

#

Students.

St_id	email	Phone	name
int	String			

Super Key

St_id

St_id, name

email

Phone

email, phone

Phone, name

=====

=====

=====

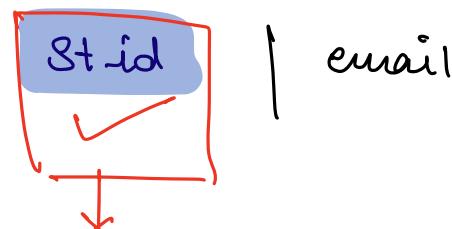
Candidate KeyMinimal Super Key

St_id

email

Phone

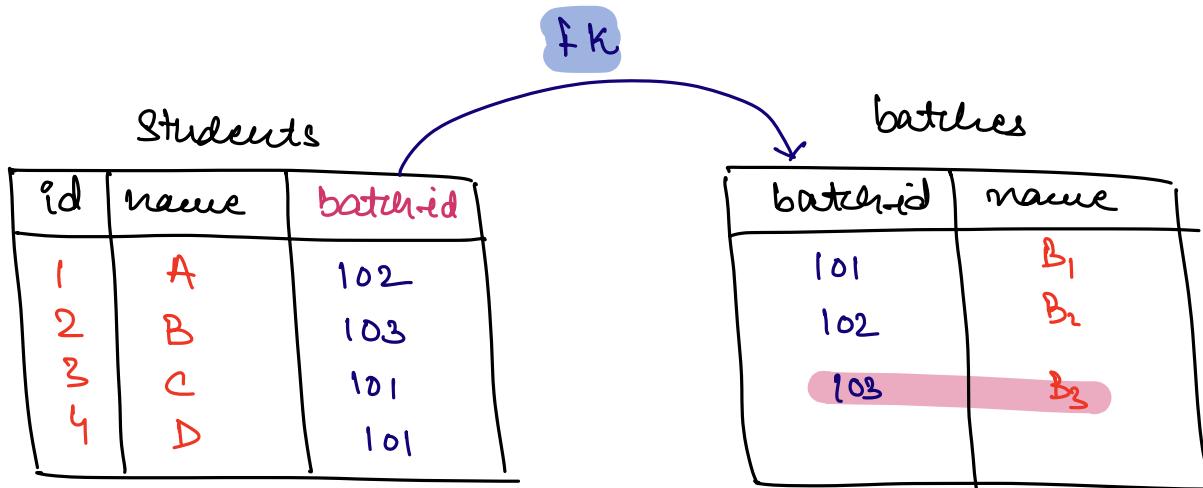
email, phone X

Primary Key

Better choice for

Primary Key

foreign key



⇒ If we try to delete the batch with id = 103, then :

- I) Not Allowed
- II) Delete all the students belongs to batch id = 103.
- III) Mark the batch_id as NULL for all students belongs to batch id = 103.
- IV) No action

CASCADE TYPES.

```
select * from film;
-- select : Print

select "Hi Everyone, Good evening. ";

select * from film where film_id = 10;

-- * -> all the columns
select * from film where release_year = 2006;

select film_id, title from film where film_id = 20;

-- DISTINCT
select DISTINCT rating from film;

select DISTINCT rating, release_year from film;

-- select rating, DISTINCT release_year from film; NOT A VALID QUERY

select * from film where rating = 'PG-13' and length < 100;

-- AND, OR, NOT
-- != <> : Not equal to
select * from film where rating <> 'PG-13';

select * from film where (rating = 'PG-13' OR release_year = 2006) and length > 100;

-- IN Operator

select * from film where rating = 'G' OR rating = 'PG' OR rating = 'PG-13';

select * from film where rating IN ('G', 'PG', 'PG-13');

select * from film where rating NOT IN ('G', 'PG', 'PG-13');

-- By default the table gets sorted based on PK of the table.
-- If we want to sort the data in other format, we can use ORDER BY
select * from film ORDER BY length;

select * from film where title = 'AGENT TRUMAN';

-- Select all the movies whose title starts with a/A
-- LIKE Operator

-- first character - A, after A, any character can come any number of times.
-- % : Any character any number of times.
-- _ : Only single character.
```

-- A% : starts with A, and after A any character can come any number of times.
select * from film where title LIKE 'A%';

-- Select all the movies whose title ends with k
select * from film where title LIKE '%K';

-- Select all the movies whose title containns 'TH'
select * from film where title LIKE '%TH%';

-- Select all the movies whose title starts with BU
select * from film where title LIKE 'BU%';

-- Select all the movies whose title starts with BU and contains SO
select * from film where title LIKE 'BU%SO%';

-- Select all the movies whose title starts with DE and last second character A
select * from film where title LIKE 'DE%A_';

-- select the films whose release year is from 2006 to 2010
select * from film where release_year >= 2006 and release_year <= 2010;

-- BETWEEN

-- age >= 18 and age <= 25 -> age between (18, 25)

-- BETWEEN a and b : >= a and <= b

select * from film where release_year BETWEEN 2006 AND 2010;

-- Select all the movies whose title starts with DE and last fourth character R
select * from film where title LIKE 'DE%R____';

-- INSERT INTO film(....) VALUES

-- (.....)