Schema Design
Pictorial representation of database.

Scaler Case Study
    1. There are several batches at scaler. Each batch has id, name,
       current instructor.
    2. Each batch has multiple classes. Each class has id, name and
       instructor.
    3. Every student has id, name, email, graduation year etc.
    4. A student can move from one batch to another batch via pausing. We
       need to store entry and exit date of students related to each batch
       they were part of.
    5. Each student has a mentor. Every mentor has name and date of birth.
    6. Every instructor has name and date of birth.

Step Based Process
    1. Handle Entities
    2. Handle Relationships

Handle Entities
    1. Identify Nouns in requirements.
    2. If we are storing some information about it, create a table for it.

Scaler Case Study
    1. There are several `batches` at scaler. Each batch has id, name,
       current instructor.
    2. Each batch has multiple `classes`. Each class has id, name and
       instructor.
    3. Every `student` has id, name, email, graduation year etc.
    4. A student can move from one batch to another batch via pausing. We
       need to store entry and exit date of students related to each batch
       they were part of.
    5. Each student has a `mentor`. Every mentor has name and date of birth.
    6. Every `instructor` has name and date of birth.

Students		Batch		Classes
ID: Integer		ID: Integer		ID: Integer
Name: String		Name: String		Name: String
Email: String		Curr_Instructor		Instructor
Grad_Year: Integer		classes		
Prev_Batch				
Curr_Batch		Mentor		Instructor
Mentor		ID: Integer		ID: Integer
		Name: String		Name: String
		DOB		DOB

```
Handle Relationships
    1. Identify Relationships.
    2. Identify Cardinality.

Process to determine Cardinality
    1. Approach from both sides.

Cardinality: Cardinality tells us how many of one entity is related to how
many of other entity.


One student can part of how many batches?
Ans: 1

One Batch can have how many students?
Ans: Many
```
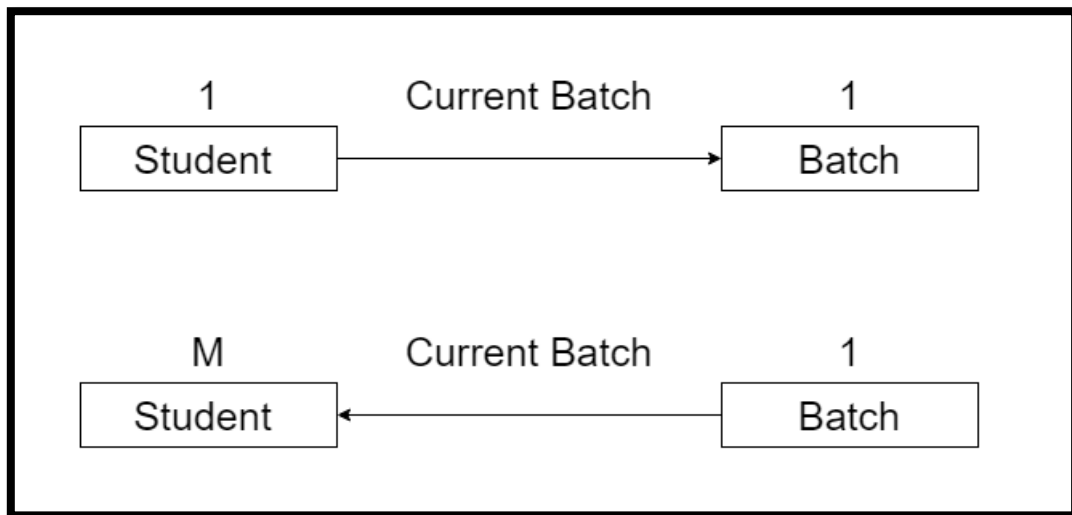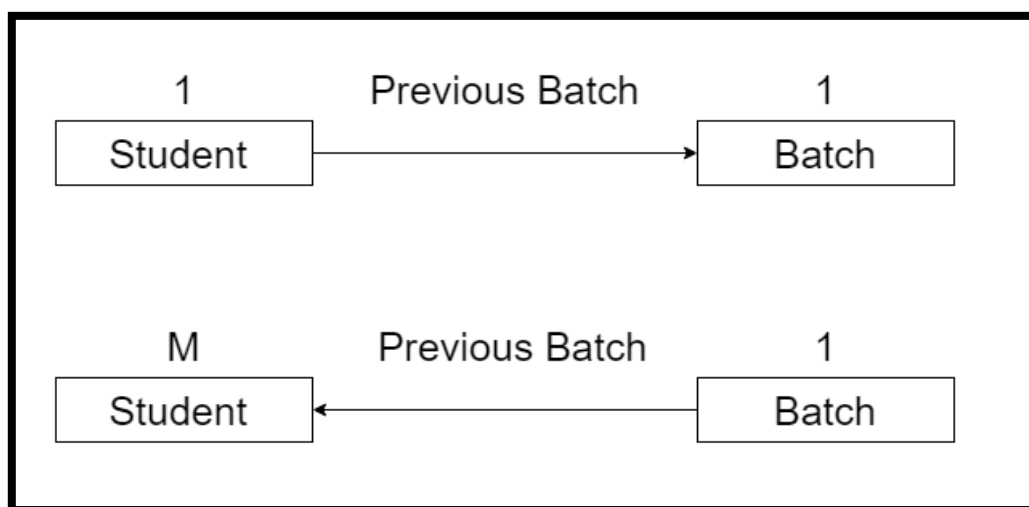
```
  1        Current Batch        1
┌─────────┐                  ┌─────────┐
│ Student │ ───────────────► │  Batch  │
└─────────┘                  └─────────┘


  M        Current Batch        1
┌─────────┐                  ┌─────────┐
│ Student │ ◄─────────────── │  Batch  │
└─────────┘                  └─────────┘
```

```
Now, take maximum from both sides. i.e. M:1. Therefore, relationship from
student to Batch is M:1.
```

```
  1        Previous Batch       1
┌─────────┐                  ┌─────────┐
│ Student │ ───────────────► │  Batch  │
└─────────┘                  └─────────┘


  M        Previous Batch       1
┌─────────┐                  ┌─────────┐
│ Student │ ◄─────────────── │  Batch  │
└─────────┘                  └─────────┘
```

Box 1

1		1
Student	$\longrightarrow$	Mentor

M		1
Student	$\longleftarrow$	Mentor

Box 2

1	Curr_Instructor	1
Batch	$\longrightarrow$	Instructor

M	Curr_Instructor	1
Batch	$\longleftarrow$	Instructor

Box 3

1		M
Batch	$\longrightarrow$	Class

1		1
Batch	$\longleftarrow$	Class
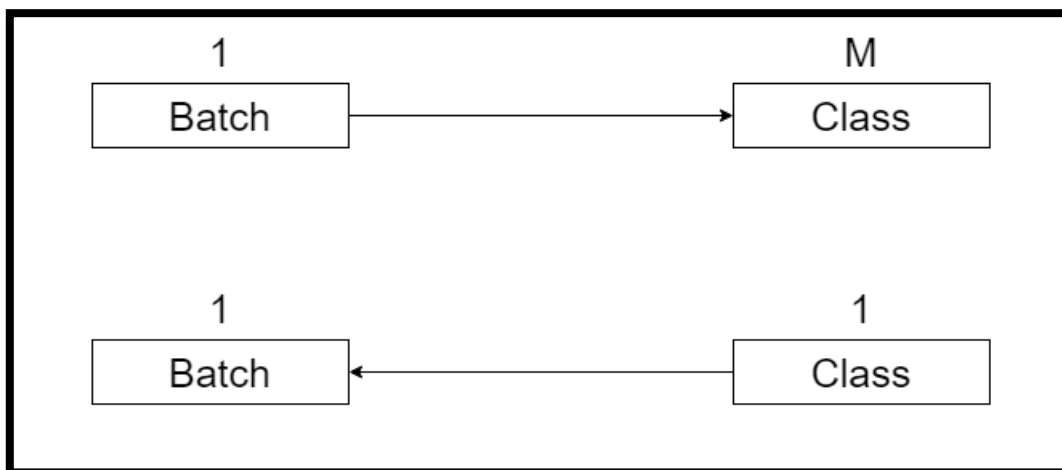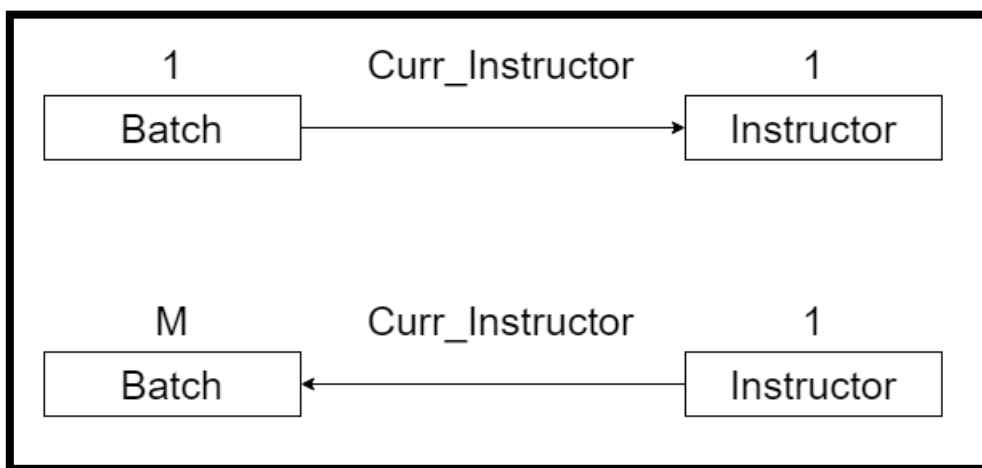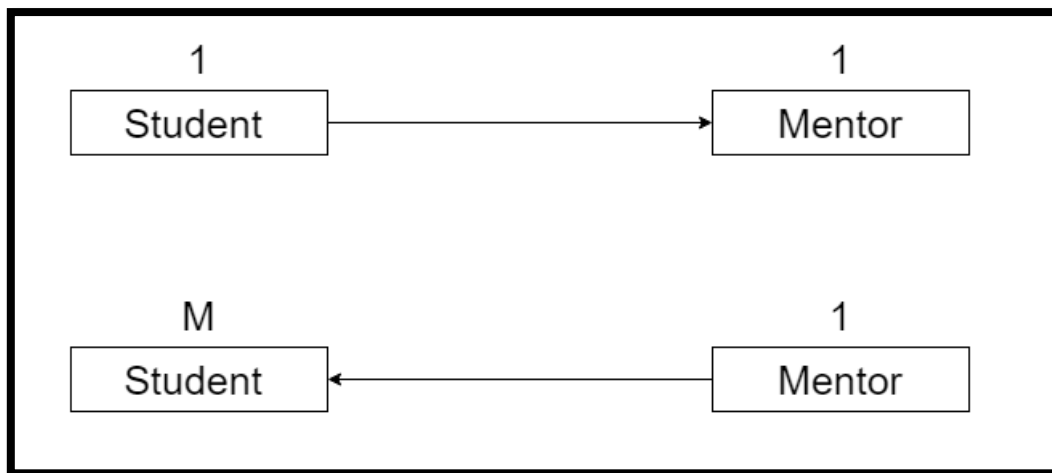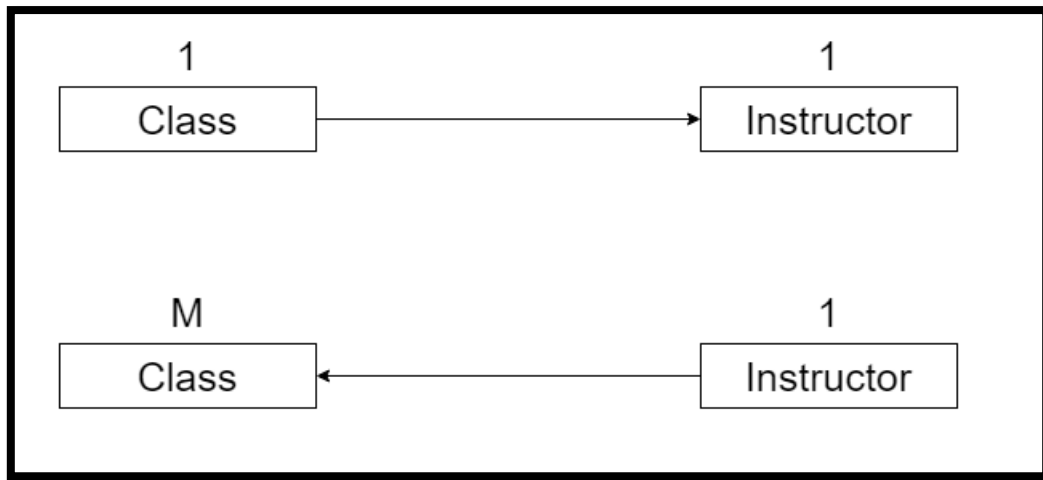
How to model different cardinalities?

1:1

Husbands

Id				

Wives

Id				HID

Options
   1. Put husband id in wives table.
   2. Put wife id in husbands table.

1:M

Batches

Id				

Classes

Id				BID
				1
				2

Options
   1. Store batch id in classes table.

M:1

Current Batch

Batches

Id				

Students

Id				BID
				1
				2

Options
   1. Store batch id in students table.

M:M

Previous Batch

Students

Id				

Batches

Id				

Options
   1. Create Mapping table and stick with composite key.
   2. Create Mapping table and add a new id column as primary key.

Students-Previous-Batches

SID	PBID
1	127
1	1045
1	1896
2	188
2	256
2	10450

Here SID, PBID together makes a composite key.
Here data is first stored by SID then it is stored by PBID.

Students-Previous-Batches

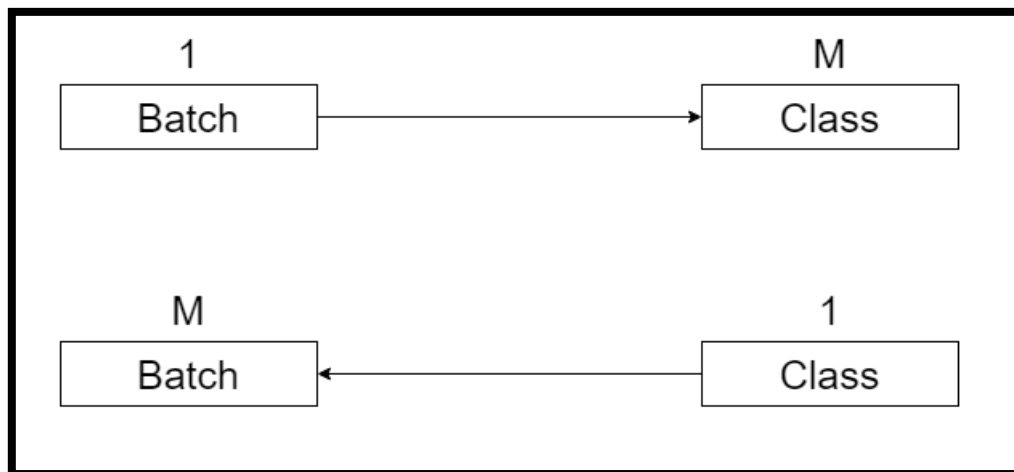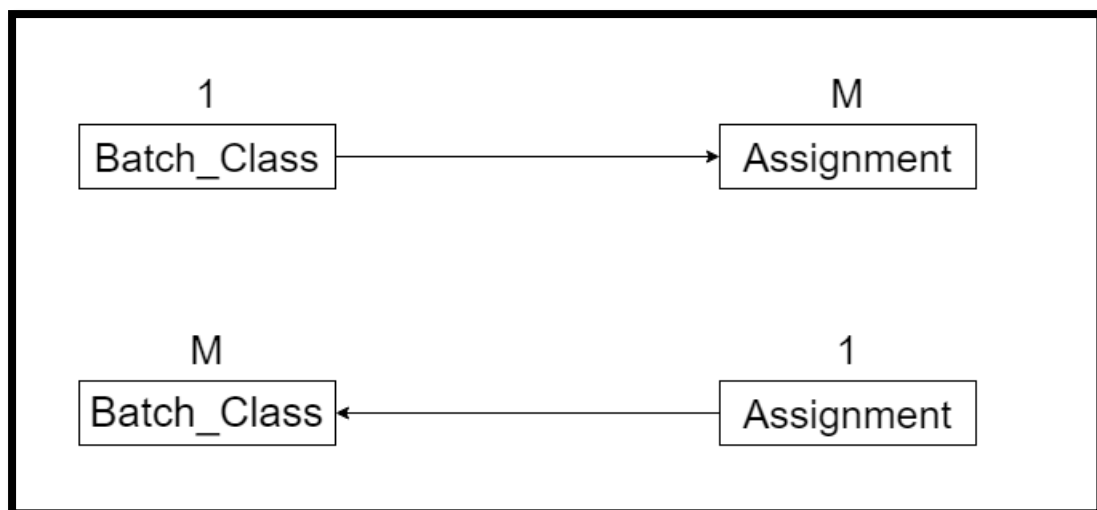| ID	SID	PBID
1	1	4
2	1	10

Create Mapping table and add a new id column as primary key.
   1. Mapping table starts acting like an entity.

ID	SID	PBID	PB_start_date	PB_start_date
1	25	1	16th Aug, 2022	18th Sep, 2022
2	25	2	25th Nov, 2022	3rd Mar, 2023

   2. When mapping table wants to create relationship with other table.

Batch-Class

ID	BID	CID

Batch-Class-Assignment with dedicated ID

BCID	AID

Batch-Class-Assignment with composite key

BID	CID	AID

Batch-Class-Assignment with dedicated ID is more easier to join.

Caveats
    1. Sparse tables.

1:1

Husbands

Id				WID

Wives

Id				

Suppose we have 100 million husbands and 1 million wives then at max 1 million marriages are possible. When we store wife id in husband table there will be at least 99 million spare entries. Thus, we are wasting 396mb memory.

We can create Mapping Table instead of storing id.

Husband-Wife

HID	WID

1:M or M:1
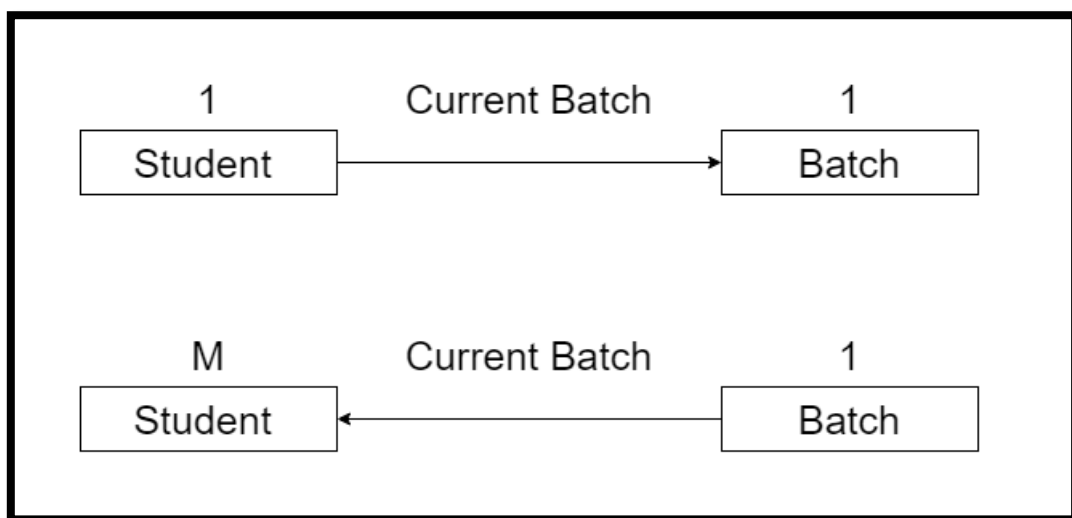
Students

ID	Name	Email	Curr_BID

Now, Suppose scaler takes a master class and 25k enrolled with master class. Out of 25k, 1k registers with scaler. This will results in 24k nulls. So, Here also we can create mapping table.

Students

ID	Name	Email

Students-Current-Batch

SID	CBID

1:1

Wife

ID	Name

Husband

ID	Name	WID	Marriage_date	Marriage_city	Honeymoon_city

Wife

ID	Name

Husband

ID	Name

Husband-Wife-Marriage

ID	HID	WID	Marriage_date	Marriage_city	Honeymoon_city

1:M or M:1

Student

ID	Name	CBID	Curr_batch_joining_date

Student

ID	Name

Batch

ID	Name

Student-Batch

ID	SID	CBID	Curr_batch_joining_date

Condition	Normal	Caveat#1 Sparse Table	Caveat#2 Relationship becomes entity
Cardinality			
1:1	Put id of any side on other side	Mapping Table	Mapping Table
1:M or M:1	Put id of 1 side on M side	Mapping Table	Mapping Table
M:M	Mapping Table	Mapping Table	Mapping Table

Final Scaler Schema Design

Students
| ID: Integer |
| Name: String |
| Email: String |
| Grad_Year: Integer |
| Prev_Batch |
| Curr_Batch |
| Mentor |

Batch
| ID: Integer |
| Name: String |
| Curr_Instructor |
| classes |

Classes
| ID: Integer |
| Name: String |
| Instructor |

Mentor
| ID: Integer |
| Name: String |
| DOB |

Instructor
| ID: Integer |
| Name: String |
| DOB |
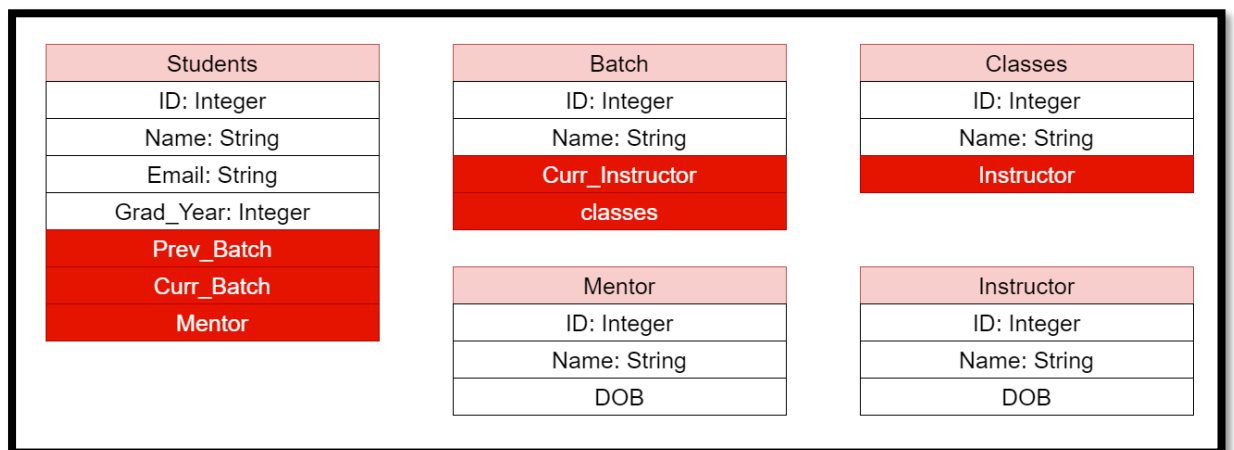
Students

ID: Integer
Name: String
Email: String
Grad_Year: Integer
Mentor_ID: Integer

Batch

ID: Integer
Name: String
Curr_Instructor_ID

Classes

ID: Integer
Name: String
InstructorID: Integer
BID: Integer

Mentor

ID: Integer
Name: String
DOB

Instructor

ID: Integer
Name: String
DOB

Student_Previous_Batch

ID: Integer
SID: Integer
BID: Integer
Batch_Joining_Date:
Batch_Ending_Date:

Student_Previous_Batch

ID: Integer
SID: Integer
BID: Integer
Batch_Joining_Date:
Batch_Ending_Date:

Student_Current_Batch

ID: Integer
SID: Integer
BID: Integer
Batch_Joining_Date: