We begin at 9:10 PM.

Good Evening.

Updates

1. Notes → L1 & L2
2. Assignment
3. Agenda
   ↳ Primary key, Composite key
   Foreign key
   → Schema Design
   → Case Study
   → SQL Data types

Schema Design

1. What is it?
2. Why is it required?
3. When is it done? Before Coding

Designing Phase

Design Document

→ Requirements → Classes
APIs
~~Tabs~~ Schema

Schema Design

- Pictorial Representation
- Tables
  - Columns (Data Types)
- Relationship btw tables

Case Study

Business Requirements

1. There are batches. Name, Start Month, Current Instructor, students.
2. Students. name, grad year, university, email, phone number, ctc
3. Students have mentors
4. Mentors have name, dob.
5. Student are allowed to shift across batches (st_date, end_date)
6. Batches will have classes.
7. Class will have name & an instructor.

Tasks

1. Find the entities

2. " " properties of those entities

Cardinality 1-1, 1-m, m-m

3. Find relationships btw entities.

4. Express the above 3 in forms of tbls & cols

Task1 (Finding the entity)

→ Nouns

→ For which information is reqd.

Students
Batches
Classes
Instructors
Mentors

Task 2

email, phone

1. Students - name, grad-year, university, etc
current_batch, previous batches,
mentor, PSP, status

2. Batches - Name, Students, classes,
start, end

3. Classes - name, instructor, date, st. time
end-time
Batches, Students

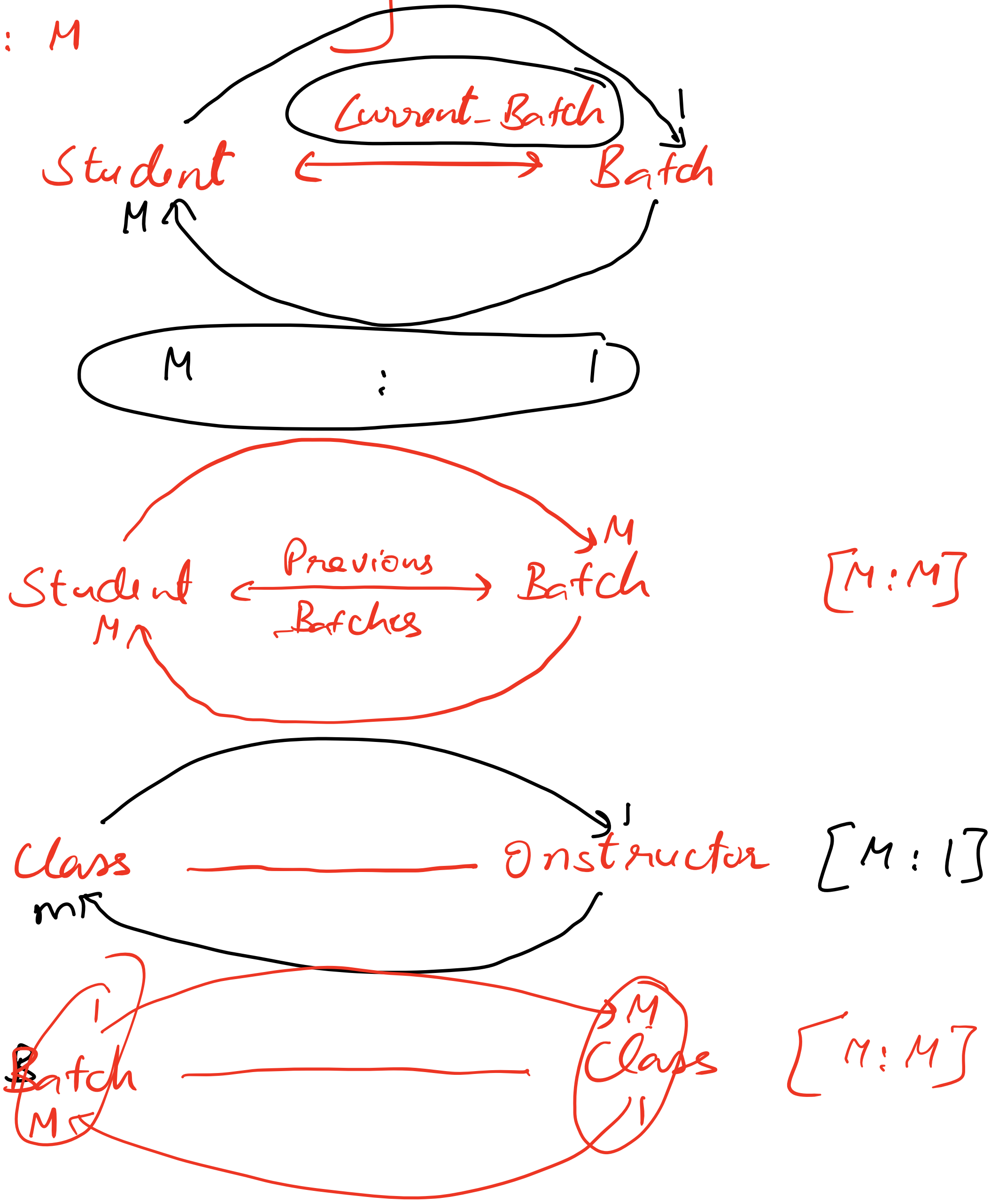
4. Instructor - name, dob, [illegible] ✓

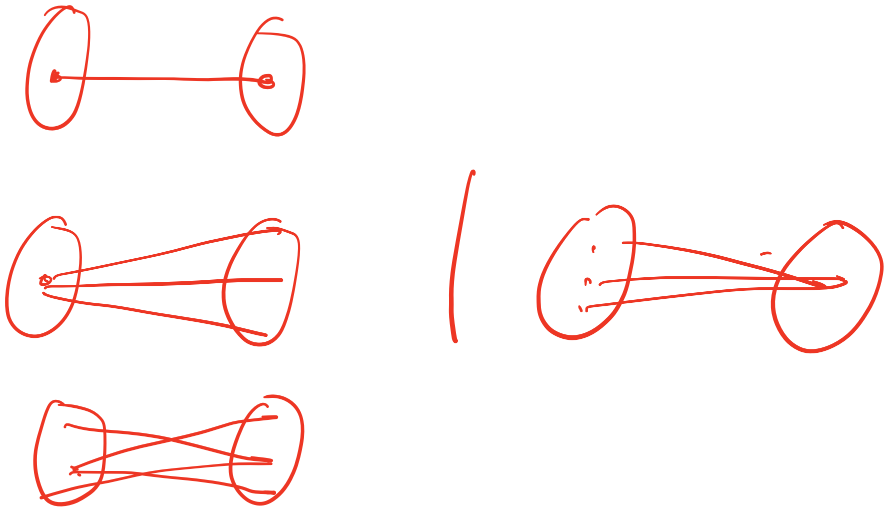
5. Mentors – name, dob, org ✓

Task 3 (Relationships)

1. One-One
2. 1 : M or M : 1
3. M : M

Cardinality

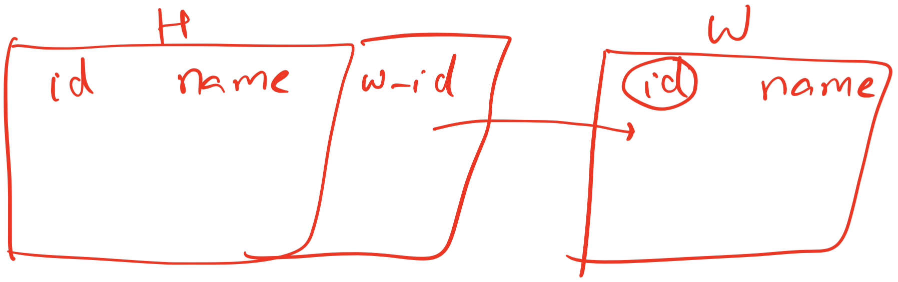




Example (1-1)

1

Husband-Wife



id	name	w-id

id	name

Or

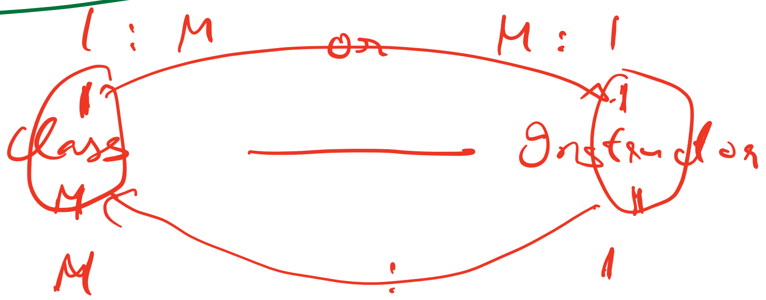
id	name

id	name	n-id

☆ Separate table is reqd. ?

1 : M or M : 1

2



Class
- id
- name
- instructor_id

Instructor
- id
- name

Option 1

Class
- id
- name

Instructor
- id
- name
- classids [ ]

Atomicity

Bad example

Option 2

Class
- id
- name
- instructor id

FK

Instructor
- id
- name

column on many side

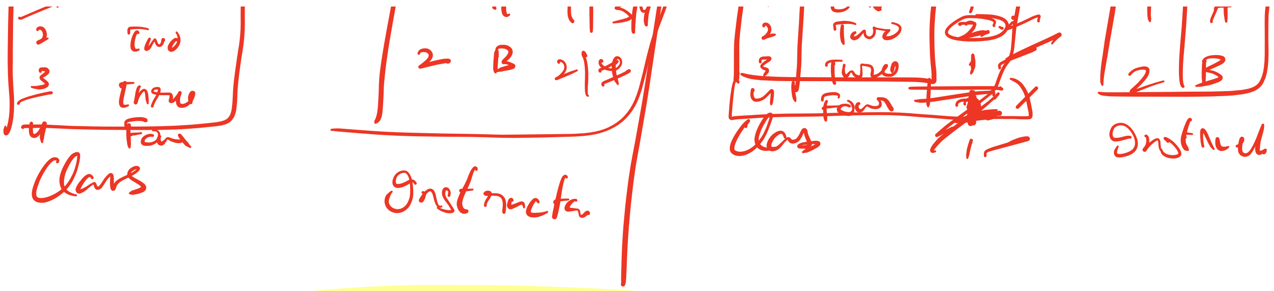
Good example

id	name
1	One

id	name	cids
1	A	1,2,..

id	name	iid
1	One	1

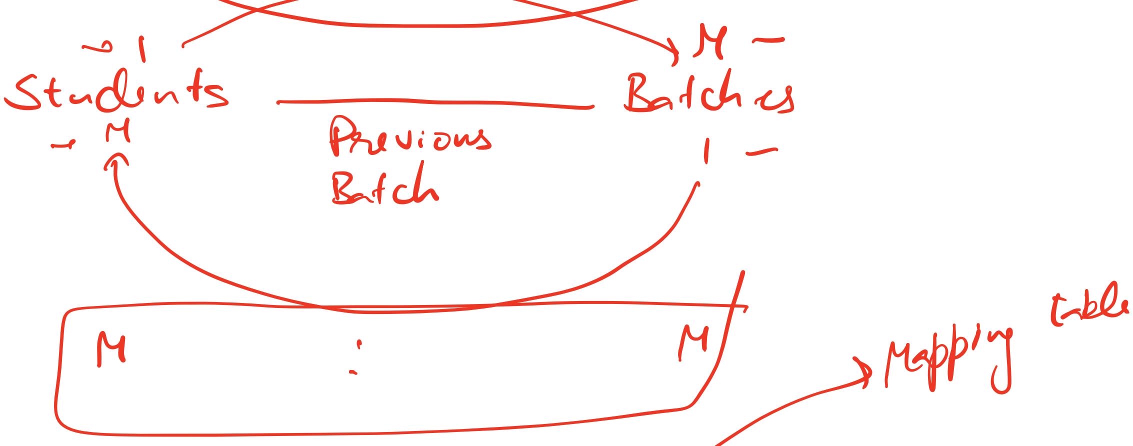
id	name
1	A

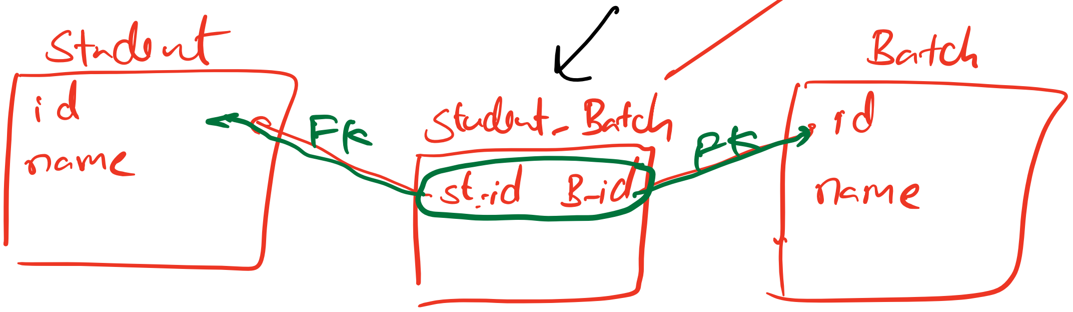


Moral : Add col on many side

M : M relation

3





1:1 → Add col on either side

1:M, M:1 → Add col on many side

M:M → Have a mapping table

id	name
1	One
2	Two
3	Three
4	Four

Student_Batch

s_id	b_id
1	[illegible]
1	3
2	1
2	2
3	2
3	3
4	1
4	2
4	3

Batch

id	name
1	B1
2	B2
3	B3

1 → 1, 3

2 → 1, 2

3 → 2, 3

4 → 1, 2, 3

Is mapping table even reqd. for 1:M or 1:1 ?

→ Case 1 : If the relation is sparse [i.e. lot of NULLS]

id	name	w_id
1	H1	W2
2	H2	NULL
3	H3	NULL
4	H4	NULL
5	H5	W1

Husbands

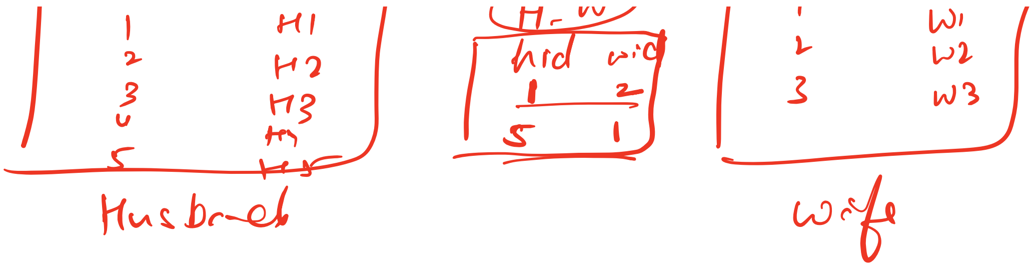
id	name
1	W1
2	W2
3	W3

Wife

id	name

H_id

id	name



1	H1
2	H2
3	H3
4	H4
5	H5

Husband

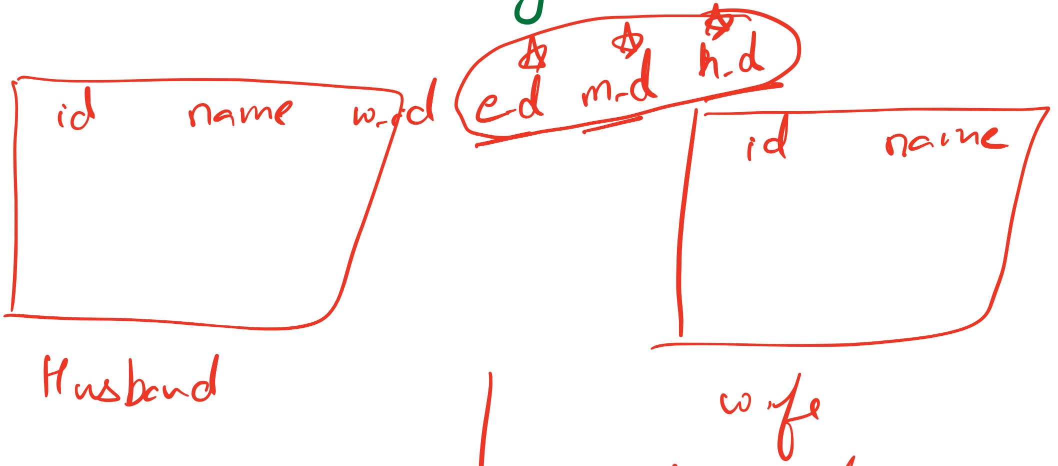
hid	wid
1	2
5	1

1	W1
2	W2
3	W3

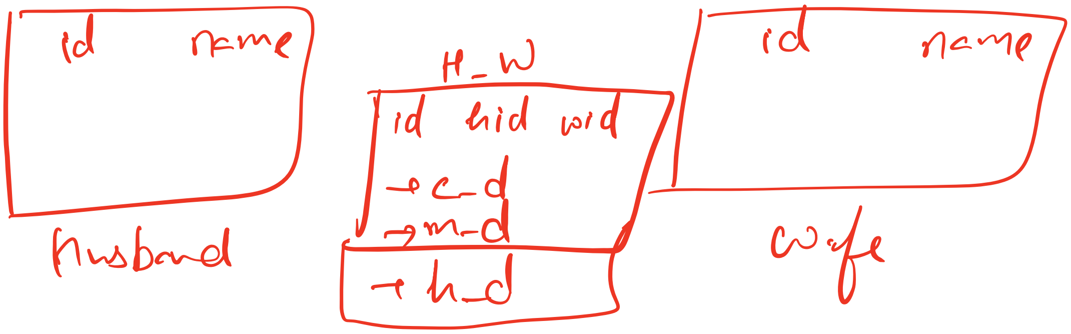
Wife

→ Case 2 [Mapping table for 1:1 or 1:M]

If there are a lot of attributes which get added but doesn't belong to the entity but the relationship



Wiser Approach



② → PK

10:41 to 10:50

→ FK
→ UK
→ CK

[1. Case Study & finish it

3. Data Types

Break till 10:50

mentor_id

- id
- name
- email
- phone
- grad
- univ
- cbid

Student

- id
- name
- start
- end

Batch

S_PB

id	sid	pbid	sd	ed
		0		

- id
- name
- iid
- date
- st
- et

Class

- id
- name
- dob

Instructor

- id
- name
- dob

Mentor

id
1
2
3

B_C

bid	cid
1	1
2	1
3	1

B_C_A

bcid	aid
1	1
1	2

Assign

id	name
1	A1
2	A2
3	A3

Composite key → A key made of more than one columns.

Super key → fname, email, phon, lname

→ (fname, email)

→ lname, email

fname
email
phon
lname

Candidate key → Minimal no of attributes

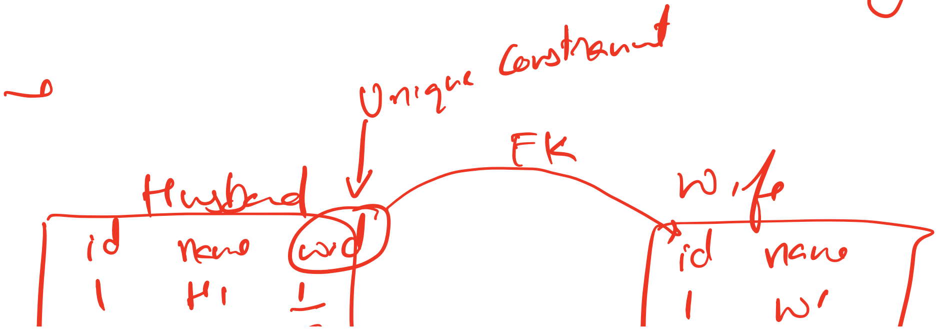
✓ email , phone ✓

email , phone X

Composite key → A key on multiple columns.

PK → A chosen candidate key

UK →



2	M2	2
3	M3	NULL
4	H4	

2	W2
3	W3

↑

1 : M

0 : M

1 per table

PK → Implicitly unique, does not allow nulls.

UK → Explicitly added, to enforce ~~implem~~ 1:1 relationship, allow nulls.

→ Many allowed in a table

Data Types

Normalization