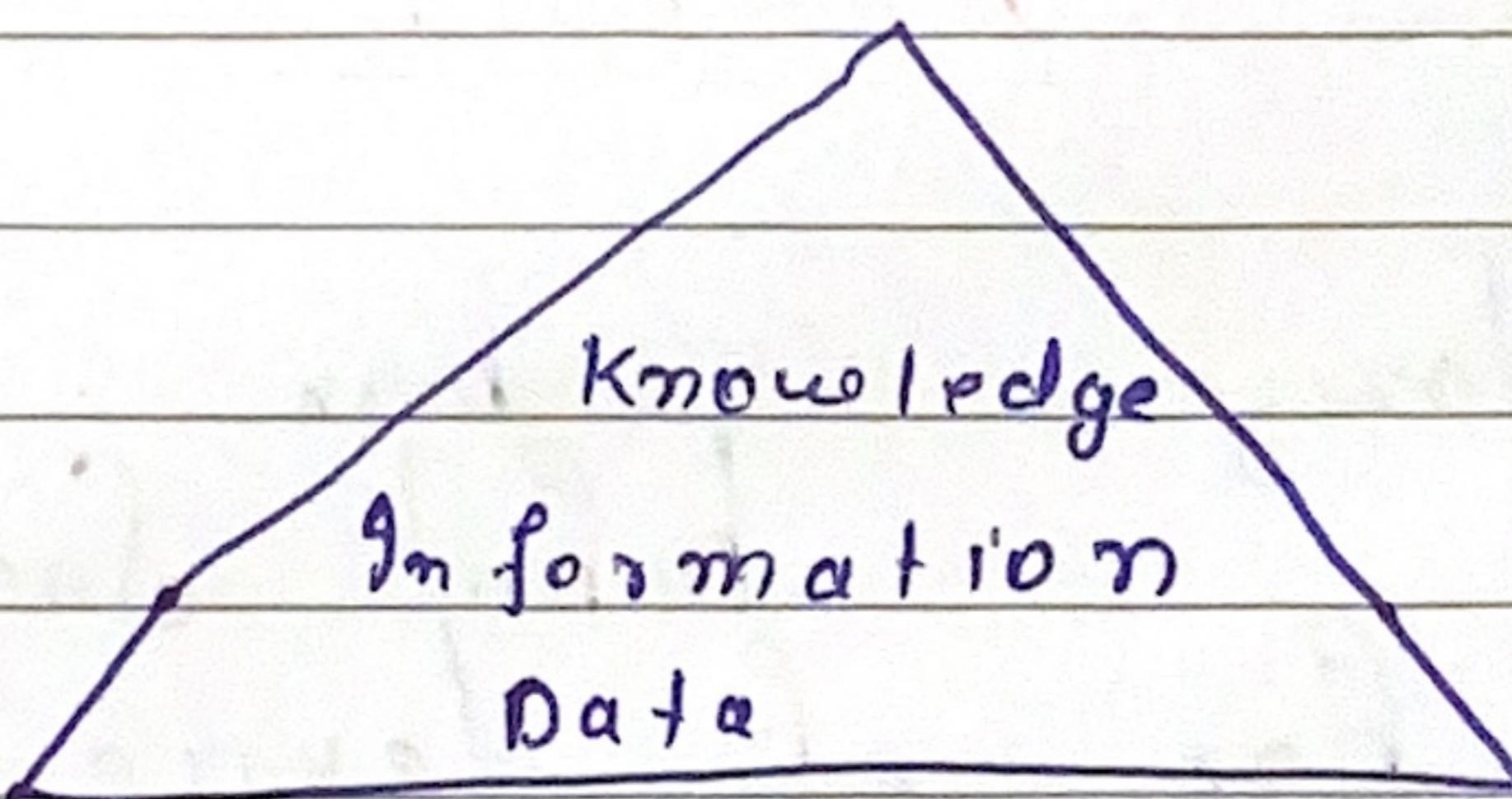


## RD BMS

- Data are raw facts 1, 2, 3  
Not conveying anything
- Jab hum data ko process karate hain  
we get information
- Information se we get knowledge



Jab data inter related mein hota hai  
we call it database

Tabular form → Relational Database

DBMS => General purpose software system  
that facilitates process

- defining = data-type, constraint, structure
- constructing = storing data
- manipulating = specific data, queries, reports
- Sharing = Allowing multiple user to access data

Protection =>

System => against hardware or software malfunction

Security => against unauthorized or malicious access

### File System Drawbacks.

- Data Redundancy - Har ek data ki bohot saari copy ham safi hai har ek device pe.
- Accessing Data - Specific data or report generation ke liye har bar ek new program.
- Data Isolation - Data scattered in various files may be in different formats waiting new application program to retrieve appropriate data is difficult.



CAP Theorem to decide which DB to use.

- Integrity problems - The data value stored in dB must satisfy certain type of constraints
- Atomicity - Transaction must be atomic it must happen entirely or not at all
- Security - Not every user of database should be able to access all data

## View of Data

- Major purpose of data base system is to provide user with an abstract view of data that is hide certain details from certain user.
- Data Abstraction  
 → Instances and Schemas

## Data Abstraction

Physical Level → Lowest level of abstraction describe how data are actually stored ; Data Base ka structure .

Logical Level → Matlab data stored karna  
kya Relationship hai

View Level → highest level of abstraction  
only access a part of  
data base

## Instance & Schema

Instance → The collection of information stored in database at a particular moment is called instance of database (Actual data)

Schema → Overall design of database. Rows, Column's  
Schema is all about that.

## Database Language

- \* Data Definition Language - We specify a dB schema by a set of definition jahon ke hum commands fo tabel design karno mein help koote hai  
create dB, Add Column, Del Column

\* Data Manipulation Language - It is a language that enable user's to access or manipulate data as organized by appropriate data Model.

→ Procedural DML what data & flow to get

→ Declarative DML what data only

## Db User & Administrator

Naive Ye db ke saath interact karte hai through application program jo previously written kare ja chuke hai

Application Ye vo log jo UI of Application design karte.

Sophisticated Ye vo fo Query likhte hai Yani management ka use karte hai They submit each Query to Query processor.

**Data Base Administrator** A person who has such central control over the system is called a database Administrator . (DBA)

- Schema Definition
- Storage Structure
- Schema & physical organization modification
- Granting of authorization for data access

### Storage Manager

**Authorization Integrity** Ye check karta hai ki user ko access karne ki authorization hai aur agar data Entry mein correct constraints satisfy ho rahi hai.

**Transaction** System failure hone par bhi data base correct state mein hona chahiye

**File** Data disk mein kaise store hogi aur allocation kis bases pe hogi

**Buffer** which is responsible for fetching data from disk storage into main memory .

## Query Processor

DDL Interpreter

DML Interpreter

Query evaluation = which execute  
low level instructions  
generated by DML Compiler

## Disadvantages

- 1 Centralization → security Vulnerability, failure
- 2 Cost of Software
- 3 Cost of Hardware
- 4 Backup & Recovery → Regular backup  
Kyoki data centralised hai

## Introduction to Database Design

Ye Ek Six Step process Hai

1 Requirements Analysis → The very first step of designing a database application is →

- Kya Requirements Hai, Konsa datatype
- Key Queries Konsi Hai
- Konsi Application ke uper built hogi.

2 Conceptual Database Design → Information ko Gathered kiya

- High Level description data ka yani kya data hogi kitne columns
- Constraints ke baare mein sochna
- \* • E-R Diagram ka use

3 Logical Database Design → Now in this step we convert our E-R Model to Relational Model

4 Schema Refinement - Ab Yeho he hum collection of Relations ko observe harte hai aur hum potential problems ko refine karte hai.

5 Physical DB Design - Typical Expected workload ke according further refine ho aur vo workload ko bear kare Jisme hai Indexing & Clustering

6 Application & Security - Ab isme Jo Database hota hai kaun uska kya data access kar sakte hai haonsi security hai kaise Authorization & Authentication hogi

E-R

E-R = Entity - Relationship Model

Entity  $\Rightarrow$  Real world objects with its attributes.

$E_1 = \text{Monish} \Rightarrow (\text{Att}) [ \text{facid}, \text{c-no}, \text{Add}, \text{class} ]$

$E_2 = \text{Monish} \Rightarrow (\text{Att}) [ \text{Regno}, \text{Course}, \text{Section} ]$

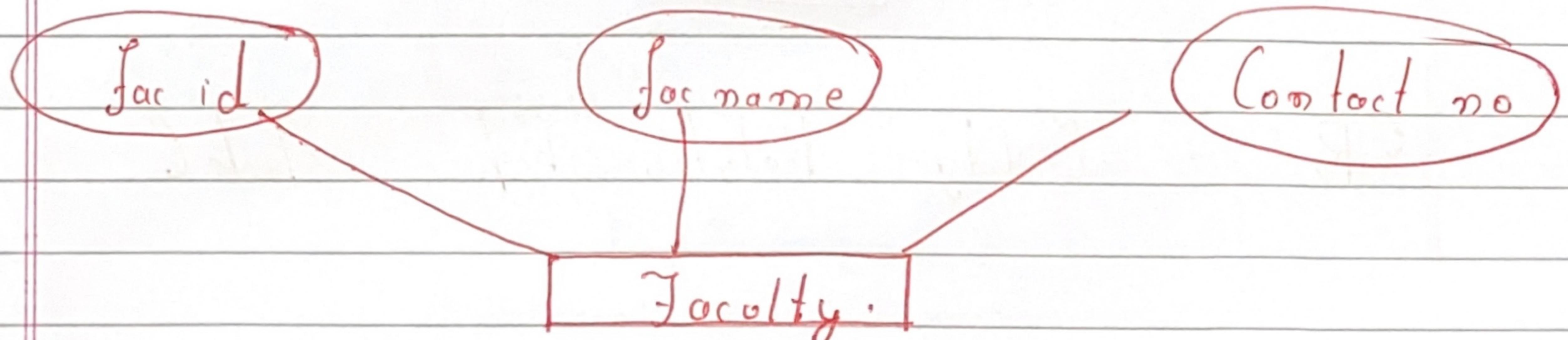
07

$E_1 \neq E_2$  (Attributes are diff)

Entity Set  $\Rightarrow$  It is a set of Entities of same type, that share the same properties or attributes

Faculty

Attributes



Super key => Set of attribute to  
ki har ek entity ke  
diff honge

- 1 Regno, No, Address # Not NULL
- 2 Regno, No.

Ishka Minimal Subset Yani Sabse chota  
subset ka element jo Unique &  
NOT NULL (Candidate Key)

Candidate key ka har Element Ban sakti  
hai primary key (Primary Key)

### Attribute Types

- Simple & Composite

Simple Address [ 2531 — 1 — ]

Composite Address [ 2531B  
Street  
Mathura ]

- Single & Multi valued

Single Only 1 hi Entry

Multi Valued More than One Entry

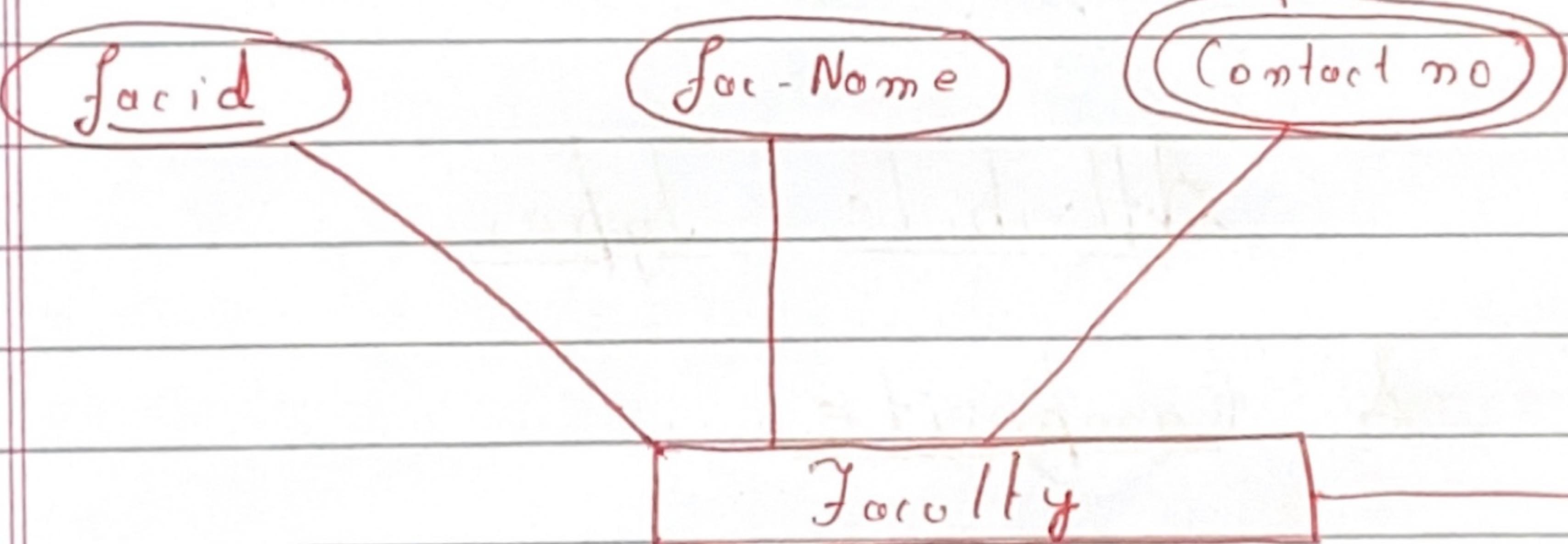
- Derived Attributes

Yani user se mat kuch automatic derive ho Jayegi

$$\text{Age} = \text{Current date} - \text{DOB}$$

Primary Key

Multivalued



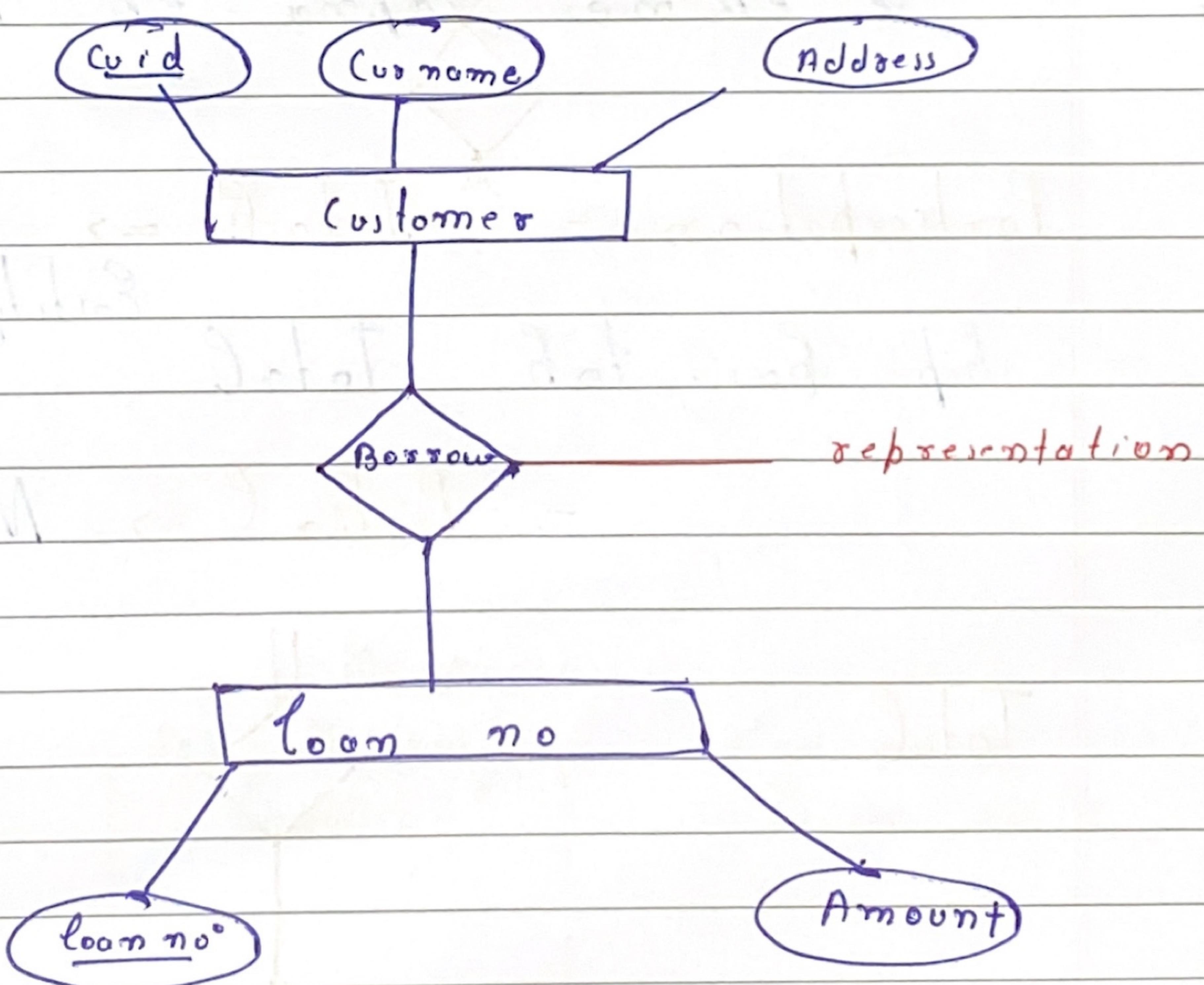
#

Remember Notation

## Relationship

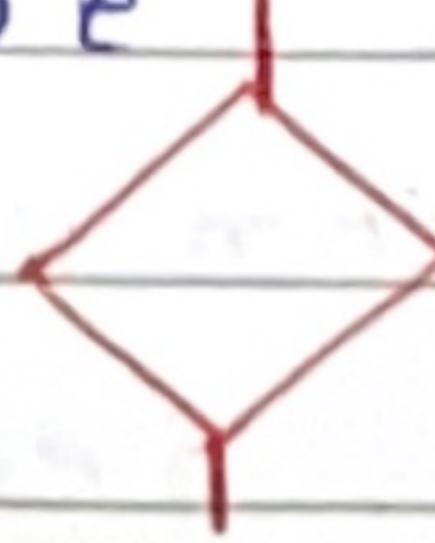
- \* A relationship is an association among several entities.
- \* A relationship set is a set of relationships of same type.

Cardinality  
=> One to One  
=> One to Many  
=> Many to Many  
=> Many to One.



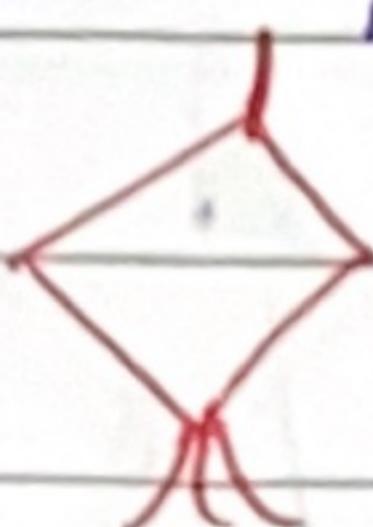
One to One

हो एक customer ने atmos एक loan लिया



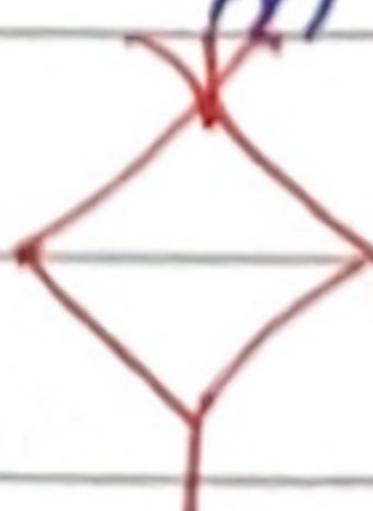
One to Many

एक Customer एक se फ्यादा भी लाने सकता है।



Many to One

2 Ya 3 Customer me milke joint  
loan liya है।



Many to Many

Yani 2 ne milke एक लिया था उनमें  
se ekne kabna ek aur le liya.

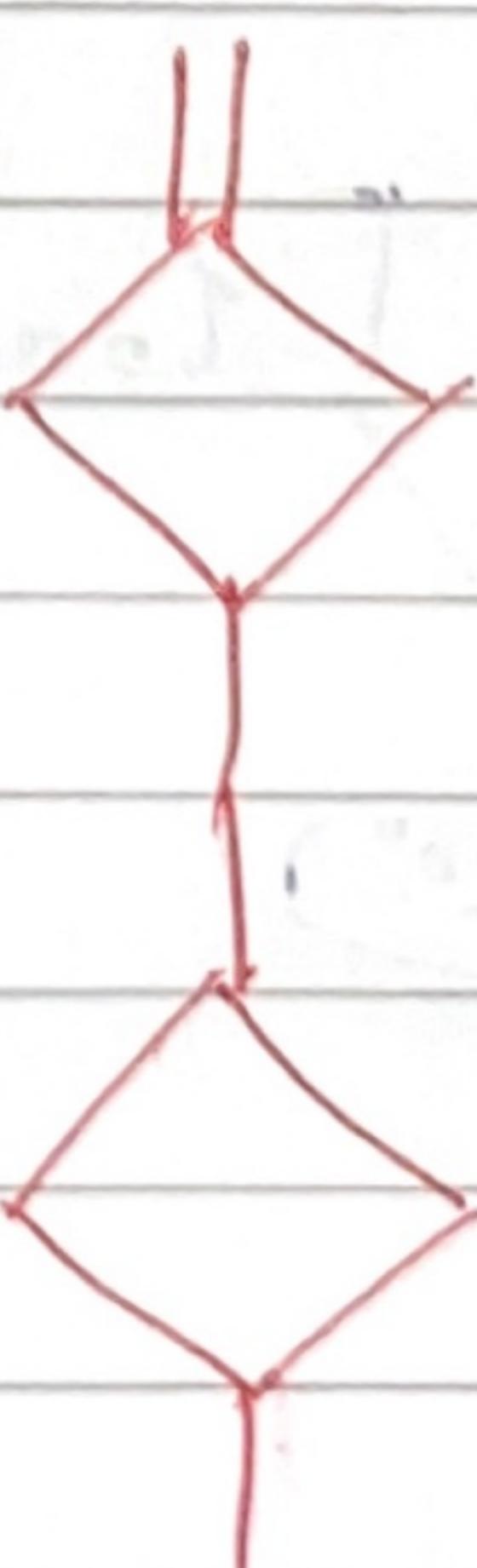


Participation - Total C  $\Rightarrow$  Agar हो एक  
Entity का relation  
ship है तो Total C

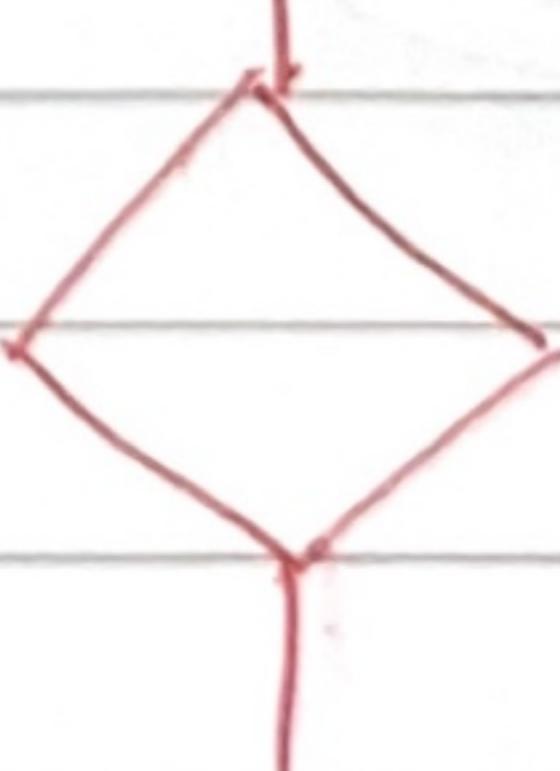
- Participation  $\Rightarrow$  Nahi toh participation.

Total C

=



Participation =



## Relationship Set

Maaano ek relation  $E_1, E_2, E_3 \dots E_n$  entities ko connect kar raha hai

tob uss Relation ki jo tabelisme kya agaya.

\* Relationship ke khudke attributes Nahi hai

PrimKey ( $E_1$ )  $\cup$  PrimKey ( $E_2$ )  $\cup \dots$

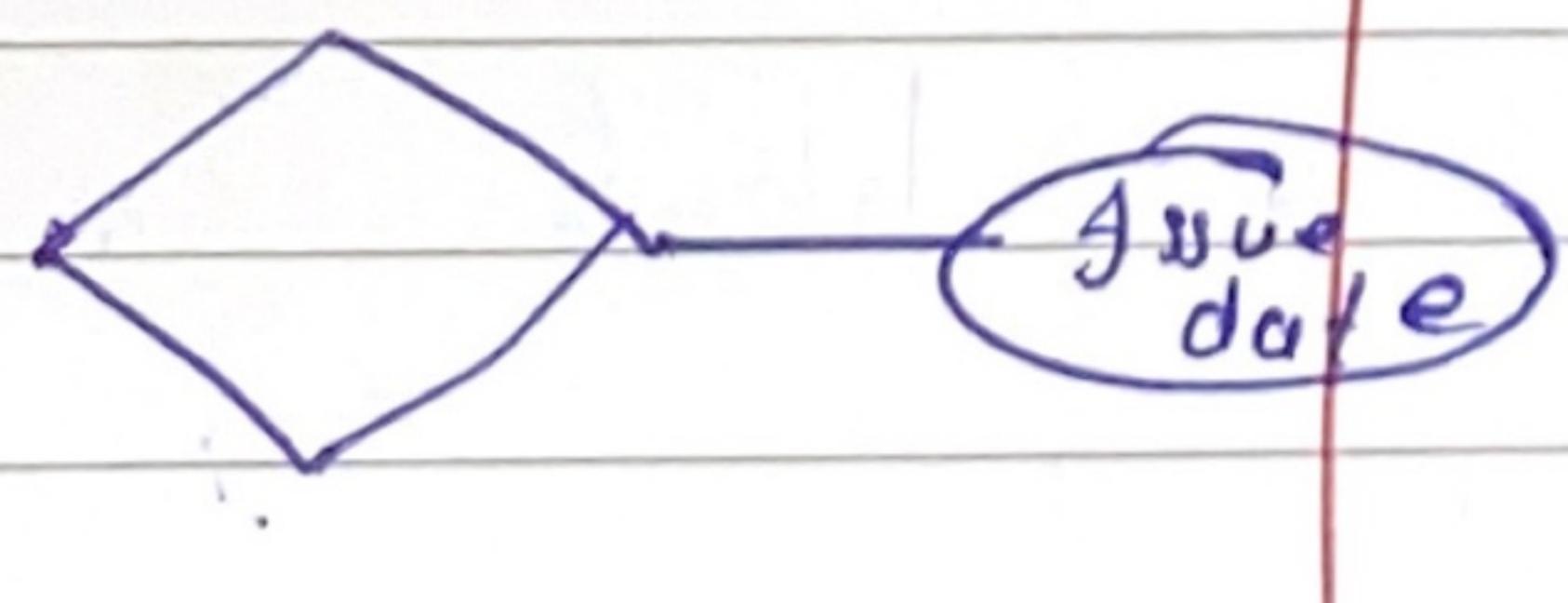
\* Agar uske attributes hai  $a_1, a_2, a_3$

Prim Key ( $E_1$ )  $\dots \cup a_1 \cup a_2 \dots$

Ex

Borrow

Primary key of customer	P.K of loan	Khudka attribute. Issue date
Cuid	Loan no.	



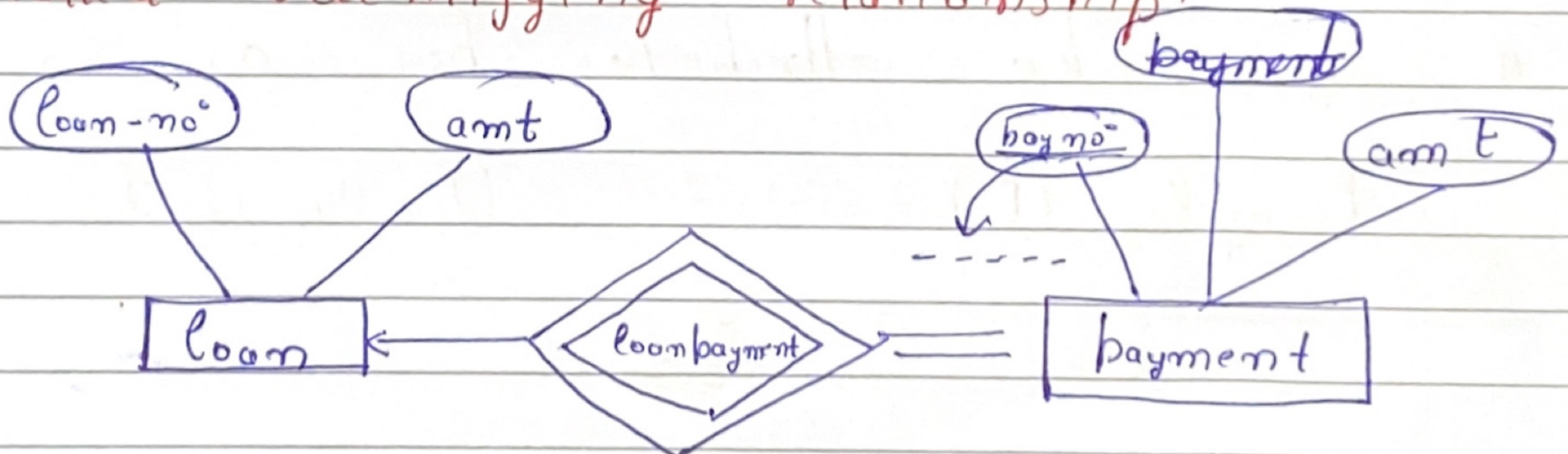
## Weak Entity Set

Agar ek arise entity ho jisse enough attributes na ho jisse ek primary key ban jaye toh wo set hai Entity set

## Strong Entity Set

Jiske baas strong Entity set it means uske baas primary key hai.

- \* The relationship associating the weak Entity set with the identifying entity set is called Identifying relationship.

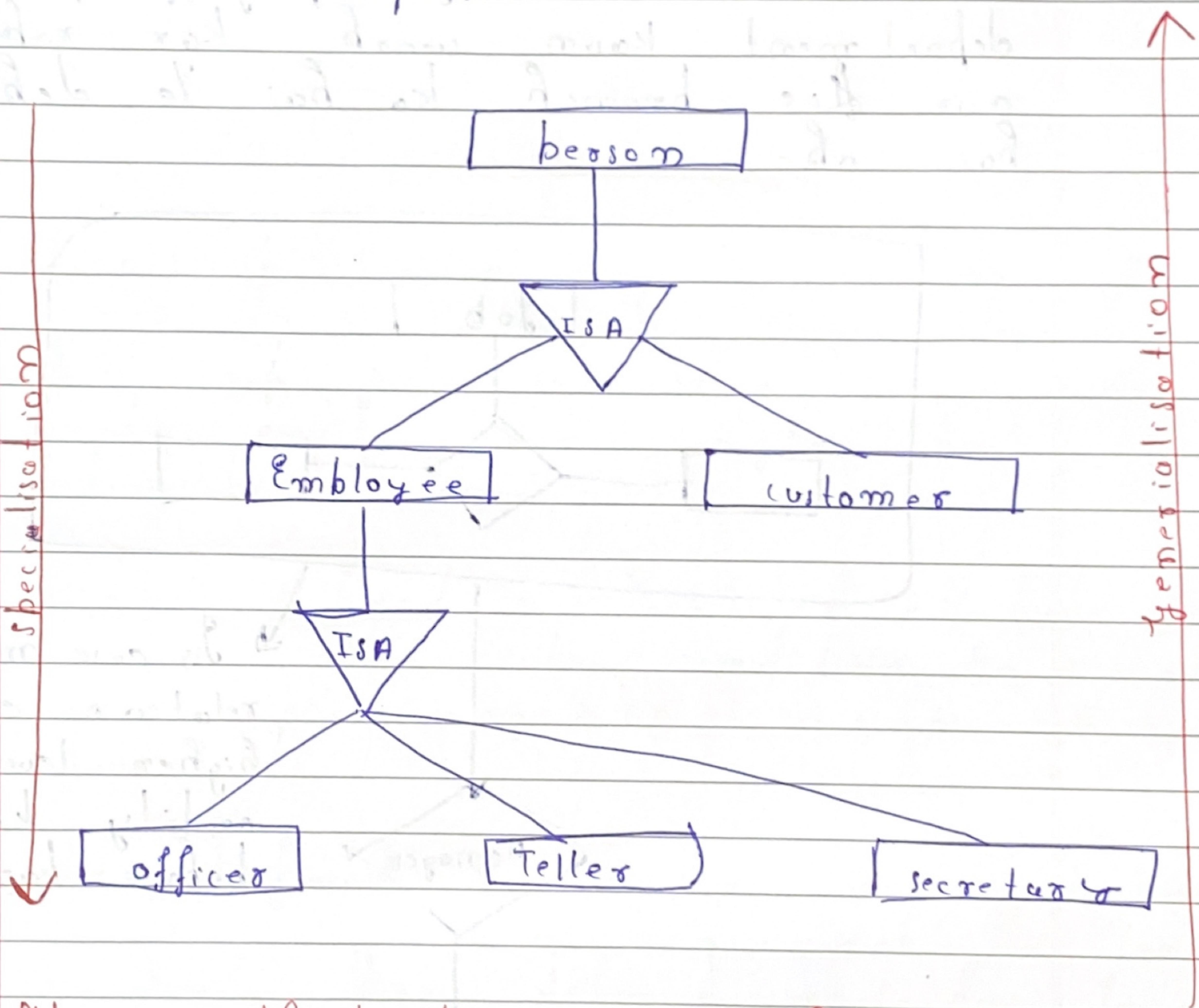


- \* Ek weak entity set ko ek owner Ya identifying entity set ke saath relation hogा.

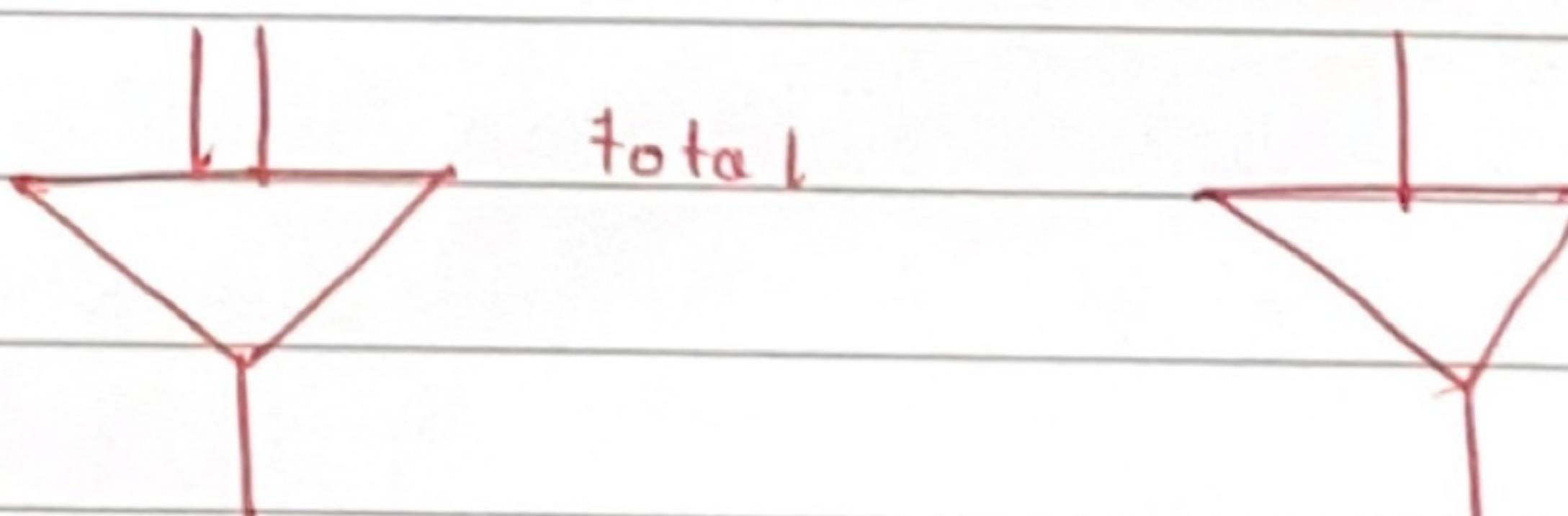
- \* Weak Entity = partial key

## Extended E-R feature

Specialization => Ab ek entity mein bhi subgroupings ho sakti hai toh unko represent

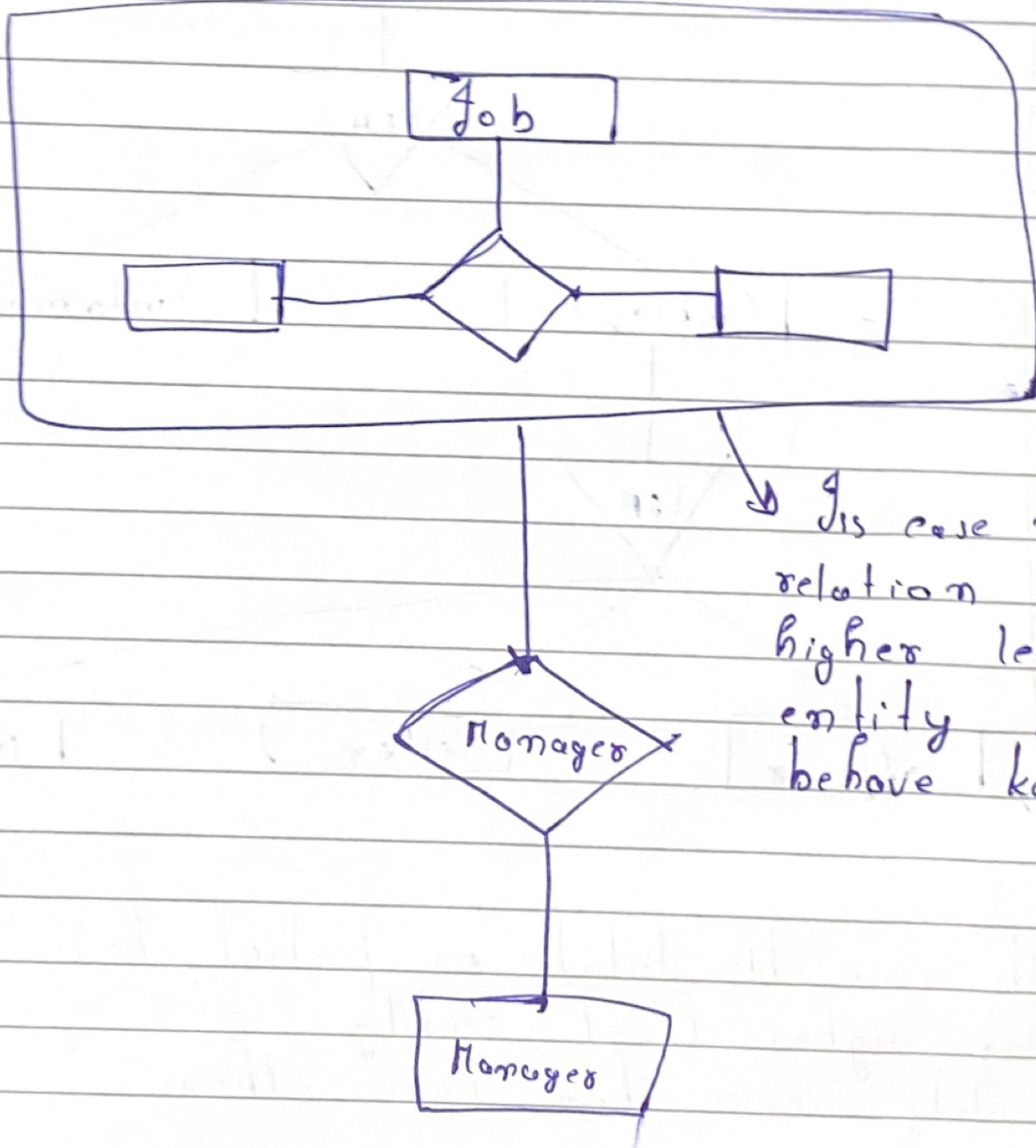


Ab isme bhi total or partial hai. Agar har higher level entity → given lower entity mein hée hui then



## Aggregation

Multiple managers ka ek record rakhna  
hai or HR ka jo project he kissa  
department kaun work kar raha hai  
aur is branch ka hai Ye dekhna  
hai ok.



Attribute Inheritance  $\Rightarrow$  Higher Level Entities  
Lower Level attributes  $\xrightarrow{\text{Ke attributes hain}}$  inherit karte hain

### Constraints on GS.

Membership  $\Rightarrow$  ab isme kya hai ki 2  
tarike se types hote hain:  
Jisey apne pata kar sake ho ki  
lower entity kaunsi higher Entity ko  
belong karti hai.

\* attribute defined  $\Rightarrow$  Yani ek attribute pe  
koi condition ke according  
for ex person  $\cdot$  age  $> 20 \Rightarrow$  teacher otherwise  
student

\* User defined  $\Rightarrow$  Ab isme koi constraint  
on attribute nahi hai Yani  
randomly user defined karta hai Jaise  
Employee  $\rightarrow$  Part time or Full time.

Disjoint Constraint =>

- \* Disjoint Constraint - Ek higher level entity  
entity ko belong karegi.  
sif ek hee lower level
- \* Overlapping - Ab isme kya hai ki ek  
entity one se jada lower  
entity mein participate kar sakti hai

## Relational Data Model

tuple

Row in a table

Column  
headers

Attributes

domain

Data type of column

$$\delta(R) \subseteq \text{dom}(A_1) \times \text{dom}(A_2) \times \text{dom}(A_3)$$

Relation  
Schema

R(A<sub>1</sub>, A<sub>2</sub>, A<sub>3</sub> --- A<sub>n</sub>)

Degree

No. of Attributes

Relation  
State

The set of tuples of a relation at  
any particular instance of time

- Ordering of tuple within a relationship.
- Ordering of values within a tuple matters

- \* A Tuple is a set of  $\langle \text{Attr}, \text{Value} \rangle$  pairs
- \* It is also ordered list of Values.

## # Interpretation of relation

- facts about Entity
- Details about Relationship

## Relational Model Constraints

- \* Implicit or Inherent Model based
- \* Explicit constraints or schema based
- \* Application based or Semantic Constraint

## Schema based Constraints.

Domain  
Constraint

Within each tuple the value of each attribute must be an atomic value of each that belongs to domain  $\text{dom}(A)$

It cannot be multivalued or composite

Key  
Constraint

No two tuples in a state of relation can have same combination of values for all attributes (No duplicates)

Superkey

Set of attributes which are unique  
No two tuples can have same combination

Default Superkey  $\Rightarrow$  Set of all attributes

Key

Minimal Superkey Yani ek aisa set of attributes jisse agar ek attribute toh wo Superkey nahi ban saka

Minimal Super Key

## Trick for E-R

- \* Noun are Entity Set
- \* Adjective are its attributes
- \* Verbs are relation
- \* Adverb are relationship Attribute