

INTRODUCTION TO RELATIONAL MODEL

* Agenda:-

- ① What is Database?
- ① What is DBMS?
- ① Different types of databases:-
 - ↳ Relational Databases
 - ↳ Non-Relational Databases
- ① Relational Database overview
- ① Keys
 - ↳ Super Key
 - ↳ candidate key

* Setup of Machine:-

- ① MySQL 8
- ① MySQL workbench
- ① Sakilla Database (optional)

(* What is Database?)

→ (Excel, Notes, Notion, Google Docs)

Students

↳ name

↳ email

↳ PSP

↳ Ph No

↳ current company

↳ current Role

↳ Grad Year

Contests

→ leaderboard

→ problems

Students.csv

⁰ Name, ¹ batch, PSP, attendance, ...

Meenakshi, 1, 80, 100

Vishal, 2, 100, 40

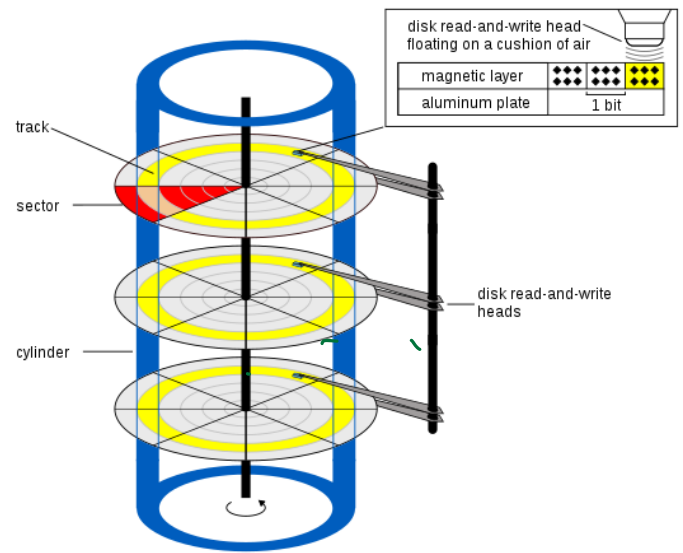
→ [ABC 4 full 20]

→ [AB, C, 2, 80, 20]

0 1 2 3 4

"C"

1. Read file
2. Line by line check
3. Parse (POP)
4. Return.



* CONS :-

① Insufficient $\rightarrow O(N)$
(super slow!)

② Security :-

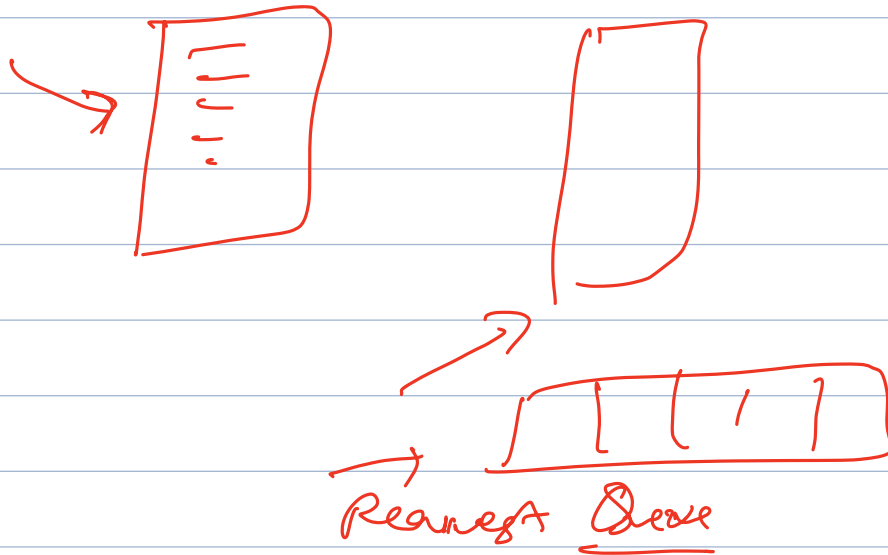
- ① No access control.
- ① Data is Readable.

③ Data Integrity :-

- ① wrong data type
- ① wrong for matching

④ Concurrency Issues :-

2 people login at
same time



⊗ What is Database?

AIRBASE

Military Base

Database

[It's a collection of interrelated
data.] (based on a business use case)

(*)

What is DBMS?

(Database Management System)



It's a piece of software that allows efficient ways to store and retrieve data, along with other necessary features such as :-

① Data Integrity

① concurrency

① security

① Access control

(*) Types of Databases:-

(*)

Relational Database:-



Table



- ① Collection of table related to each other
- ① Each row will be a different item.
- ① " col " " attribute.

① Non-relational DBs :-

- ① Graph
- ① key-value
- ① Document
- ① Columns

Break till 08:08 AM

② Properties of RDBMS:-

- ① Collection of tables
- ① Every row should be unique.
- ① All the values in a column should be of same type.

① The values should "atomic".

↑
indivisible

① Column sequence
is not guaranteed.

(never assume!)

① The order of rows → NOT Guaranteed.
(ORDER BY)

② No two columns have same Name.

name	name

③ Keys :-

Students

First-name	last-name	PSP	batch	email	Phone
Abhishek	Kumar	80	1	abcd@gmail.com	9876. -
Abhishek	Verma	70	2	xyz@gmail.com	4321. . .
Pankaj	Sharma	90	1	a1@gmail.com	975. . .
Rahul	Kumar	100	2	a2@gmail.com	432. . .
⋮	⋮	⋮	⋮	⋮	⋮

80

④ Super key :-

(first-name) X

(last-name) X

(PSP) X

(batch-id) X

(email) ✓

(Phone) ✓

Constraint

Visual schema

Visual schema

(first-name, last-name) ✗

(lastname, PGP) ✗

(batch_id, email) ✓

(batch_id, phone) ✓

(email, phone) ✓

(PGP, email, lastname, ...)

(b., e., l.)

⋮

⇒
⇒
uniqueness

⑧ Super keys

↳ a column or set of columns of a table whose values are guaranteed to be able to identify a row uniquely.

(*) Candidate keys :-

Students

First-name	last-name	PSP	batch	email	Phone
Abhishek	Kumar	80	1	abcd@gmail.com	9876.~
Abhishek	Verma	70	2	xyz@gmail.com	4321.~
Pankaj	Sharma	90	1	a1@gmail.com	975...
Rahul	Kumar	100	2	a2@gmail.com	432...
⋮	⋮	⋮	⋮	⋮	⋮

$(\text{lastname}, \underline{\text{email}})$ ✗

(email) ✓

Candidate keys

↳ Super key of smallest size

(last-name) ✗

(email) ✓

(phone) ✓

✓ $(\text{first-name}, \text{last-name})$

① Candidate key \Rightarrow super key; if we remove any column, that key will no longer be a SUPER key!

(where all columns defined are needed to identify rows uniquely)

(first_name, last_name)

X (first_name)
X (last_name)

✓✓ (first_name, last_name) X \Rightarrow Candidate key!

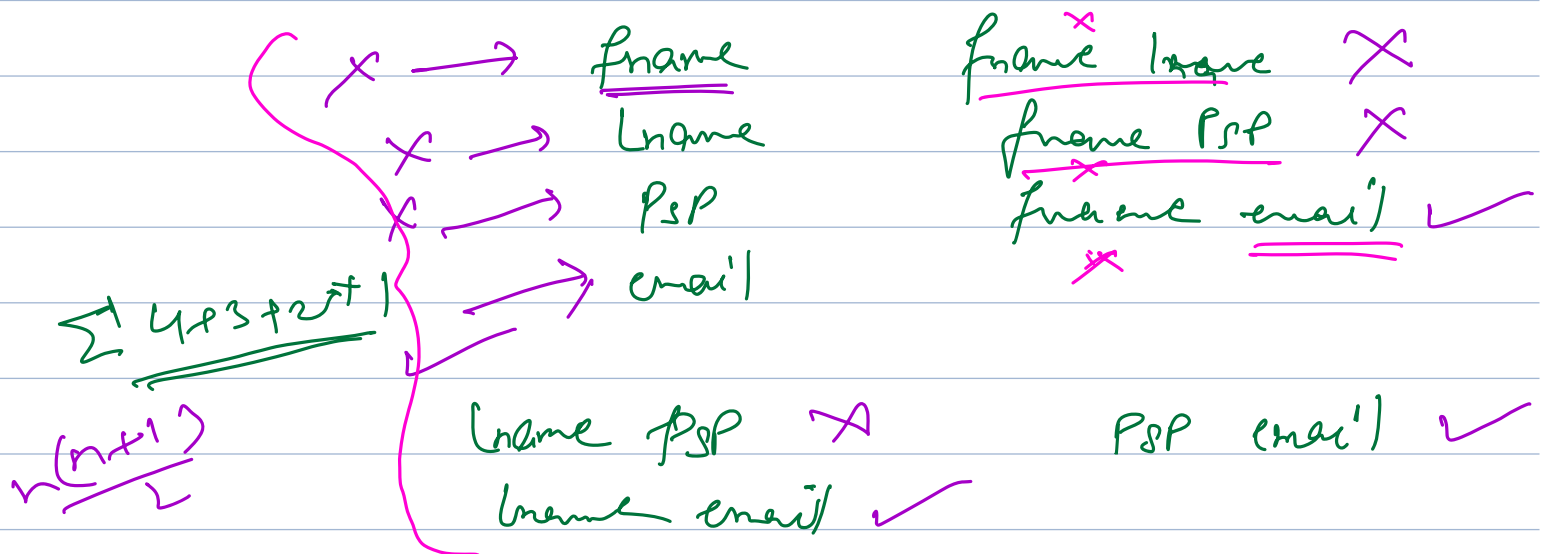
X (EmpId, Department)
↑ ↑
X

All possible combinations

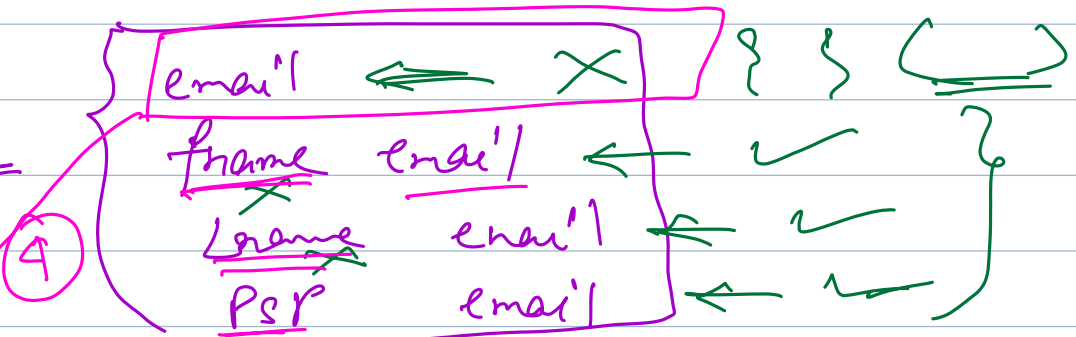
super key

candidate keys!

{ name, lname, PSp, email }



Super keys



PK!

email
(phone) (Candidate Key!)
(Candidate Key!)

⇒ { email
 fname, email } X
 (name, email) X
 PSP, email } X

superkey

(Phone , email)
 ↑ ↑
 ✓ X
 > ✓

PK!

(first name , (name)) ⇒
 ✓ ✓

empId
PK

Candidate Key! → (empId , email) → X
 → (empId) X
 → (email)
 Both PK & C ✓