Keys

candidate keys

(email) ✓ CK.

(phone, email) CK

(phone, name)

(phone, ~~~~~)

Redundancy.

key: uniquely identifying a [ROW]

candidate keys: Super keys of The smallest size

(email) CK.

(phone) CK.

attendace Table.

id	student_id	class_id	attendance %
1	400	191	80%.
2	401	191	40
3	402	201	~~100~~ 30
4.	402	202	~~100~~ 60

st-id, attendance

(student_id) super X.

(class_id) X.

(student_id, attendance) X

(student_id, class_id) super key ✓

candidate key.

length 2

((id)) super - CK.

(email, ⌢⌢⌢), super

candidate : super key from which
you cannot drop any more col^m
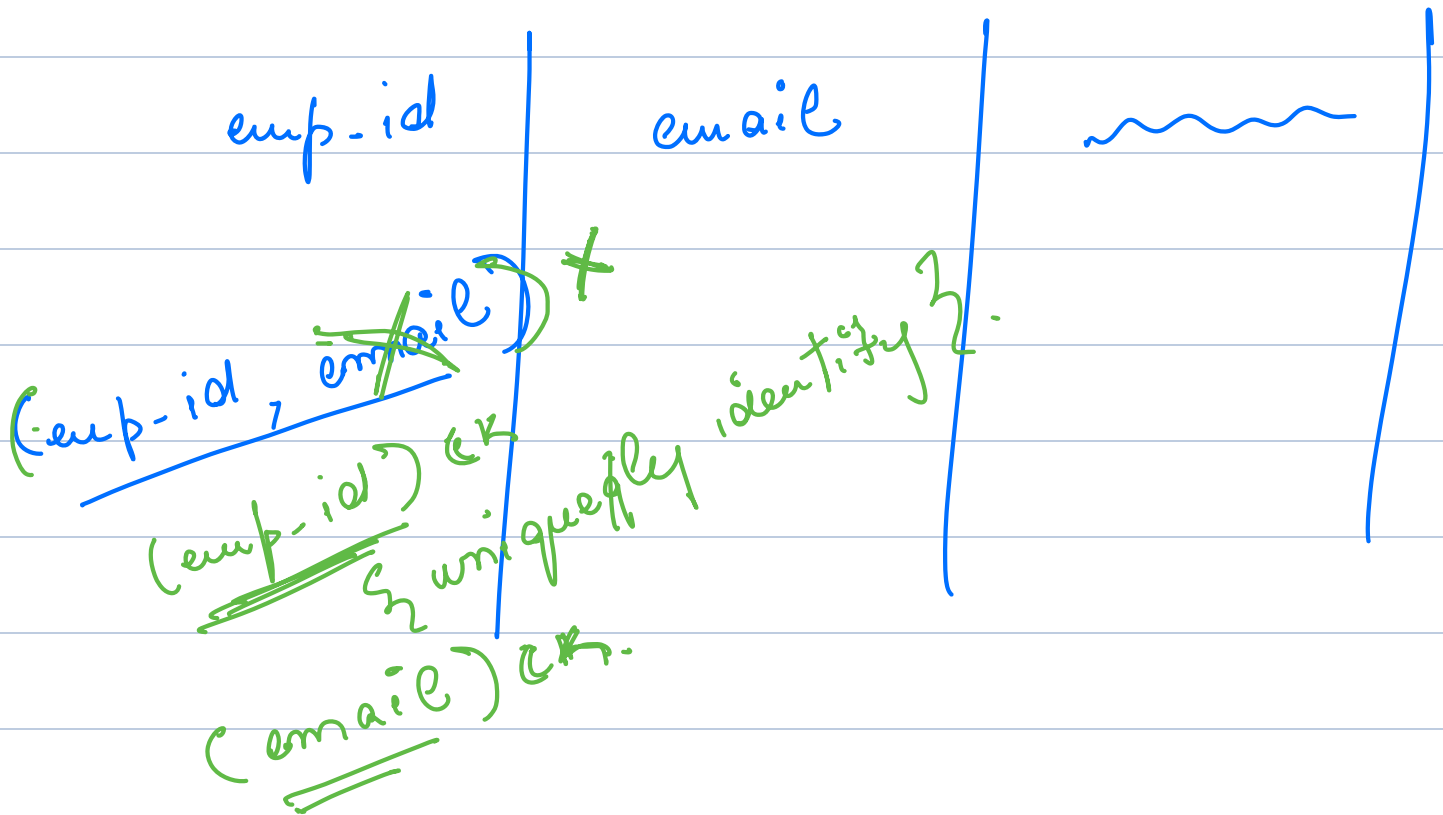without losing m uniquely

identifying a row.

→ Is super key always a candidate key?
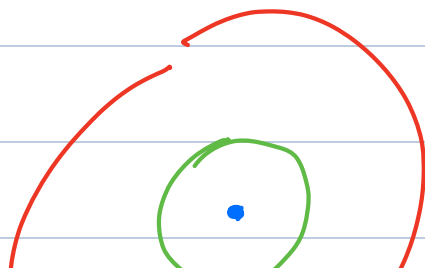   definite no.

→ candidate key for employee Table
    → (emp-id, department) CK.✗
    → (email)   ~~department~~ redundant colⁿ.
                CK. {no redundant colⁿ.}

emp-id	email	～～～～

(emp-id, ~~email~~) ✗

(emp-id) CK.
    { uniquely identify }

(email) CK.

key : uniquely ROWS.

— super key
— candidate

→ Database, 1 Primary, Table.

Primary keys -

→ database will use PK to store
sweetopate data on the disk..

id

1

2

2

:

102

:

index = PK
default

>100

<100

sorting

102, (~~~).

1, (), 2()
99().
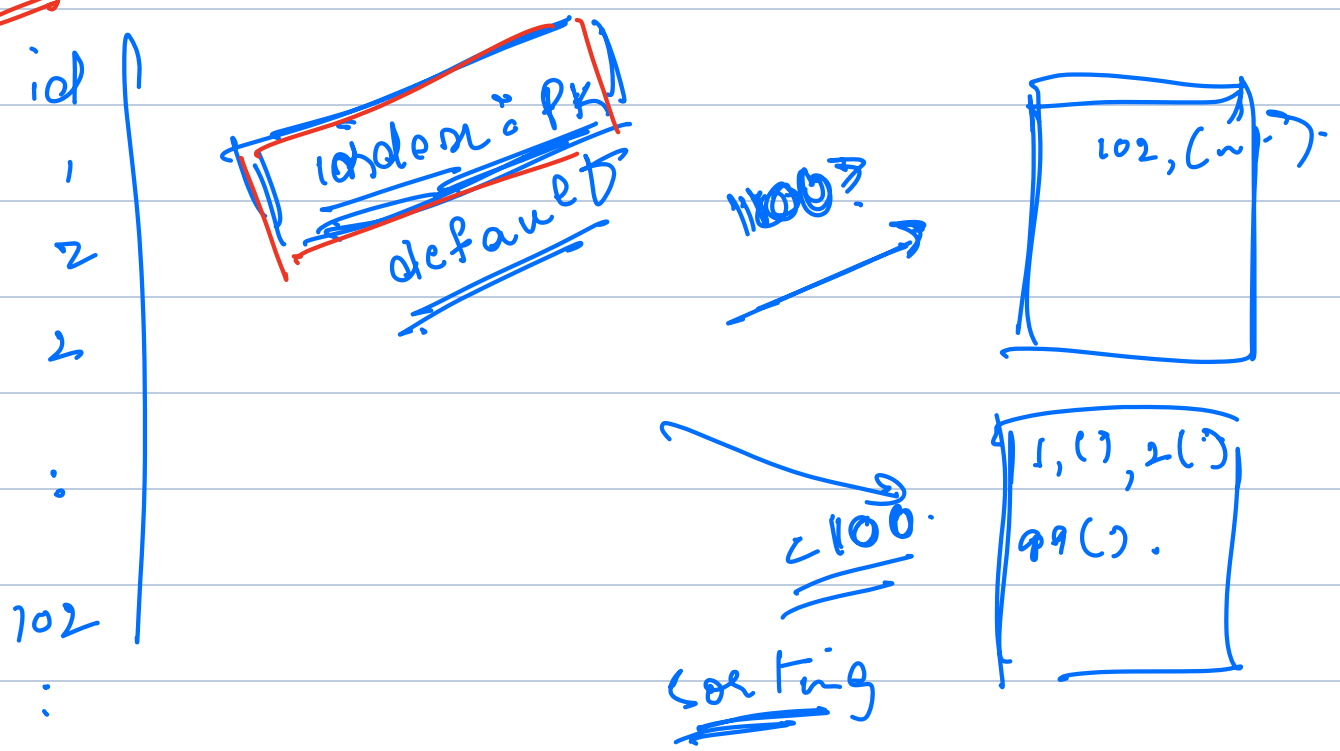
int ?

(~~~ , ~~~ , ~~~).

① int col
② single col
③ Should ~~not~~ change (Idealy).
④ should not be null.

} good to have

$($ phone number $)$

$($ email $)$ $\longleftarrow$ ayush1 $\longrightarrow$ f

ayush2

students

id	name	email	phone	batch_id
1	Ayush			
2	Sid			
3	Novis			

Auto-increments

	SK	CK	P
name	✗	✗	✗
email	✓	✓	
phone	✓	✓	
batch_id	✗	✗	

→ cluster (Int).

→ easily identifiable

→ unique

composite keys.

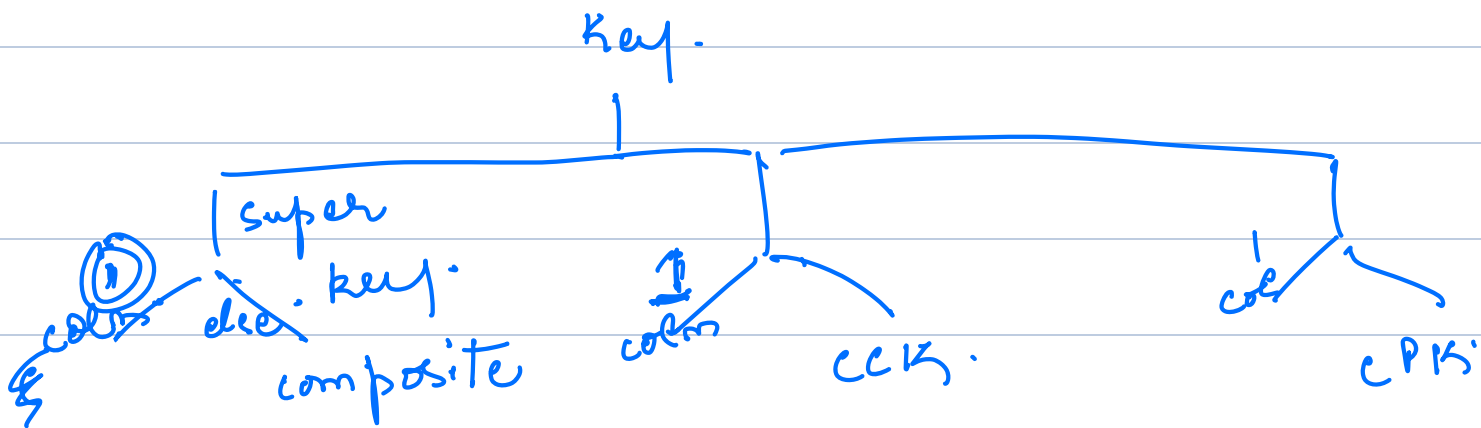A key with 2 or more columns.

(email, phone).

(email, address) Super key
composite super key.

(email) SK, CK.
column (CK ∞)

(student_id, batch_id)
SK composite candidate key.
composite Primary key.

Key.

super
key.
composite

column
CCK.

col
CPK

foreign keys { foreign key constraints }

student

id	name	email	Phone	batch_id
1	Ayush	xyz@	1234	1
2	Arsalan	~	2345	1

null

referential
integrity

student cons.

batch

id	name

1   batch_name   100

foreign key: col^n or set of colums that help uniquely identify a row of another Table ..

user Table

user_id	email	phone	wallet_amount
1	nyz	123	2000/-

deleting user ✗

$-100 ✗$
$-100+20$

Transaction

discount name.

Tron-id	amount	vendor	userid	discountid
✗ 1	100	swiggy	(1) null	3 (20/-)
2	₹920	Hacker	1	74.
3	1920	Hacker	3	3

discount

discount id	name	amount
3	dimali Dhamaka	20

while deleting entry in user:

→ null ⟹ SET NULL ⎫ replicate ⎫
→ delete ⟹ CASCADE ⎬ somewhere ⎬
→ error: ⎭ ⎭
→ NO ACTION

In SQL, you can specify ..what to do
in case of an update or. delete
while creating FK.

Install
→ MySQL 8.
→ SQL client (of your choice).
→ work bench
→ Pgadmin
→ dbeaver.
→ Sakila database