SQL-02

Keys

Agenda

- ↳ Super-key
- ↳ Candidate key
- ↳ Primary key
- ⇒ SQL Introduction

↳ Foreign key

○ Composite key

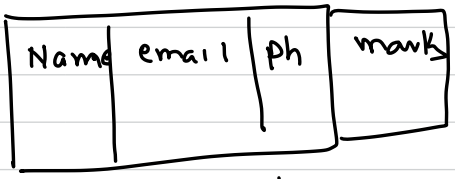
Super key:

{email}

{phone No}

{~~Name~~}

{~~marks~~}



Student

{Name, email}

{Name, email, mavks}

Candidate-key: is a super key from which no col can be removed and it will still have the property of uniquely identifying a row.

minimal super key

{ ~~Name~~, email }

{ ~~Name~~, email, ~~major~~ }

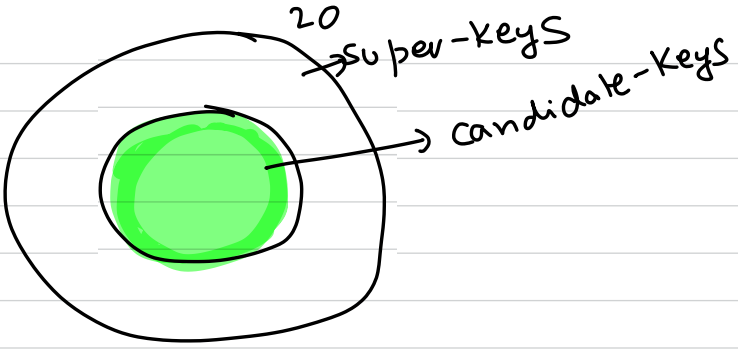
{ email }

{ phone }

{ email, phone NO }

↑ Remove OR ↑ Remove

	Super Key	Cand Key
{ ~~Name~~, email }	Yes	NO
{ ~~Name~~, email, ~~major~~ }	Yes	NO
{ email }	Yes	Yes
{ phone }	yes	Yes
{ email, phone NO }	Yes	NO



Every CK is SK but not every SK is CK.

Will CK be always one column?

Attendace → of every student for every class.

Studentid	class id	attd.
1	2	50
1	3	75
2	2	60

	SK	CK
{ student_id, class_id, attd }	Yes	NO
{ student_id }	X	X
{ class_id }	X	X
{ student_id, class_id }	Yes	Yes

↳ a CK can have more than 1 colm.

Quiz 3:

5 ME
11 ME

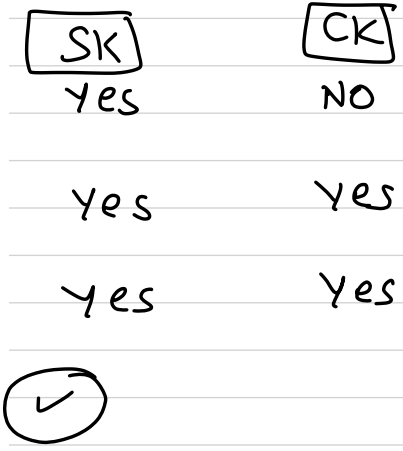
Candidate key

	SK	CK
☐ { Employee Id, ~~Dept~~ }	Yes	NO
☑ { Email }	Yes	Yes
☐ { First Name, Last Name }	X	X
☐ { Last Name, Dept }	X	X

If both EmployeeID and Email are unique for each employee, which of these could be a Candidate Key for the Employee table?

35 users have participated

A	{EmployeeID, Email}	17%	
B	{EmployeeID}	6%	
C	{Email}	3%	
✓ D	Both B and C	74%	



PRIMARY Key

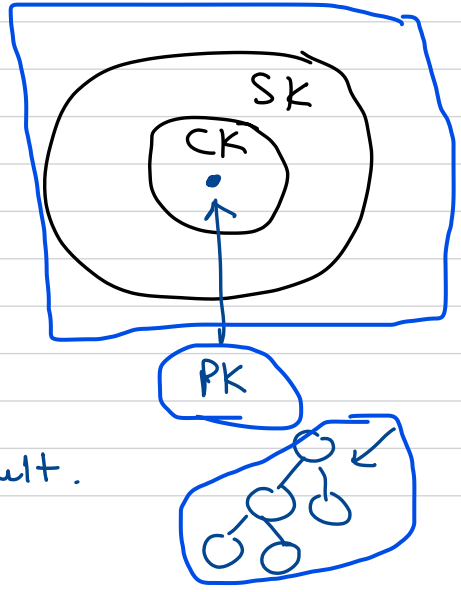
multiple SK : Yes
multiple CK : Yes

primary key : → It is a candiate key that is chosen to be the key for that table.

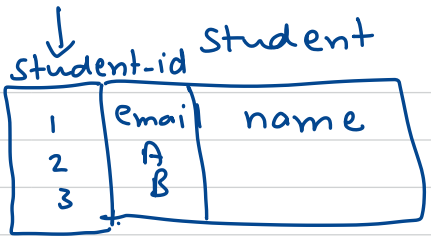
A good primary key

- ✓ small size
- ✓ fast to sort upon (results are sorted acc to PK)

→ DB create 'index' on PK by default.



if student email is updated, will also have to update the tree structure (index)



✓ should not get changed.

Candidate keys

{ student-id }
{ email }
{ phone no }

→ Primary Key
↳ small
↳ fast
↳ will not get changed.

Attendance

class_id	student_id	attendance
2	5	50%
2	6	60%
3	5	40%
3	6	70%

st_id → 6

cl_id → 2

(class_id, student_id, att)

(class_id, student_id)

(2, 6)

Super-key: Yes — Y

Can-key: Yes — N

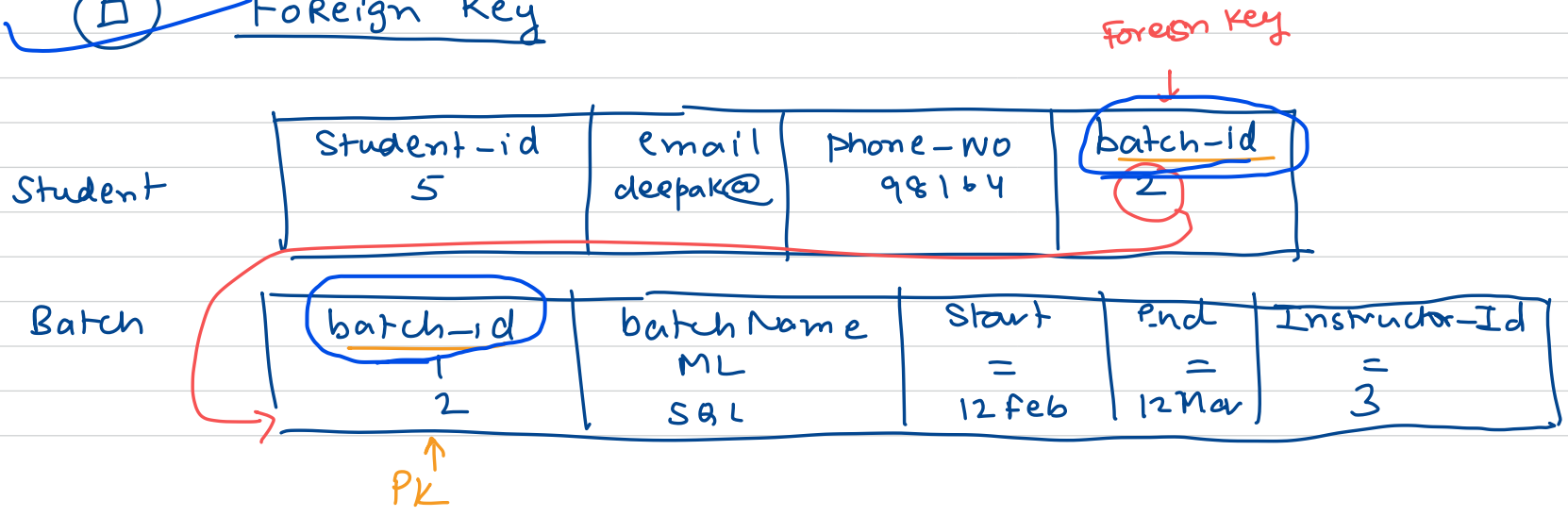
Primary-key: yes, and composite — N

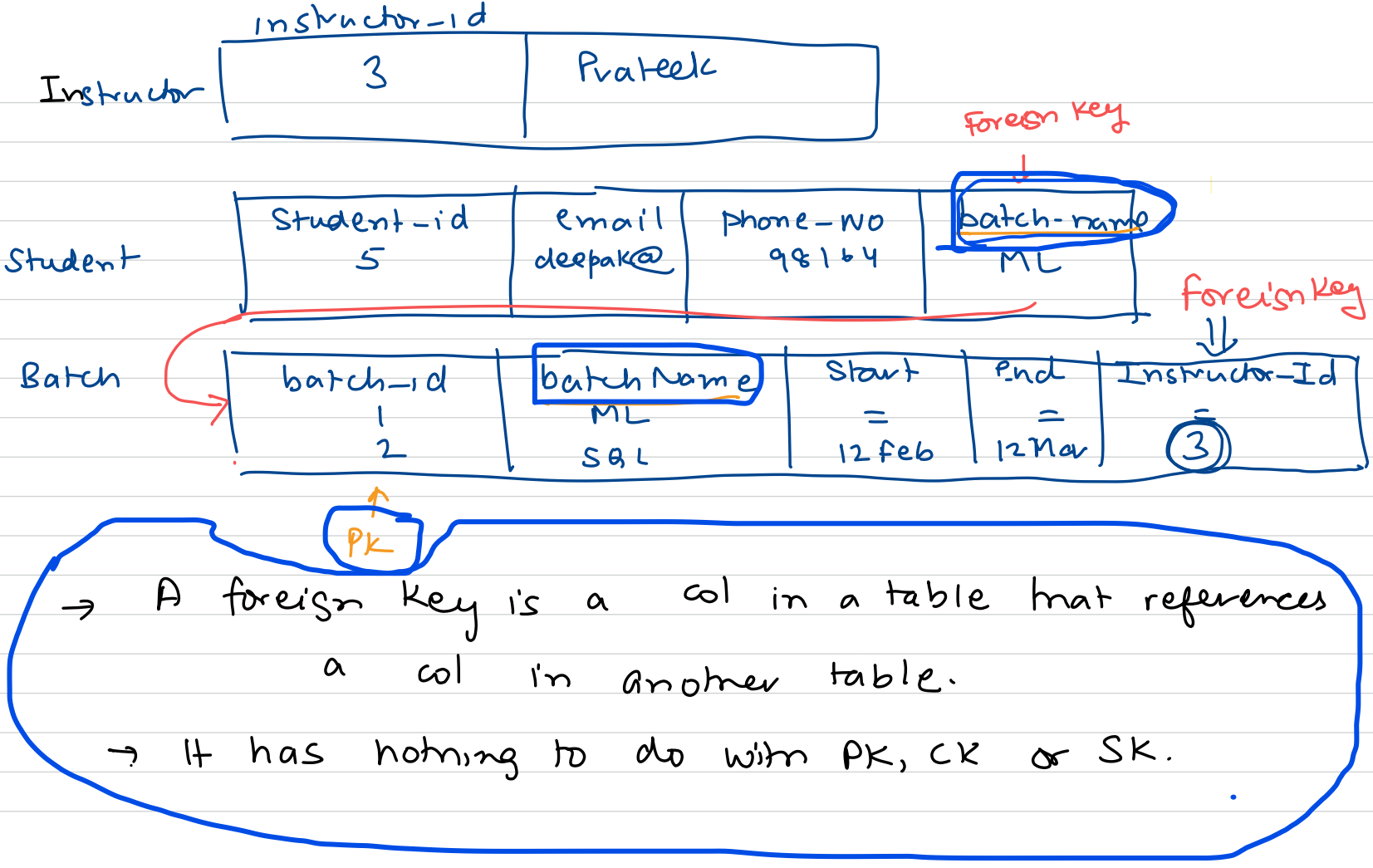
Composite Key: (Terminology)

Any key which has more than one column is termed as composite key.

A foreign key is a field (or a set of fields) in one table that uniquely identifies a row in another table. The purpose of a foreign key is to create a link between the data in two tables. It should be candidate key.

Foreign Key




instructor-id
Instructor
3
Prateek
Foreign key
Student
Student-id
5
email
deepak@
phone-NO
9816 4
batch-name
ML
foreign key
Batch
batch-id
1
2
batch Name
ML
SQL
Start
=
12 feb
End
=
12 Mar
Instructor-Id
=
3
PK
→ A foreign key is a col in a table that references a col in another table.
→ It has nothing to do with PK, CK or SK.

batch_id	batch_name
1	Batch A
2	Batch B
3	Batch C

student_id	first_name	last_name	batch_id
1	John	Doe	1
2	Jane	Doe	1
3	Jim	Brown	2
4	Jenny	Smith	3
5	Jack	Johnson	2

Foreign key

Foreign key

mentor_id

5
7
5
6
5

Mentor

Ment Id	Ment Name
5	Prateek
6	Naman
7	Deepak

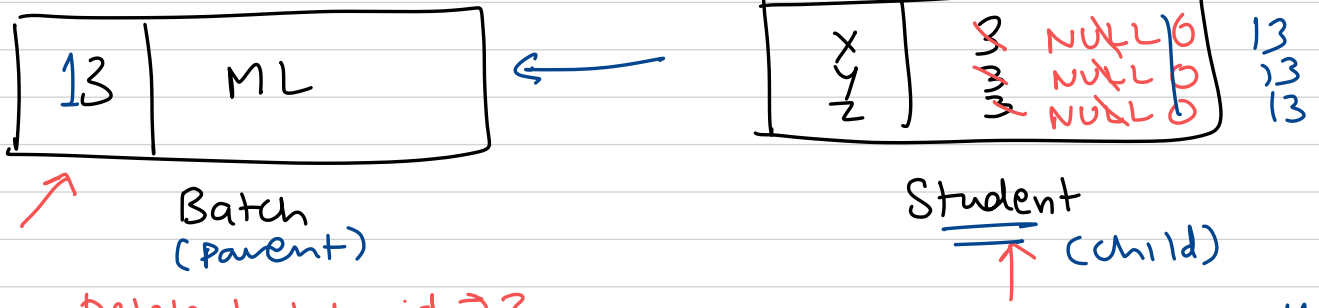
Multiple FKs can be declared as FK referencing different tables.

Intro to SQL

10:10

Issues

- ~~Delete~~ a batch
- Update



Delete batch_id → 3

- CASCADE — ① Delete students
- SET NULL — ② Set NULL
- RESTRICT — ③ Don't Allow / Restrict
- SET DEFAULT — ④ Set Default → 0

Default Behaviour

Update Studen

Key Points about Foreign Keys

Referential Integrity: Foreign keys ensure that values in the column in one table (the child table) must exist in another table (the parent table). This helps maintain data consistency.

On Delete/On Update: Foreign keys can be set with rules for what happens when the referenced data in the parent table is deleted or updated. Common rules are ON DELETE CASCADE (deletes related child rows), ON DELETE SET NULL, or ON DELETE RESTRICT