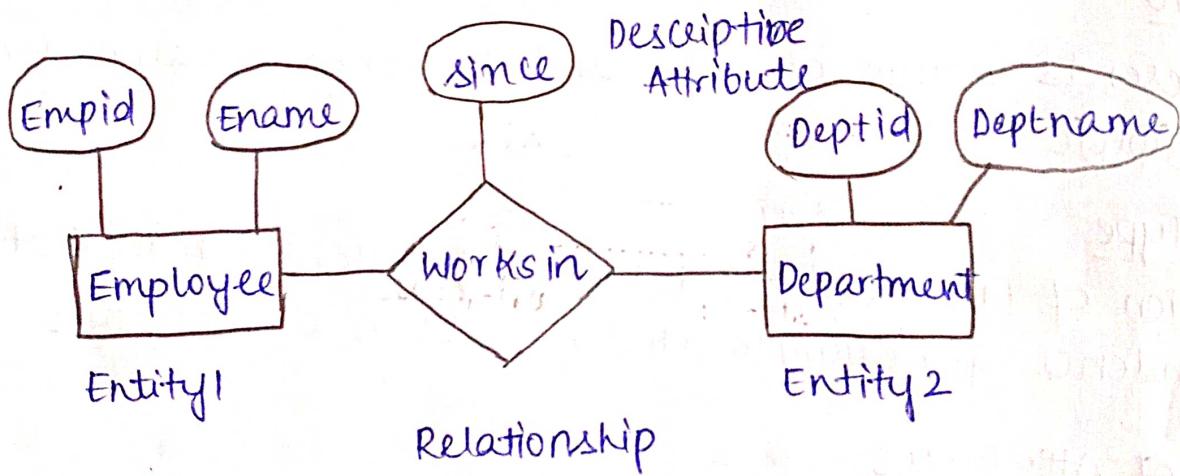
2.) Composite Attribute The value of attribute can be divided into multiple parts Eg:- Studname, firstN, MiddleN, lastN

3) Single valued & Multi-valued :- The attribute has single value for an entity Eg:- Roll No, marks etc

The attribute contains multiple values for an entity
Eg:- Telephone no consisting of STD code followed by landline No, country code, mobile No

4) Derived Attribute :- The value of attribute is used to compute (or) to calculate the value of other attribute, these values are not stored by it is computed whenever required
Eg: Marks \rightarrow Percentage DOB \rightarrow Age
 \downarrow Avg

5) Descriptive Attribute :- It is used to record the information about the relationship in between entities



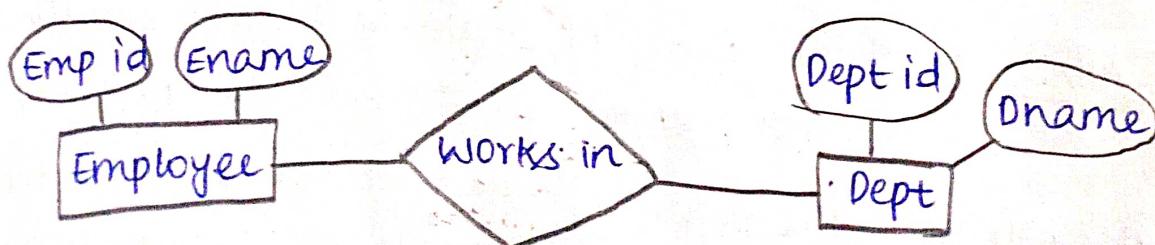
[10]

Relationship & Relationship sets

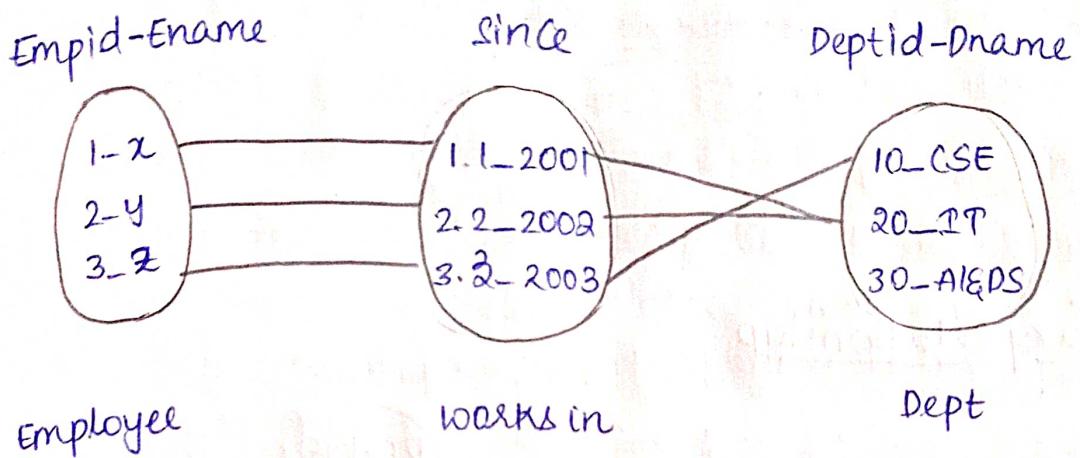
Relationship :- It the association in b/w two or more entities

Relationship sets :- It is the collection of relationship of same type

Eg:- Employee works in a department



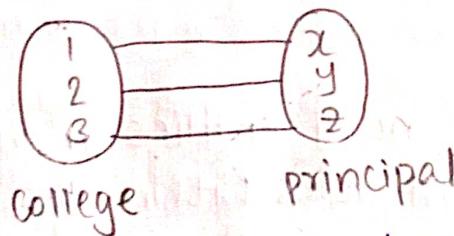
Relationship Instance



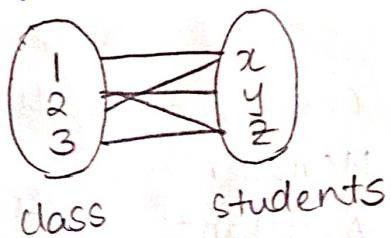
Mapping Cardinalities

1.1 Binary Relationship :- the relationship in between the no. of entities in the entity set is called Mapping cardinalities
There are two entities involved in relation.

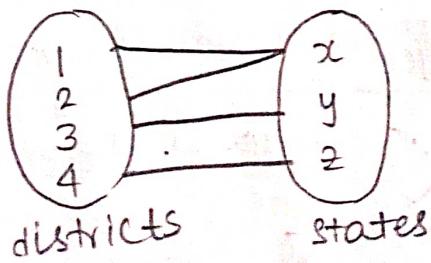
(1) One-One :- Every college is headed by one principle



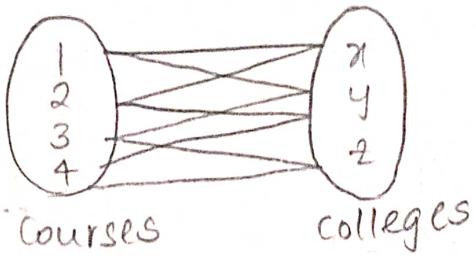
(ii) One-Many = Each and every class have many students
Many students couldn't able to attend multiple class at same time



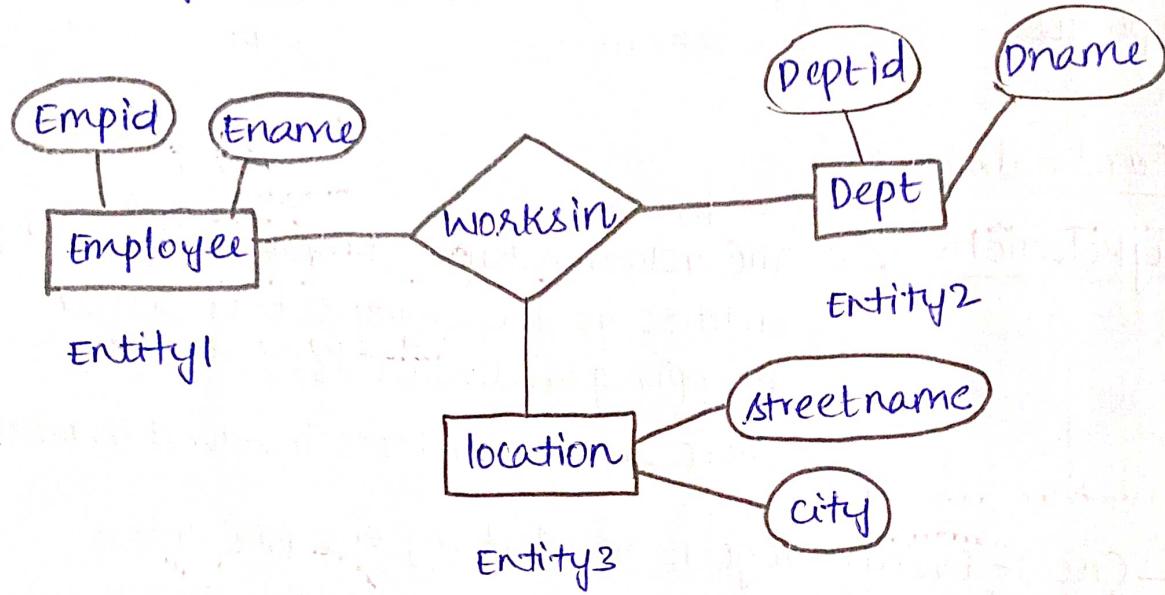
(iii) Many - Many: - Many districts can form a state but many states cannot form a district



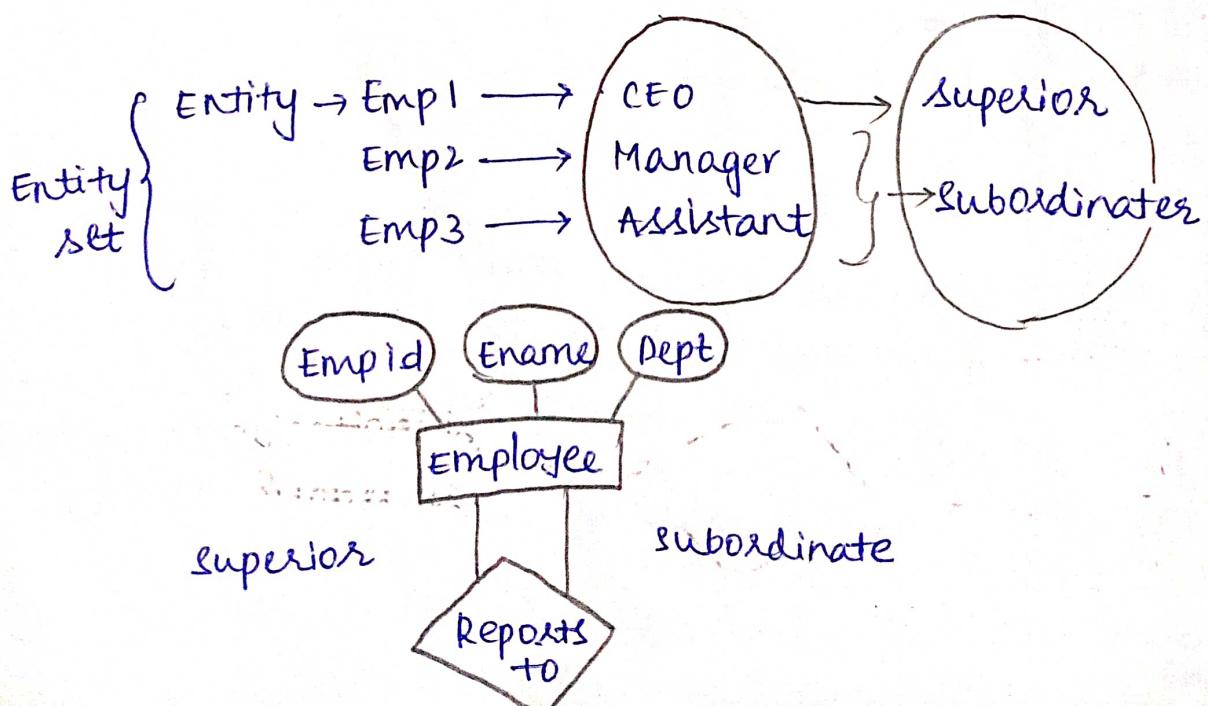
(iv) Many-Many: Many courses are offered by many colleges



2.7 Ternary Relationship



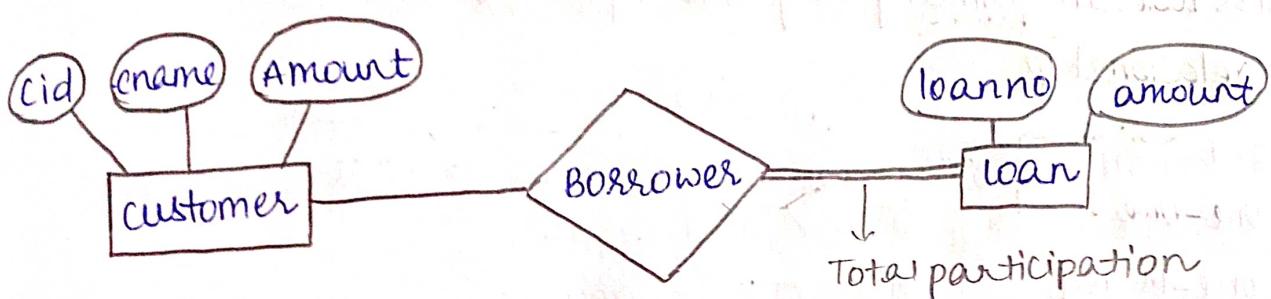
Role Indicator: If the no. of entities in the entity set occupy different roles within the entity set then by using the functionality we can identify the role of entity



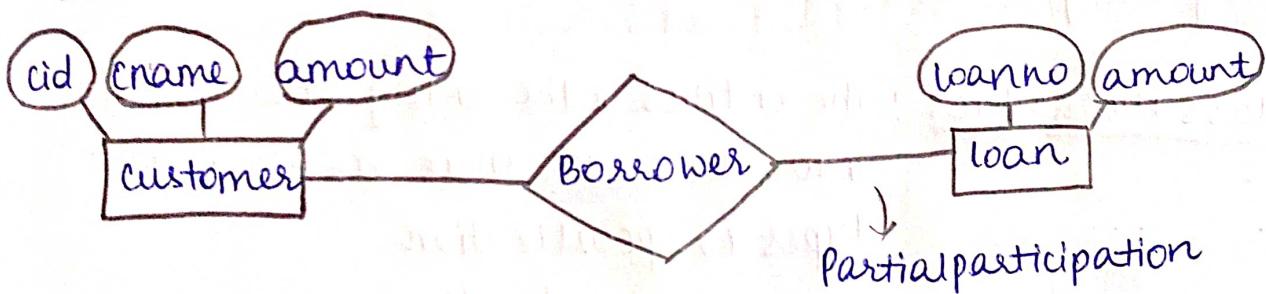
Participation Constraints :- If the no. of entities in the entity set are participating in another entity set It is called participation constraint.

It has two types 1.) Total participation 2.) Partial participation

1.) Total participation : All customers in the entity set are taken the loan. It is represented by using double lines.



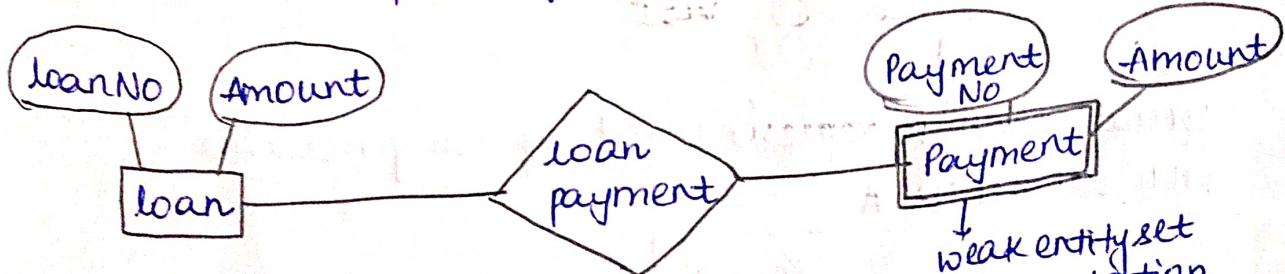
2.) Partial participation : Only few costumer have taken the loan



Strong Entity set :- The attributes in the entity set can have primary key & foreign key

Eg:- Rollno, customer id, employee id etc

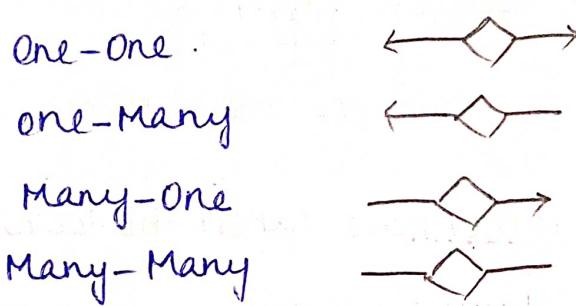
Weak Entity set :- the attributes in the entity set can't have primary key



Primary key <loanno, paymentno>

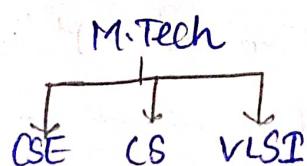
weak entity set representation
double box

- To Convert weak entity set to strong entity set; initially it must satisfy whether the participation of entity is total or not then attach the primary key from strong entity set to the non-primary key of weak entity set.
- All the payments are under the single loan therefore it is total participation
- The payment no is updated for every transaction such that it is not an primary key this representation offers 1 to many relationship

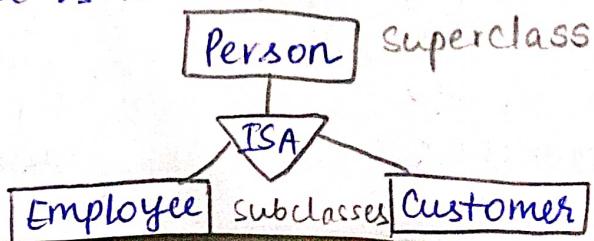


[7] class hierarchies: The entities in the entity set are divided into sub classes. There are two different types 1. specialization
2. Generalization

specialization: A super class is divided into multiple sub classes and every sub class is different from other classes

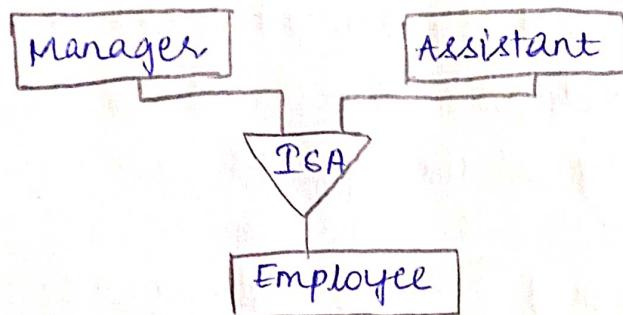


specialization is represented by using rectangle with a label is A

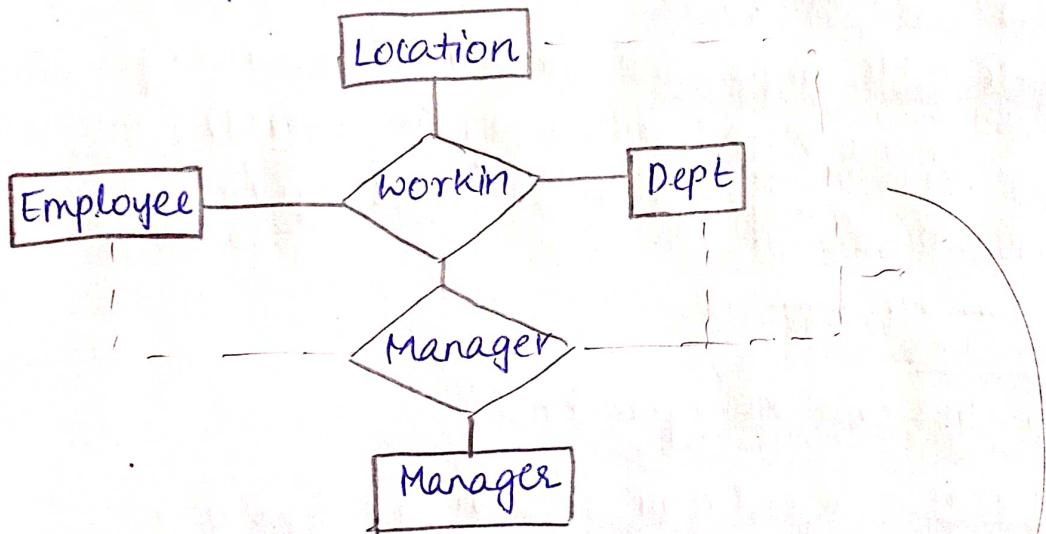


Generalization: The sub classes of entity set are derived into single super class.

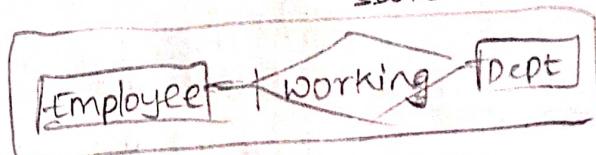
The super class contains the common properties of sub class.



Aggregation: To create a relationship in between relationship and entities set in the VR model it creates duplication of data.



Aggregation creates abstraction for the relationships to create high level data entities



Conceptual design with E-R Model



Entity set



Attributes



Relationship set



Derived
attribute



Multivalued



weak entity set



Total participation



partial
participation