

# RDBMS

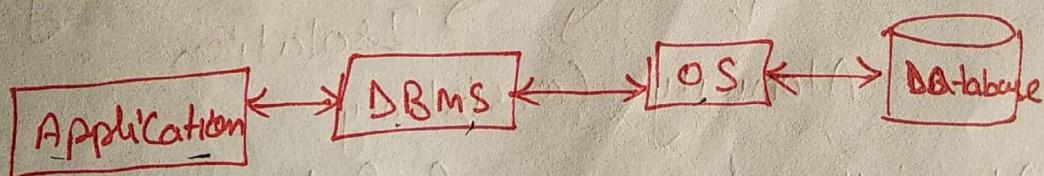
- DBMS  $\Rightarrow$  interrelated data collection होता है।
- यह भी word के मिलकर आता है।

Database + Management System = DBMS

- यह inter related information collection है, जिसका use different user different purposes के लिए मार्ग है,

Rollno.	Name	Branch	Address
1	A	IT	X
2	B	CS	Y
3	C	EC	Z

- database management system Application program के operating system के interface provide करता है।



Applications of DBMS  $\Rightarrow$

- 1) College
- 2) hospital
- 3) Banking
- 4) Reservation

- 5) Telecommunication
- 6) Sales
- 7) finances.

## Advantages of DBMS $\Rightarrow$

1) No Repetition / Less Redundancy

2) Data Independence.

3) More Secure.

4) Less Development time.

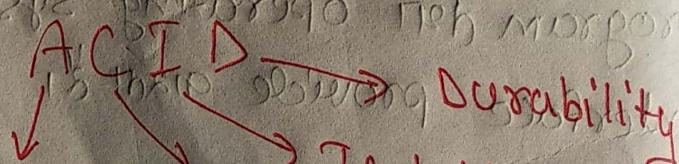
5) Easy Access.

## Disadvantages $\Rightarrow$

1) Cost of software is high.

2) Cost of hardware is high.

3) Complexity of Backup & Recovery.



• Atomicity  $\Rightarrow$  No loss of data.

$A = 1000$   
read(A);

$A = A - 500$ ,  
write(A);

$A = 1000$   
 $A = 500$

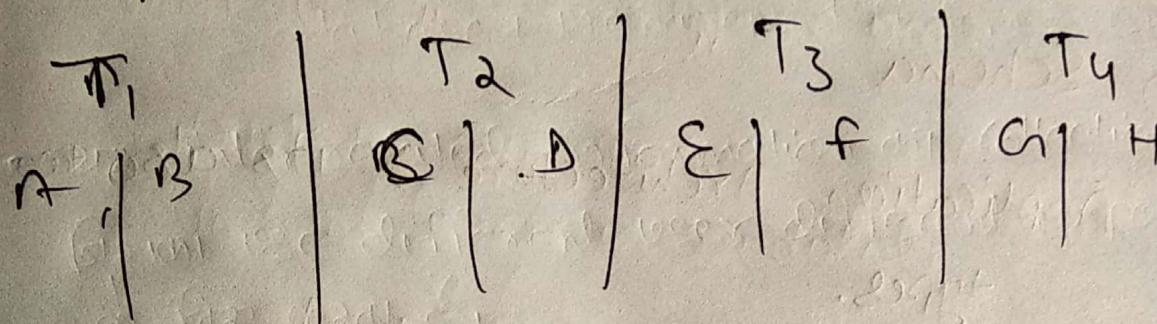
$B = 1000$

read(B);  
 $B \neq B + 500$ ,  
write(B);

$B = 1500$   
 $C = 200$

. Consistency  $\Rightarrow$  initial and last data is equal  
initial sum = final sum.

. Isolation  $\Rightarrow$  data is independently perform parallelly.



. Durability  $\Rightarrow$  Duration of time to complete the transaction.

### Difference between DBMS & file processing

	DBMS	file processing
1) Repetition problem	Not found	found.
2) Atomicity problem	Not found	found
3) ACID problem	Not found	found.
4) more secure		Less Secure.
5) data accessing	Easily	difficult,

## -11 Database Concept & Architecture ⇒

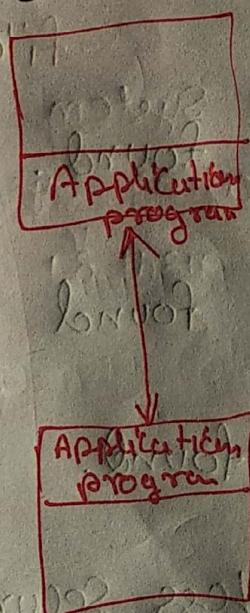
# Architecture of DBMS  $\Rightarrow$  designing structure of database are called Architectures.

it's two types

## 2) Schema

1) Application Architecture  $\Rightarrow$  it's also two types.

two tier Architecture  $\Rightarrow$  दोनों Client  
or Server दोनों से Application program दोनों से directly communicate करता है।



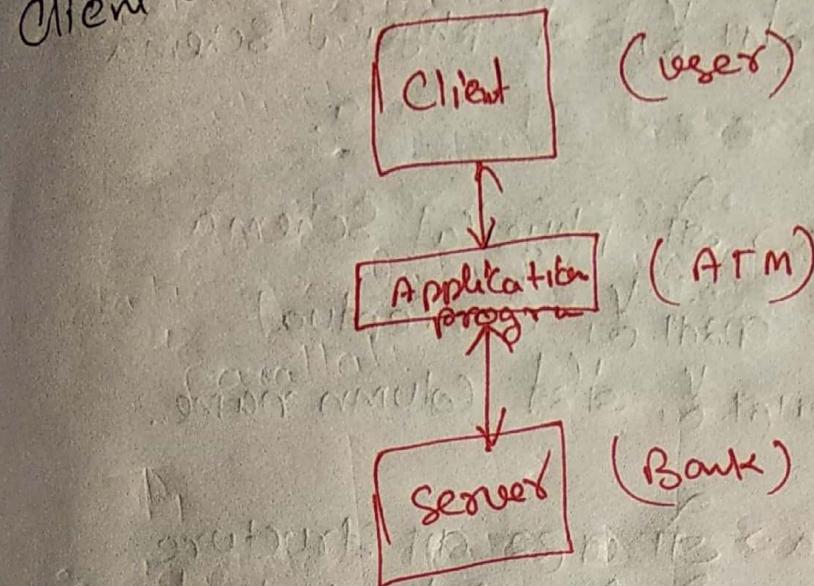
Client (user)

1800 1800 1800 1800 1800

Serves (Bank)

2) Three tier  $\Rightarrow$  client  
After application program  
Work perform at client & server  
Client server at client & server

Client or send query



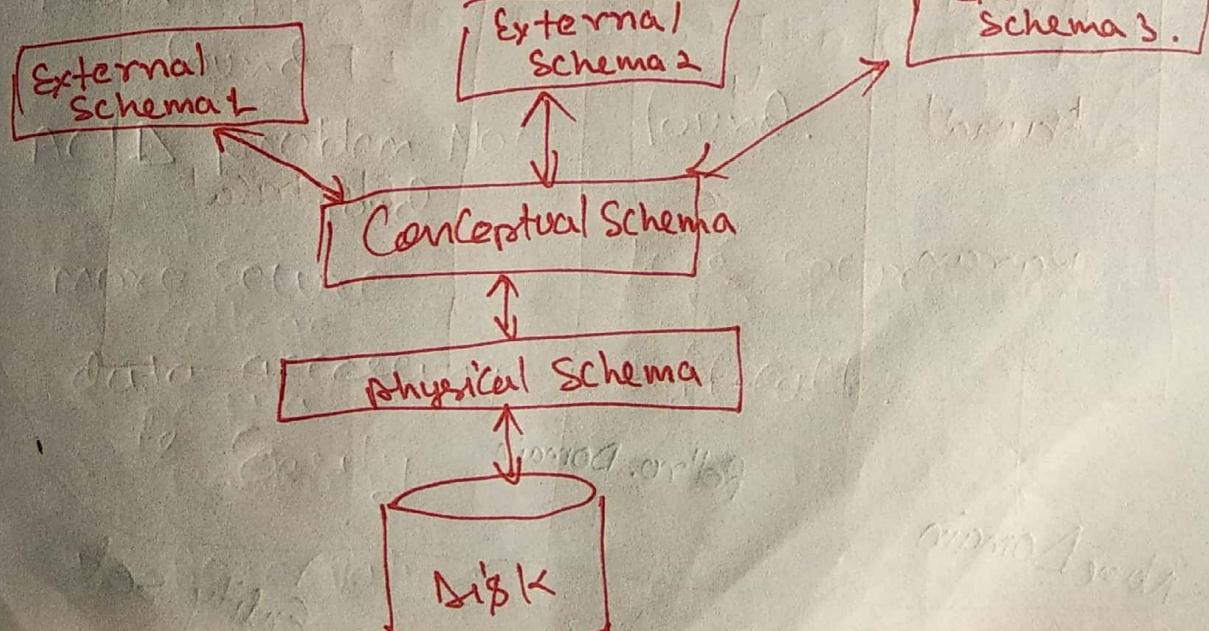
# Schema Architecture  $\Rightarrow$  Overall designing of database are called schema

architecture.

(Internal)  
Physical schema / physical Level

1) Conceptual schema / Logical

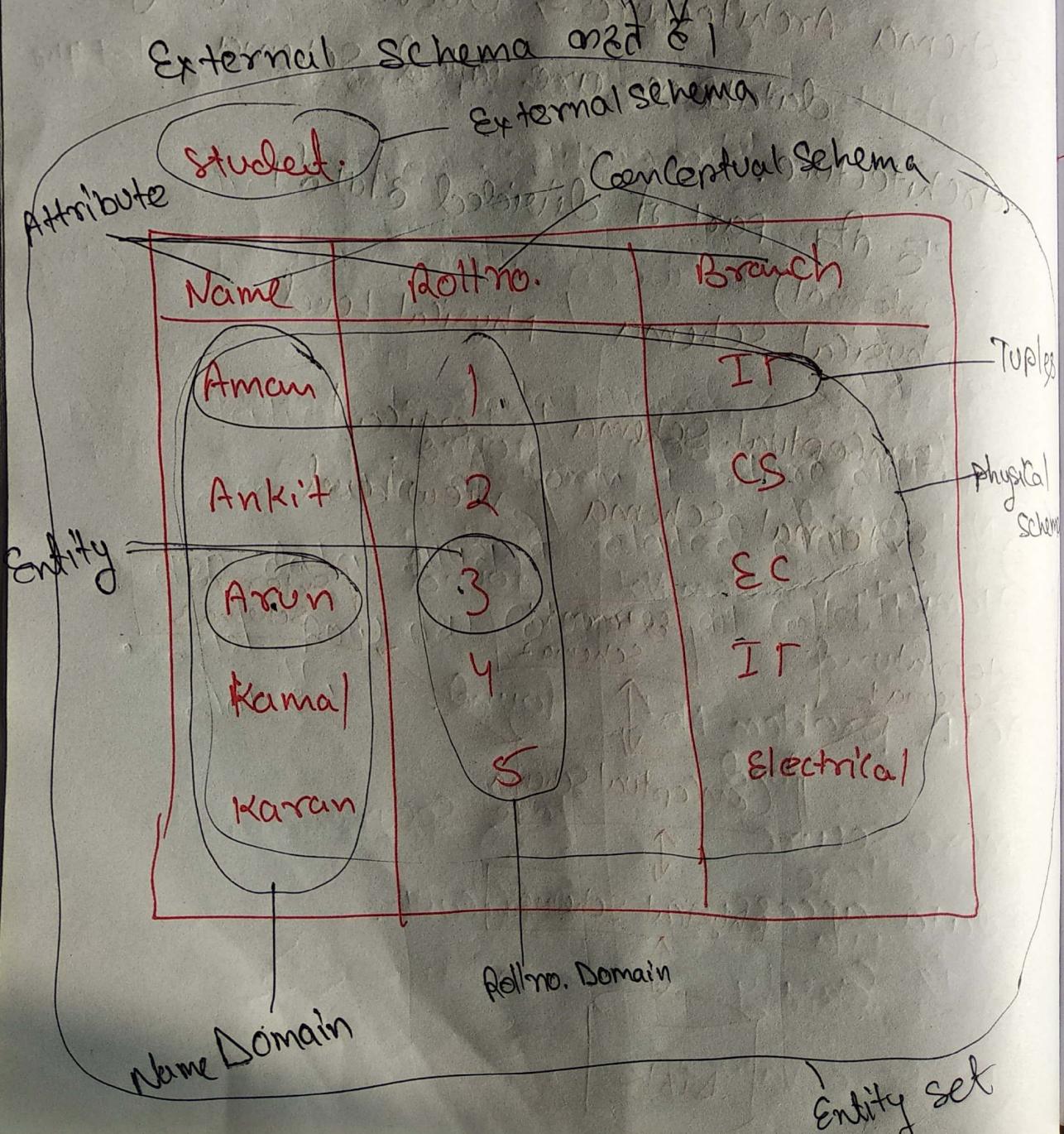
2) External schema / view Level



1) Physical Schema  $\Rightarrow$  It's Actual data in database  
It stores data & its physical schema  
and it's,

2) Conceptual Schema  $\Rightarrow$  It's physical schema  
It represent chart & Conceptual  
Schema consists of, it's column name.

3) External Schema  $\Rightarrow$  It's part of structure  
of database It's part of chart & at last



Data Independence  $\Rightarrow$  यदि डेटा Schema Level  
में किसी changes करते हैं तो उसके  
Effect दुसरे Schema Level पर नहीं होता data

Independency.

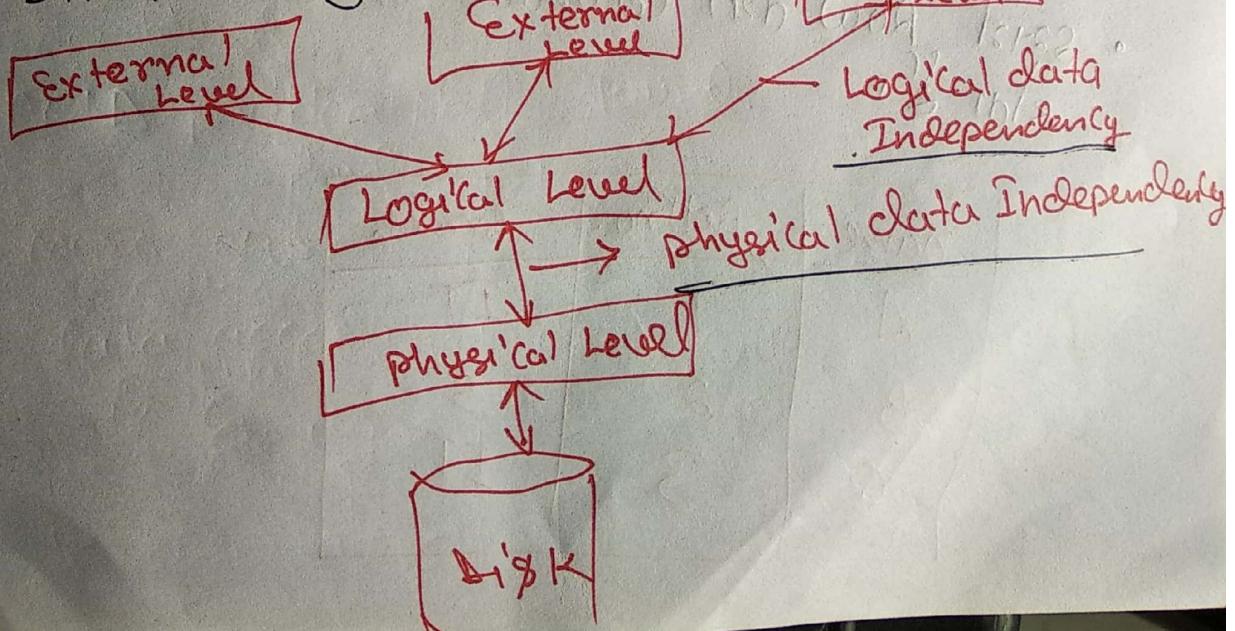
It's two types-

1) Physical data Independence

2) Logical data Independence.

1) physical data Independence  $\Rightarrow$  यदि Physical  
Level में किसी change करते हैं तो उसके  
effect Logical Level में नहीं होता, physical  
data Independence बनता है।

2) Logical data Independence  $\Rightarrow$  यदि Logical Level  
में किसी change करते हैं तो उसके effect  
External Level में नहीं होता, logical data  
Independency बनता है।



# Data Models  $\Rightarrow$  Data models means  
to represent of data.

- Data art ~~an~~ structure Create ~~an~~  
data model ~~an~~ ~~an~~

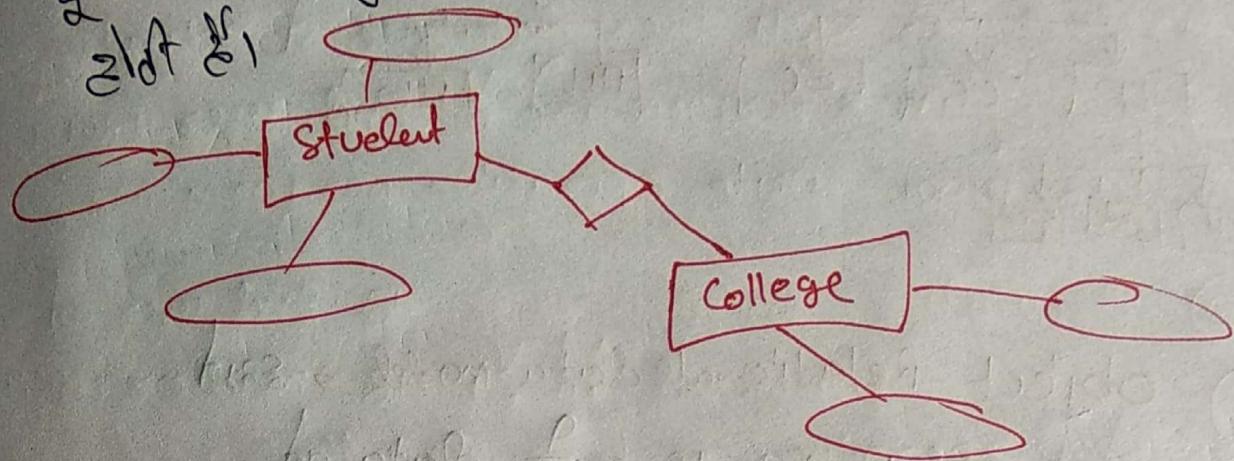
### Types of Data models.

- 1) Relational data model.
- 2) Object-oriented data model
- 3) Network data model
- 4) Hierarchical data model
- 5) Object relational data model

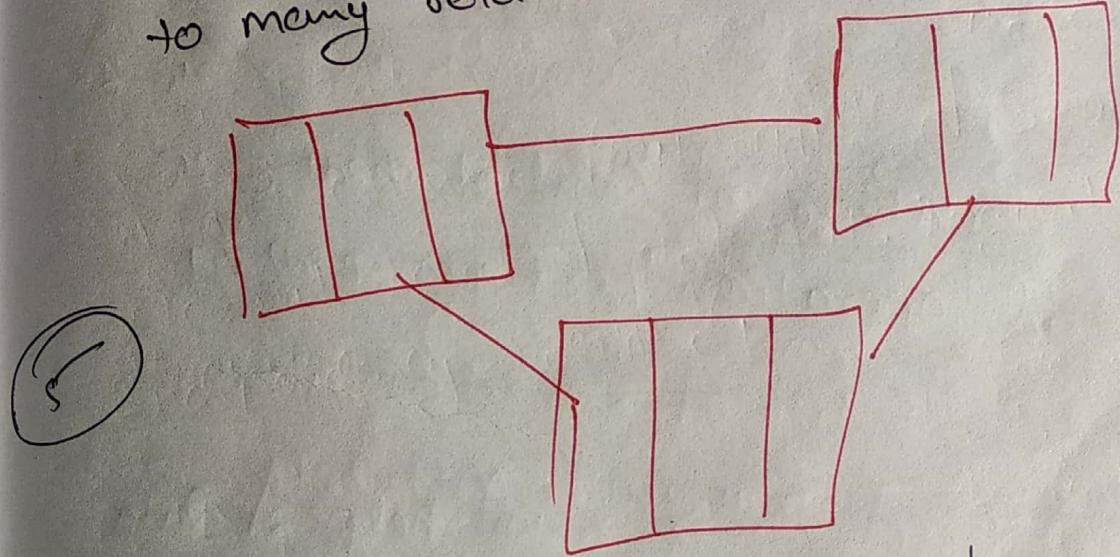
1) Relational data model  $\Rightarrow$  ~~an~~ data  
in table art form ~~an~~ store ~~an~~  
data ~~an~~ related ~~an~~  
• ~~an~~ Row ~~an~~ Column ~~an~~ Collection  
~~an~~

name	Rollno.	Branch
A	1	IT
B	2	CS
C	3	IT

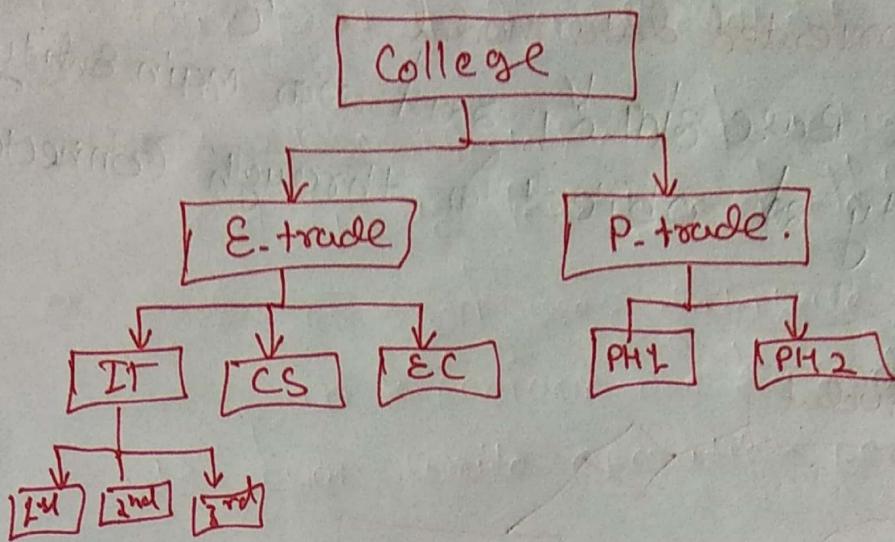
⑥ 2) Object-oriented data model  $\Rightarrow$  यह E-A model  
 इस Base होता है, जिसके main Entity  
 जैसी Entity के object के through Connected  
 होते हैं।



3) Network data model  $\Rightarrow$  सभी data many to  
 many relation of through Connected  
 होते हैं, जिनमें से एक table होता है, जो many  
 to many relation के connected होते हैं।



4) Hierarchical data model  $\Rightarrow$  सभी data one  
 tree structure के होते हैं।  
 एक root data होता है जिसके Subchild होते हैं।

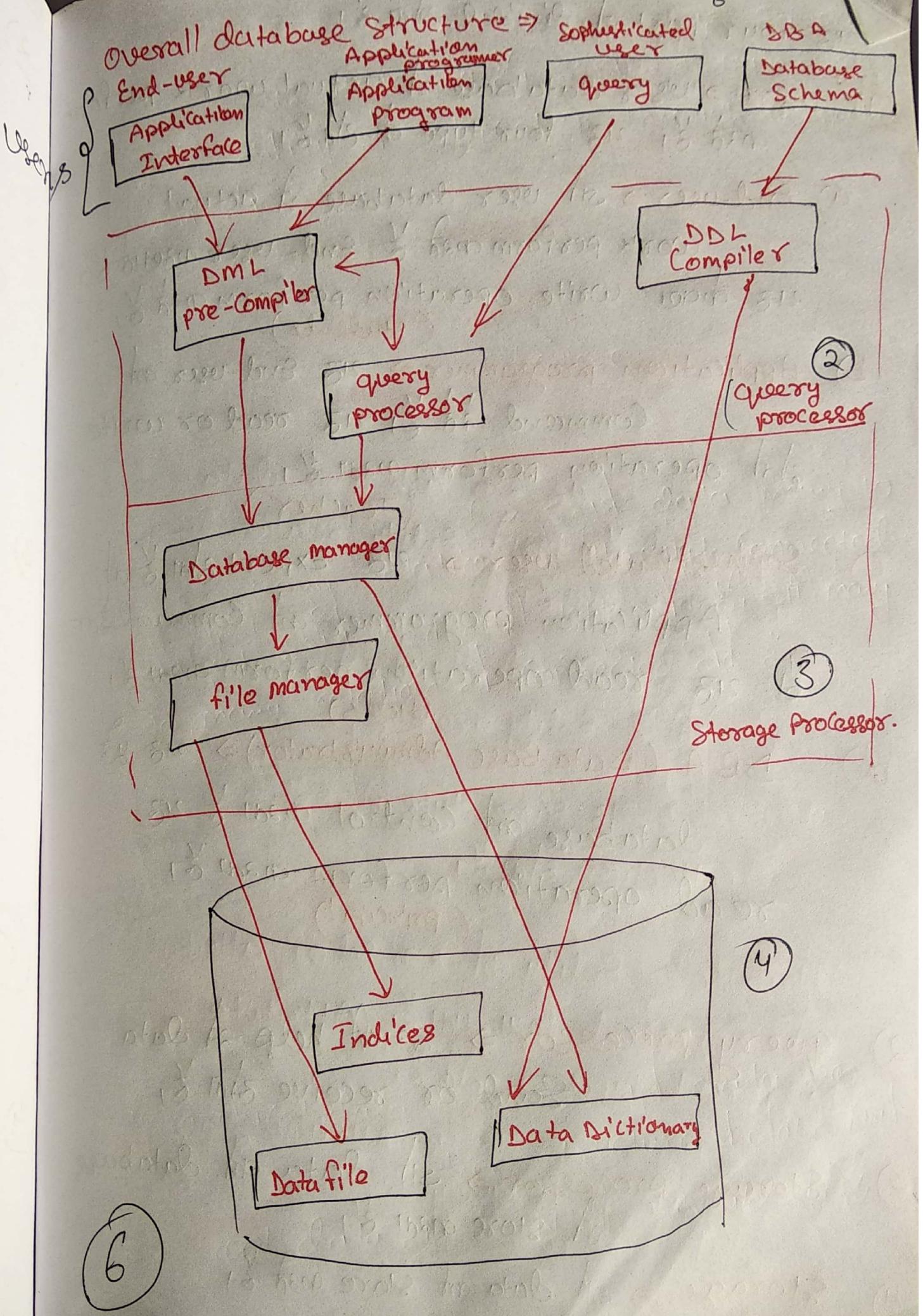


5) Object - Relational data model  $\Rightarrow$  OR

Object Create on the data and  
Access or retrieve data

use ORDBMS

Ex  $\Rightarrow$  C++, Java.



यह four part के divided होते हैं।

Imp.

1) **Users**  $\Rightarrow$  user database of actual work  
मात्र है, यह four type के होते हैं।

a) **End-user**  $\Rightarrow$  sit user database of actual  
work performance मात्र है End-user  
यह direct write operation perform करते हैं  
(students)

b) **Application programmer**  $\Rightarrow$  यह End-user के  
Command के हैं, यह read or write  
operation perform करते हैं  
(Teacher)

c) **Sophisticated user**  $\Rightarrow$  यह expert होते हैं  
Application programmer के Command  
में यह read operation perform करते हैं  
(Hods)

d) **DBA (Data Base Administrator)**  $\Rightarrow$  यह  
database के control करते हैं यह  
read operation perform करते हैं  
(principal)

2) **query processor**  $\Rightarrow$  search help at data  
Send or receive होते हैं,

3) **Storage processor**  $\Rightarrow$  sit data at database  
at store करते हैं।

4) **storage**  $\Rightarrow$  sit data at store करते हैं।

▷ BMS Languages  $\Rightarrow$  Database of four languages ही है।

(7)

1) DDL (Data Definition Language)

2) DML (Data Manipulation Language)

3) DCL (Data Control Language)

4) DQL (Data Query Language)

1) DDL  $\Rightarrow$  संस्करण की विधायत अवधारणा देता है। But data के बदलाव को नहीं करता है।  
Changes नहीं करता है।  
Ex  $\Rightarrow$  table name change  
 $\Rightarrow$  table को remove करता है।  
So that DBA perform करता है।

query  $\Rightarrow$  Create, Alter, Drop.

2) DML  $\Rightarrow$  संस्करण की विधायत अवधारणा देता है। But data के बदलाव को करता है।

query  $\Rightarrow$  Insert, update, delete etc.

3) DCL  $\Rightarrow$  संस्करण की विधायत अवधारणा देता है। Access करने की विधायत है।

query  $\Rightarrow$  Rollback, privacy, grant, revoke

DQL  $\Rightarrow$  इनपुट डेटा के दाता अवलोकन के लिए एक विशेष भूमिका है।

query  $\Rightarrow$  Select.

Data modeling using the E-R  
Model

E-R model  $\Rightarrow$  यह एक high level data  
model है, जिसके Entity - Relationship  
पर आधारित है।

Entity  $\Rightarrow$  जो एक व्यक्ति की जाति होने के लिए जानी चाहिए है।

Entity set  $\Rightarrow$  वह same type के entity का  
Collection होता है।

Attribute  $\Rightarrow$  Entity का एक set of attribute  
के represent करता है। Attribute  
different types के होते हैं।

Key Attribute  $\Rightarrow$  वह attribute जो कि एक  
table के uniquely identify करता है।  
Ex - Roll no.?

2) multivalue Attribute  $\Rightarrow$  એક attribute બાંધી value  
2005 એ સ્થાની હતી હૈ,

(8)

Ex  $\Rightarrow$  phone no.

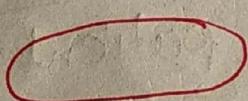
3) Composite Attribute  $\Rightarrow$  એક attribute એવું subpart  
જે divide કરી શકતું હૈ।

Ex  $\Rightarrow$  Name  $\leftarrow$  fname  
                          Lname

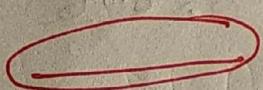
4) derived attribute  $\Rightarrow$  એક attribute એવું જે

Attribute એવી છાડી શકતું હૈ,  
જે - ~~is~~ Date of Birth એ અને age findout એ માટે,

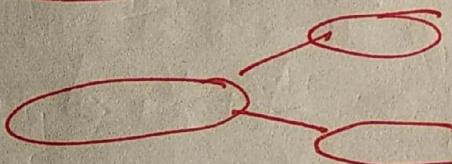
E-R diagram Notation  $\Rightarrow$



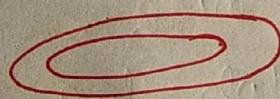
Attribute



Key Attribute



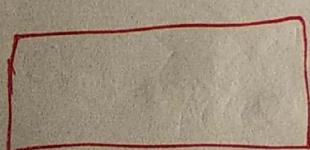
Composite  
Attribute



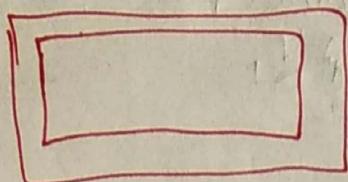
multivalue Attribute



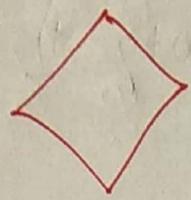
derived Attribute



strong Entity set



Weak Entity Set

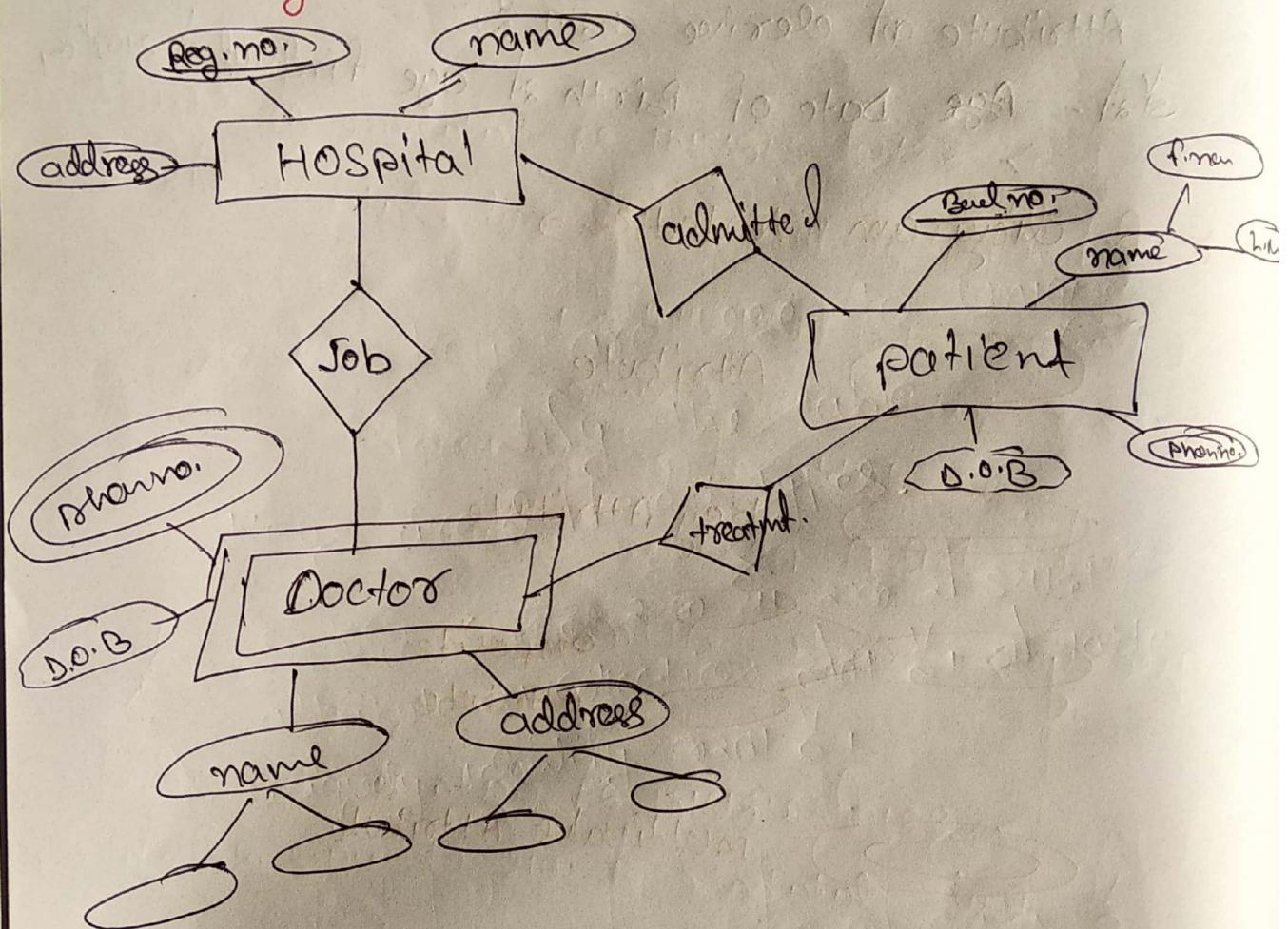


relationship



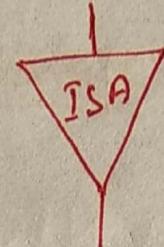
Link.

Ex ⇒ design a E-R diagram of Hospital.



# Generalization & Specialization $\Rightarrow$

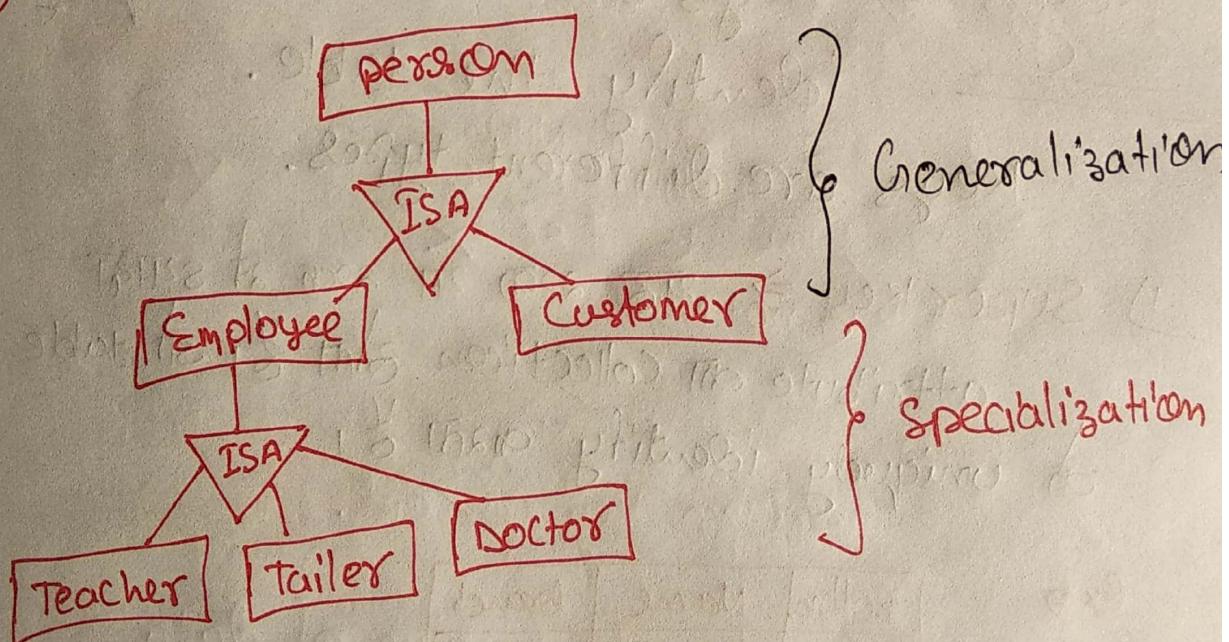
Generalization भाफी एक प्रोसेस अव ग्रेनरल  
फिल्ड होती है, जो इसके General Value व  
divide करती है तथा ISA एवं denote  
प्रोटो टाइप है।



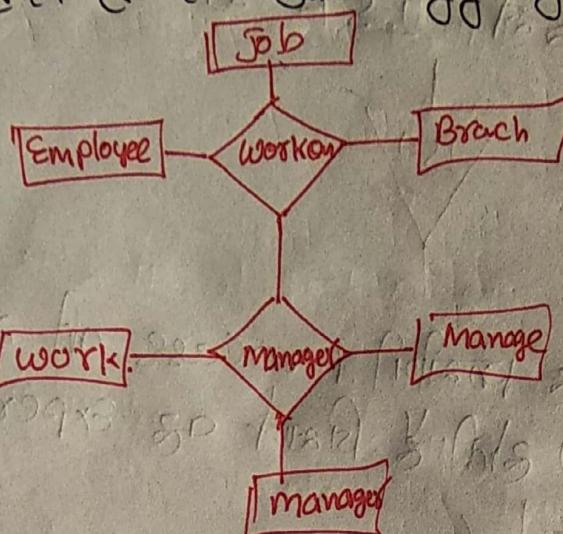
Specialization  $\Rightarrow$  यह किसी प्रोसेस अव 2005  
special field होती है जिसमें एक expert होते हैं



Ex :-



Aggregation  $\Rightarrow$  Relationship among  
relationship. यह एक relationship  
2<sup>nd</sup> क्षेत्रीय relation direct connected  
होता है at 3<sup>rd</sup> aggregation एवं 4<sup>th</sup>



Keys  $\Rightarrow$  Keys are used to uniquely  
identify the table.

Keys are different types.

1) Superkey  $\Rightarrow$  यह एक या एक अपरिवर्तनीय  
attribute or collection का है। जो table  
को uniquely identify करता है।

Ex  $\Rightarrow$

Rollno	Name	Branch
1	A	IT
2	B	CS
3	C	EC

Ex  $\Rightarrow$   $S_1 = \text{Rollno}$ ,

$S_2 = \text{Rollno, name}$

$S_3 = \text{Rollno, Branch}$

sy - Rollno, Name, Branch

2) Candidate key  $\Rightarrow$  निम्नलिखित कॉमन्डेटी के संख्या का मान होता है।  
 Super key का अर्थ यह है कि यह एक बड़ी संख्या है जो एक वर्ग का अधिक संख्या का मान होता है।  
 Uniquely identify इसका अर्थ है कि यह एक वर्ग का अधिक संख्या का मान होता है।

(10)

Ex  $\Rightarrow$   $C_1 = \text{Rollno, Name}$

$C_2 = \text{Rollno.}$

$C_3 = \text{Rollno, Branch}$

3) Primary key  $\Rightarrow$  निम्नलिखित कॉमन्डेटी के संख्या का मान होता है। यह एक वर्ग का अधिक संख्या का मान होता है। और यह एक वर्ग का अधिक संख्या का मान होता है।

Ex  $\Rightarrow$   $P_1 = \text{Rollno.}$

4) Foreign key  $\Rightarrow$  यह एक वर्ग का अधिक संख्या का मान होता है। यह एक वर्ग का अधिक संख्या का मान होता है। यह एक वर्ग का अधिक संख्या का मान होता है। यह एक वर्ग का अधिक संख्या का मान होता है।

foreign key  $\Rightarrow$  foreign key P.K

P.K	Rollno.	Name	ID	Branch	address
1	A	4	4	IT	X
2	B	6	7	CS	Y
3	C	8	9	EC	Z

foreign key = ID

# Integrity Constraints  $\Rightarrow$  data ont database  
 जो store chat of DB Rules कहते हैं।  
 इन्हें Integrity Constraints कहते हैं।  
 ये निम्न तरीके होते हैं।

1) Entity Integrity Constraints

2) Referential Integrity Constraints

3) Domain Constraints

1) Entity Integrity Constraints  $\Rightarrow$  यदि कोई

Attribute ont primary key define

फॉरम वे कोई नहीं बना सकते।

P.K	Rollno.	name	Branch
1		A	IT ✓
2	NULL	NULL	✓
3	NULL	CS	✓
NULL	B	EC	X

P.K is not Null

2) Referential Integrity Constraints  $\Rightarrow$

foreign key concept.

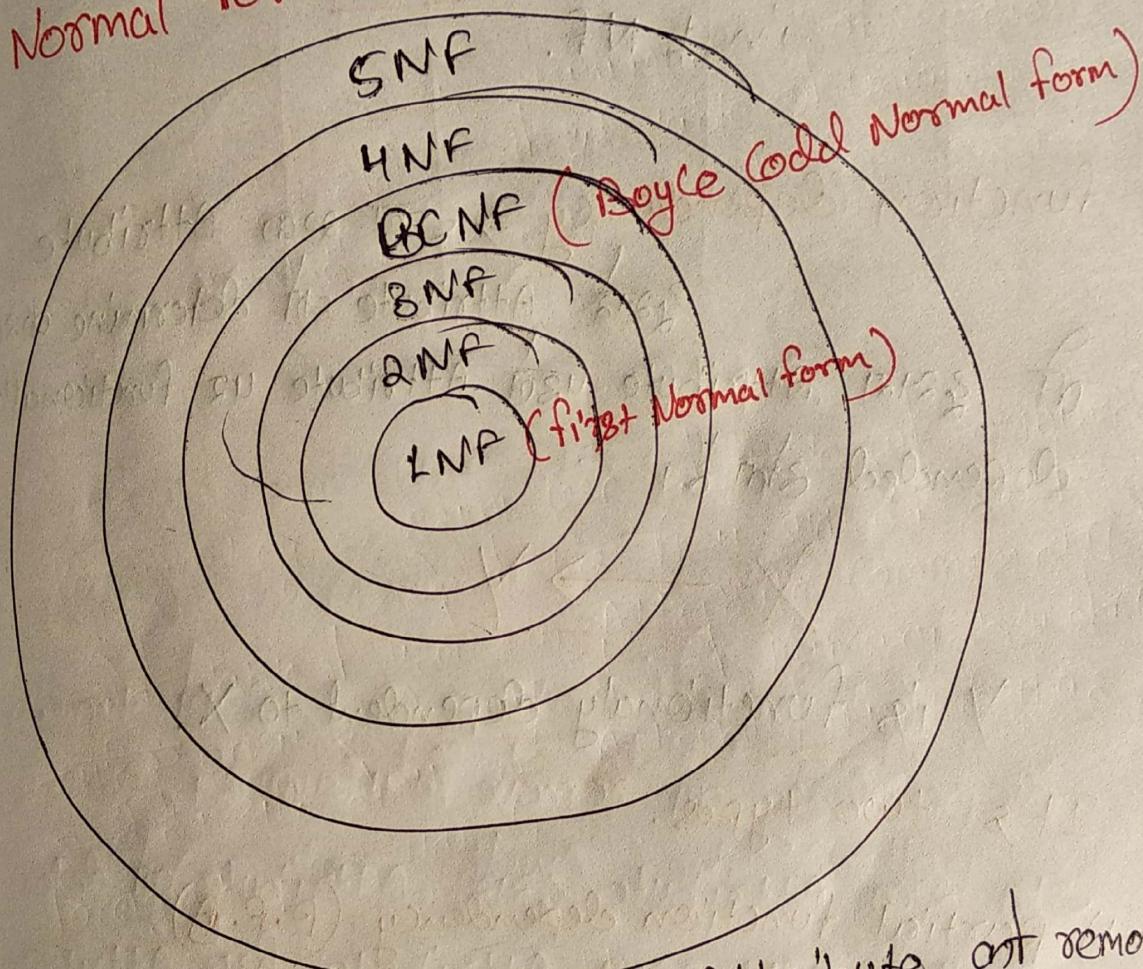
3) Domain Constraints  $\Rightarrow$  Value of data  
 should be atomic. Column of  
 chart 200 की value होनी चाहिए।

Name	Rollno.	Branch
A	2	IT, CS X
B	3	EC ✓

Normalization

Normalization  $\rightarrow$  Technique  $\&$  Standard  $\rightarrow$  to manage database table  $\&$  manage data.

Normal forms  $\Rightarrow$



1) 1NF  $\Rightarrow$  यदि multivalue Attribute  $\&$  remove  
 form  $\&$  तो क्या होगा ?  
 (जट दिखता है, table  $\&$  ~~जटी~~ data Atomic  
 form  $\&$  होते चाहिए)

Name	Rollno.	Address
A	1	Dehradun, N. Neogar
B	2	Rishikesh, Dehradun
C	3	Rishikesh. <u>Not in 1NF.</u>

o Convert it LNF by us.

Name	rollno.	address
A	1	Dehradun
A	1	N.Nagar
B	2	Rishikesh
B	2	Dehradun
C	3	Rishikesh

In LNF.

= function dependency  $\Rightarrow$  यदि एक Attribute  
जो किसी Attribute का determine करता है।  
तो उसे Attribute का द्वारा उसे functionally  
dependent कहते हैं।

$$X \rightarrow Y$$

Y is functionally dependent to X.

It's two types.

- 1) partial function dependency (P.F.D)
- 2) fully function dependency (F.F.D)

1) partial function dependency  $\Rightarrow$  यदि एक Attribute  
जो किसी Attribute का determine करता है।

$$X \rightarrow Y$$

2) Fully function dependency (F.F.D)  $\Rightarrow$  सत्ता  
 यदि Attribute नियम भवति तो Attribute का  
 determine करके वह individual का determine  
 कर सकता है।

$$A \text{ } B \rightarrow C$$

$$\begin{matrix} A & \not\rightarrow C \\ B & \not\rightarrow C \end{matrix}$$

(12)

Student.

Name	Rollno.	Branch	Address
A	1	IT	X
B	2	CS	X
C	3	EC	Y
A	4	IT	Y

P.F.D      Name  $\rightarrow$  Branch

F.F.D

Name Rollno.  $\rightarrow$  address

Name  $\not\rightarrow$  address

Rollno.  $\not\rightarrow$  address

# 2 NF (Second Normal form)  $\Rightarrow$  सभी Fully functional dependency वा partial function dependency नहीं होती हैं।

Student 1

Name	Branch
A	IT
B	CS
C	EC

Student 2

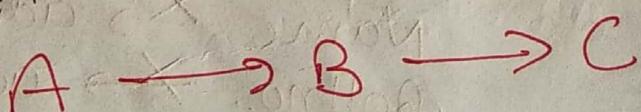
Name	Rollno.	Address
A	IT	X
B	CS	X
C	EC	✗
A	IT	Y

2 NF

← Back table Student

# 3 NF (third Normal form)  $\Rightarrow$  सभी transitive dependency को remove करना है।

Attribute वाले Attribute को determine करने की क्षमता है।



(A, B)      (A, C)

Rollno.	Name	Branch
1	A	IT
2	B	CS
3	C	EC
4	D	EEC

NOT in 3NF  
Rollno. → Name → Branch.

it divide into two table.

student 1

student 2

Rollno.	Branch
1	IT
2	CS
3	EC
4	EEC

Rollno.	Name
1	A
2	B
3	C
4	D

it's in 3NF

# BCNF  $\Rightarrow$  2E, 3NF are extended form

Same 3NF



# SQL

Unit - 6

(Structure Query Language)

1) Create  $\Rightarrow$  Create table tablename ( columnname1 datatype, Columnname2 datatype -- ColumnnameN datatype ), don't write other possible

Ex  $\Rightarrow$

Create table Student ( name varchar(20), Branch varchar(20), Rollno. int(10) ),

2) Insert  $\Rightarrow$  Insert into tablename ( columnname1, Columnname2 -- ColumnnameN ) Values ( value1, value2 -- - valueN ),

Ex  $\Rightarrow$  Insert into Student ( name, Branch, Rollno. ) value ('Ayush', 'IT', 02),

3) Select  $\Rightarrow$  1) whole table Selection  
2) particular row Selection  
3) particular Column Selection

1) whole table Selection  $\Rightarrow$

Select \* from tablename;

Ex  $\Rightarrow$  Select \* from Student;

(14)

2) particular row selection  $\Rightarrow$

Select \* from tablename where condition;

Ex  $\Rightarrow$  To find Rollno. 2 student data.

Select \* from Student where Rollno. = 2;

Select \* from Student;

particular Column selection  $\Rightarrow$

Select Columnname1, Columnname2 ... from tablename;

Ex  $\Rightarrow$  Select name from Student;

Select name from Student;

4) update  $\Rightarrow$

update tablename Set Columnname = "value"  
where condition

Ex  $\Rightarrow$  Update Student Set Name = 'Aman'  
where Rollno. = 02;