

ER, R Diagram

Contents

Chapter 1: Entity Relationship Diagram	2
1. Definition:	2
2. Type of relationship:	2
3. Practical:	3
Chapter 2: Relational Diagram.....	6
1. Definition:.....	6
2. Convert ERD to RD:.....	6

Chapter 1: Entity Relationship Diagram

1. Definition:

- An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database.

- The Entity /Relationship Model:

+ **Entity:** is an object, represent name of table and is represented by **rectangles**.

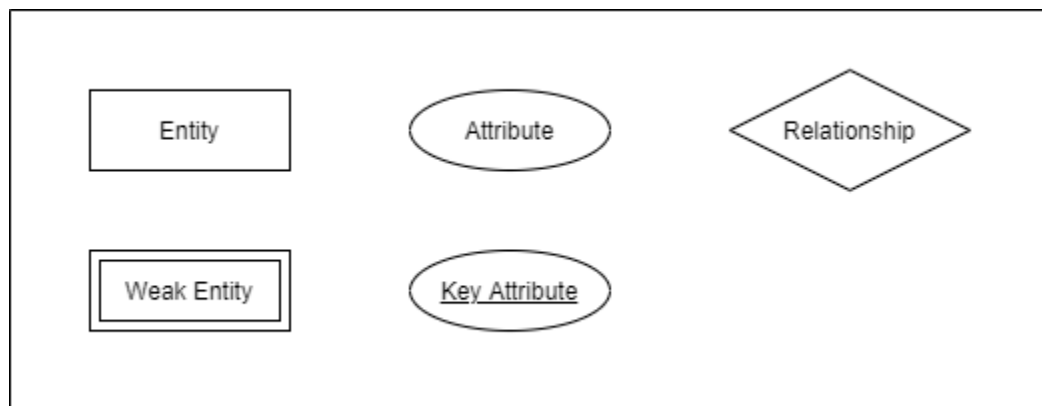
+ **Weak entity:** is an entity that must defined by a foreign key relationship with another entity.

+ **Attributes of an Entity:** are column headers and represented by **ovals**. **Key attributes** represented by **ovals** and **underline**.

+ **Relationship:** represents the relationship between 2 entities, that is represented by **diamonds**.

+ **Edges:** connect a relationship to its entity sets.

+ **Foreign Key:** is a column (or combination of columns) in a table whose values must match values of a column in some other table. **Don't need to draw**, because it will duplicate attribute with 2 tables (**Note:** but must be add when create database).

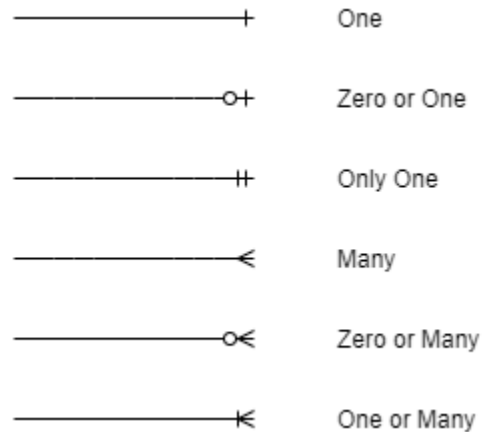


2. Type of relationship:

- **Cardinality:**

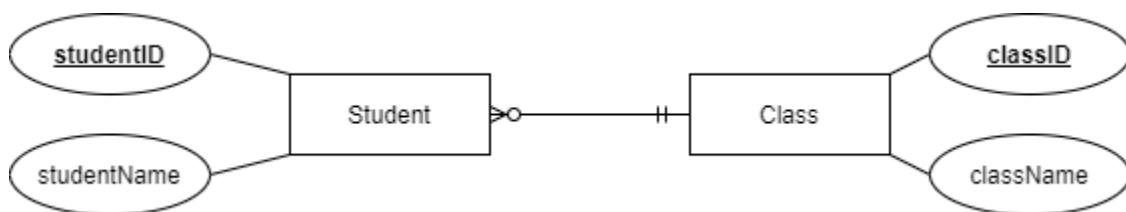
- MANY-MANY relationship

- ONE-MANY relationship
 - ONE-ONE relationship
- **Participation:**
- Mandatory
 - Optional



3. Practical:

- **First example:** Come back to high school, now I have two tables and its attributes:
 - Student: studentID, studentName
 - Class: classID, className
- **Analysis:** Easy to see that one student must belong to one class, but one class can have zero to many students, so we can draw ERD like this:



- **Summary:**
 - + Entity: Student, Class.
 - + Attributes: studentID (Key), studentName, etc
 - + Relation: one to many(mandatory).

+ Foreign Key: classID.

- **Second example:** Now back to present, I'm in university and have one more table is Course:

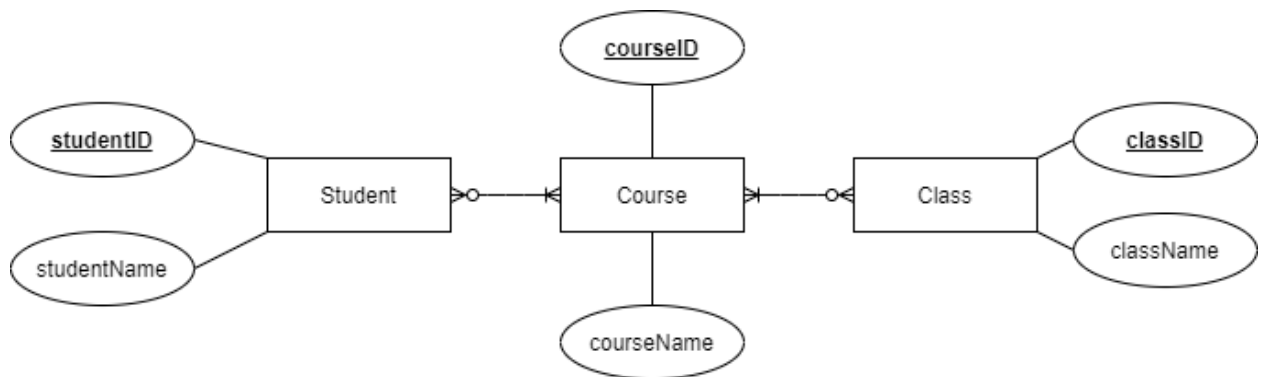
- Student: studentID, studentName
- Course: courseID, courseName
- Class: classID, className

- **Analysis:**

+ Student must enroll one or more course, but foreach course can be enrolled by zero or many students.

+ Foreach course can be taught in zero or many class and one class must have at least one course.

And the ERD will be like this:

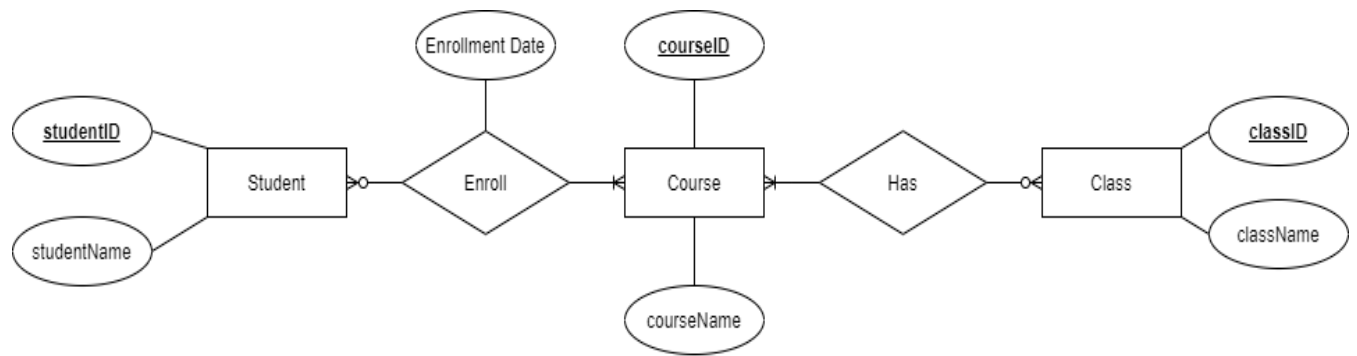


- **Problem:** but now I have a problem with this ERD:

+ With many to many relationships, I need a table to store 2 keys of the others.

+ Between 2 Entity Student and Course, I need an attribute "Enrollment date" to store the date Student enroll to course.

- **Solution:** I will create a Relationship between the entities, ERD will be like this:



- Now I have those tables:
 - Student: studentID, studentName
 - Enroll: studentID, CourseID, Enrollment Date
 - Course: courseID, courseName
 - Has: courseID, classID
 - Class: classID, className
- **Summary:**
 - + Entity: Student, Course, Class.
 - + Attributes: studentID (Key), studentName, etc
 - + Relation: one to many(mandatory).
 - + Relationship: Enroll, Has.

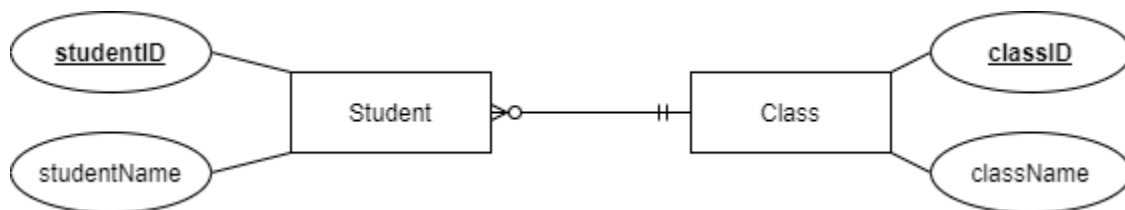
Chapter 2: Relational Diagram

1. Definition:

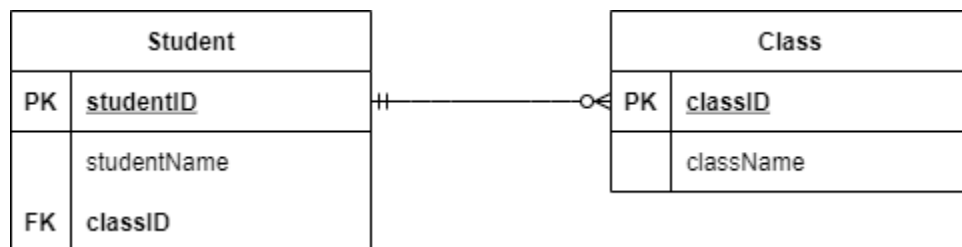
- A relational diagram is a graphical representation of relational databases' entities, attributes that are present in those entities and the relationship among these entities. (**Note:** Base on ERD).

2. Convert ERD to RD:

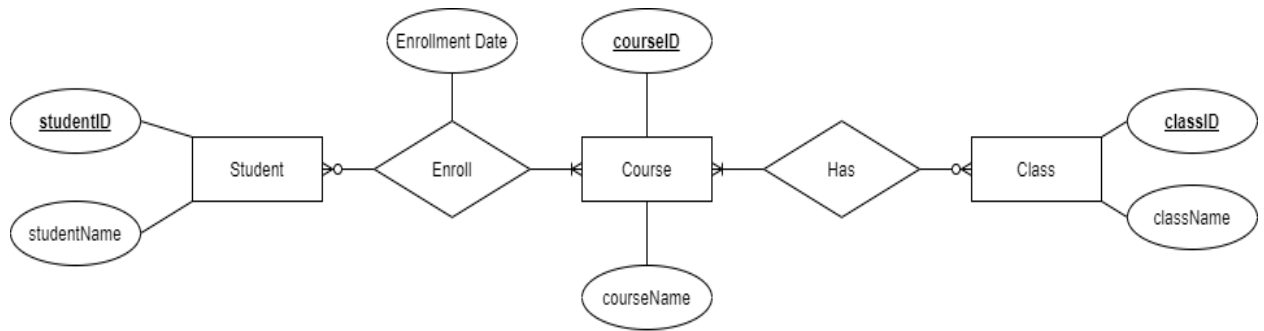
- Back to example 1 above:



- Now, how to convert this ERD to RD?
- Follow these steps:
 - + Convert all Entities and Relationships (many to many) to table.
 - + Foreach table add all attributes of that entity (**Note:** now add even more foreign key).
 - + With the relation, just reverse all relation in ERD.
- Ex: $\text{Student} \text{---} \text{Class}$ To $\text{Student} \text{---} \text{Class}$



- Continues with example 2:



- Similarly, which those steps above:

