9521279429

Today's Agenda :-

Req Gathering ✓

Class Diagram ✓

- → Schema Design.
- → Coding the models
- → Coding the controllers
  - ↳ DTO (Data transfer objects)

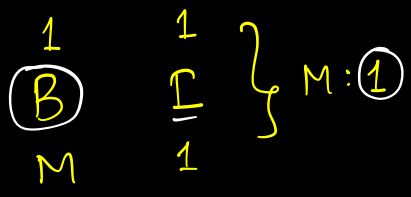
Schema Design

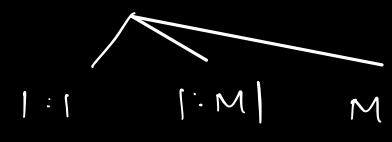
1) For all the classes in the Class Diagram, create a table for them.

2) For all the primitive attr, represent them as columns in the corresponding table.

3) For all the non primitive attr, find relationships (→ Association) / cardinalities.

Based on the cardinality we will go for inheritance or Association





(id of 1 side on any other)

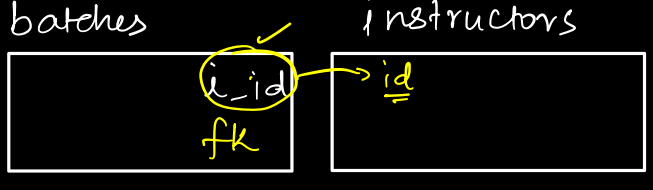
(id of 1 on m side)

M:1

(Mapping table)

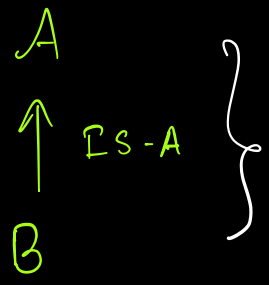
} representing fk constraints.

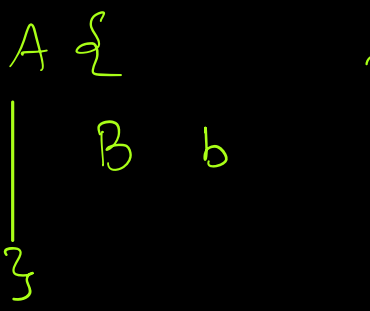
Batch {
Instructor i
}

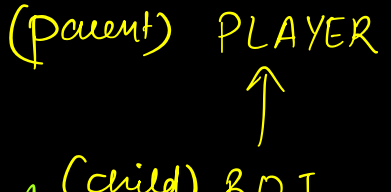


There can exist 2 types of relationships b/w classes

- Inheritance (IS-A) (parent - child rel^{n})
- Association (HAS-A)
  - Composition
  - Aggregation



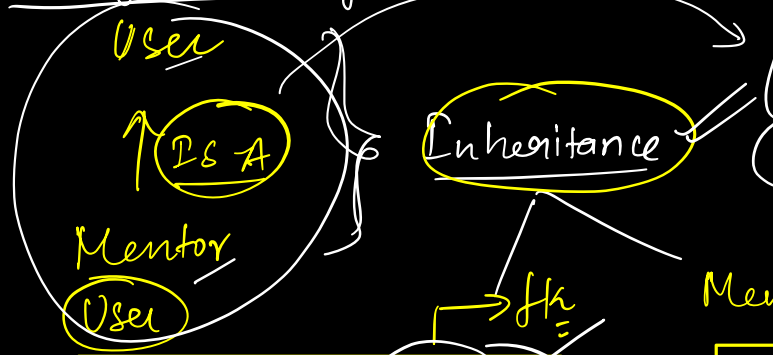


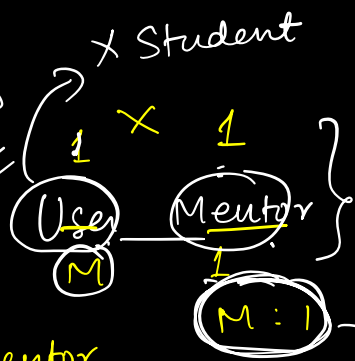


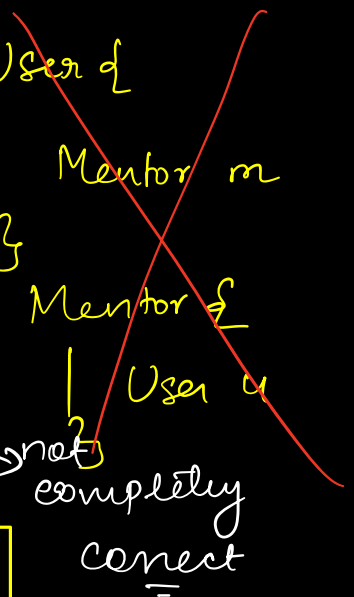
kisi class m kisi class ka object ho

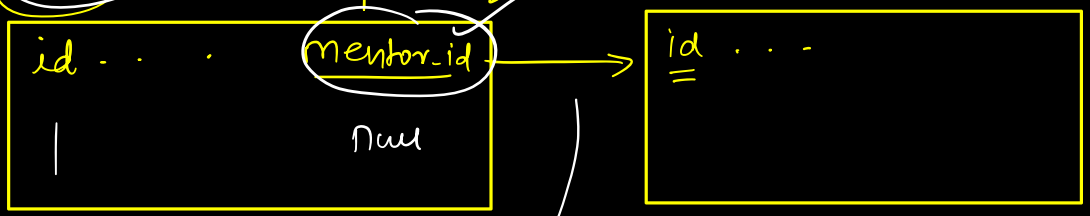
Representation of Inheritance is also done through fk most of the times

1 of the ways :-









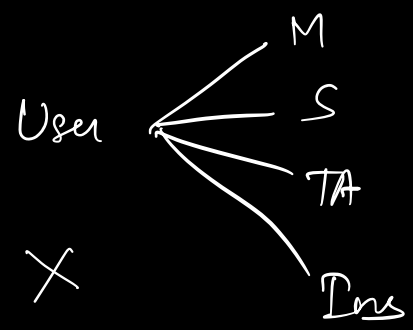
Uniquely identifies a row in the other table.

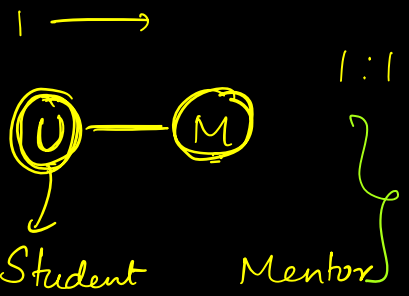
multiple users can have a mentor but 1 user can not have multiple mentors and its not necessary that every user will have mentor, he may be student or mentor itself.
so cardinality is M:1 and for some mentor_id it can be null.

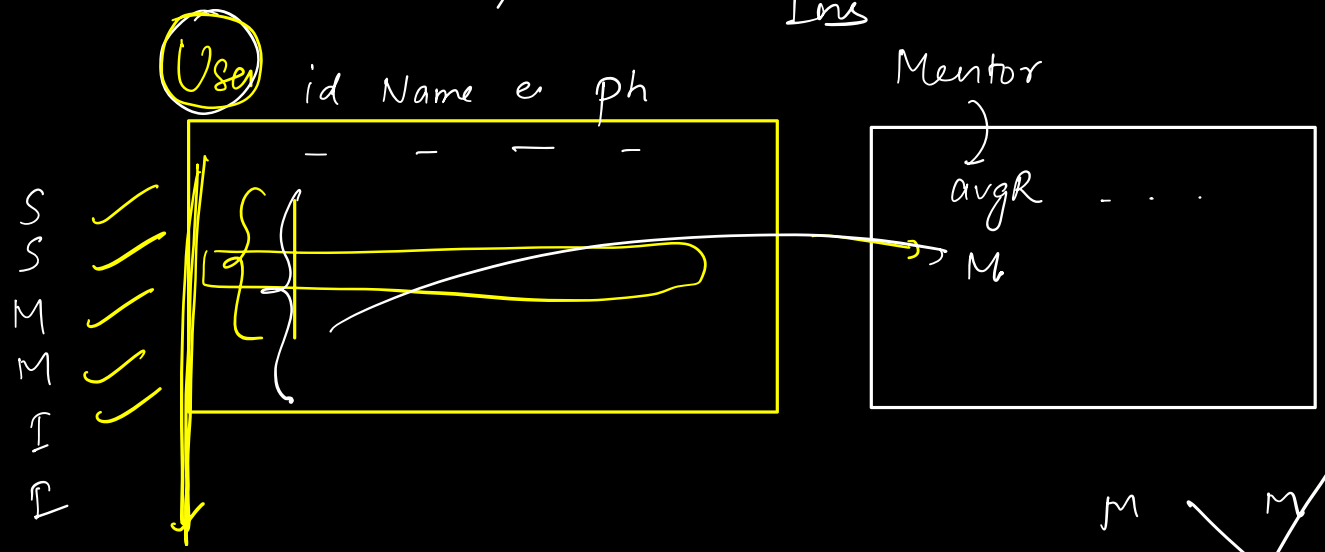
Student Mentor

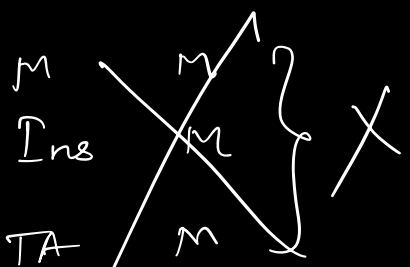
M : 1

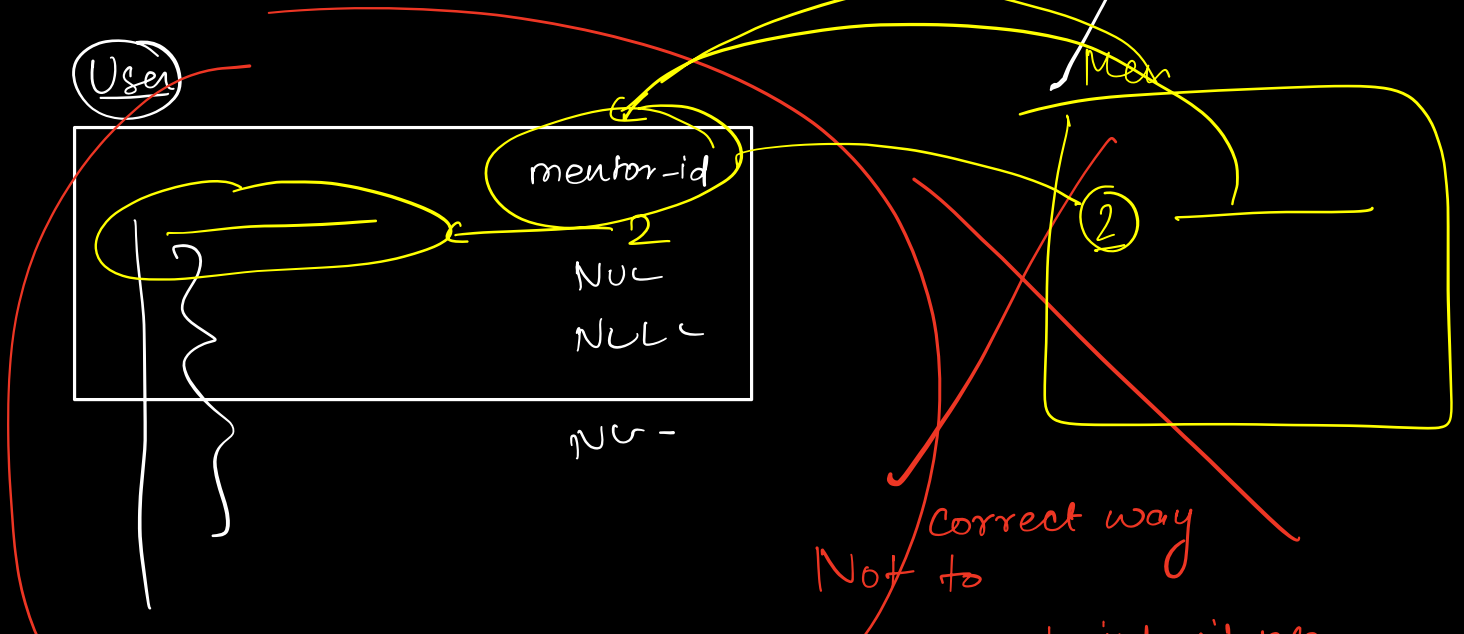
{{{ fk constraints



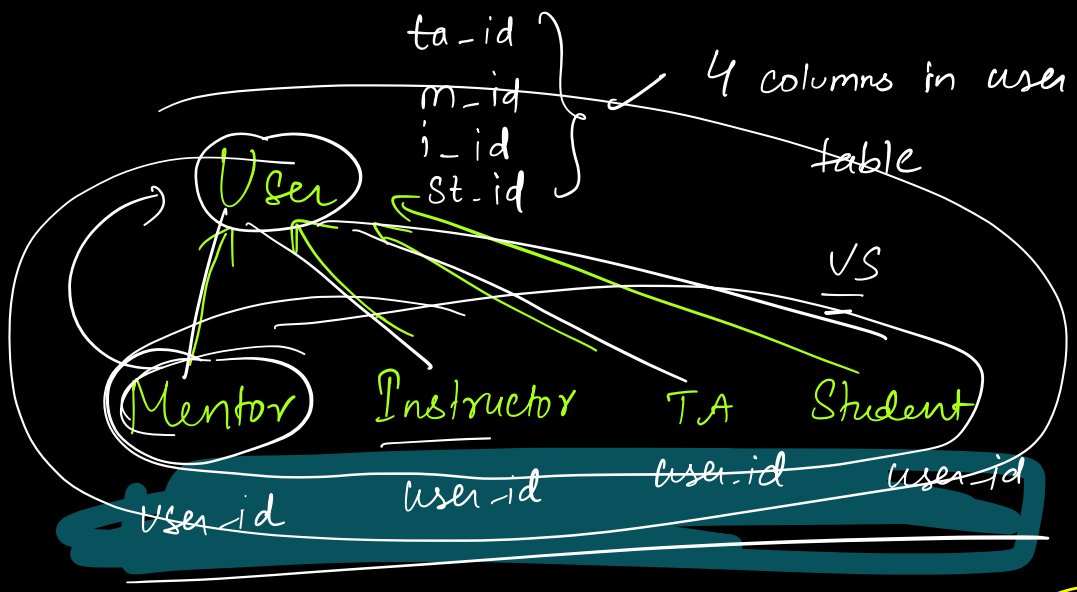








add the ref of parent in the child



Adding parent id in the child table is correct way

Normalization

Inheritance

Design

User

id	Name	email	pw	addres
21	Vmang			

Mentor

id	avgRating	s/p	expertise	user_id
2				

Instructor

id	avgR	Salary	user_id
2	4.9	10 ₹	21

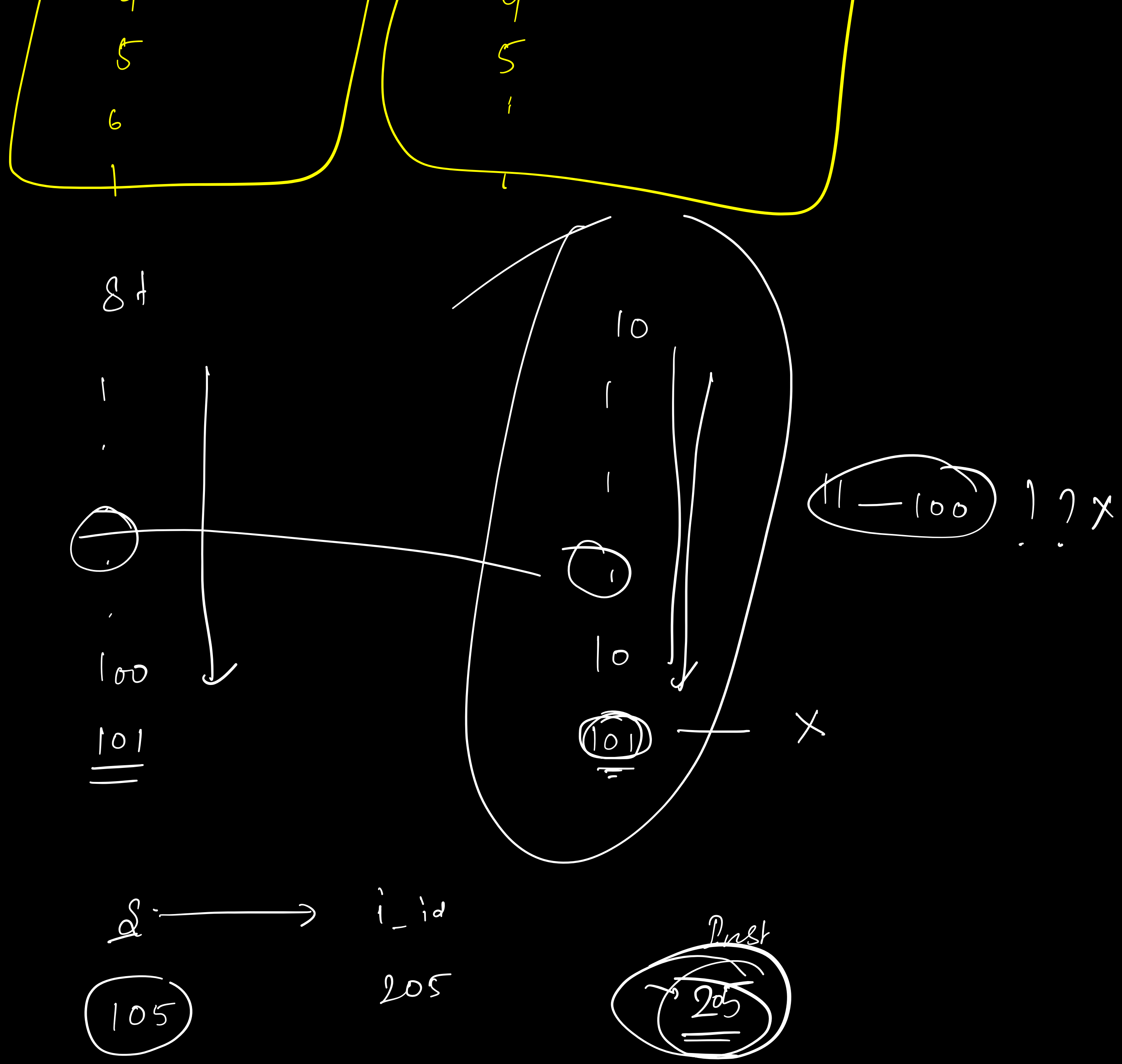
St

id	psp	. . .	user_id

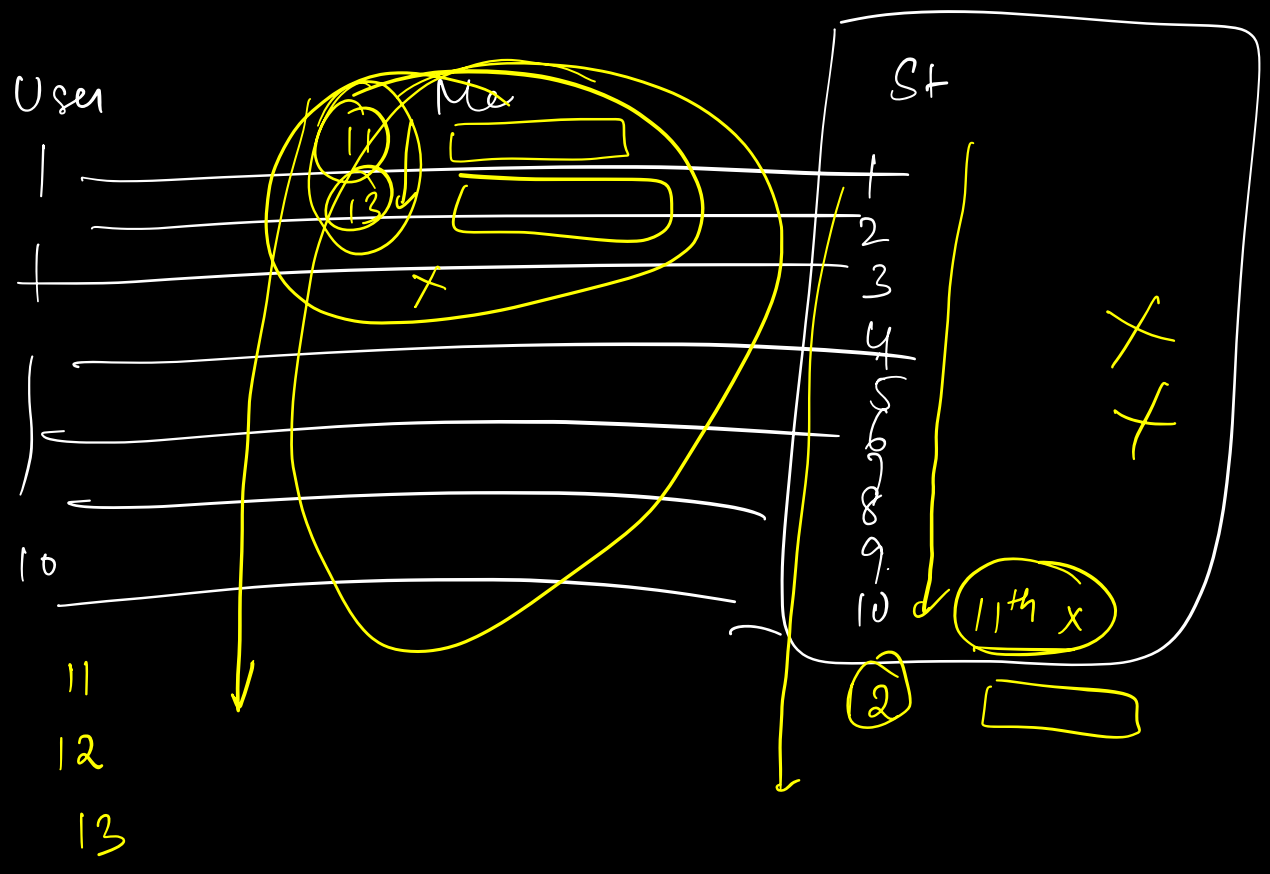
M S

1
2
3

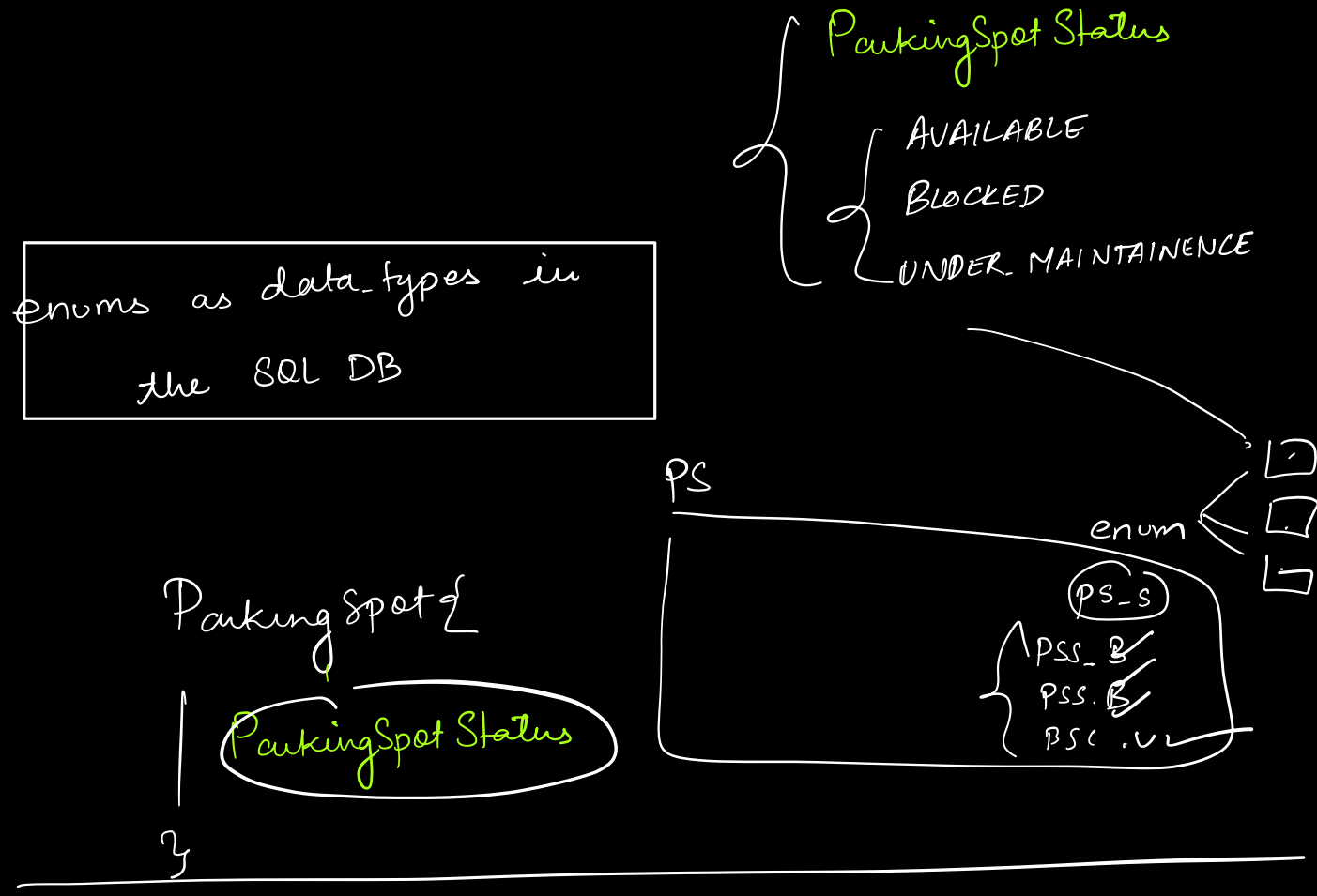
2
3



5
6
5
st
1
100
101
10
1
1
10
101
11 — 100
x
x
s → i_id
105
205
205

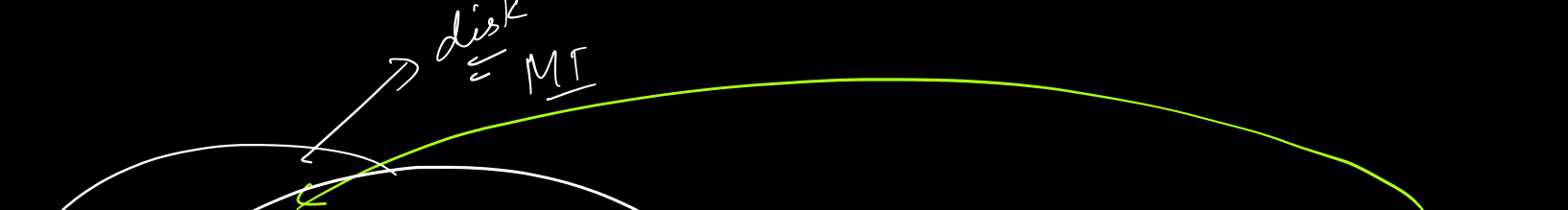

User
Ma
St
1
2
3
4
5
6
7
8
9
10
11
12
13
x
x
x
11th x
1
2
3
2

Representing enums in tables



2nd (Suggested)

1) For every enum, create a separate table
2) Every table should have 2 columns - (id, value)
3) In the tables, you can represent enum_id as fk.



ParkingSpotStatus

id	value
1	BOOKED
2	V_M
3	BLOCKED

String / varchar

ParkingSpot

int (PK)

PSS

1 2 39

2

int

Models ⇒ objects

Class
||
Blueprint of entity
||
real world object

class BaseModel {

id
createdAt
lastModifiedAt

}

Audit / logging

konsa object kb bna kha update hua sari info.. so it will help in logging purpose

extend BaseModel