
Prepared and Written By
RAZEEN AHMED

CSE-370
[DATABASE SYSTEMS]

HANDWRITTEN NOTE

Computer Science and Engineering
BRAC University

Github: github.com/razeen
LinkedIn: [linkedin.com/in/razeenahmed](https://www.linkedin.com/in/razeenahmed)

File based Approach

The data doesn't overlap.

The overlapped data isn't shared.

DBMS

↓
DB
↓
Data

+ Opt. App

meta data &

Database Approach

both of the users data are stored in the same database.

stored data stay separately.

* Using DBMS we can put different constraints

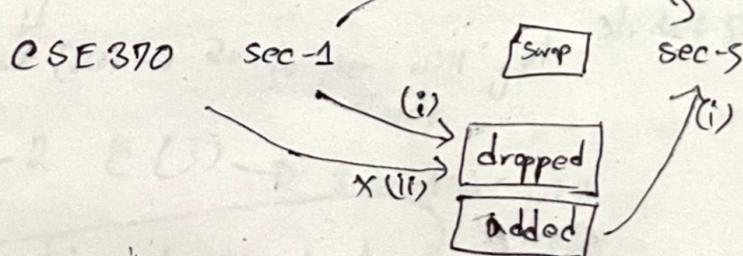
* Concurrent users → different users can process data at the same time.

Main characteristics of Database Approach :-

100% data independence is not achievable.

→ we can use data approach for data abstraction

Recover subsystem



either drop dropped + added

if dropped only ✓ then connection failure then it will make sure it go back to the prev

(the data which will show us a logical view)

Database users

workers behind the scene

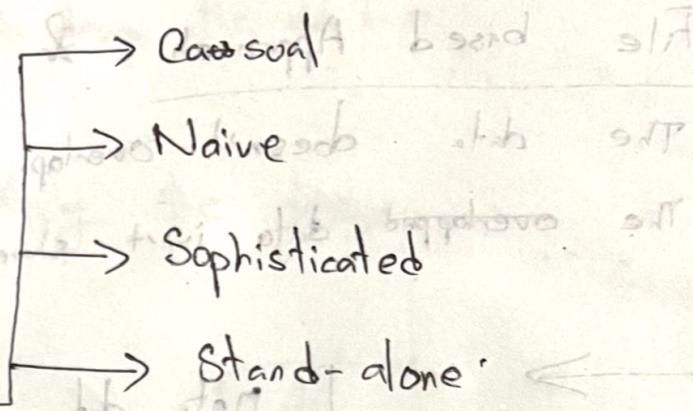
(created)

actors on the scene

Mirror server] → same data is stored in multiple serve.

→ Actors on the scene:-

- (i) Database administrator
- (ii) Database designers
- (iii) Software engineers
- (iv) End users
(final customers)

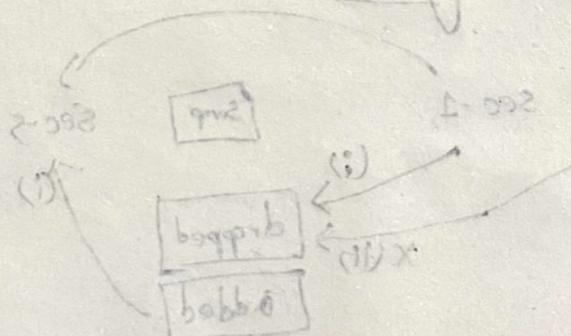


When DBMS may not suffice:-

- (i) When thousands of data comes in ms (short time)

new sendata

old data
sendata
set
inplace
(bottom)



babb0 + boppib pub -> boppib
new value boppib -> babb0

`mysql -u root -p`
 ↓ user → password
 ↓ root user

copy the command in

root and save in a file named

root user

global stub

concept

lab stub (without last digit) ← unchanged

Show databases

global stub hoagd ←

show databases;

First Name	Last Name	Email	DOB	Password
John	Doe	john.doe@gmail.com	1990-01-01	password

Create database Your_DB_Name;

use Your_DB_Name

Not adaptive to the input

Varchar(20) → 20 position will get allocated

column header

Varchar(3) → 3 position will get allocated

Select : Stu-ID, Projectmarks

from lab_grades;

adaptive to the input

= decimal(3,2)
total 3 digits

2 digits after decimal point

04/06/23

DATA MODELS:-

structure of database → what type of database and columns will be there. What type of data type

CATEGORIES OF DATA MODELS :-

→ Conceptual (high level, semantic) data model.

→ Physical data models

→ Implementation data models :-

Subspace between conceptual & physical models

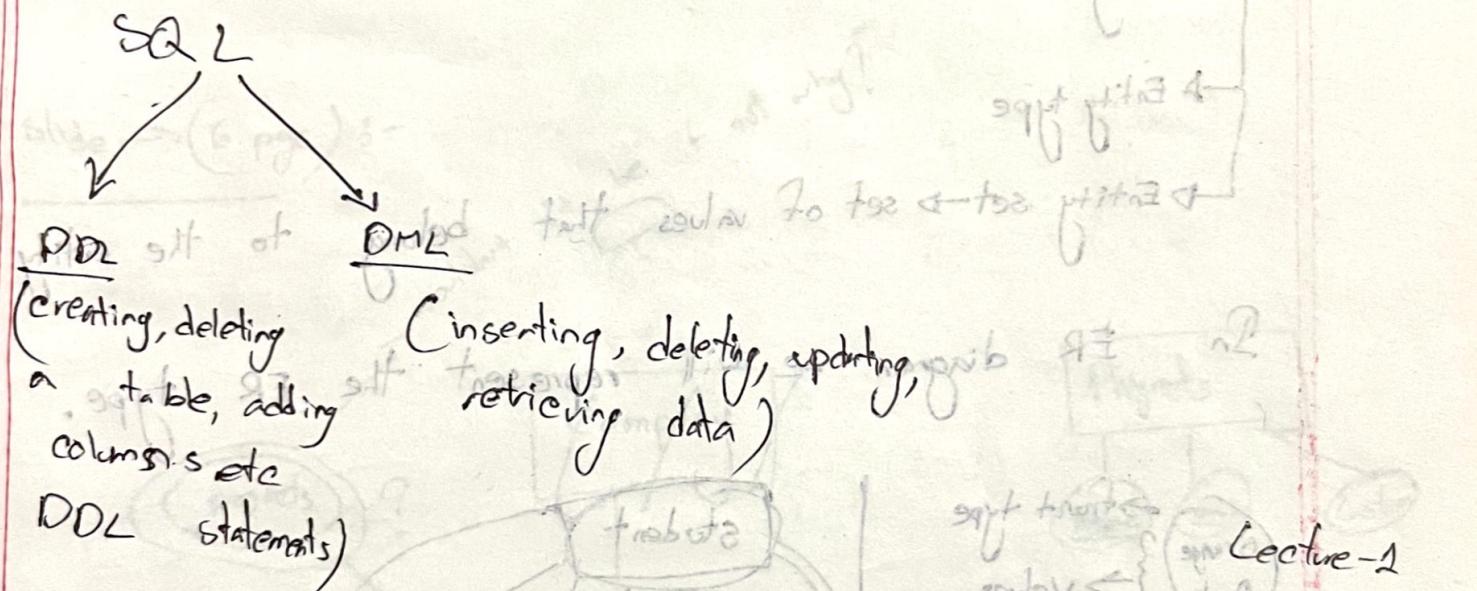
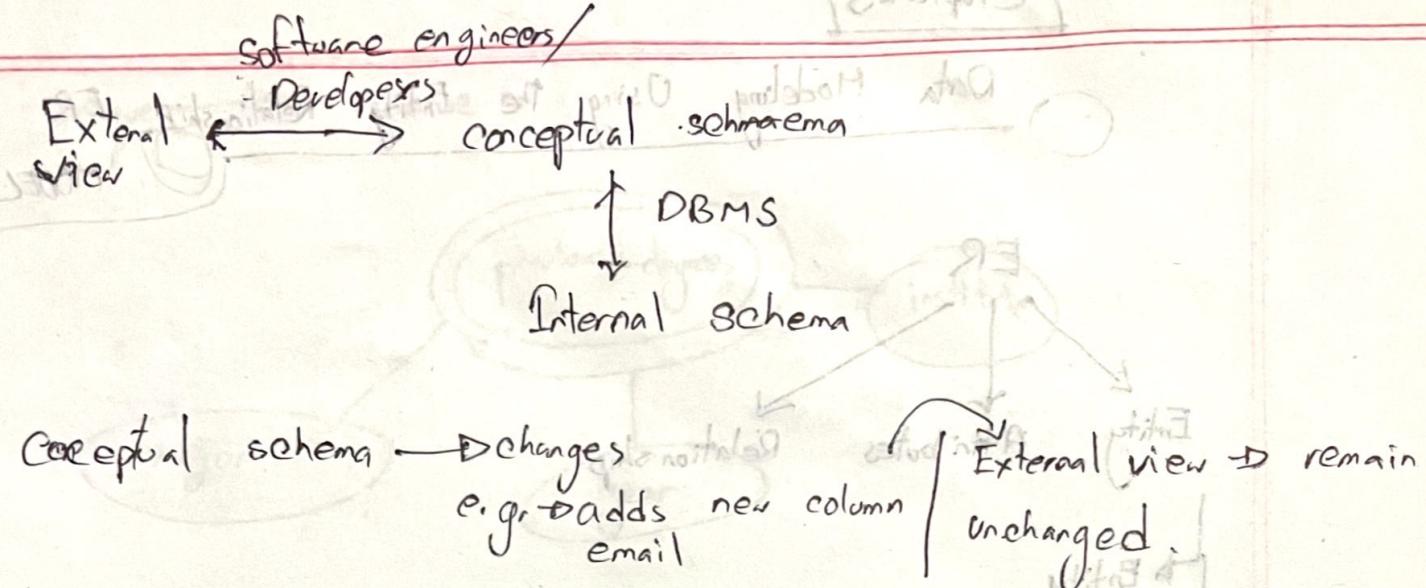
Description of database :-

meta-data → what kind table, columns, datatype will be there

intension → intend to do something.

Database state :- collection of all data. & Data stored at a particular moment in time.

Three schema architecture :- (Theoretical concept).



Question:-

Scenario given → logical independence

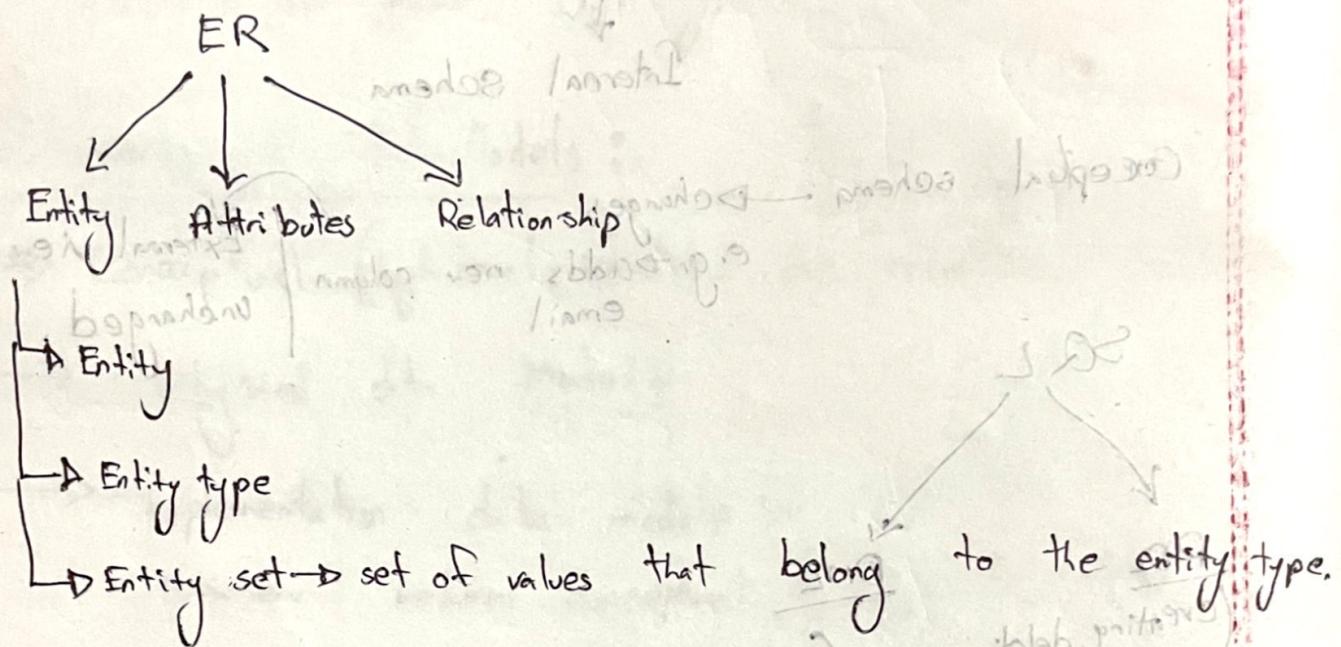
MCQ | SQL

Ques 2
Ans 1

07/06/23

Chapter-3

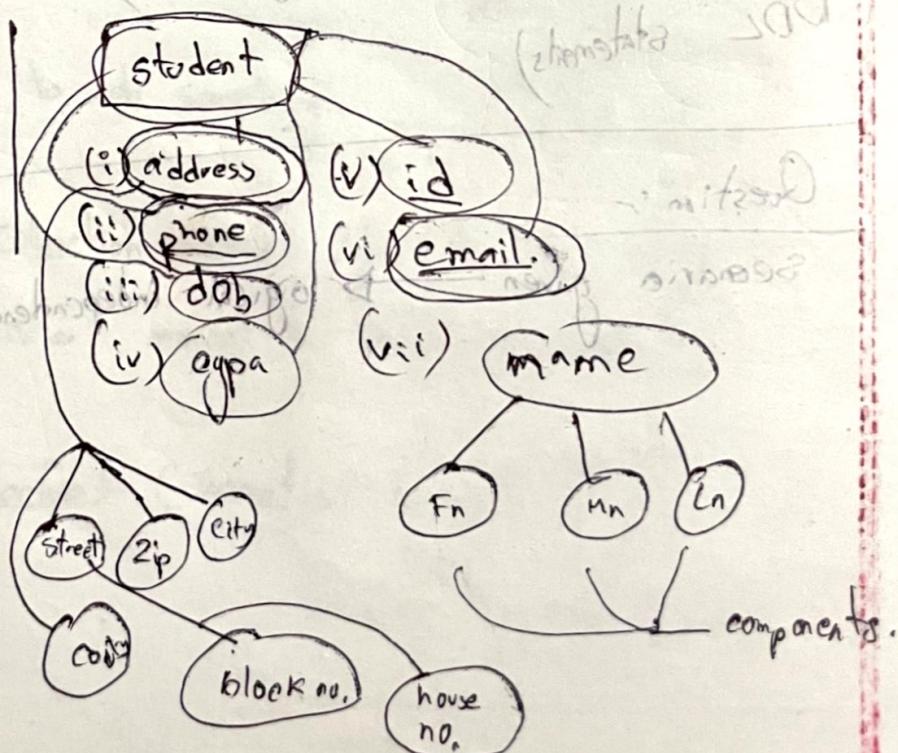
Data Modeling Using the Entity Relationship Model



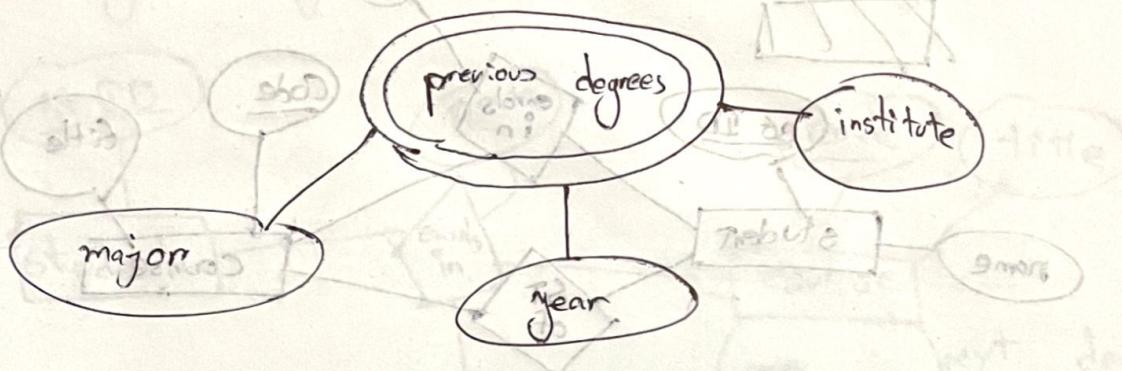
In ER diagram we will represent the ER type.

Orange, Apple } → fruit type
} → values

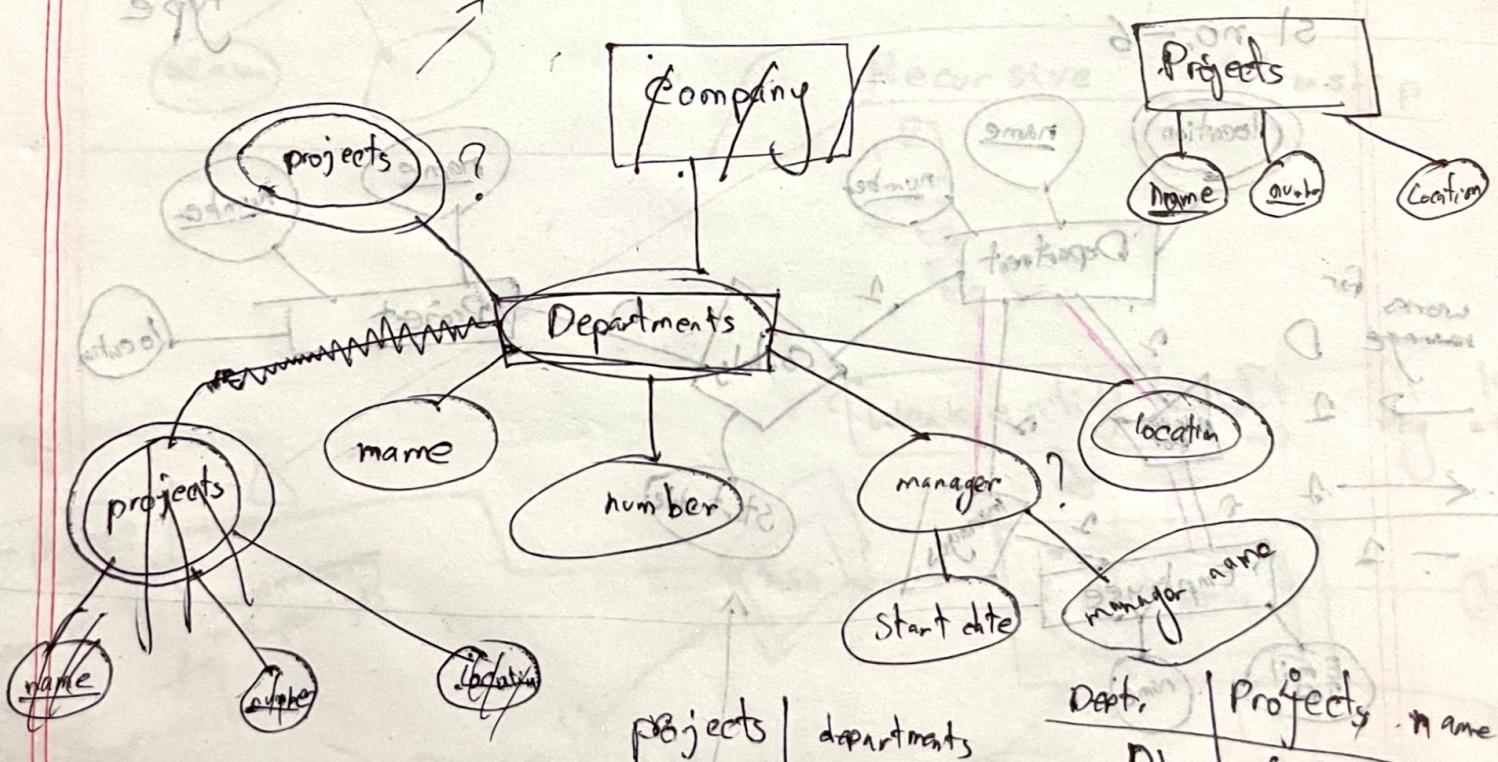
multivalued
↳ double oval



unique → underline



slide → (6 page) :-



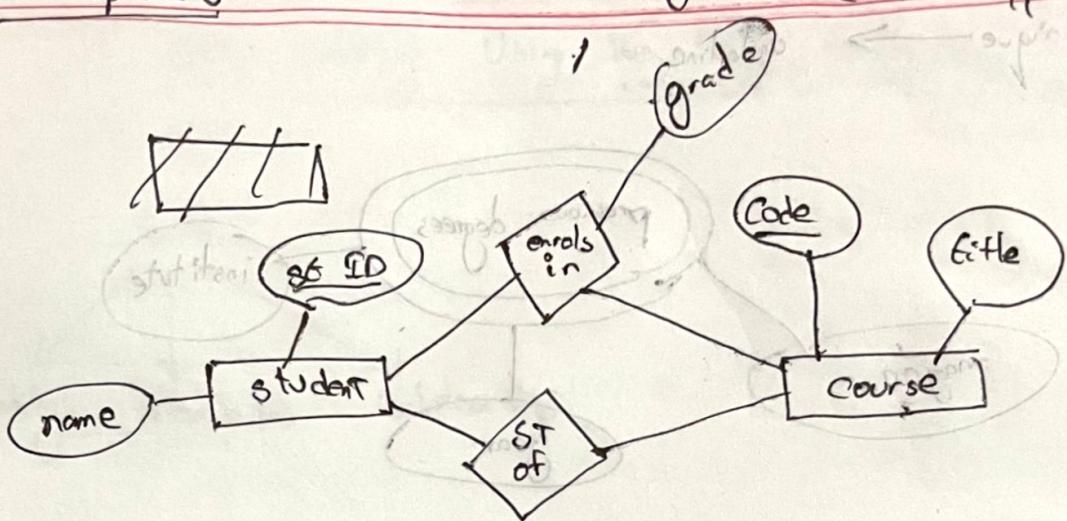
Projects	departments
P1	D1
P2	D2
P3	D3

Dept.	Projects	Name
D1	{ P1, P2 }	
D2		
D3	{ P3 }	

14/06/23

Chapter 3

student → grade (we don't know which course he got)
 course → grade (we know which student got the grade)



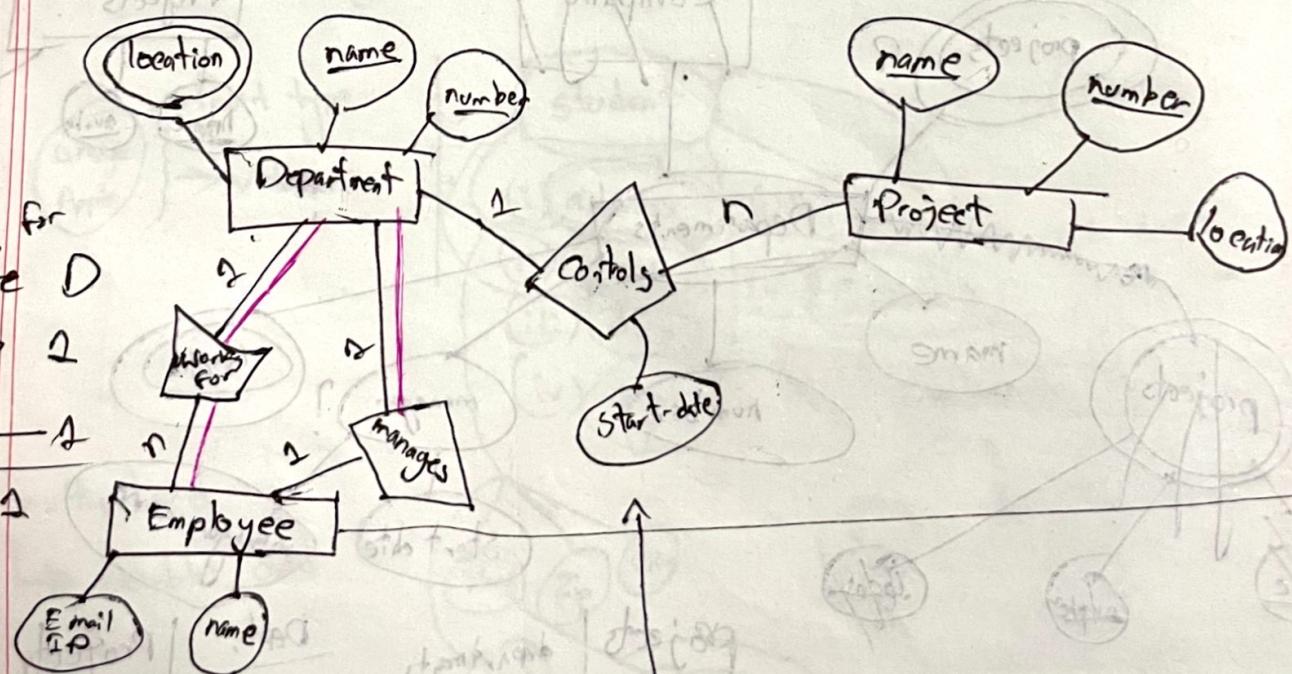
Noor - enrolls in - CSE 870

Uday - enrolls in - CSE 370

Navid - ST of - CSE 110

→ relationship type

S1 no. - 6



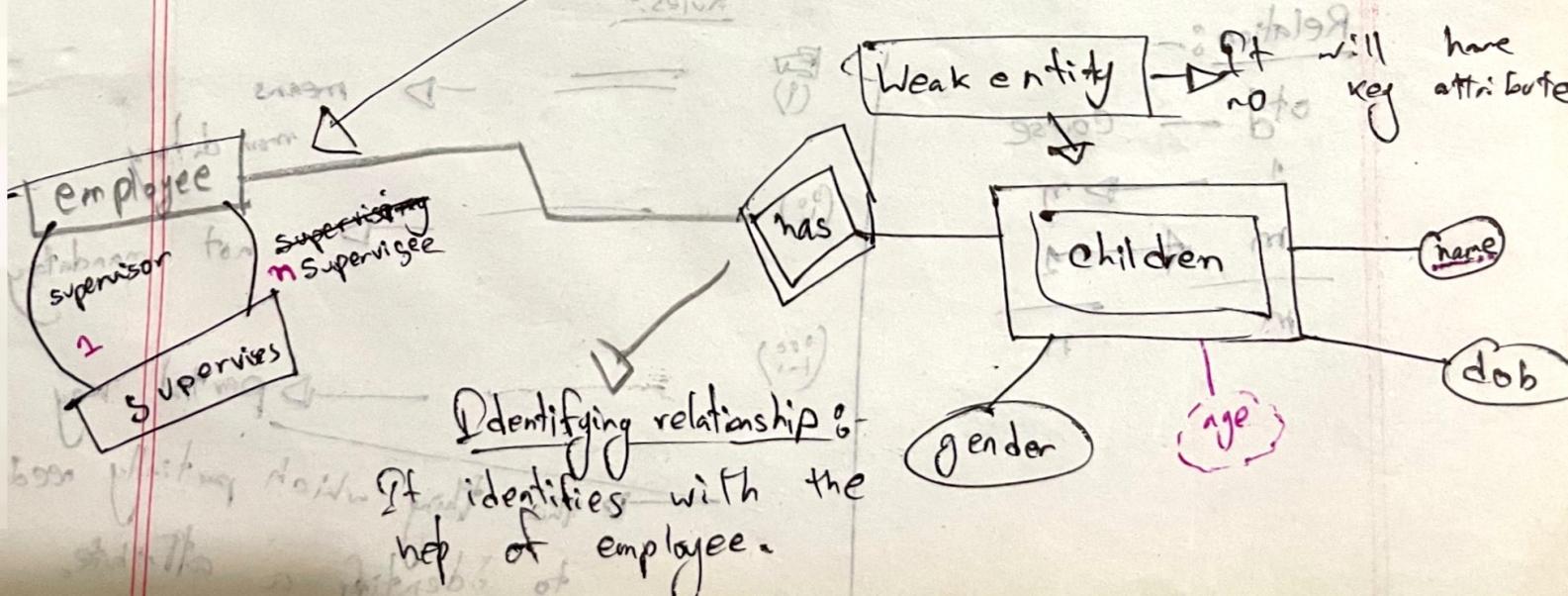
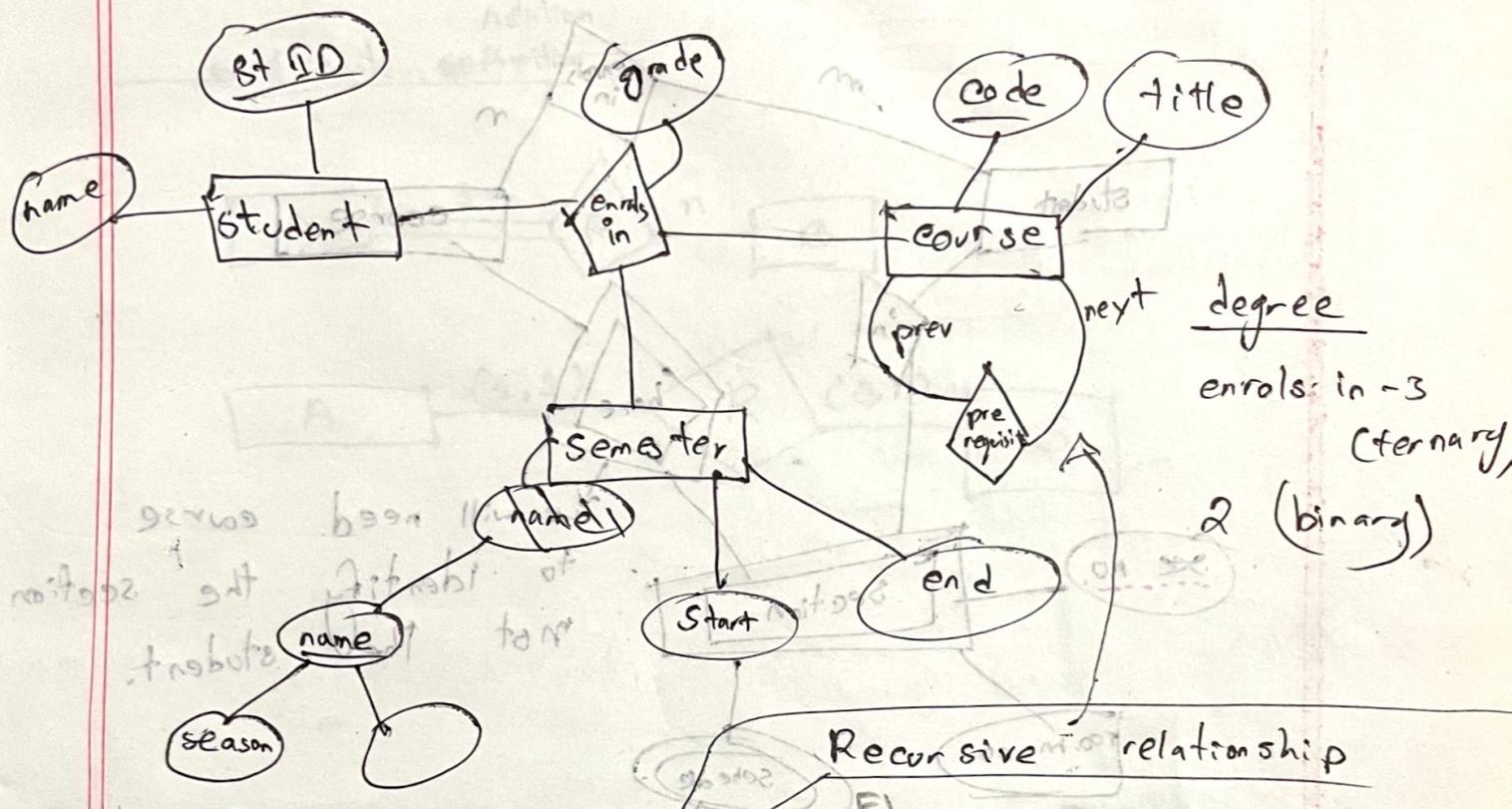
For certain entities we have to use both entity and relationship attributes to identify the entity.

→ How to understand a recursive relationship

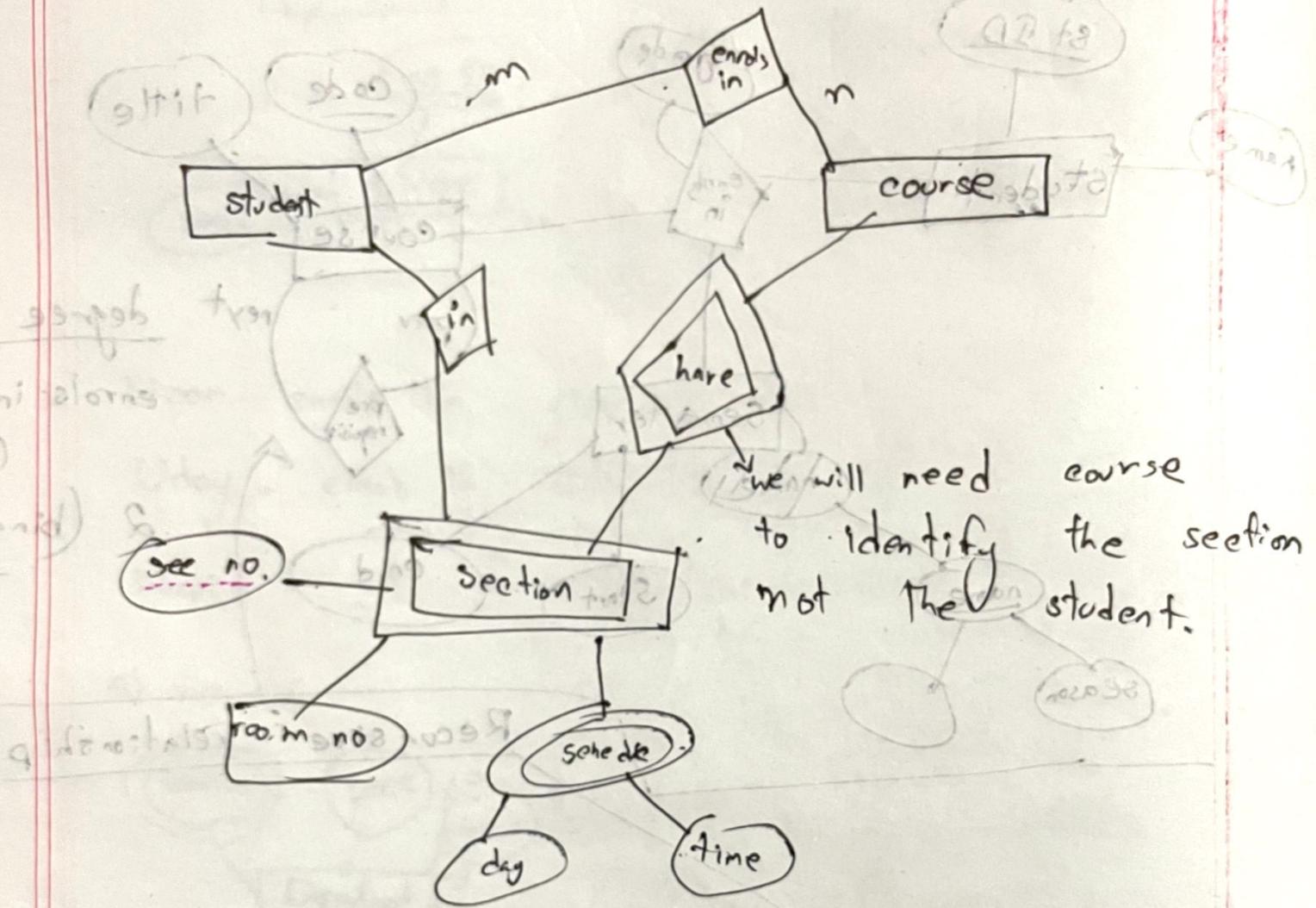
1 D → n

2 D → 2
2 → n

How many entity in a relation have they are degree.



we can part and consider a section from both



Rules:-

Relation :-

std — course

1 → n

m ← 1

m — n

→ means mandatory

not mandatory

partial key

the thing which partially needs to identify a attribute

studied the food chain

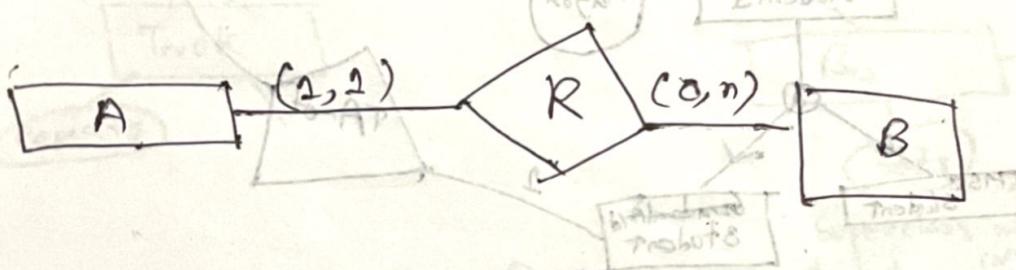
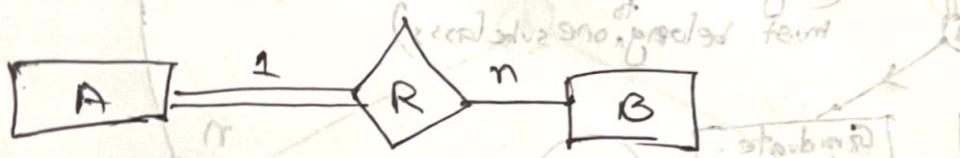
encoder

2 encoder & 2 encoder

If we can do we can derive the existing relationship.

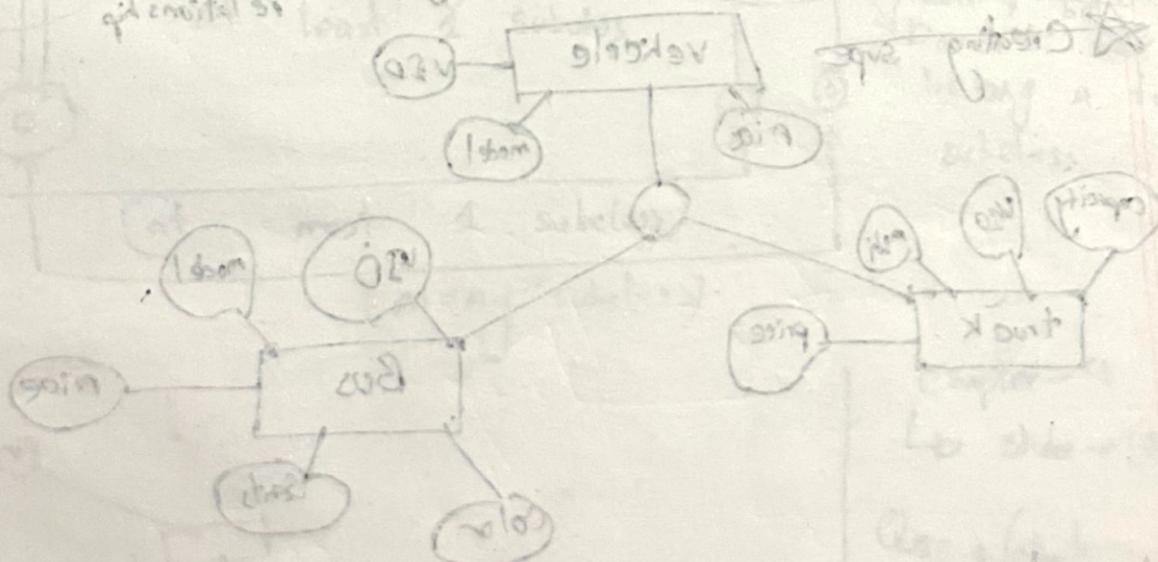
Alternate definition

notation



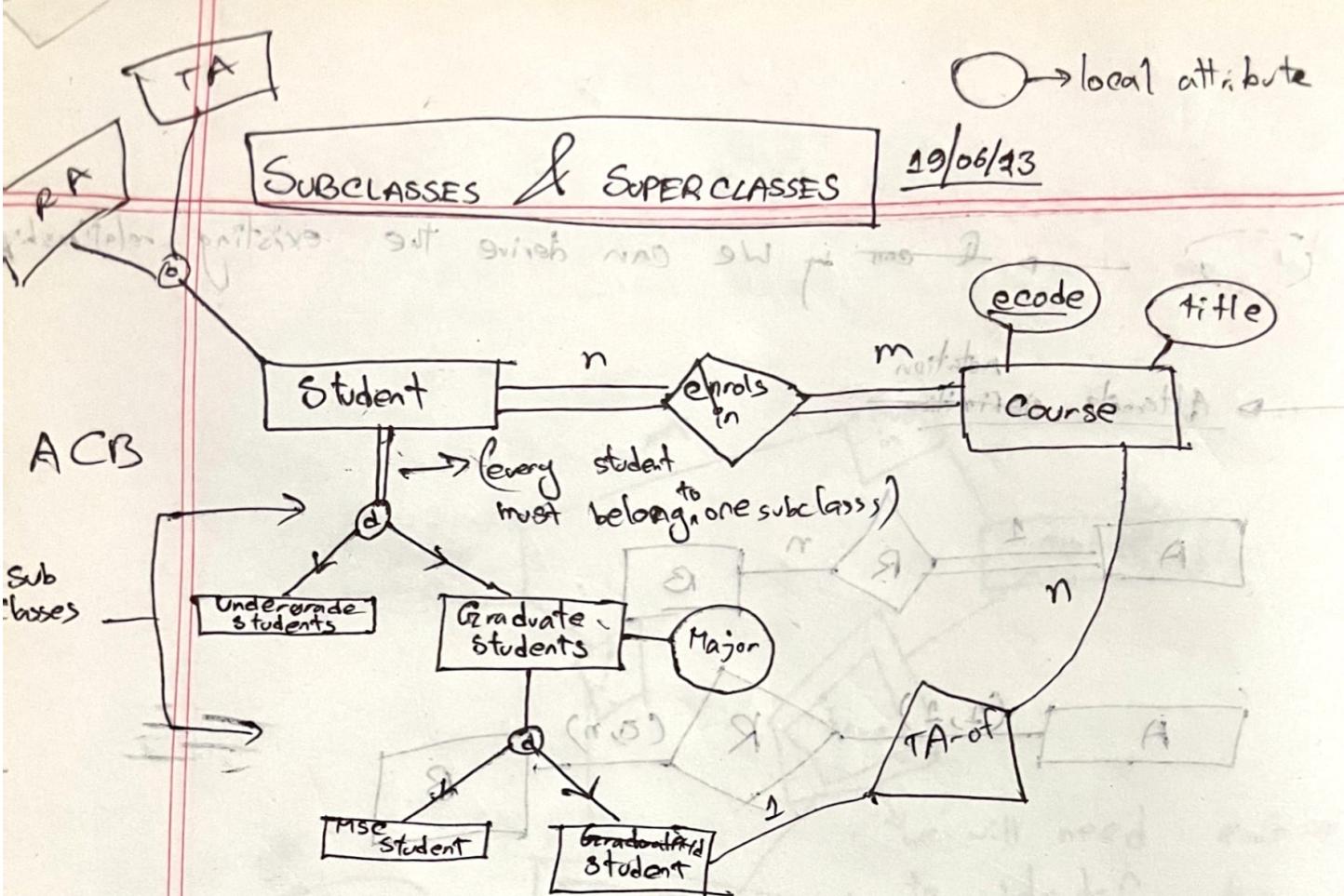
attribute items - caught out to two are correlated

1 dimensional



local attribute

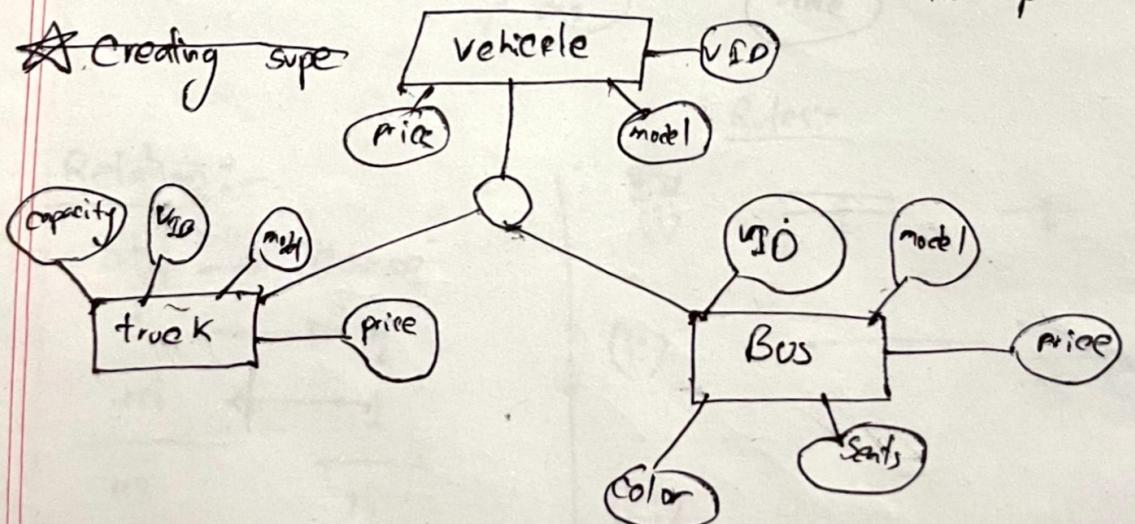
19/06/13



* Process of creating subclasses from superclasses is called specialization.

Subclasses are not of two types:- empty subclass relationship "

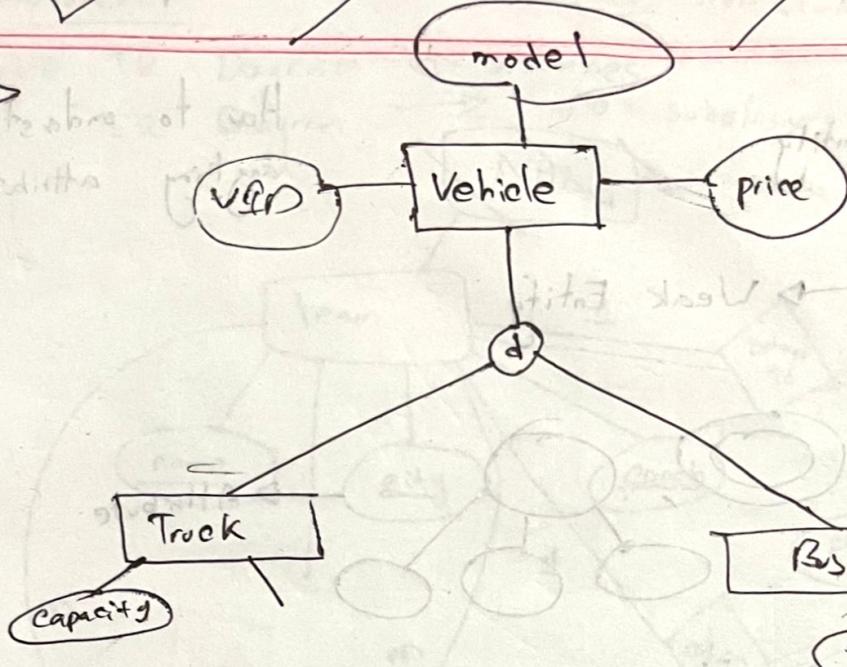
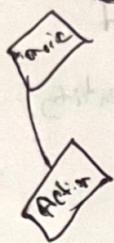
* Creating super



* Creating superclass from subclass is generalization.

data requirement
→ draw the draw
ER diagram

Design the
ER diagram



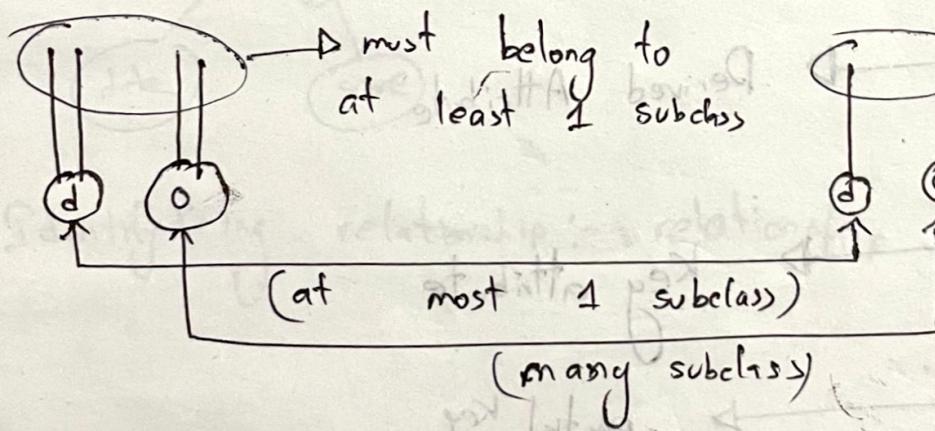
Overlapping: 1. super class
can belong to more
than one sub class

Each

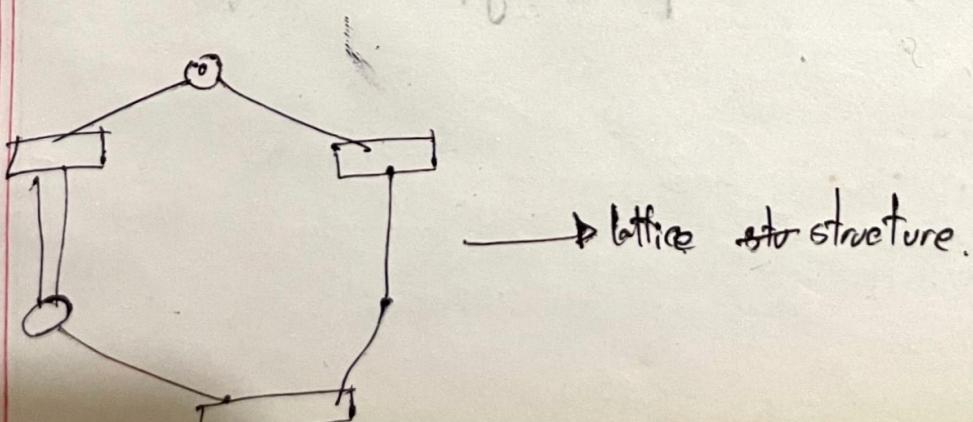
Disjoint constraint:

Superclass member
at most
one student can't
belong to more
than one subclass. → ②

overlapping: one student can
belong to more than one subclass. → ①



most belong
needn't be
Doesn't need to
belong to any
subclass

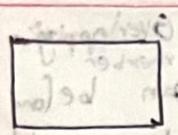


Chapter - 4
↳ slide → (4-26)
Quiz → (chapter - 3+4)

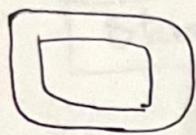
21/06/23

3 student
3 project
3 Lab
etc

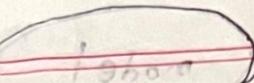
How to understand if anything attribute or entity



Entity



Weak Entity



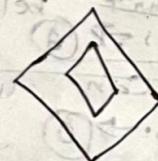
Attribute

Attribute

Attribute



relationship



identifying relationships.

attribute after relationship

Derived Attribute

Attribute

Key attribute

(candidate key)

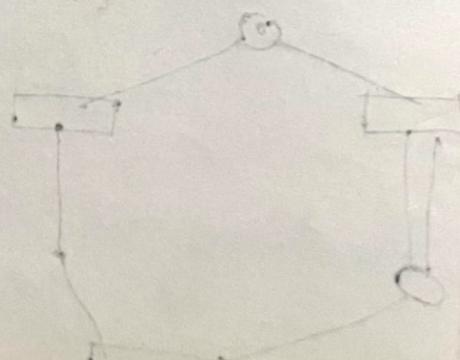
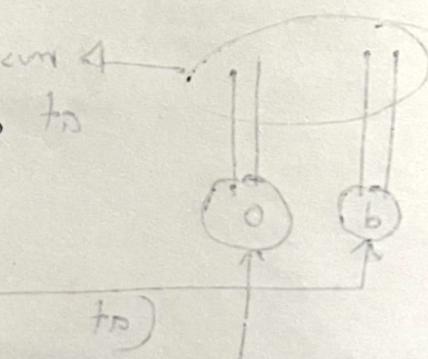
partial key

P - typed

(ss, p) while

(115 - 110) + 320

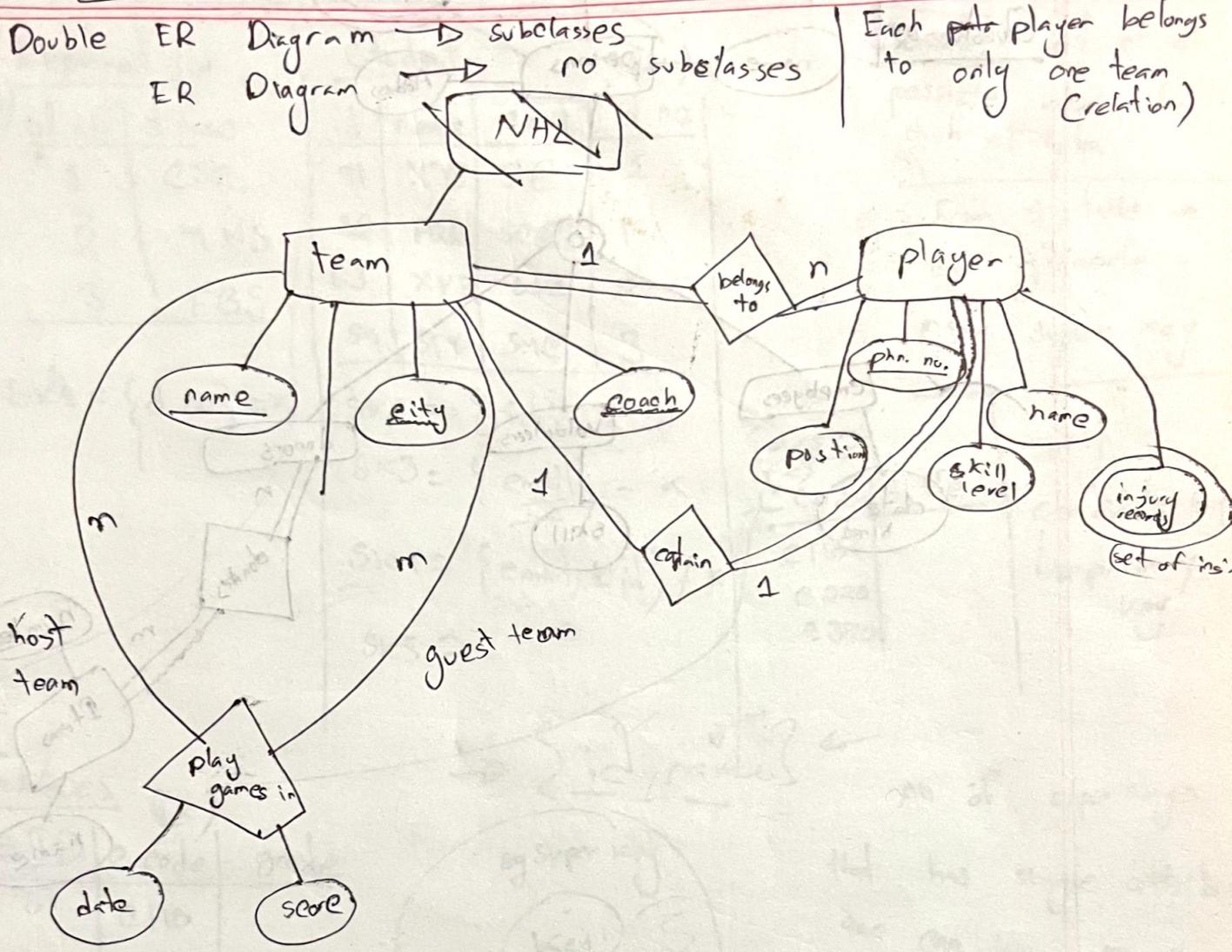
autonumber with



Property of something → Attribute

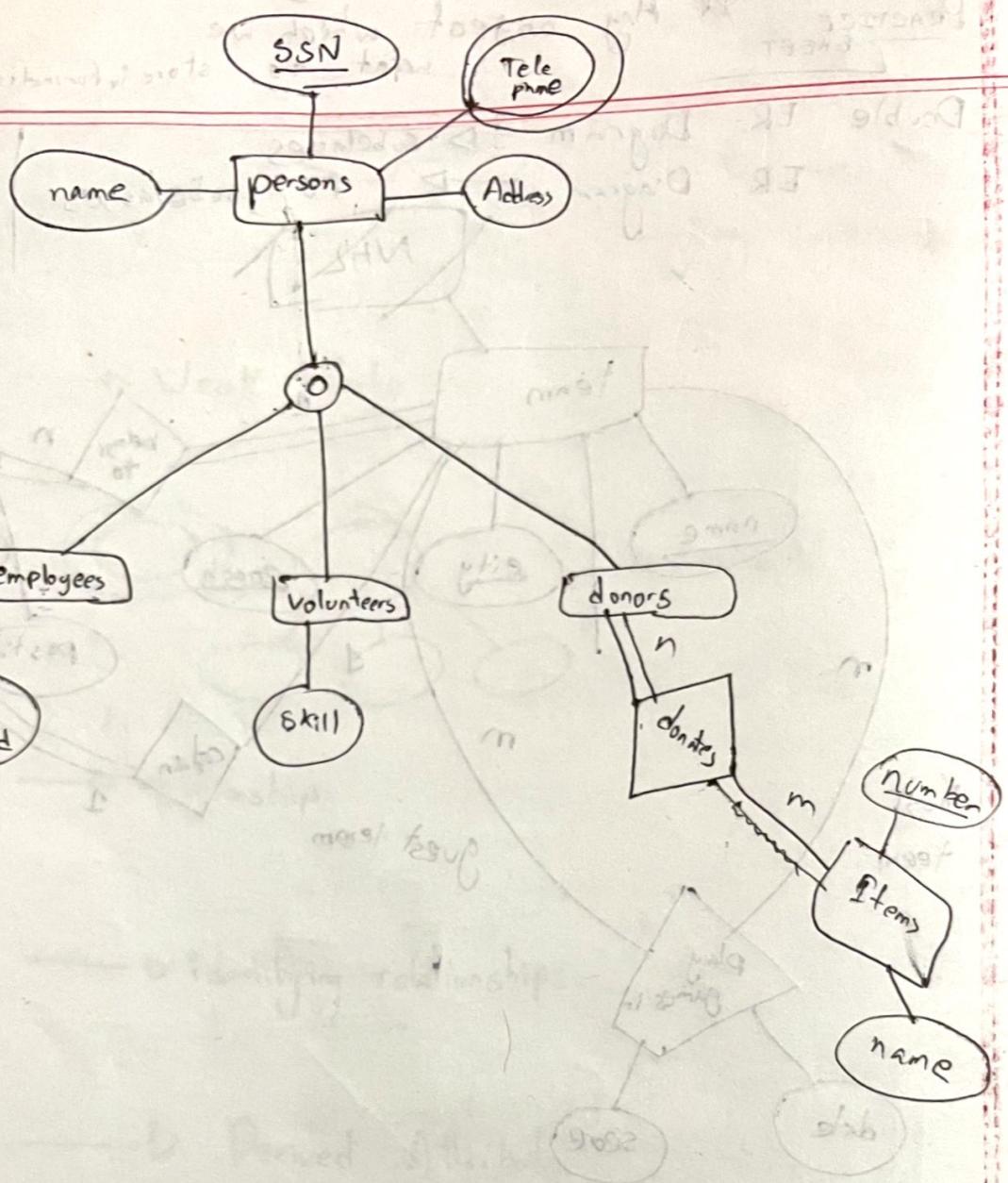
PRACTICE SHEET
Any object which we want to store information → Entity.

Question-1



Identifying relationship :- relationship between weak & strong f

Question-02



single attribute set or set of attributes which is unique is superkey

05/07/23

Relational Integrity Constraints

Department	
sl no.	d name
1	CSE
2	MNS
3	EBS

$$\text{SK1} = \{d \text{ no}\} = K$$

Student	
	(fk)
id	name
S1	XYZ
S2	PQR
S3	XYZ
S4	STY

$$\text{SK2} = \{id\} = K$$

$$\text{SK3} = \{\underline{\text{email}}\} = K$$

$$\text{SK4} = \{\text{email}, \underline{\text{id}}\} \neq K$$

$$\text{SK5} = \dots$$

Domain: set of all possible values for each attribute.

From 1 table we can generate many super key

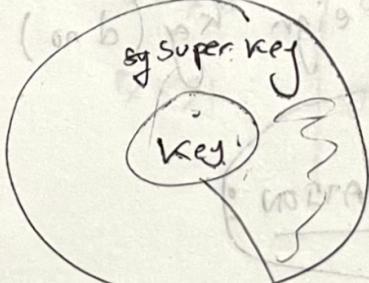
Courses	
	(fk)
c code	title
C110	
C220	
C370	

only 1 candidate key
↳ primary key

GRADES	
(fk) S ID	(fk) c code
S1	C110
S2	C220
S3	C370
S4	CSE10 CSE370

These has to be some values
there can't be null
There should be a primary key.

These two because
can be a key
and its not unnecessary



from the key we can make a super key
↳ both of fit

All of superkeys that has single attribute are can be a key & (but if every 1 one of them is necessary then it can be a key)

Foreign keys are usually the primary keys of another table.

Domain

any attribute

key, entity integrity

primary key
Entity

referential Integrity

foreign key

Create table student

id	char(2)	= 2x2
name	varchar(100)	= 2x2
email	varchar(100)	= 2x2
d no	int	

primary key(id)

foreign kg (d no) references Dept (dno)

DELETE OPERATION

(i) on delete restrict

(ii) on delete cascade

(iii) on delete set null

just null replaces with null.

allowed to delete row from the table

delete everything related to that row

UPDATE OPERATION

on update restrict / cascade / set null

	small	big	
small	A	B	
big	C	D	
INSERT →	✓	X	ON DELETE CASCADE ↳ DELETES ON BOTH TABLE
DELETE →	X	✓	ON DELETE SET NULL
UPDATE →	X	X	ON UPDATE CASCADE ↳ UPDATES ON BOTH TABLE

Dept-Code	Dept-name
10	QSE
20	EEE
81	MNS
52	BSS
62	ARC

Std-ID	Std-name	Dept-Code →
1234	Alice	10
4567	Bob	81
5678	Carol	52
7891	John	10

small	big	small	big
A	B	C	D
D	E	F	G
H	I	J	K
L	M	N	O
P	Q	R	S
V	W	X	Y

reg

reg-id	Name
1	A
2	B
3	C
4	D

logins

login-id	Name
1	X
2	A
3	Y
4	Z

Joins

innen

outer

left

right

#Select * from registration

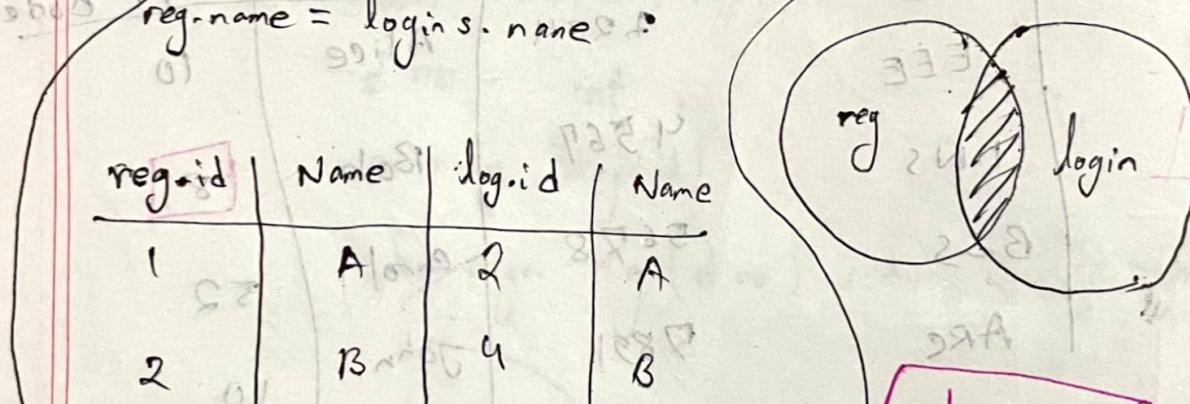
reg-name = logins.name

1st table

2nd table

join

logins on

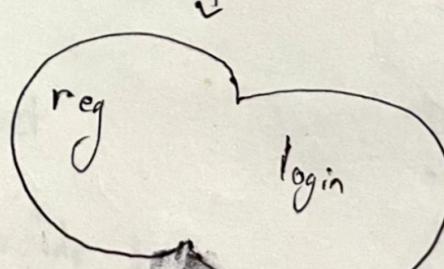


inner join

reg

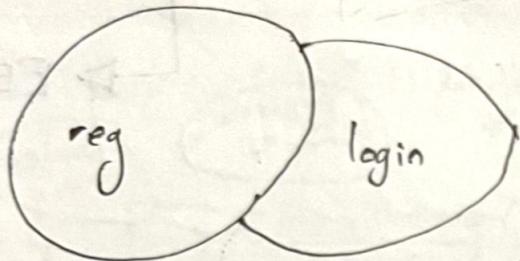
logins

outer join



reg-id	Name	login-id	name
1	A	2	A
2	B	4	B
3	C	null	null
4	D	null	null
null	null	1	X
null	null	3	Y

Select reg.id, Name, login.id from registration left join logins on reg.name = logins.name



reg

Select reg.id, Name, login.id from registration ~~right~~ inner join logins on reg.name = logins.name
C ~~inner~~ join on reg.login.id = logins.login.id

reg.id	Name
1	A
2	B
3	C
4	D

Pg-9 | (Chapter-7)

FOR EVERY ENTITY THERE WILL BE SEPERATE TABLE

Employee

(2) ↳	SSN	Bdate	Address	Sex	Salary	Fname	Minit	Lname	Dno
-------	-----	-------	---------	-----	--------	-------	-------	-------	-----

Project

Name	Number	Location
------	--------	----------

Department

Name	Number	mgrssn	Startdate
------	--------	--------	-----------

Dependent

SSN	Name	Sex	Bdate	Relationship
-----	------	-----	-------	--------------

bring
in
identifying
entities

Works-on :

Pro	SSN	hours
-----	-----	-------

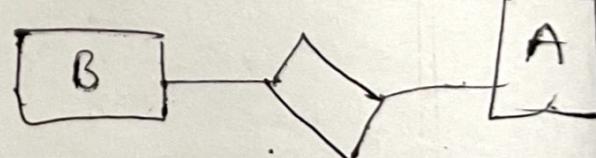
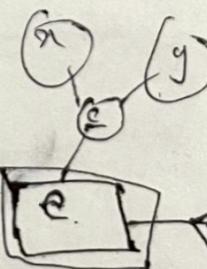
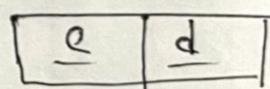
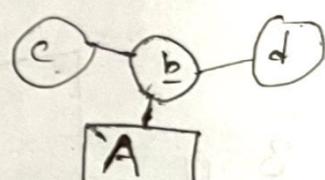
(3)

Choosing
the smaller
Primary
key

(3)

(1)

1



EE 9018

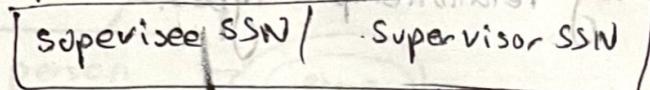
- ① Cross referencing
② FK approach

(P-std 15)

Many $(M \rightarrow n)$ relation for Supervision

↑
Supervisor

Supervision



SSN

1 to 1 (Manages)

free option

FK approach

Cross referencing approach

Both approach
(merge)

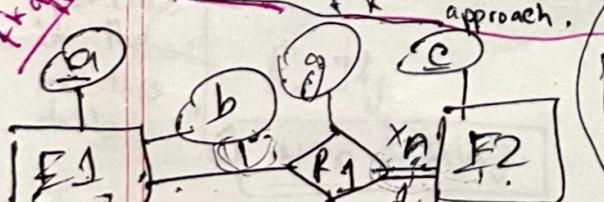
m to n

Cross referencing

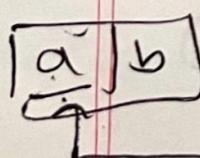
1 to m

Cross referencing

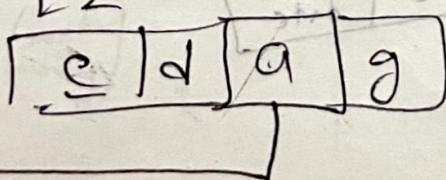
FK approach



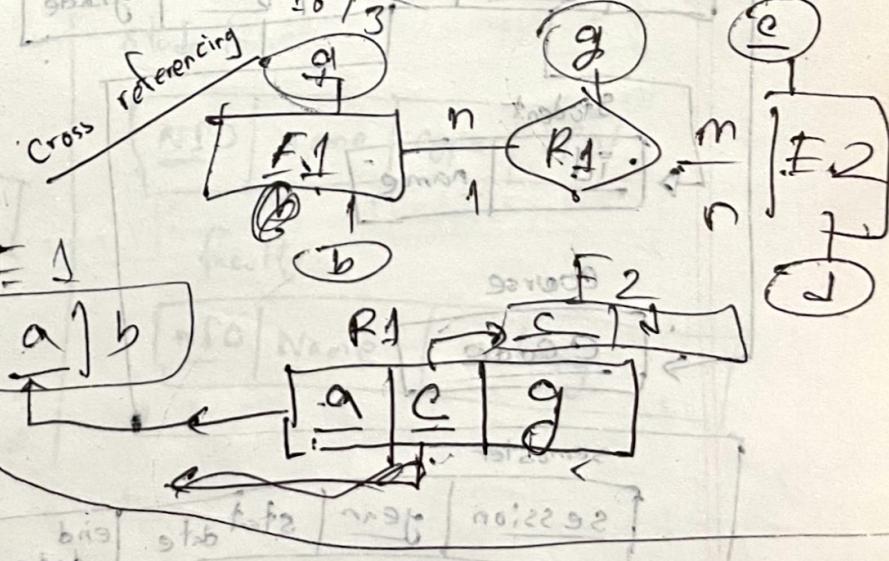
E1



E2



FK approach



31/07/23

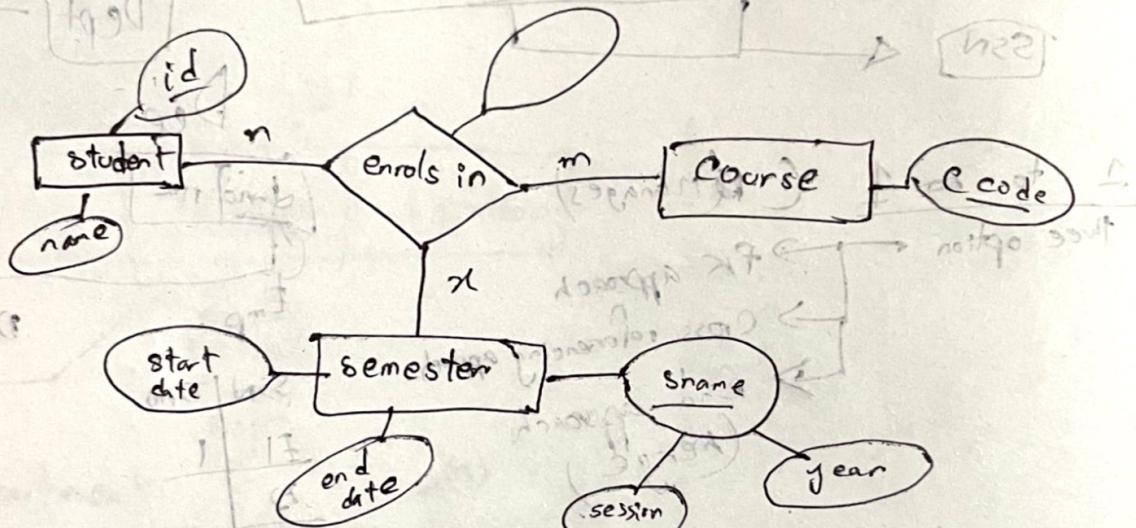
(Chapter - 7)

Mapping :-

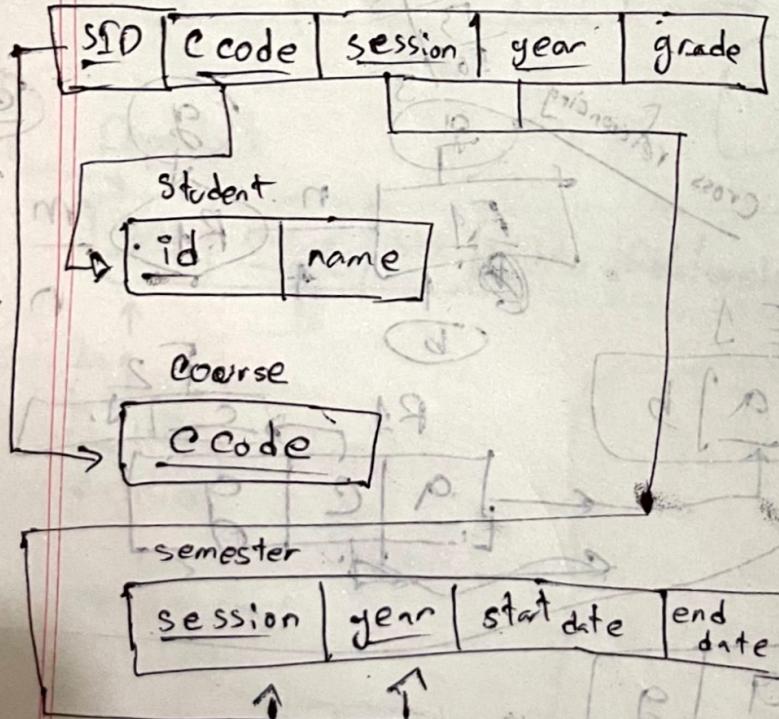
Step 7 :-

Mapping of

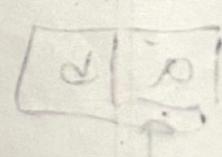
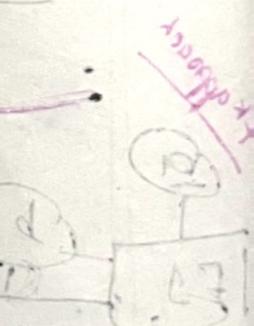
N-ary relationship types.



enrolls in

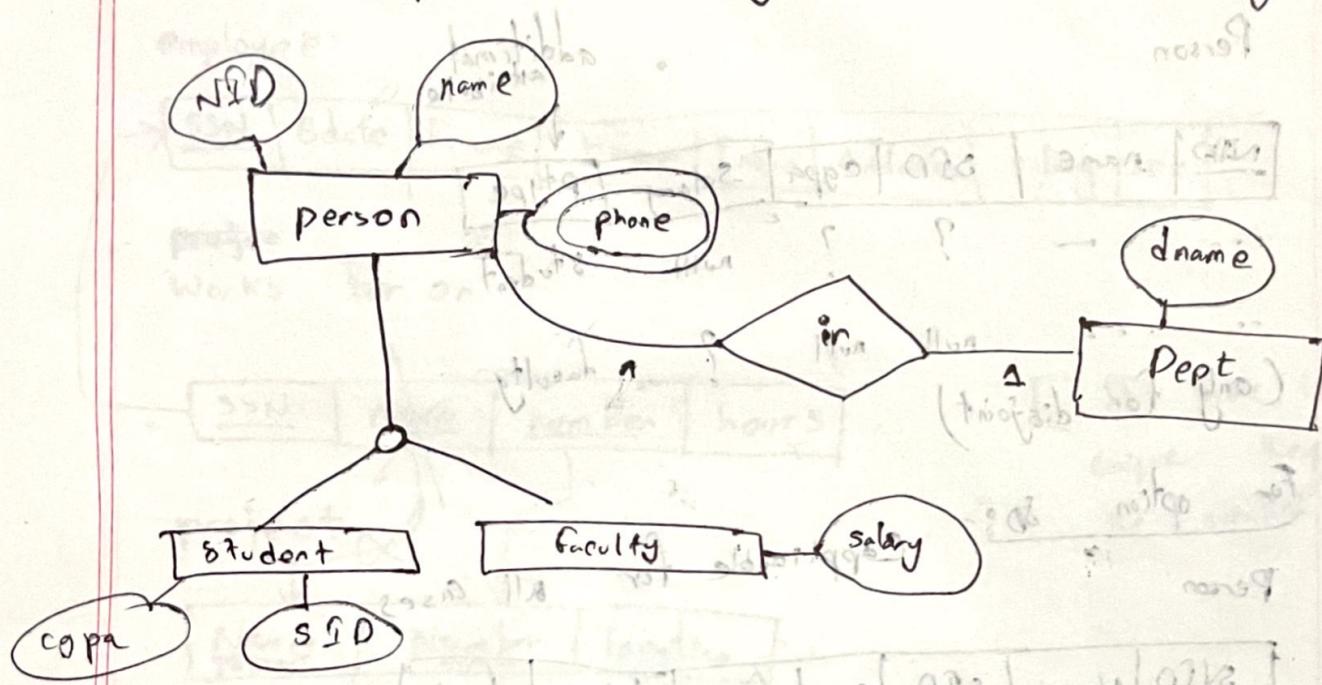


case of
In 1 fk approach
we will not understand



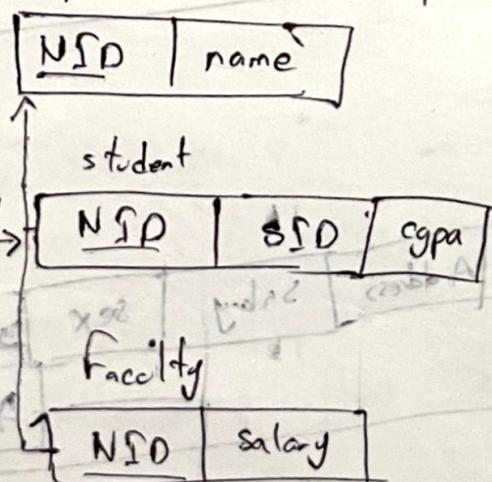
building for 8A & mapping it
 (problem part) ←
 time on listing of →
 . 98 96

Step-8 Options for mapping specialization or generalization.



(applicable
for all
cases)

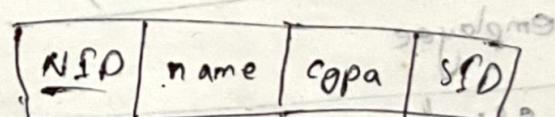
For 8A option :-
 person subclass &
 superclass



primary key
of the →
superclass
in the
subclass
table

For 8B option :-
 (only for subclass)

student not 1.) don't do



Ques -3 (next
monday)

fk approach → fk not underlined
 (not mandatory)
 → For partial we can't use 8B.

For option 8C in

program not existing 8-9578

Person

<u>NID</u>	name	<u>SSN</u>	legpa	Salary	Ptype	noeessq
------------	------	------------	-------	--------	-------	---------

additional attribute
 ↓

null

student

? ? null null

? faculty

(only for disjoint)

For option 8D :-

Applicable for all cases

Person

<u>SSN</u>	Name	<u>SSN</u>	legpa	faculty	student flag	faculty flag
------------	------	------------	-------	---------	--------------	--------------

PRACTICE

FK approach (1 to n)

employee

<u>SSN</u>	Bdate	Fname	Mname	Cname	Address	Salary	sex	name
------------	-------	-------	-------	-------	---------	--------	-----	-----------------

Department

<u>name</u>	<u>Number</u>	locations
-------------	---------------	----------------------

only 1 primary key will be chosen

<u>Dno</u>	<u>location</u>
------------	-----------------

Cross referencing approach (m to n)	insert into	join
employee	insert into	join

SSN	Bdate	Fname	Mname	Iname	Address	salary	sex
-----	-------	-------	-------	-------	---------	--------	-----

project

works for on

SSN	name	number	hours
-----	-----------------	--------	-------

project

Name	Number	Location
------	--------	----------

[If there is two unique key pick the smaller one]

02/08/23

Chapter-10

INFORMAL GUIDELINES :-

(not in) → exam die larva in chole
khate hai →
→ Burger sponge

Guideline-1

name	DOB	gender	marks	status	roll
xyz	12/12/1999	male	90	pass	422

Insertion Anomaly:-

name	DOB	marks	status	roll
abc	12/12/1999	90	pass	422

Deletion Anomaly:- When I am trying to deleting some thing and some data get also deleted and eventually having data loss.

Update Anomaly:- If I update in one row it won't get updated in another row.

Guideline-3:- We should avoid null value as much as possible.

Guideline-4:- After joining new data is generated and that data is incorrect

CLASS
PRACTICE

Services

ID	start-time	start-date	duration	price
----	------------	------------	----------	-------

Resume Interview Mentorship

ID

ID

ID

Customer

NSD	email	skills	strength	weakness	phone	name
-----	-------	--------	----------	----------	-------	------

Employee

emp ID	name	phone	email
--------	------	-------	-------

career advisor client Manager

emp ID

emp ID

Purchase

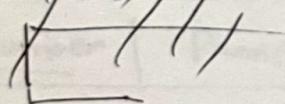
ID	NSD
----	-----

Industry

NSD	name	email	name	phone	expertise
-----	------	-------	------	-------	-----------

empl.

talks / Ap



Team

Tman	CIB	coach
------	-----	-------

Plays with

H Tname	C Tname	J	S C
---------	---------	---	-----

Chapter - 10 until slide 35

First Normal Form :-

Full dependency :-

{SSN; Pnumber} → hours

More than one partial dependency that then it is not in 2NF form.

2NF

Emp - Proj

SSN	Pnumber	Hours	Ename	Pname	Place
FD1					
FD2					
FD3					

2NF form

Employee

SSN	Ename

Project

Pnumber	Pname	Place

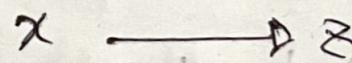
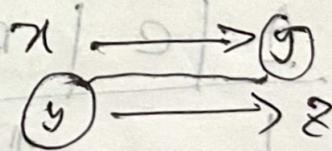
Emp - proj

SSN	Pnumber	Shares

Transitive dependency

x	y	z
---	---	---

prime attr.



transitive dependency

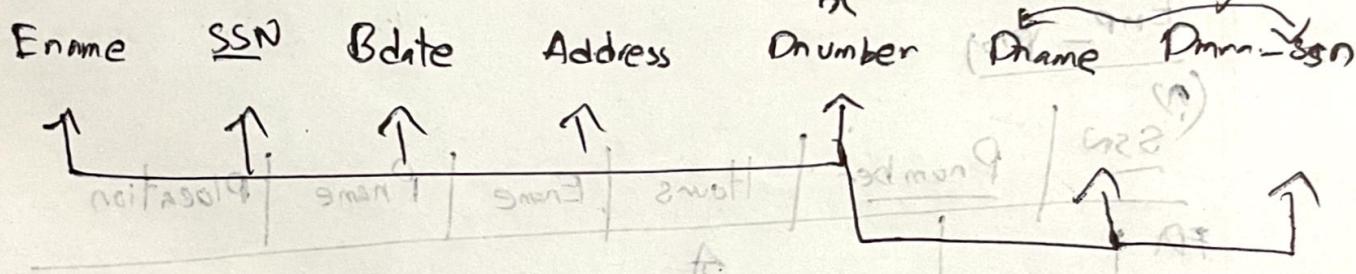
22 slide 11th, Q & A

Transitive dependency cannot exist if we have to remove it from 3NF.

we have to remove it from 3NF.

(b)

EMP - DEPT

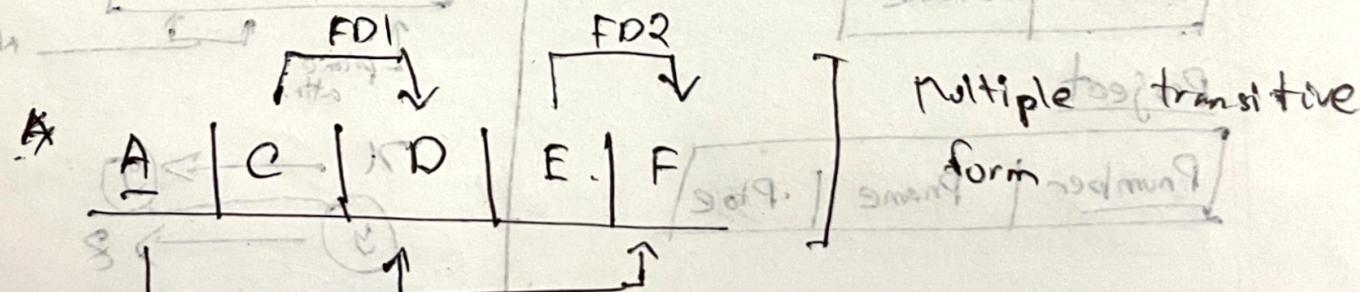


Emp

SSN	Ename	Bdate	Address	Dnumber	Dname
-----	-------	-------	---------	---------	-------

Dept

Dnumber	Dname	Dmgr-SSN
3	6	



A	C	E
3	4	5

C	D
2	3

E	F
1	2

G	H	I
1	2	3

1 NF → There should not be any :-

- (i) Multivalued attribute
- (ii) Composite attribute
- (iii) nested table

2NF → There :-

- (i) Cannot have partial dependency

1st cond.

2nd cond.

3NF

→ There will no

- (i) Transitive dependency

THE
color and
other describ

color and
other describ

other describ

Question - 1 - (Chapter 10 prac sheet)

2NF

Cars

Car ID	Car Model	Car year	Price	Co. & company
--------	-----------	----------	-------	---------------

Salesman

SID	SM Name	Com percentage
-----	---------	----------------

Car ID	smID	disc	date sold
x	z	y	

3NF

Car sales

<u>Car ID</u>	<u>sm ID</u>	date sold
		start between (ii)

disc info

<u>date sold</u>	<u>disc</u>	listing start formed (i)
		start between (ii)

disc
on. disc start → disc
positive negative (i)

(books long (i) - no book) → books

books

books

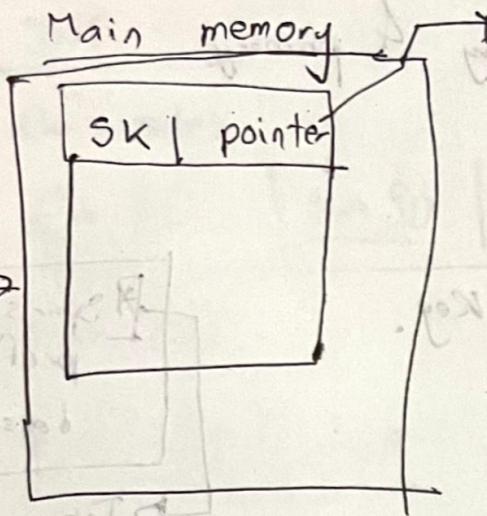
books	long	no book	books	books
-------	------	---------	-------	-------

books

books	long	no book	books
-------	------	---------	-------

books	long	no book	books
b	l	n	b

EXAMPLE B⁺-Tree

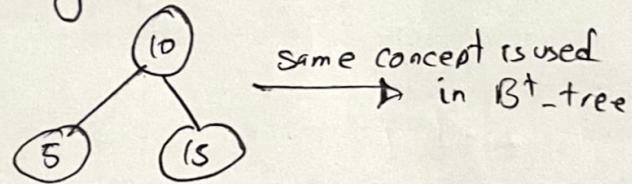


Spare
dense

we create index to speed up the search, delete, create.

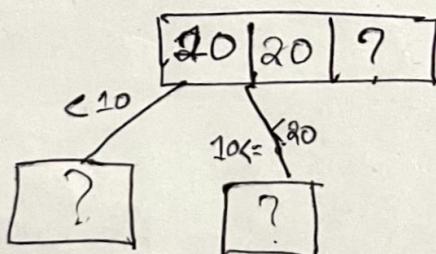
DBMS uses B⁺-Tree (common data structure for DBMS → efficient way)

Similar to
Binary Search Tree



B⁺-tree

Always Dense index
never sparse



n = 9, 6 ...

Order=n	Non root node	root node
values	$\frac{min}{n-1}$ to $\frac{max}{n-1}$	1 to n
children	$\frac{n}{2}$ to n	0 to n

Updates on B⁺-Trees:

- (i) Delete, ✓ each syllabus
- (ii) Insert,
- (iii) modify

EXAMPLE

Order, n=3



Insertion sequence

8, 5, 1, 7, 3, 12, 9, 6

[8]

[5 | 8]
1 5 8

Why split?

→ because node value is filled up

→ we will split on the basis of middle of value

[5 | 8]
5 3 8

[1]

[1 | 2]

[2 | 3]

[5]

[7 | 8]
7 8 12

(i) copy
(ii) move up

[8 | 9 | 10]
8 9 10

[10 | 11 | 12]
10 11 12

[7]

[5]

[1 | 3]

[5 | 6]

[7]

[8]

[9 | 12]

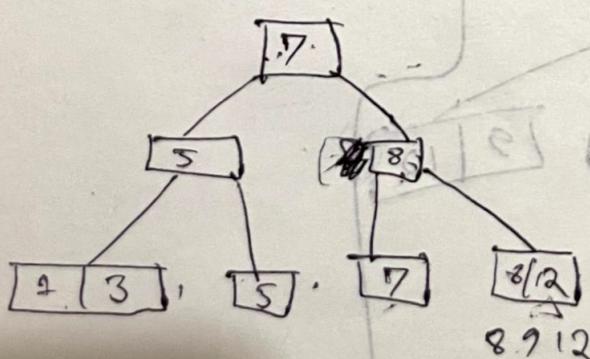
[1 | 2]

[3]

[4 | 5]

[6 | 7 | 8]
6 7 8

[9 | 10 | 11]
9 10 11

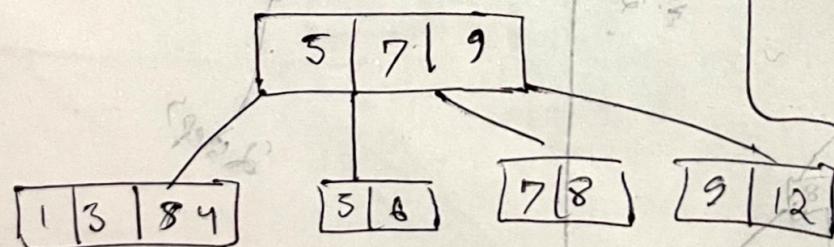
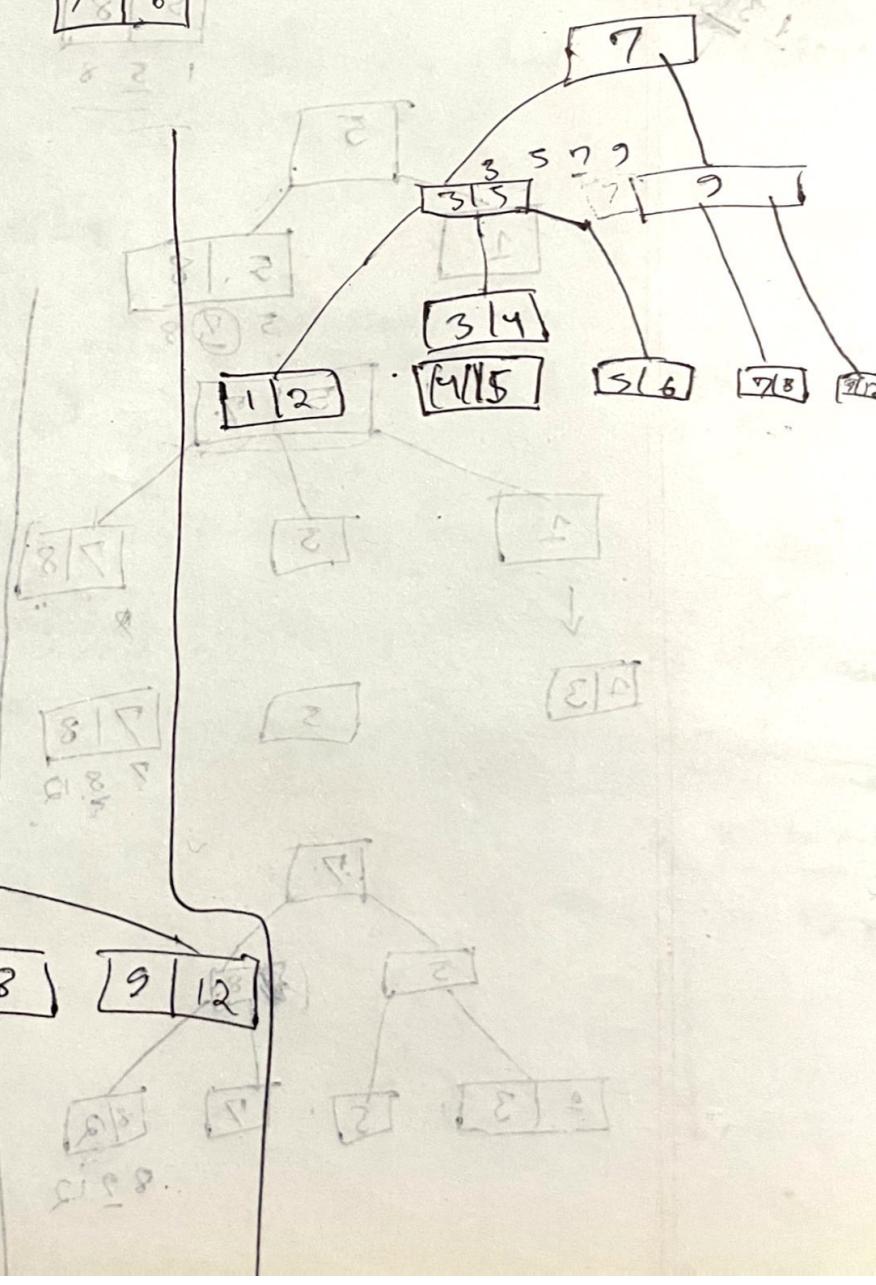
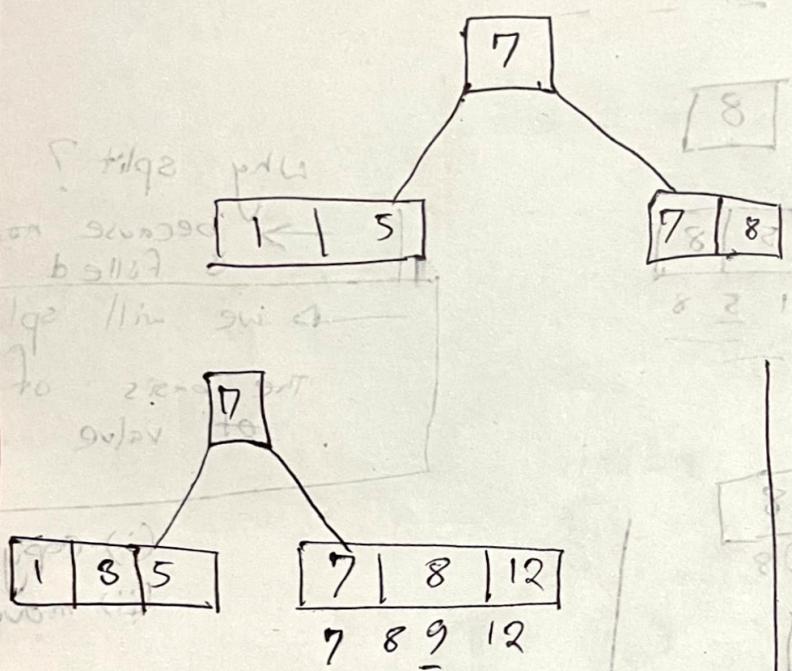


8, 5, 1, 7, 3, 12, 9, 16

Order n = 4

4, 2
newly added

1	5	8
1	5	7 8



1 3 2 3 8 4

HASHING

28/08/25

$h(id) \rightarrow$ Function

(i) Sorting

(ii) Bt tree

(iii) Hashing

(1 slot or table)

EXAMINE

Indexing slot (file) 2 slots 2 bits
bits applied on file organization.

hash index
Applied on File

How hash index works?

b



o

[50]

[30]

[60]

i

[51]

[71]

2

[32]

hash

$h = id \% 5$

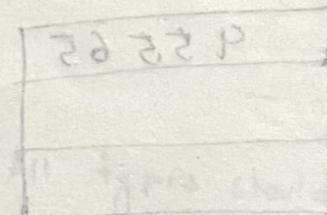
50

51

32

71

(forward/overflow)



22) 8 of 8

QUESTION

EXAMPLE

(Side no. - 51)

bucket ind = 2

$n = \text{first } \% 5$
digit

s bucket will

will take

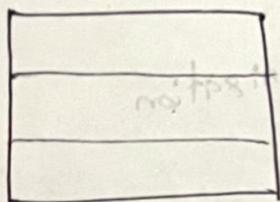
and

put place

in index 3

in every bucket

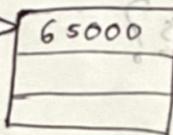
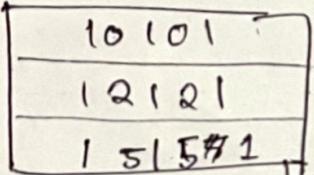
0



65000 $\% 5$

$6 \% 5 = 1$

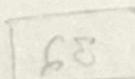
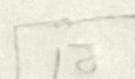
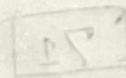
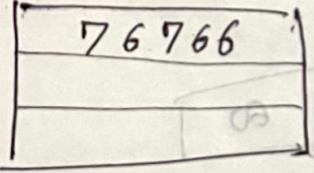
1



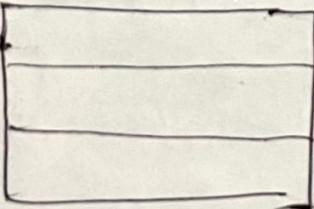
64565

$6 \% 5 = 1$

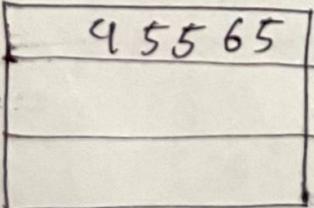
2



3



4



Whenever I want to join there must have primary K. and foreign K. relationship.

Team name
city
Captain Name

Select T.name, T.city, P.name from
Teams T inner join Players P
on T.Captain-phone = P.phone

Team name
city
Captain name
injury records
 $(A+B)+C$
 $(A+B)+C$

Select T.name, T.city, P.name
from (Teams T inner join Players
P) on T.Captain-phone = P.phone;
inner join player-injury records PR
on P.phone = PR.phone

Final

Time: 1. 45 - 2 hrs

marks: $(40 - 45) \rightarrow 30$

Q & Q (All)

1. ER / ERR \rightarrow Schema (All types should be memorized)
2. Normalization
3. SQL
4. Indexing & hashing

Answer to the q. no.-2

Given,

$$h = (\text{sum of all digits in id}) \% 7$$

Here,

BUCKET 0

12121
33465

BUCKET 1

83821
98345
32343 58583

BUCKET 2

BUCKET 3

201021
22222

BUCKET 4

76766
7155615
76543

BUCKET 5

45565

BUCKET 6

15151

$$(i) h(76766) : \quad (xi) h(22222)$$

$$32 \% 7 = 4 \quad 16 \% 7 = 3$$

$$(ii) h(10101) : \quad (xii) h(33465)$$

$$3 \% 7 = 3 \quad 21 \% 7 = 0$$

$$(iii) h(45565) :$$

$$25 \% 7 = 4$$

$$(iv) h(83821) :$$

$$22 \% 7 = 1$$

$$(v) h(98345) :$$

$$29 \% 7 = 1$$

$$(vi) h(12121) :$$

$$7 \% 7 = 0$$

$$(vii) h(76543) :$$

$$25 \% 7 = 4$$

$$(viii) h(32343) :$$

$$15 \% 7 = 1$$

$$(ix) h(58583) :$$

$$29 \% 7 = 1$$

$$(x) h(15151) :$$

$$13 \% 7 = 6$$