Entity Relationship Model (ER – Model)

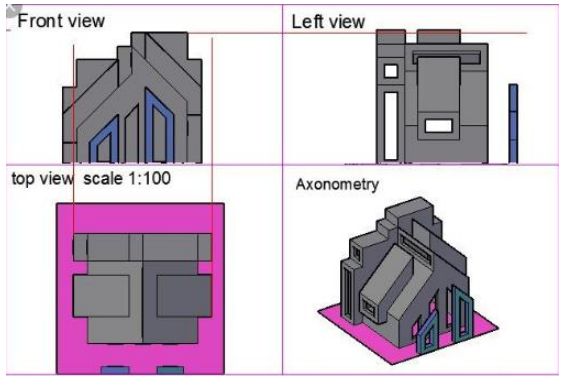
Outline

- Basic concept of E-R diagram
- Types of Attributes
- Mapping Cardinality
- Weak Entity Sets
- Extended E-R features
- Generalization and Specialization
- Constraints on Specialization and Generalization
- Aggregation
- E-R diagram of Hospital Management System
- Reduction to E-R Database Schema
- Database Models
- Integrity Constraints

Design phase

High Level view

we use diagram to explain what are the things in your database



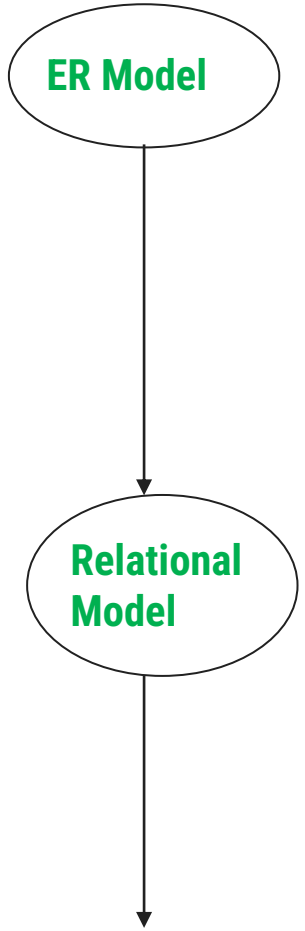
Representational view

Use relations(table)

Construction where the individual items going to fit(**Implementation**)

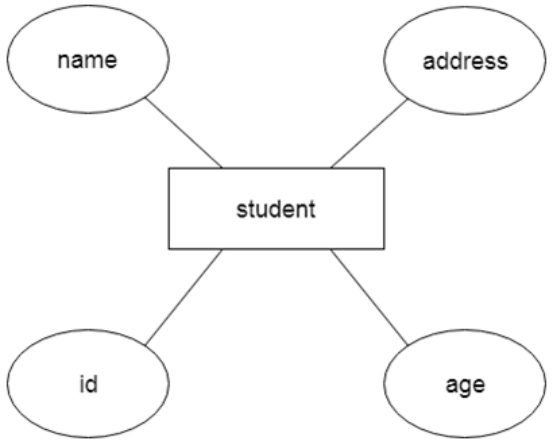
Low Level view

How we are going to store the table in HDD what structure is required, how much byte is required, how much byte is skipped for accessing records



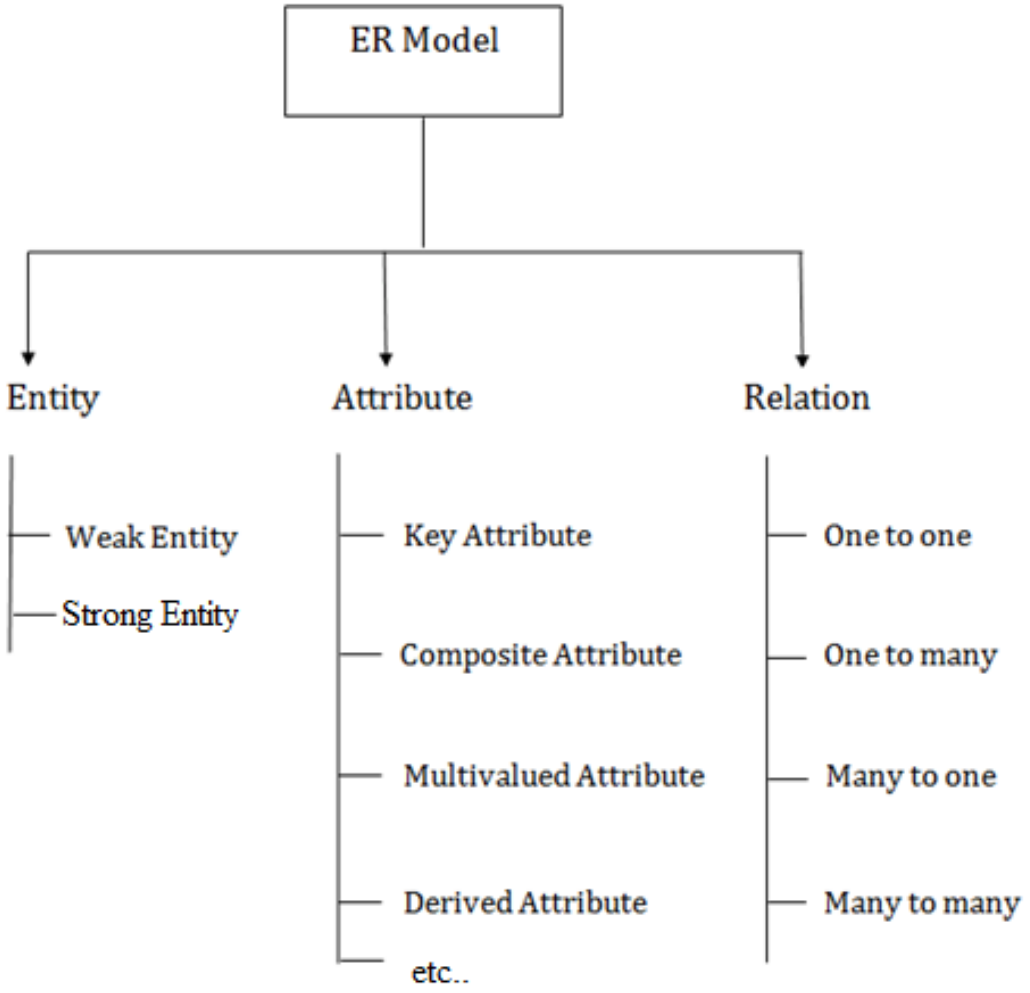
Basic concepts

- What is Database Design?
  - Database Design is a collection of processes that facilitate the **designing**, **development**, **implementation** and **maintenance** of enterprise database management systems.
- What is E-R diagram?
  - E-R diagram: (Entity-Relationship diagram)
  - It is a high-level data model or conceptual model that gives the **graphical (pictorial) representation** of the logical structure of the database.
  - It shows all the constraints and relationships that exist among the different components.
  - It uses different types of symbols to represent different objects of database.
  - **For example,** Suppose we design a school database. In this database, the student will be an entity with attributes like address, name, id, age, etc. The address can be another entity with attributes like city, street name, pin code, etc and there will be a relationship between them.



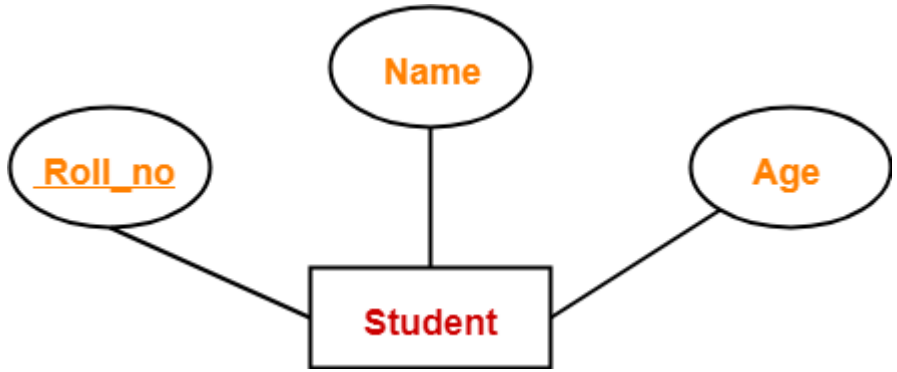
Components of ER diagram

- An ER diagram is mainly composed of following three components-
  1. Entity
  2. Attributes
  3. Relationship



Consider the following Student table-

Roll_no	Name	Age
1	Akshay	20
2	Rahul	19
3	Pooja	20
4	Aarti	19



This complete table is referred to as “Student Entity Set” and each row represents an “entity”.	Here, • Roll_no is a primary key that can identify each entity uniquely. • Thus, by using student’s roll number, a student can be identified uniquely.

Project

1. Collect the information(verbal description from client)→diagram
2. Noun → Entity{ a person owns a car : person}
3. Verb → relationship{ a employee works for department}, any verb going to describe is relationship
4. Noun describes noun → attribute {property of the entity}

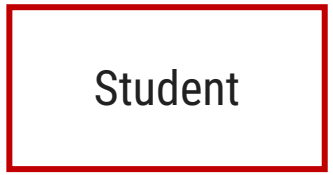
Employee works for Department:

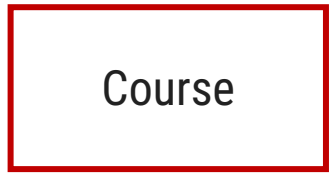
Person, employee, Department	← (thing)	→Entity
Name, age, address phone	← (properties)	→Attribute
Works_for, owns {verbs}	←(association)	→ Relationship

1(a). Entity

- An entity is a **person**, a **place** or an **object**.
- An entity is represented by a **rectangle** which contains the name of an entity.
- Entities of a college database are:
  - Student
  - Professor/Faculty
  - Course
  - Department
  - Result
  - Class
  - Subject

Symbol





Exercise Write down the different entities of bank database.

Exercise Write down the different entities of hospital database.

1(b) Entity Set

- It is a **set (group) of entities** of **same type**.
- Examples:
  - All persons having an account in a bank
  - All the students studying in a college
  - All the professors working in a college
  - Set of all accounts in a bank

76766	Crick
45565	Katz
10101	Srinivasan
98345	Kim
76543	Singh
22222	Einstein

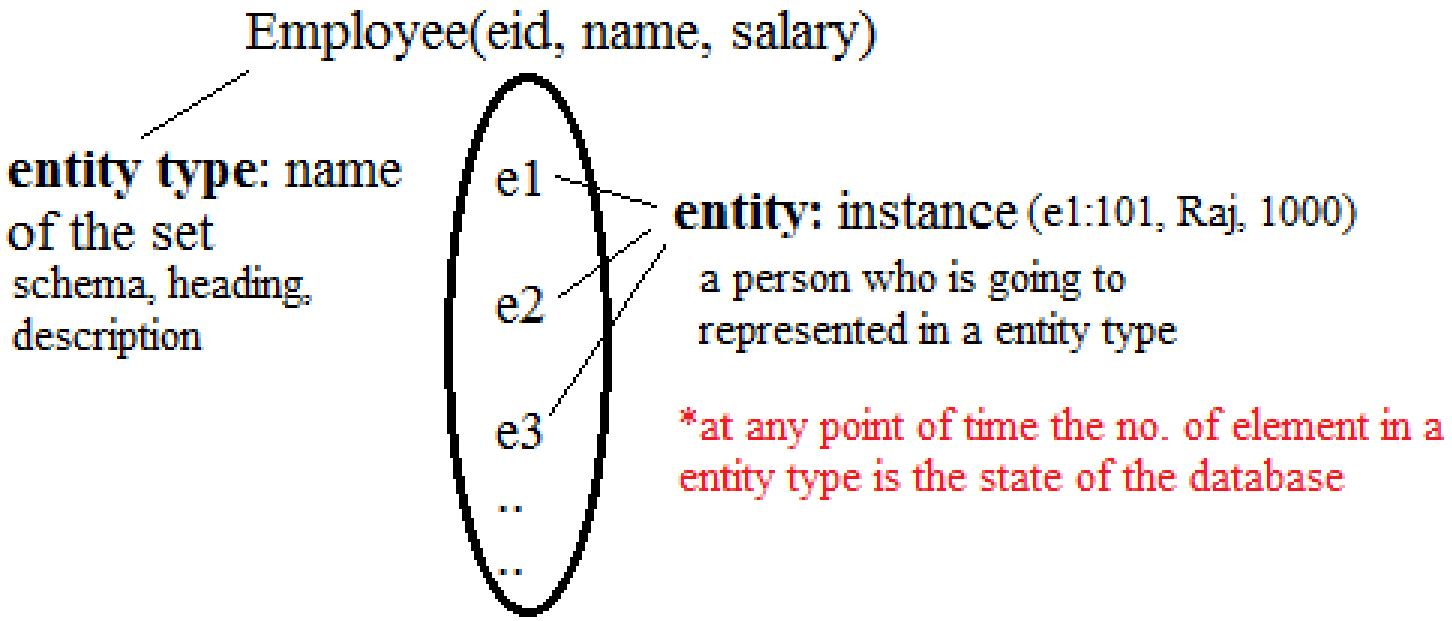
instructor

98988	Tanaka
12345	Shankar
00128	Zhang
76543	Brown
76653	Aoi
23121	Chavez
44553	Peltier

student

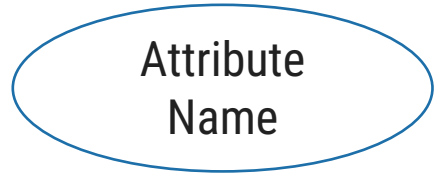
Example

- An **entity** is an object that exists and is distinguishable from other objects.
  - Example: specific person, company, event, plant
- An **entity type or set** is a set of entities of the same type that share the same properties.
  - Example: set of all persons, companies, trees, holidays

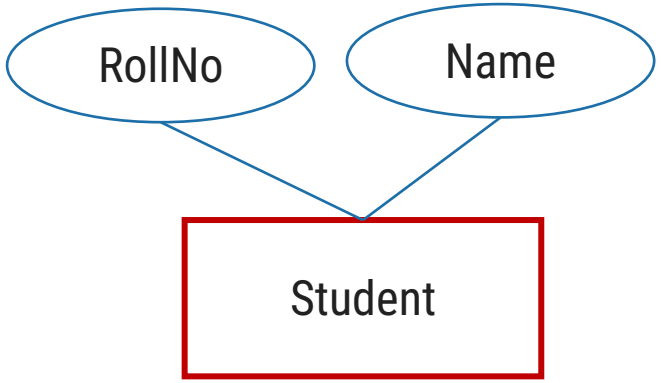


2. Attributes

- Attribute is **properties** or details about an entity.
- An attribute is represented by an **oval** containing name of an attribute.
- Attributes of Student are:
  - Roll No
  - Student Name
  - Branch
  - Semester
  - Address
  - Mobile No
  - Age
  - SPI
  - Backlogs



Symbol

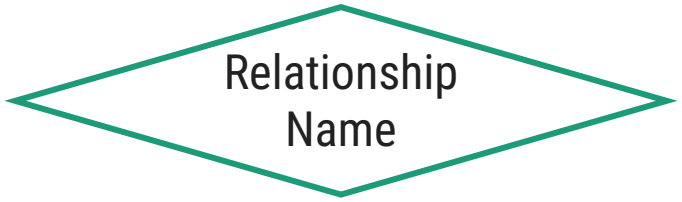


Exercise Write down the different attributes of Faculty entity.

Exercise Write down the different attributes of Account entity.

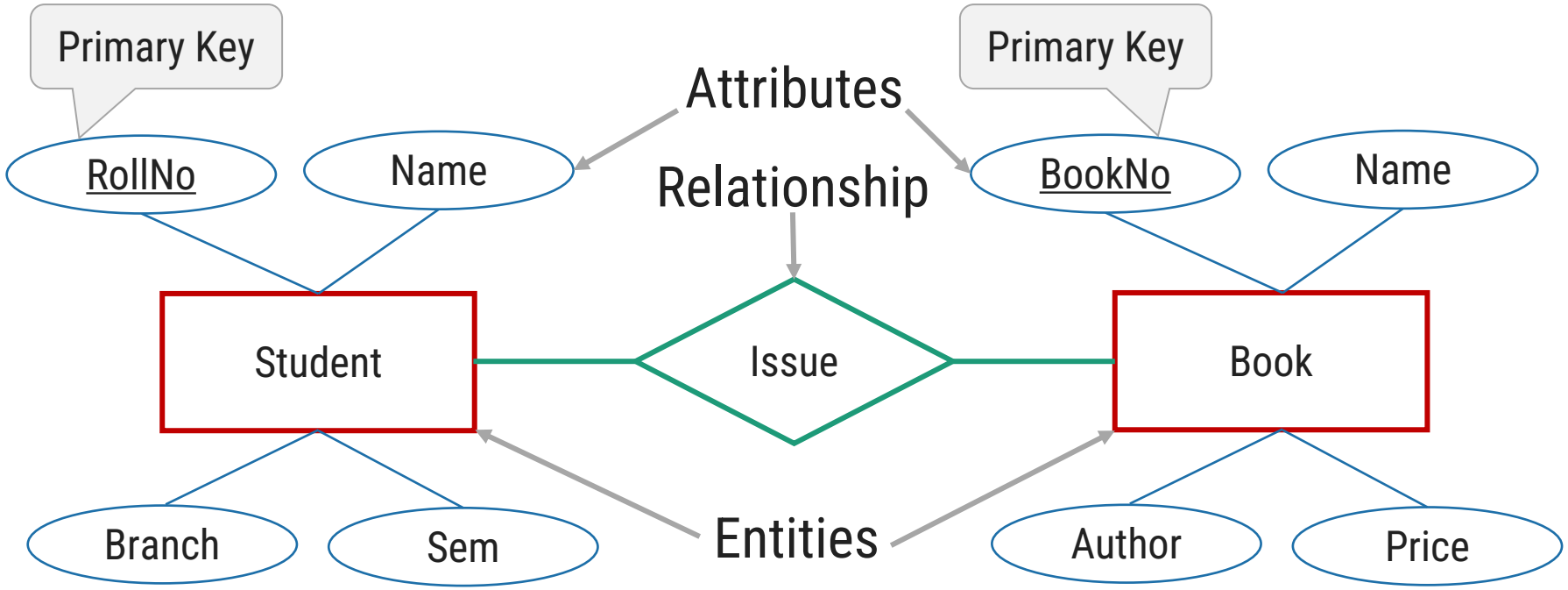
3. Relationship

- Relationship is an **association** (connection) between several entities.
- It should be placed between two entities and a line connecting it to an entity.
- A relationship is represented by a **diamond** containing relationship's name.



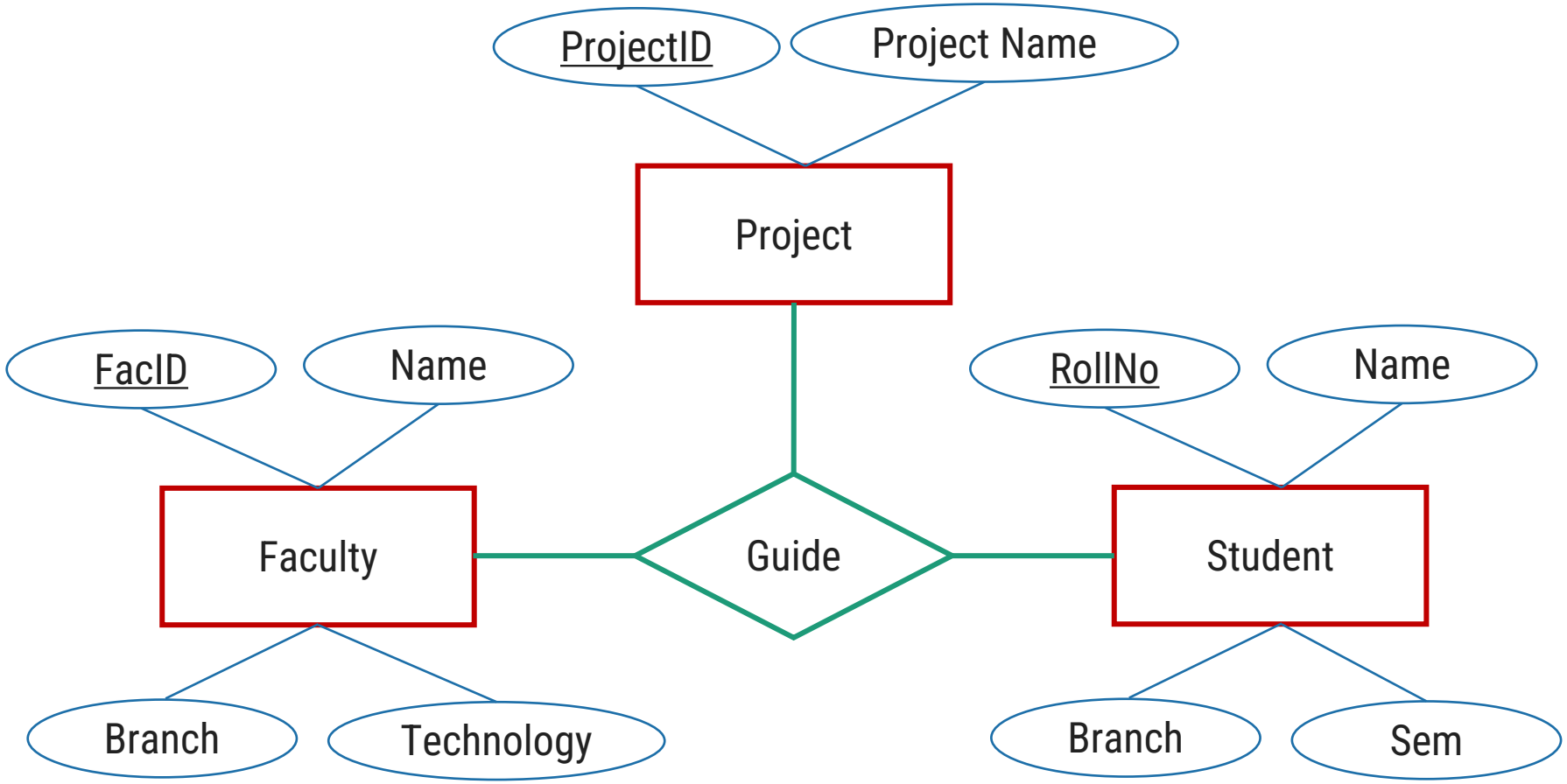
Symbol

E-R Diagram of a Library System



Each and every entity must have one primary key attribute.
Relationship between 2 entities is called binary relationship.

Ternary Relationship

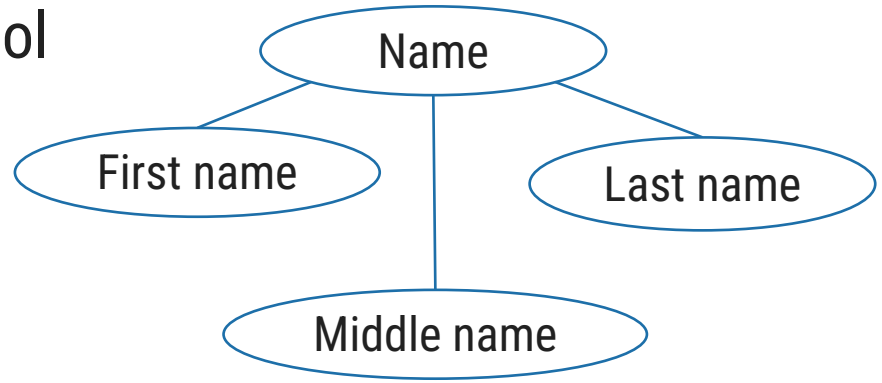


Relationship between 3 entities is called ternary relationship.

Exercise

- Draw an E-R diagram of following pair of entities
  - Customer & Account
  - Customer & Loan
  - Doctor & Patient
  - Student & Project
  - Student & Teacher
    - Note: Take four attributes per entity with one primary key attribute.
      Keep proper relationship between two entities.

Types of Attributes

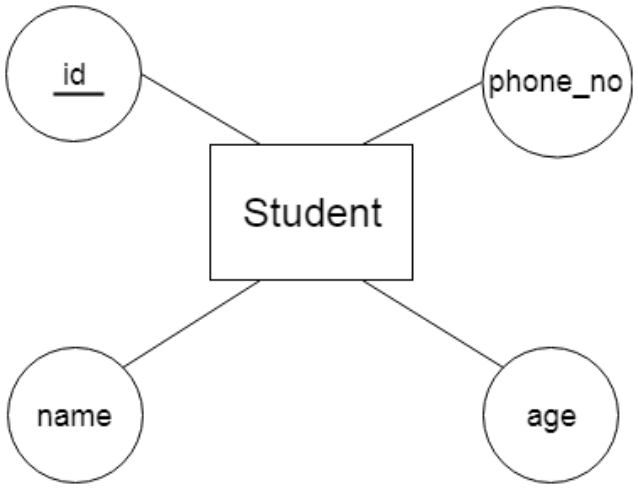
Simple Attribute	Composite Attribute
Cannot be divided into subparts	Can be divided into subparts
E.g. RollNo, CPI	E.g. Name (first name, middle name, last name) Address (street, road, city)
Symbol 	Symbol 

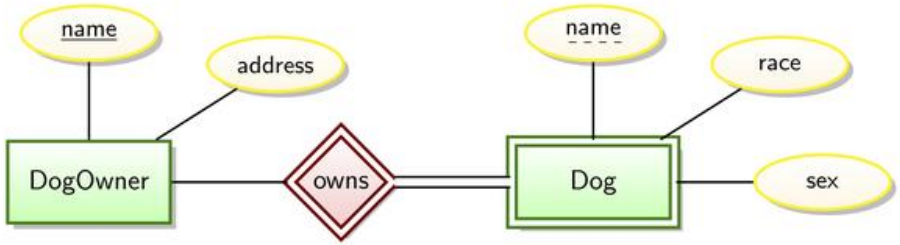
Types of Attributes

Derived Attribute	Multi-valued Attribute
It's value is derived or calculated from other attributes	Has multiple (more than one) value
E.g. Age (can be calculated using current date and birthdate)	E.g. PhoneNo (person may have multiple phone nos) EmailID (person may have multiple emails)
Symbol 	Symbol 

Types of Attributes

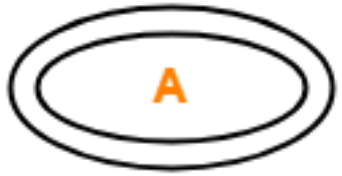
Key Attribute	Partial Attribute or partial key
The key attribute is used to represent the main characteristics of an entity. It represents a primary key. The key attribute is represented by an ellipse with the text underlined. e.g. id	It's one or more attributes that uniquely identify a weak entity for a given owner entity.



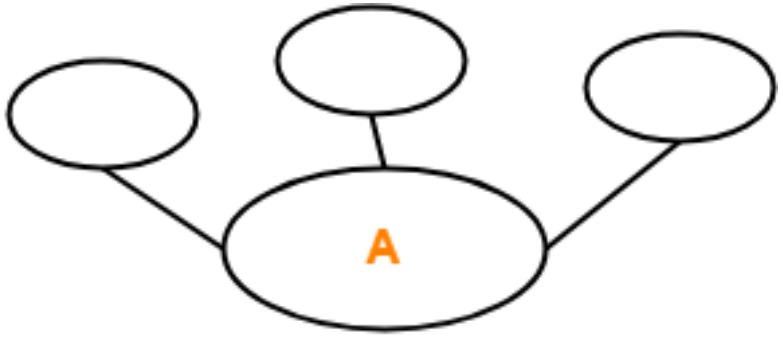


Summary

Attribute

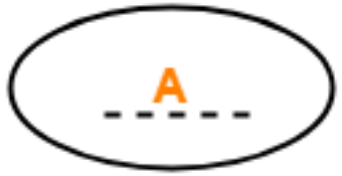


Multivalued Attribute



Composite Attribute

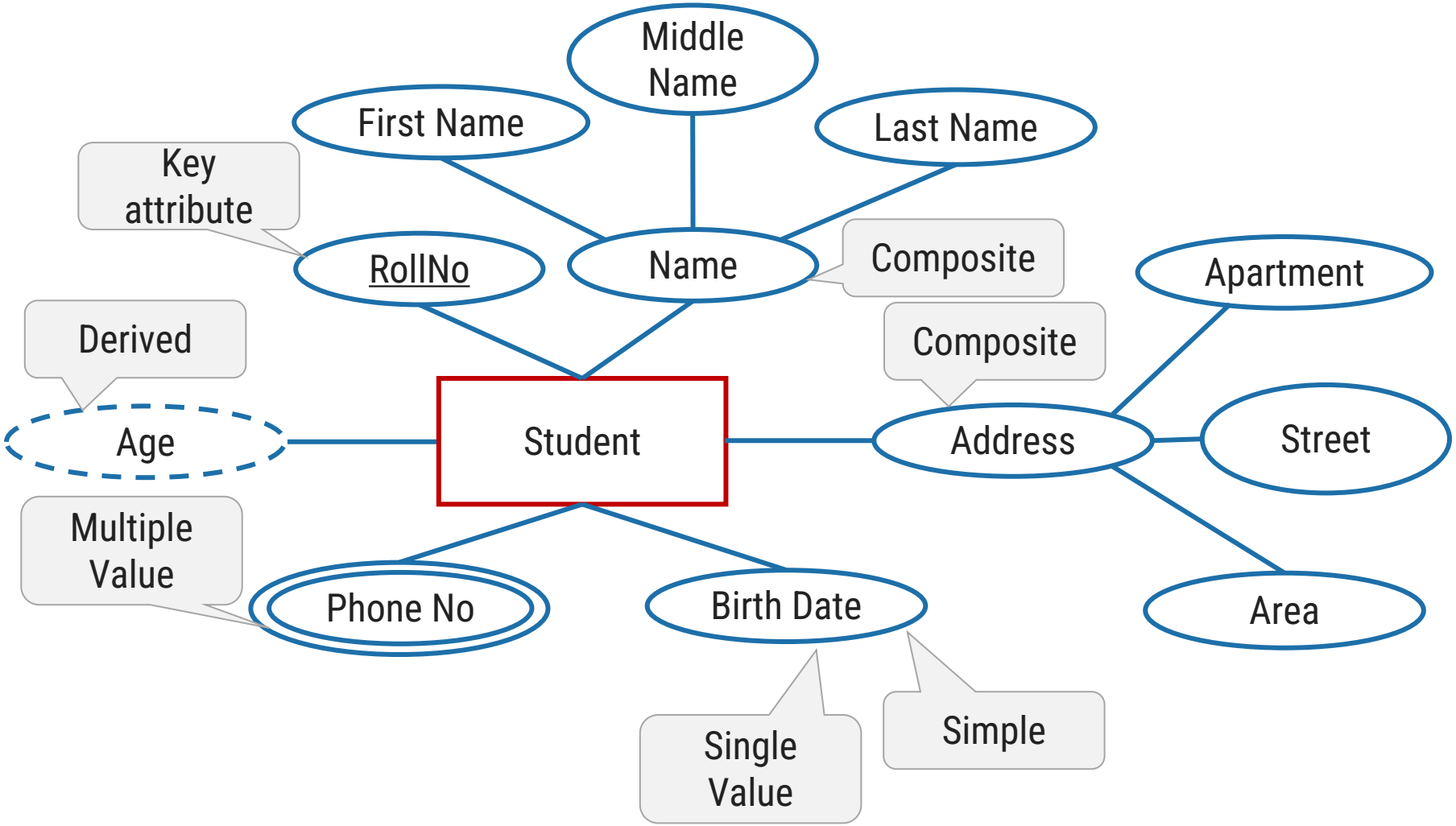
Key Attribute



Partial Attribute

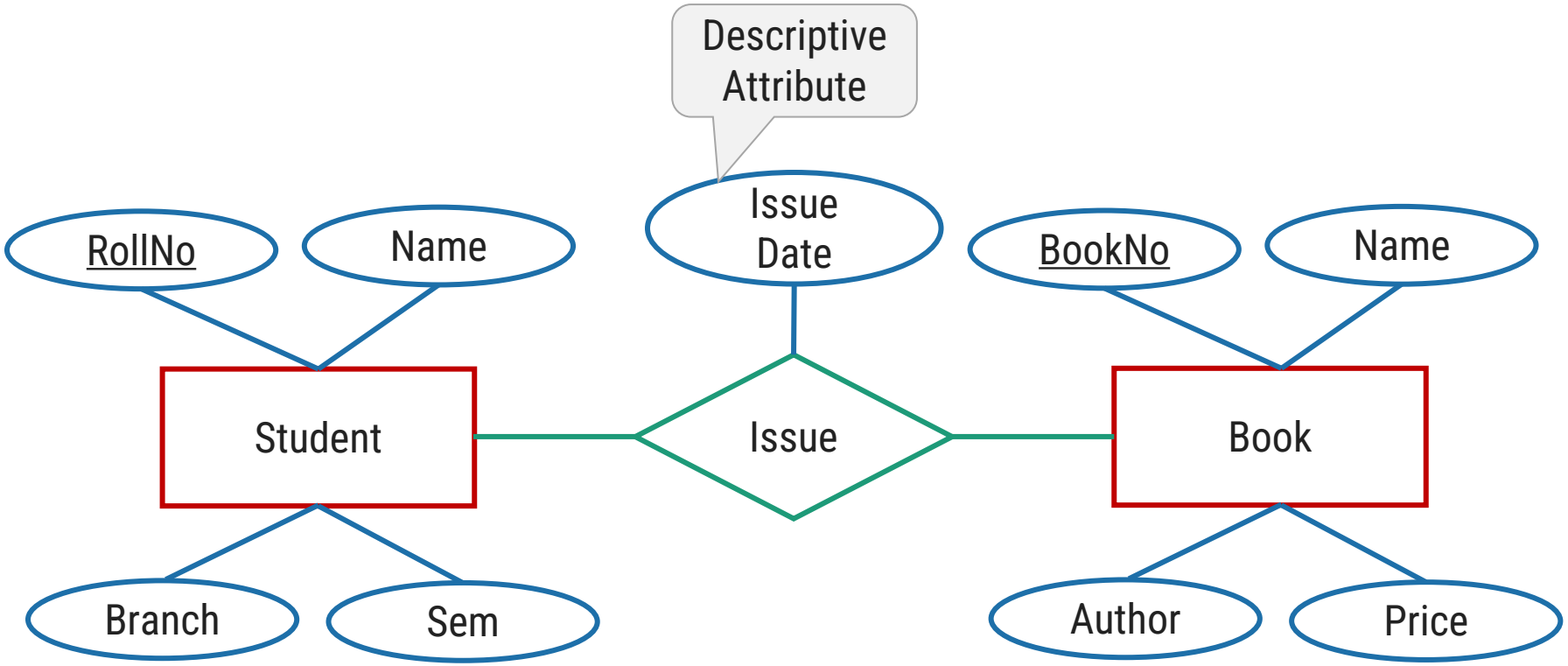
Derived Attribute

Entity with all types of Attributes



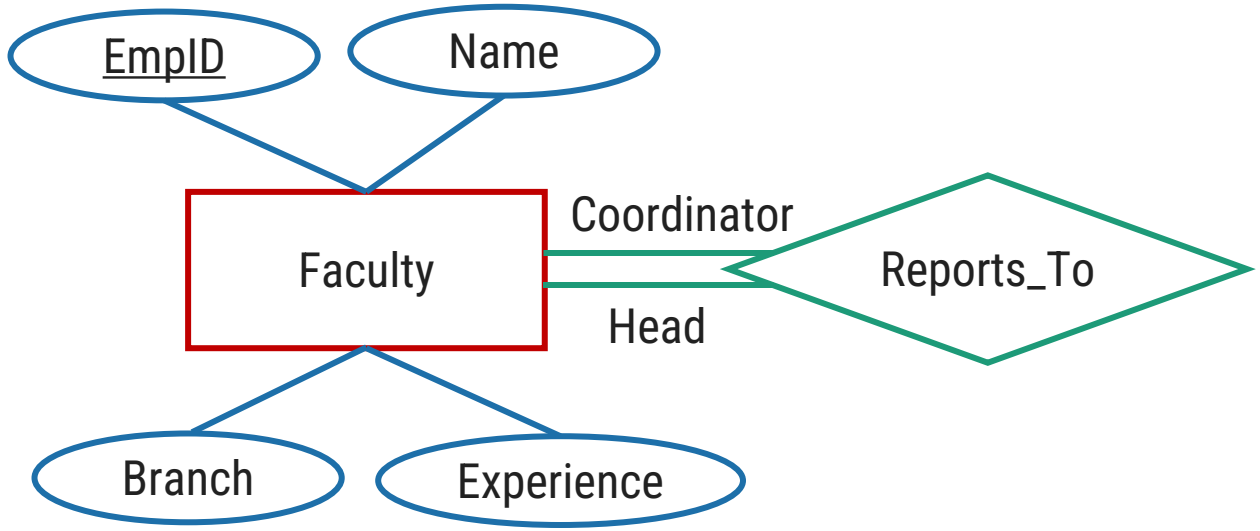
Descriptive Attribute

- **Attributes of the relationship** is called descriptive attribute.



Role

- Roles are indicated by labeling the lines that connect diamonds (relationship) to rectangles (entity).
- The labels “Coordinator” and “Head” are called roles; they specify Faculty entities interact with whom via Reports_To relationship set.
- Role labels are optional, and are used to clarify semantics (meaning) of the relationship.

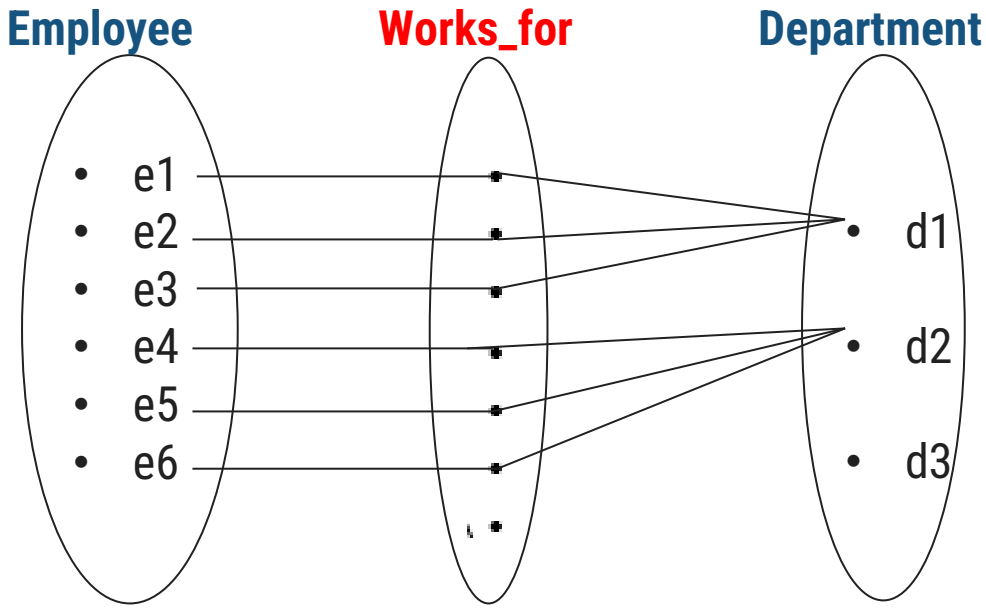


Mapping Cardinality (Cardinality Constraints)

- It represents the **number of entities of another entity set** which are **connected to an entity** using a relationship set.
- It is most **useful in describing binary relationship sets**.
- For a binary relationship set the mapping cardinality must be one of the following types:
  a) One to Many
  b) Many to One
  c) Many to Many

a) One to Many(1:N)

Verbal Description: Every Employee works for exactly one department, and a department can have many employee, new department need not have a employee. **(RA phase)**



Degree: 2
Max card: e(1), d(N)
Min card: e(1) ,d(0)

Every relation in relation set is actually an association between employee set and department set

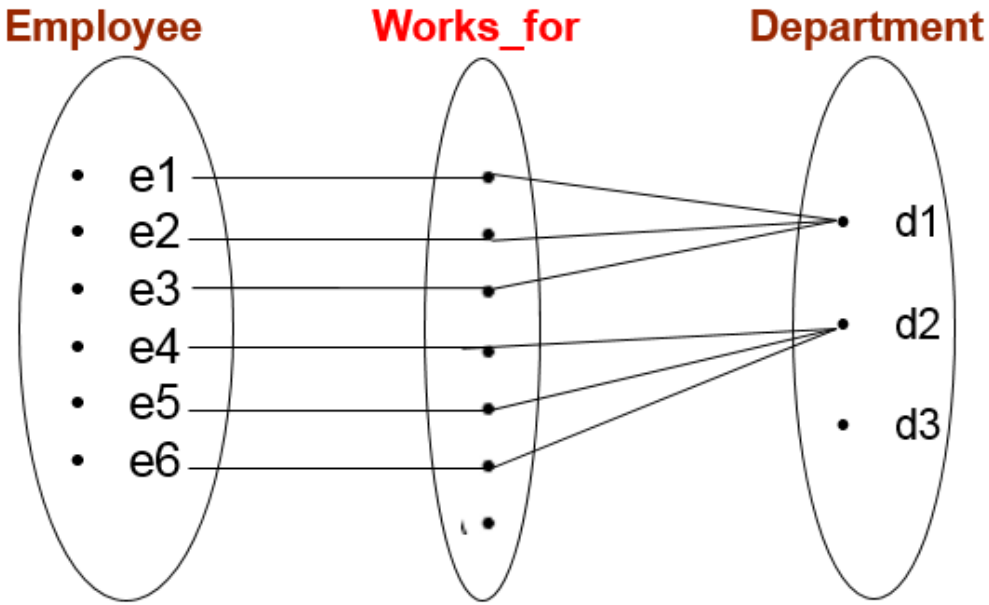
Degree: In relationship how many entity set are participating

Cardinality ratio(max card): Maximum no of relationship a entity can participate

Participation(min card): Minimum no of relationship in each an entity can participate

***Every relationship in a relationship set is an association between employee and department**
***we have to make ER diagram as readable as possible**
***we can also take department first: a department employs an employee**

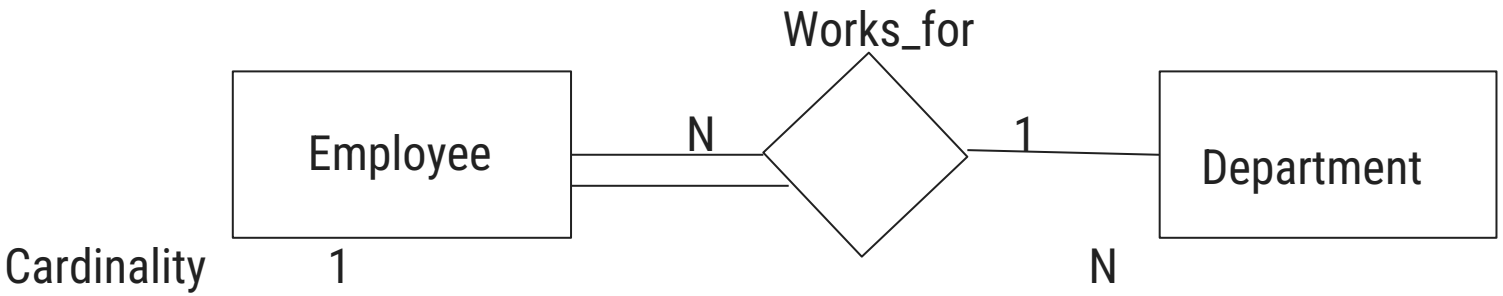
Verbal Description: Every Employee works for exactly one department, and a department can have many employee, new department need not have a employee. **(RA phase)**



Participation

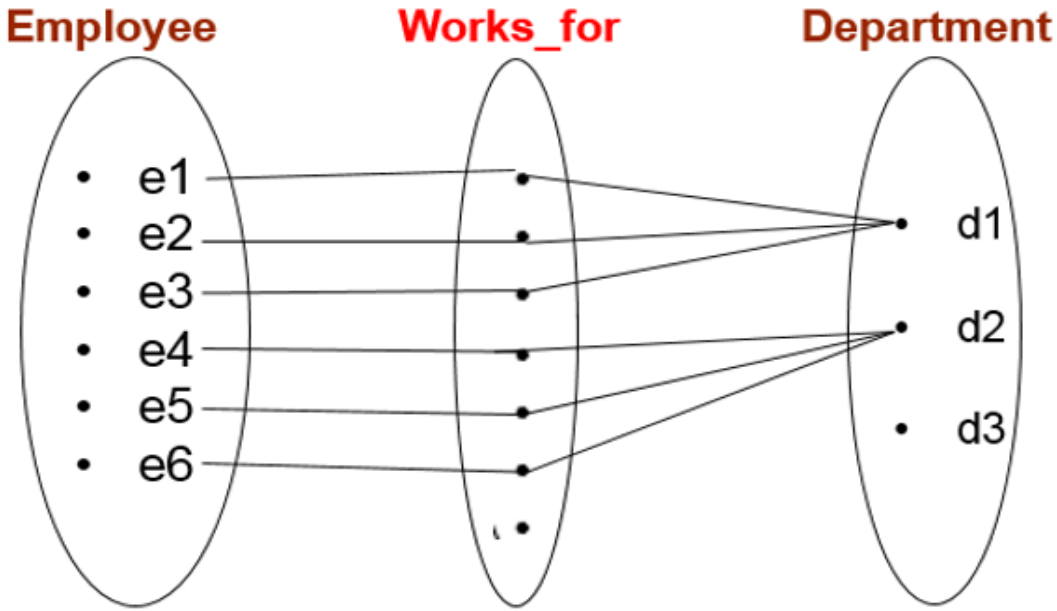
Total participation using double line(min. card=1)

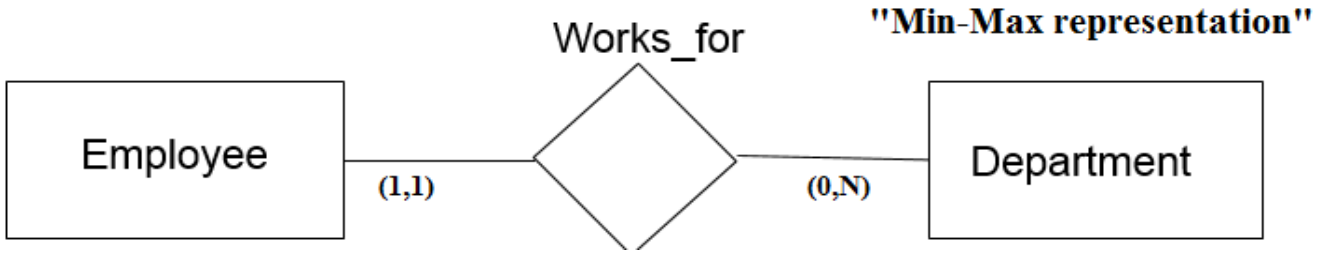
partial participation using single line(min. card=0)



Cardinality ratio/single line-double line

Min Max Representation



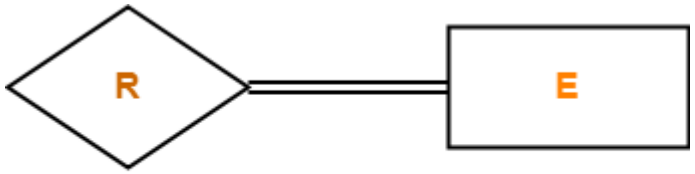


What if ? (2,N) (0,N)

Participation

1. Total Participation- :

- It specifies that each entity in the entity set must compulsorily participate in at least one relationship instance in that relationship set.
- That is why, it is also called as **mandatory participation.**
- Total participation is represented using a double line between the entity set and relationship set.



Total Participation

- Here,
  - Double line between the entity set “Student” and relationship set “Enrolled in” signifies total participation.
  - It specifies that each student must be enrolled in at least one course.

Participation

2. Partial Participation-

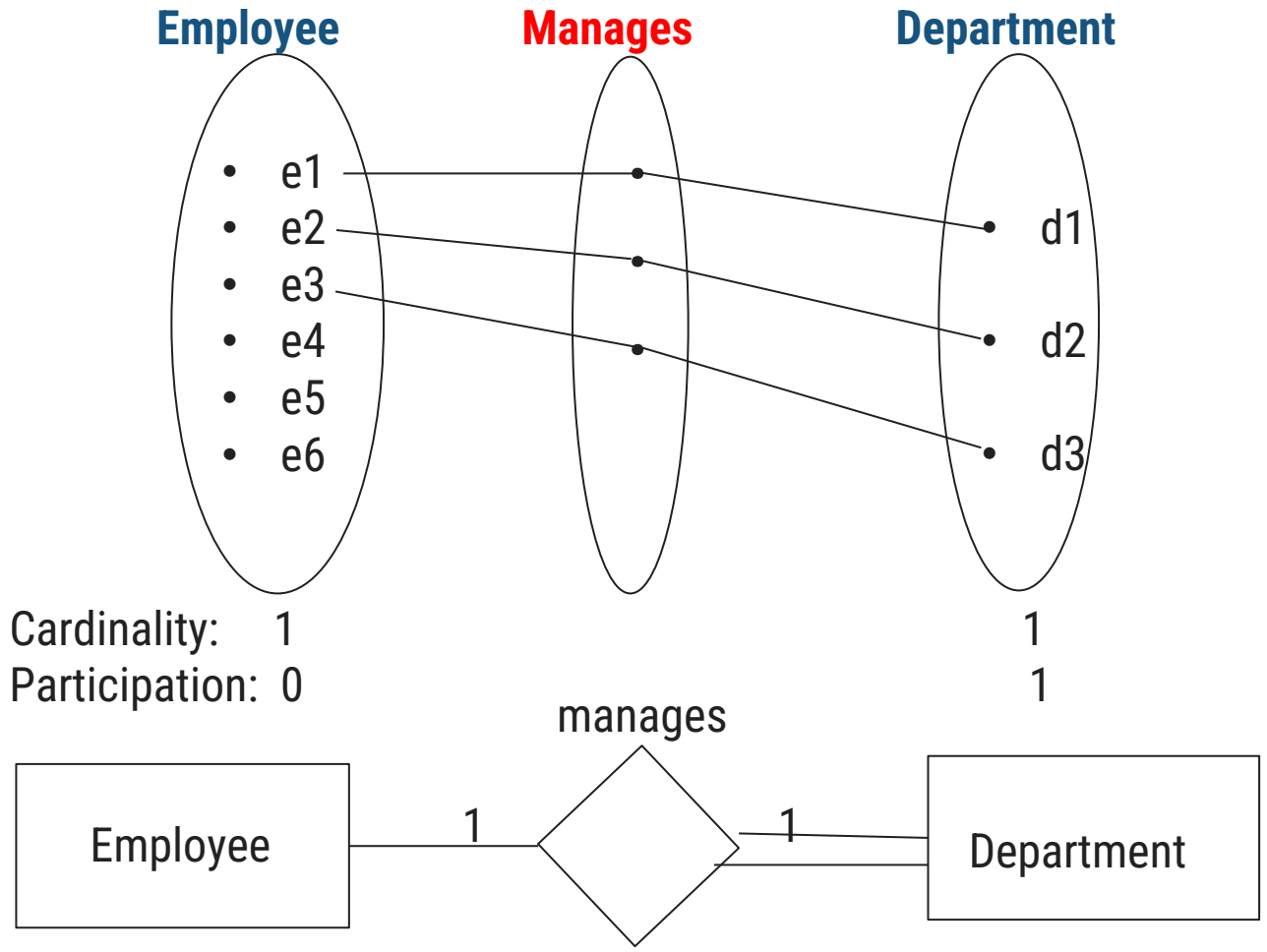
- It specifies that each entity in the entity set may or may not participate in the relationship instance in that relationship set.
- That is why, it is also called as **optional participation.**
- Partial participation is represented using a single line between the entity set and relationship set.

Here,

- ✓ Single line between the entity set “Course” and relationship set “Enrolled in” signifies partial participation.
- ✓ It specifies that there might exist some courses for which no enrolment's are made.
- ✓ Minimum cardinality tells whether the participation is partial or total.
  - ✓ If minimum cardinality = 0, then it signifies partial participation.
  - ✓ If minimum cardinality = 1, then it signifies total participation.
- ✓ Maximum cardinality tells the maximum number of entities that participates in a relationship set.

One to One(1:1)

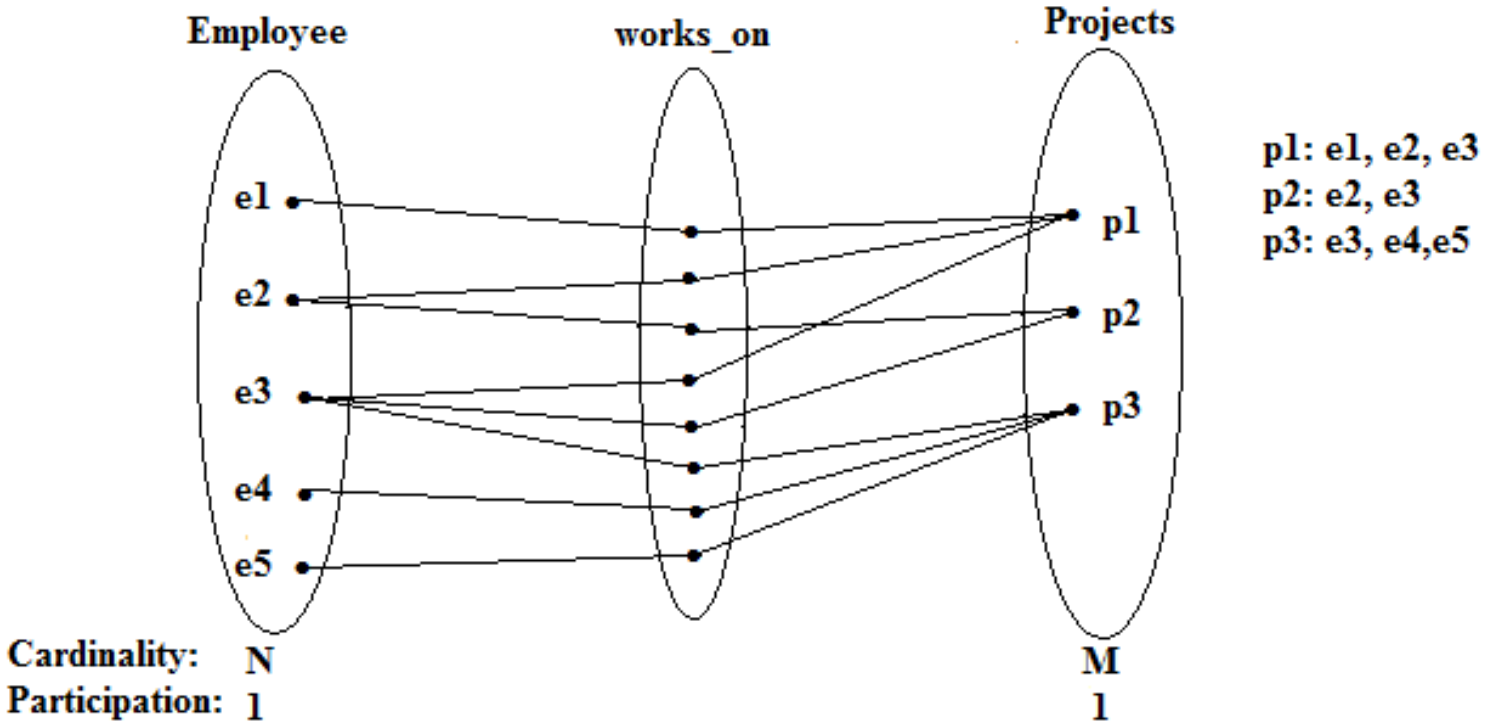
- **RA**: Every department should have a manager and only one employee manages department and an employee manages only one department.

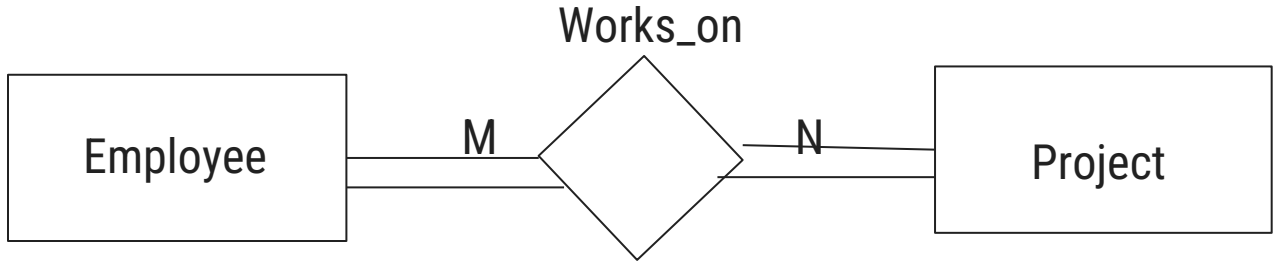


Total participation: double line
Partial participation: single line

Many to many(M:N)

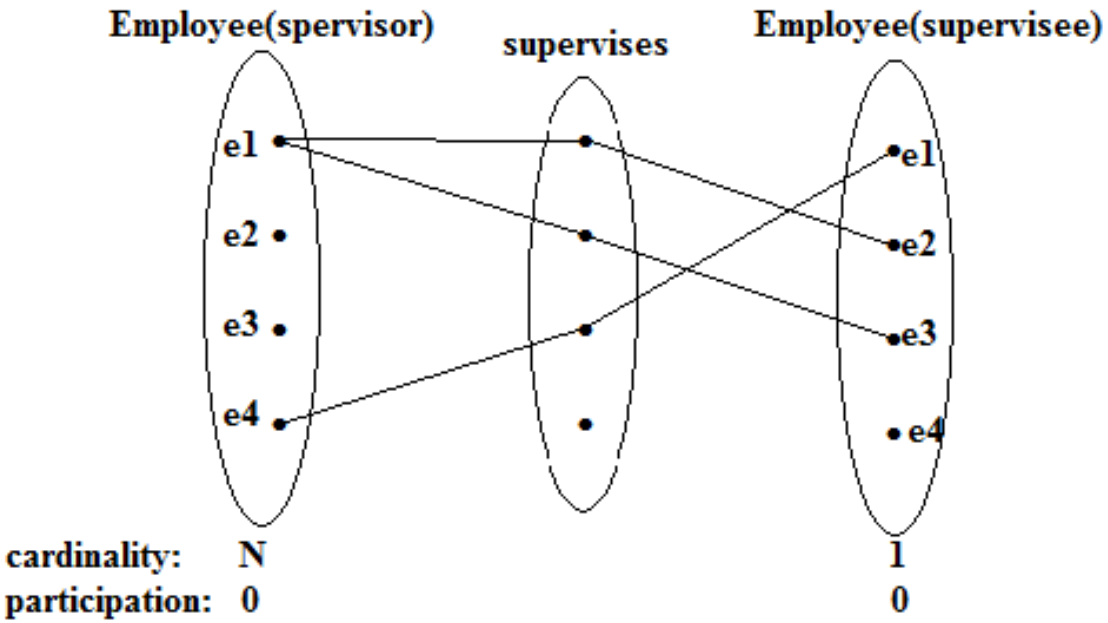
RA: Every employee supposed to work on atleast one project(he can work on many projects) As well as every project supposed to have many employees and it is supposed to have atleast one employee.

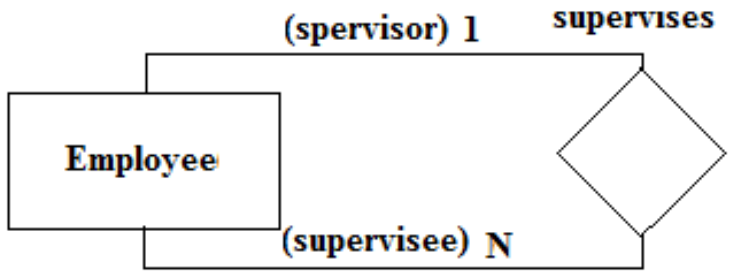




Recursive relationship

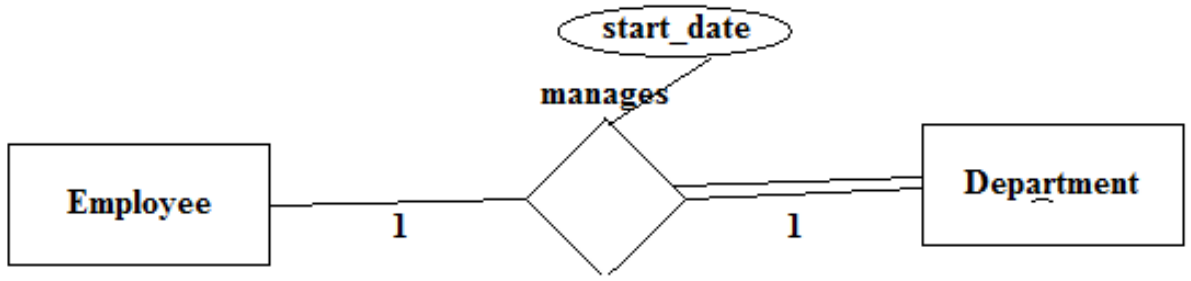
RA : Every employee suppose to report only one employee(boss or supervisor), an employee can supervise one or more employee. It is not necessary that every employee has to manage someone.





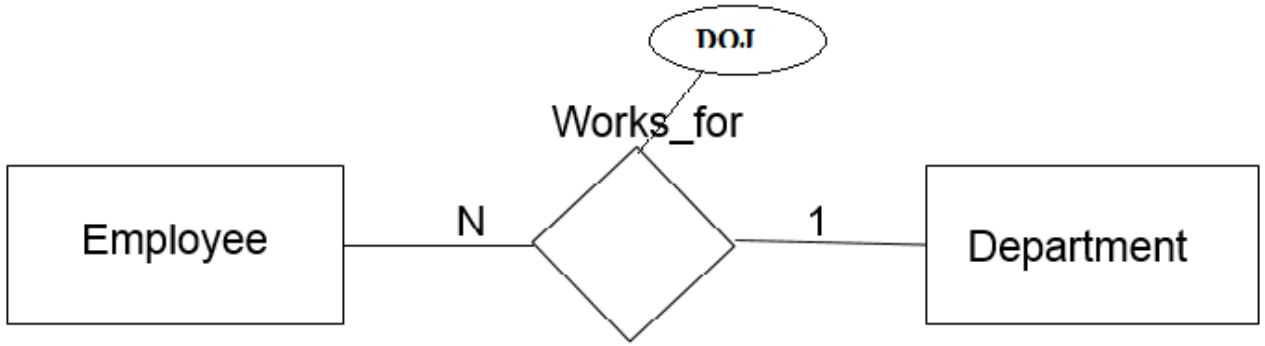
Degree=2

Attribute to relation

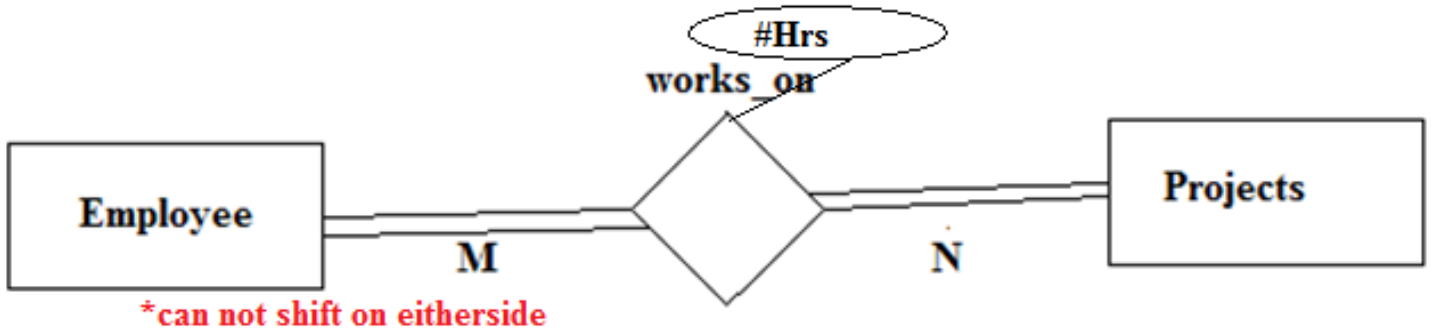


* shift start_date to either side

* It depends on the designer whether to shift the attribute or keep it separate relation



* DOJ must move towords N side

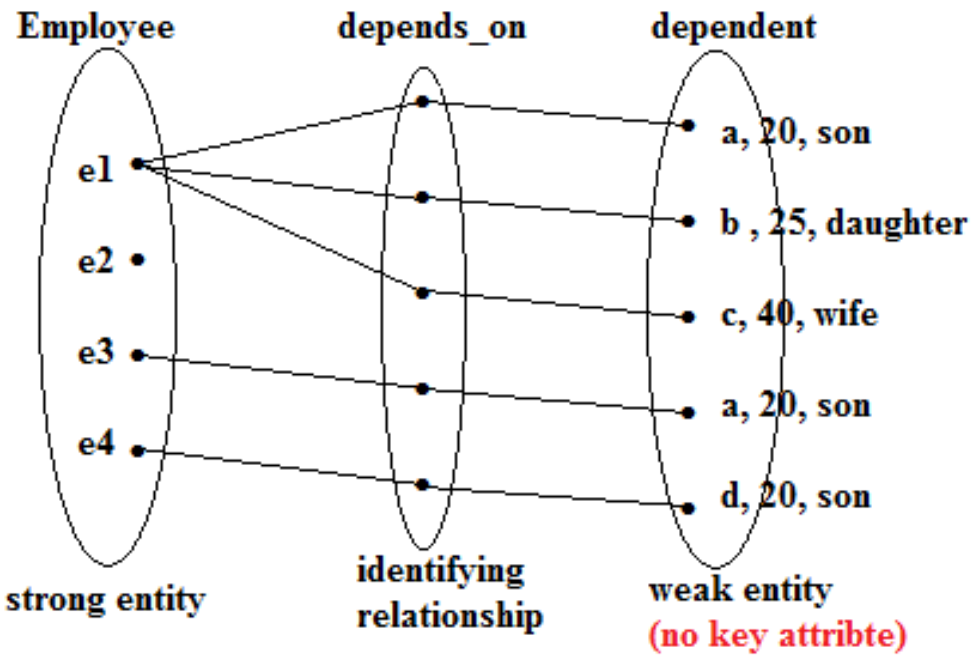


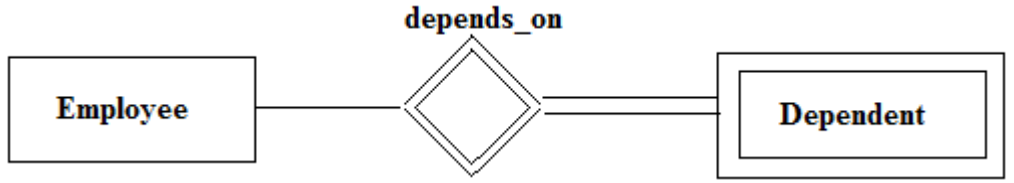
*can not shift on eitherside

Weak Entity

On the basis of requirement analysis we identified the key attribute, some times we might not have a key attribute at all.

Example: Every dependent is associated with one employee, but some employee may not have any dependent.

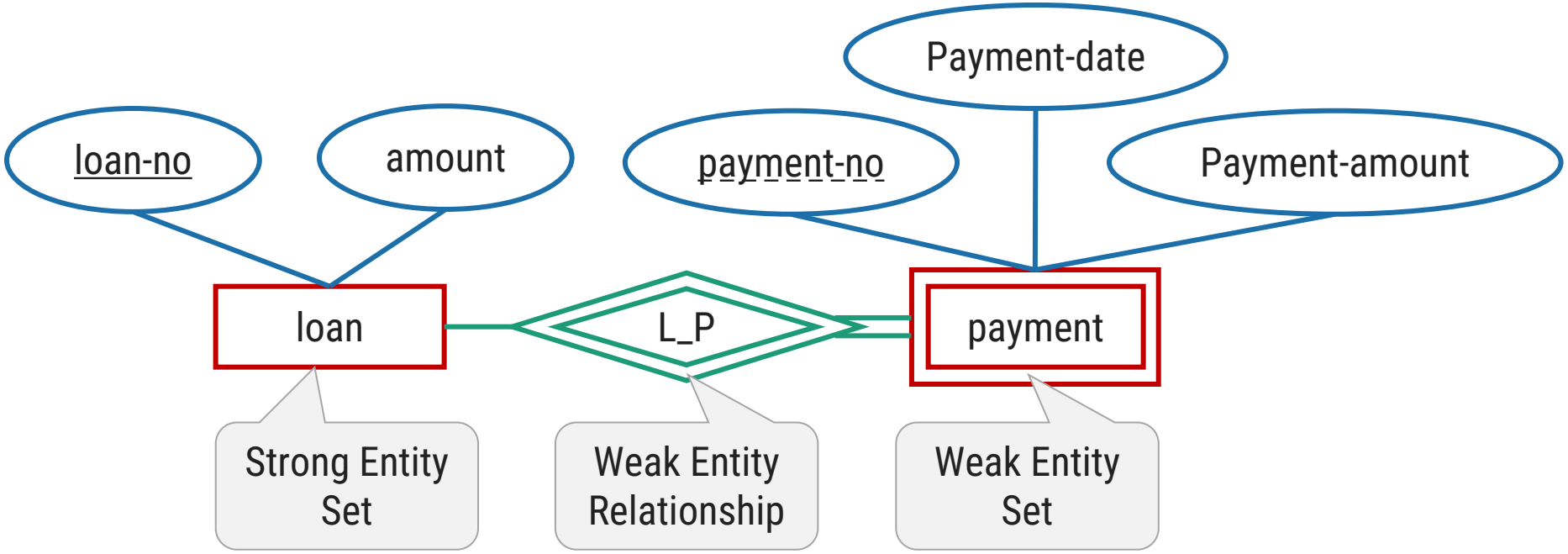




Weak entity is always in total Participation with strong entity

1. Without strong entity weak entity does not exist
2. We can identity the weak entity distinctly by taking the key attribute from strong entity

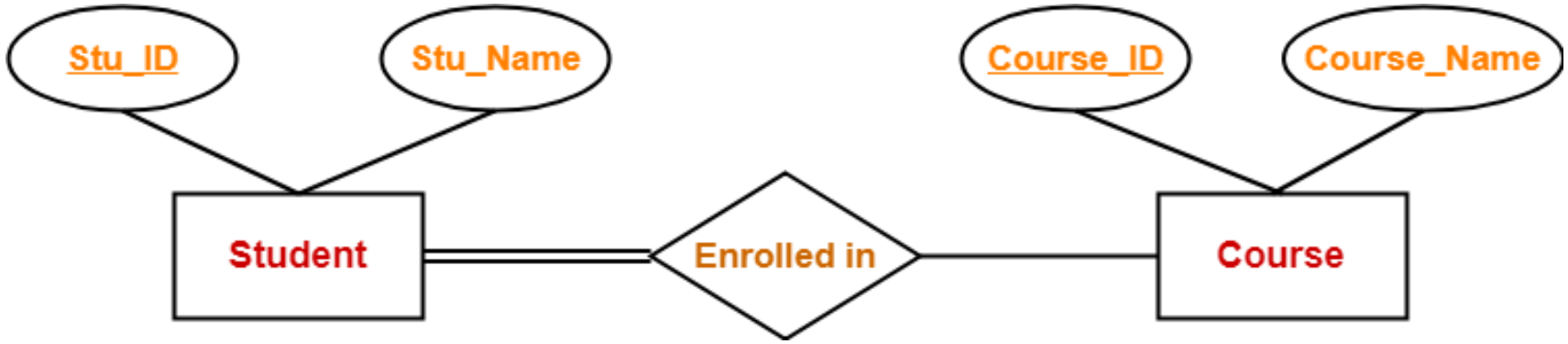
- An **entity set that does not have a primary key** is called weak entity set.



- Weak entity set is indicated by double rectangle.
- Weak entity relationship set is indicated by double diamond.

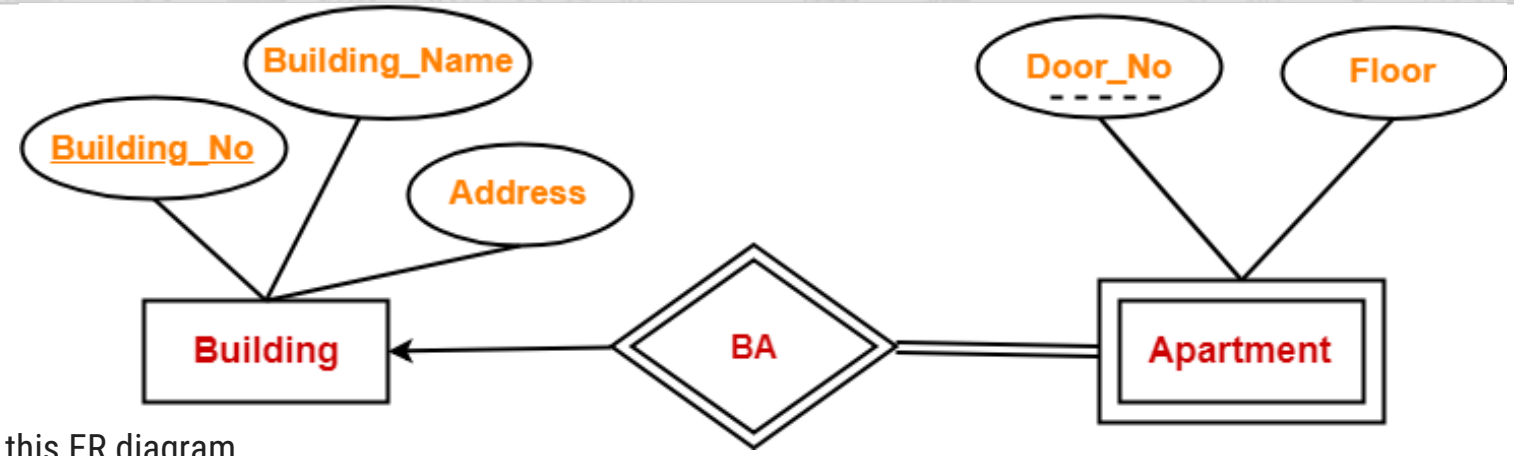
- The **existence of a weak entity set** depends on the **existence of a strong entity set**.
- The **discriminator (partial key)** of a weak entity set is the set of **attributes that distinguishes all the entities** of a weak entity set.
- The **primary key** of a weak entity set is created by **combining the primary key of the strong entity set** on which the weak entity set is existence dependent and the **weak entity set's discriminator**.
- We underline the discriminator attribute of a weak entity set with a **dashed line**.
- Payment entity has payment-no which is discriminator.
- Loan entity has loan-no as primary key.
- So primary key for payment is **(loan-no, payment-no).**

Example



In this ER diagram,

- Two strong entity sets "**Student**" and "**Course**" are related to each other.
- Student ID and Student name are the attributes of entity set "Student".
- Student ID is the primary key using which any student can be identified uniquely.
- Course ID and Course name are the attributes of entity set "Course".
- Course ID is the primary key using which any course can be identified uniquely.
- Double line between Student and relationship set signifies total participation.
- It suggests that each student must be enrolled in at least one course.
- Single line between Course and relationship set signifies partial participation.
- It suggests that there might exist some courses for which no enrollments are made.



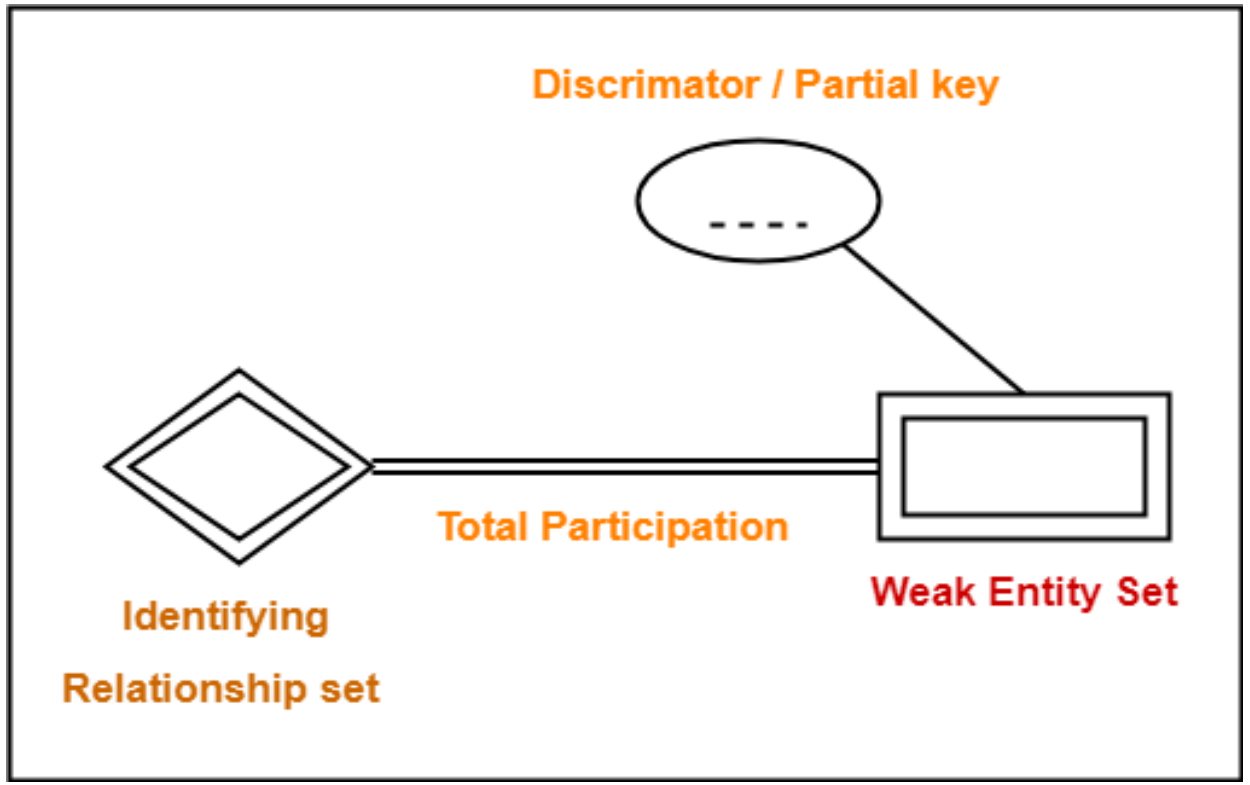
In this ER diagram,

- One strong entity set "**Building**" and one weak entity set "**Apartment**" are related to each other.
- Strong entity set "Building" has building number as its primary key.
- Door number is the discriminator of the weak entity set "Apartment".
- This is because door number alone can not identify an apartment uniquely as there may be several other buildings having the same door number.
- Double line between Apartment and relationship set signifies total participation.
- It suggests that each apartment must be present in at least one building.
- Single line between Building and relationship set signifies partial participation.
- It suggests that there might exist some buildings which has no apartment.
- To uniquely identify any apartment,
  - First, building number is required to identify the particular building.
  - Secondly, door number of the apartment is required to uniquely identify the apartment.

  - Primary key of Apartment = Primary key of Building + Its own discriminator
    = Building number + Door number

Important point

In ER diagram, weak entity set is always present in total participation with the identifying relationship set.



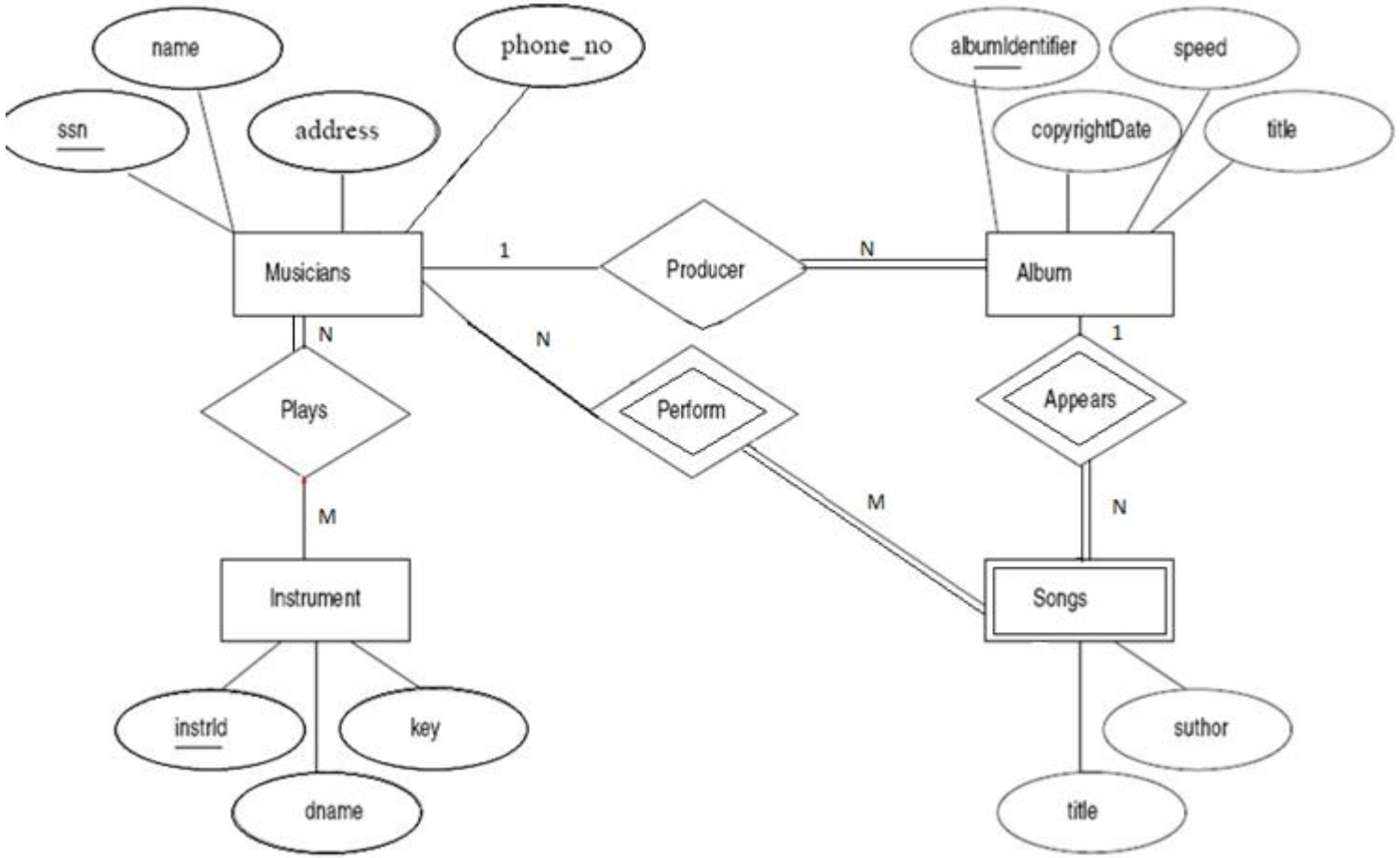
Strong vs Weak Entity

Strong entity set	Weak entity set
A single rectangle is used for the representation of a strong entity set.	A double rectangle is used for the representation of a weak entity set.
It contains sufficient attributes to form its primary key.	It does not contain sufficient attributes to form its primary key.
A diamond symbol is used for the representation of the relationship that exists between the two strong entity sets.	A double diamond symbol is used for the representation of the identifying relationship that exists between the strong and weak entity set.
A single line is used for the representation of the connection between the strong entity set and the relationship.	A double line is used for the representation of the connection between the weak entity set and the relationship set.
Total participation may or may not exist in the relationship.	Total participation always exists in the identifying relationship.

Exercise 1

- Draw an ER diagram for Sony Music Company, which store information about musicians who perform on its albums (as well as other company data) in a database.
  - Each musician that records at SonyMusic has an SSN, a name, an address, and a phone number.
  - Each instrument used in songs recorded at SonyMusic has a unique identification number, a name (e.g., guitar, synthesizer, flute) and a musical key (e.g., C, B-flat, E-flat).
  - Each album recorded on the SonyMusic label has a unique album identifier, a title, a copyright date, a format (e.g., CD or MC).
  - Each song recorded at SonyMusic has a title and an author.
  - Each musician may play several instruments, and a given instrument may be played by several musicians.
  - Each album has a number of songs on it, but no song may appear on more than one album.
  - Each song is performed by one or more musicians, and a musician may perform a number of songs.
  - Each album has exactly one musician who acts as its producer. A musician may produce several albums, of course.

Solution


name
phone_no
albumIdentifier
speed
ssn
address
copyrightDate
title
Musicians
1
Producer
N
Album
N
N
1
Plays
Perform
Appears
M
M
N
Instrument
Songs
instrId
key
suthor
dname
title

Exercise 2

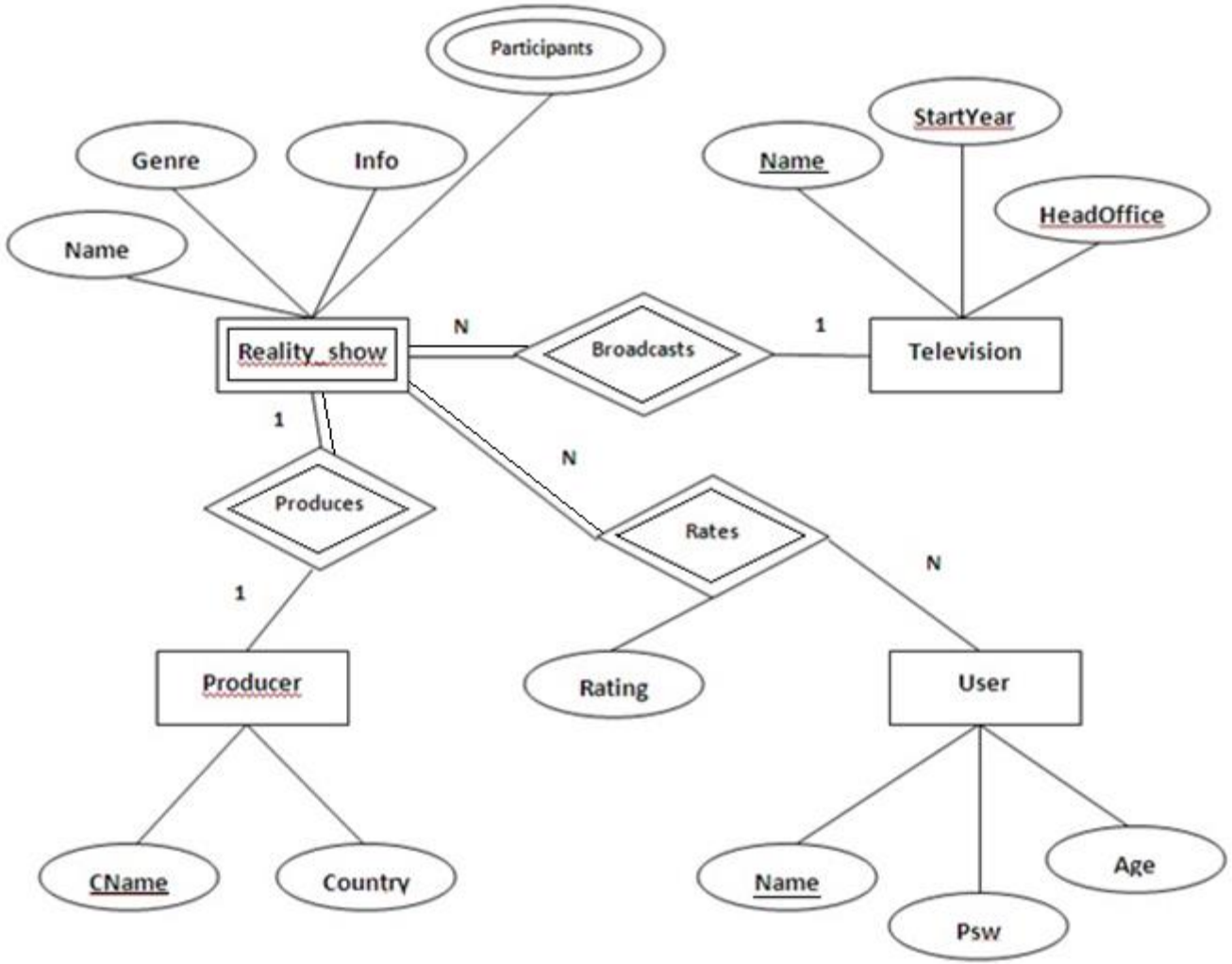
- **Draw an ER diagram for the given scenario;**

- Suppose that you are designing a schema to record information about reality shows on TV. Your database needs to record the following information:

- For each reality show, its name, genre, basic-info and participants name. Any reality show has at least two or more participants.
- For each producer, the company name, company country. A show is produced by exactly one producer. And one producer produces exactly one show.
- For each television, its name, start year, head office. A television may broadcasts multiple shows. Each show is broadcasted by exactly one television.
- For each user, his/her username, password, and age. A user may rate multiple shows, and a show may be rated by multiple users. Each rating has a score of 0 to 10.

- Draw an entity relationship diagram for this database.

Solution

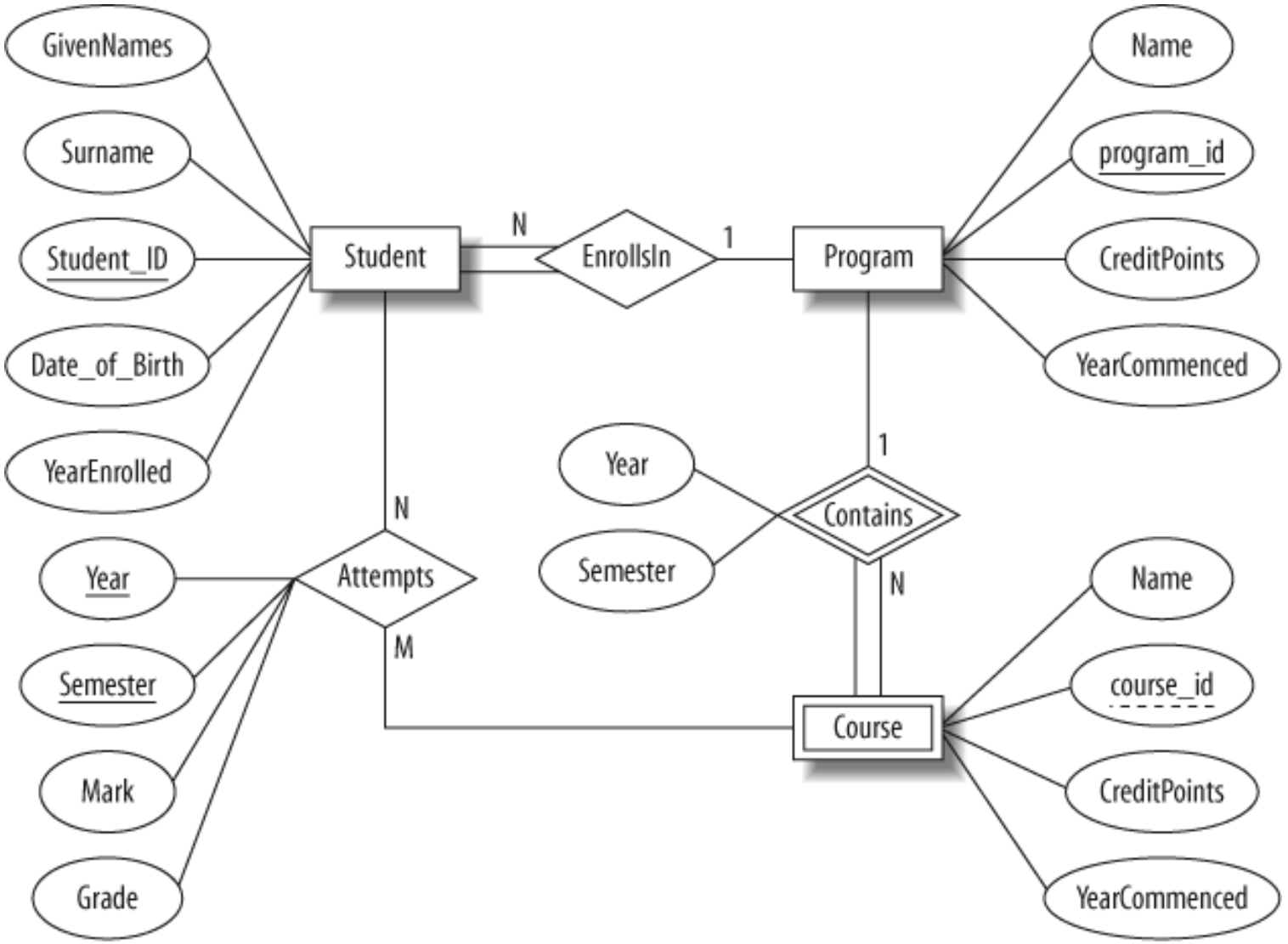

Participants
Genre
Info
Name
Reality_show
N
Broadcasts
1
Television
StartYear
Name
HeadOffice
1
Produces
N
Rates
1
N
Producer
Rating
User
CName
Country
Name
Psw
Age

Exercise 3

- Consider the following requirements list:

- The university offers one or more programs.
- A program is made up of one or more courses.
- A student must enroll in a program.
- A student takes the courses that are part of her program.
- A program has a name, a program identifier, the total credit points required to graduate, and the year it commenced.
- A course has a name, a course identifier, a credit point value, and the year it commenced.
- Students have one or more given names, a surname, a student identifier, a date of birth, and the year they first enrolled. We can treat all given names as a single object—for example, “John Paul.”
- When a student takes a course, the year and semester he attempted it are recorded. When he finishes the course, a grade (such as A or B) and a mark (such as 60 percent) are recorded.
- Each course in a program is sequenced into a year (for example, year 1) and a semester (for example, semester 1).

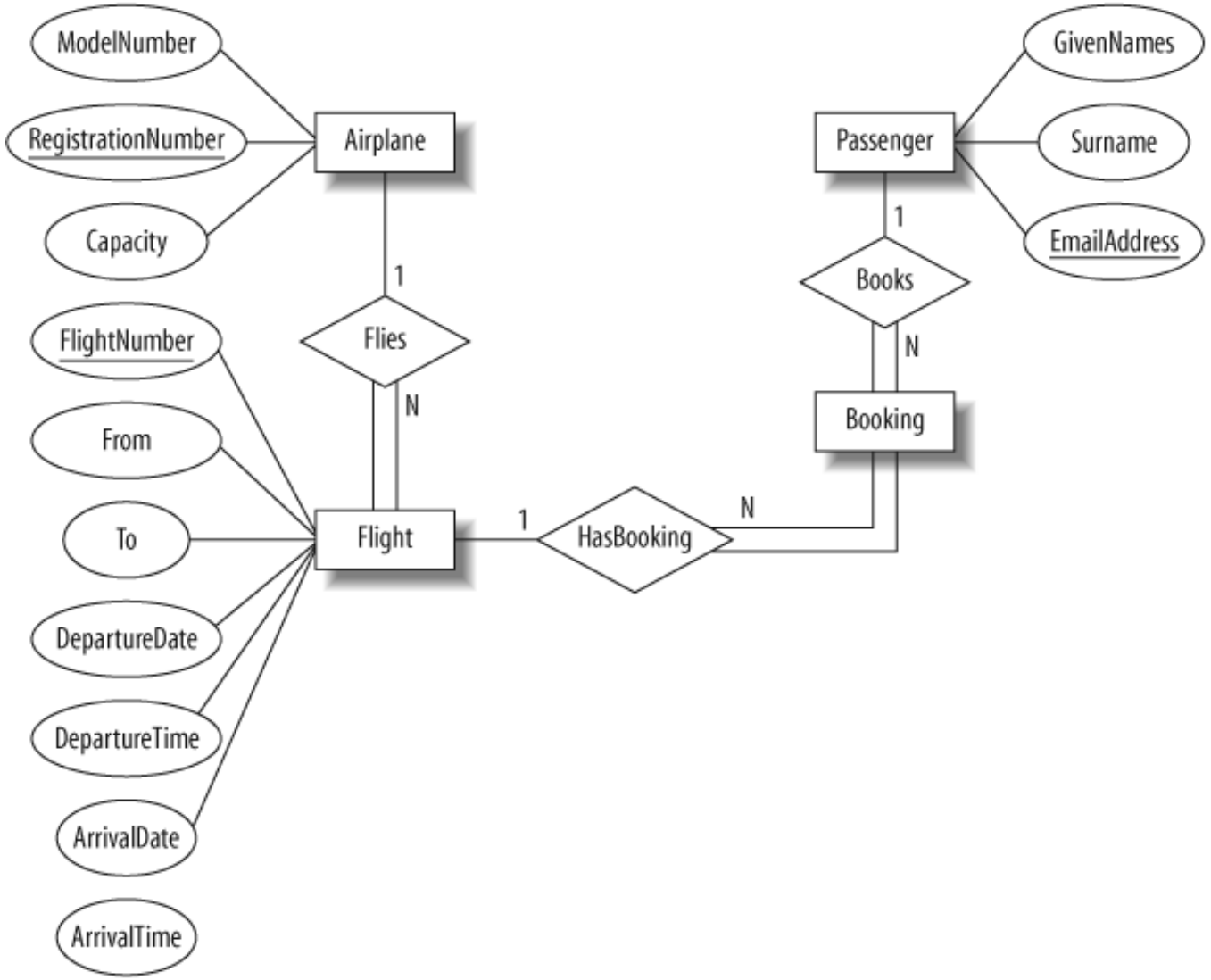
Solution


GivenNames
Surname
Student_ID
Date_of_Birth
YearEnrolled
Student
N
EnrollsIn
1
Program
Name
program_id
CreditPoints
YearCommenced
Year
Semester
1
Contains
N
Year
Semester
Mark
Grade
Attempts
N
M
Course
Name
course_id
CreditPoints
YearCommenced

Exercise 4

- Consider the following requirements list:
- The airline has one or more airplanes.
- An airplane has a model number, a unique registration number, and the capacity to take one or more passengers.
- An airplane flight has a unique flight number, a departure airport, a destination airport, a departure date and time, and an arrival date and time.
- Each flight is carried out by a single airplane.
- A passenger has given names, a surname, and a unique email address.
- A passenger can book a seat on a flight.

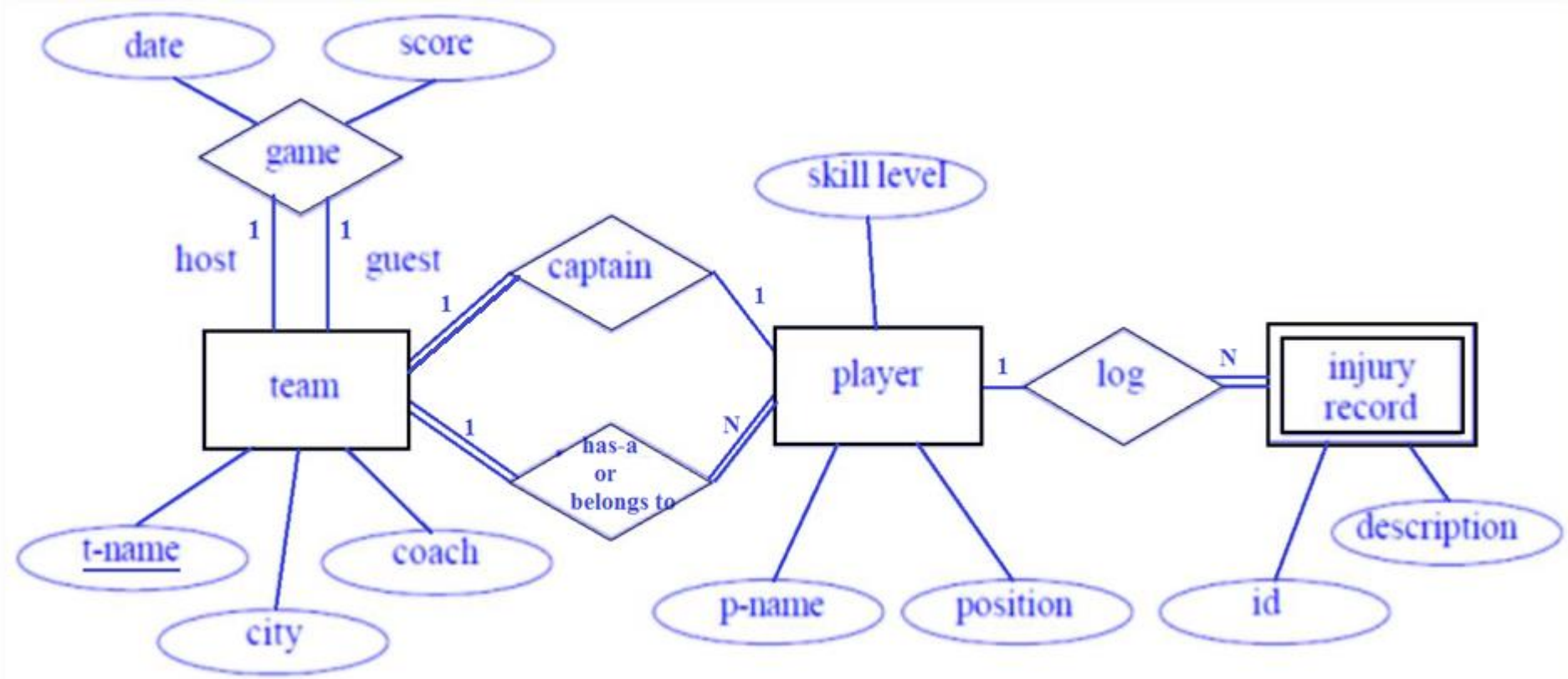
Solution


ModelNumber
RegistrationNumber
Capacity
Airplane
1
Flies
N
FlightNumber
From
To
DepartureDate
DepartureTime
ArrivalDate
ArrivalTime
Flight
1
HasBooking
N
Booking
N
Books
1
Passenger
GivenNames
Surname
EmailAddress

Exercise 5

- Suppose you are given the following requirements for a simple database for the National Hockey League (NHL):
- the NHL has many teams.
- each team has a name, a city, a coach, a captain, and a set of players.
- each player belongs to only one team.
- each player has a name, a position (such as left wing or goalie), a skill level, and a set of injury records.
- a team captain is also a player.
- a game is played between two teams (referred to as host_team and guest_team) and has a date (such as May 11th, 1999) and a score (such as 4 to 2).
- Construct a clean and concise ER diagram for the NHL database.

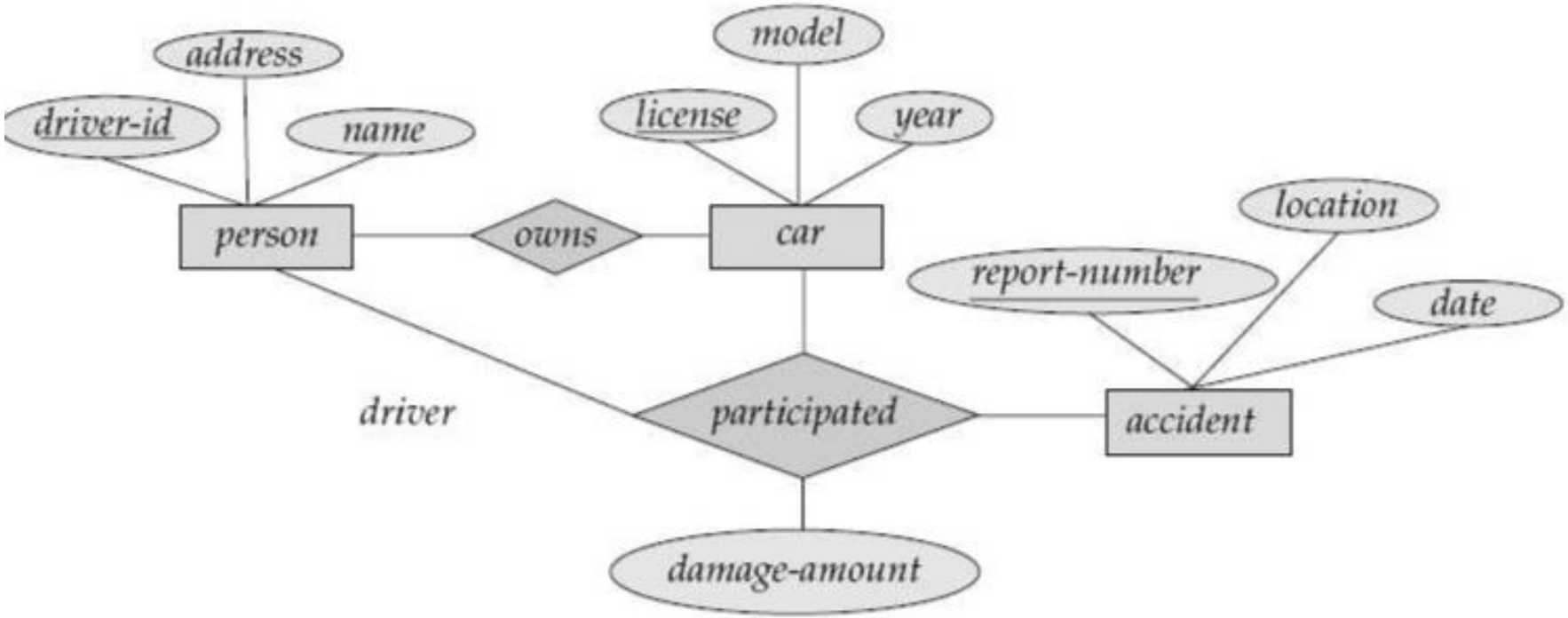
Solution


date
score
game
host
guest
1
1
captain
1
1
skill level
team
1
1
has-a
or
belongs to
N
player
1
log
N
injury
record
t-name
coach
city
p-name
position
id
description

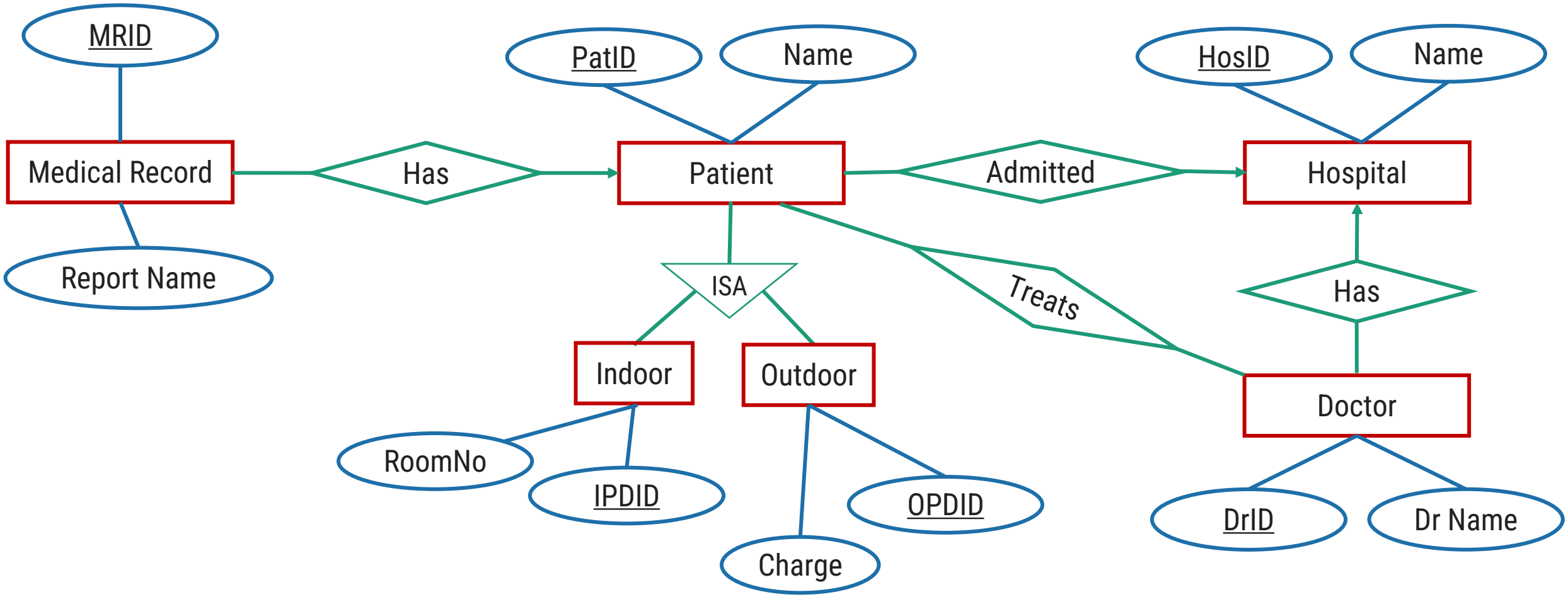
Exercise 6

- Consider a car-insurance company:
- Each customer or person have their name, address and unique driver id.
- Each car is having model, year and licence no.
- While each accident which is reported in terms of location, date and report no.
- Each customers own one or more cars.
- Each car has associated with it zero to any number of recorded accidents.

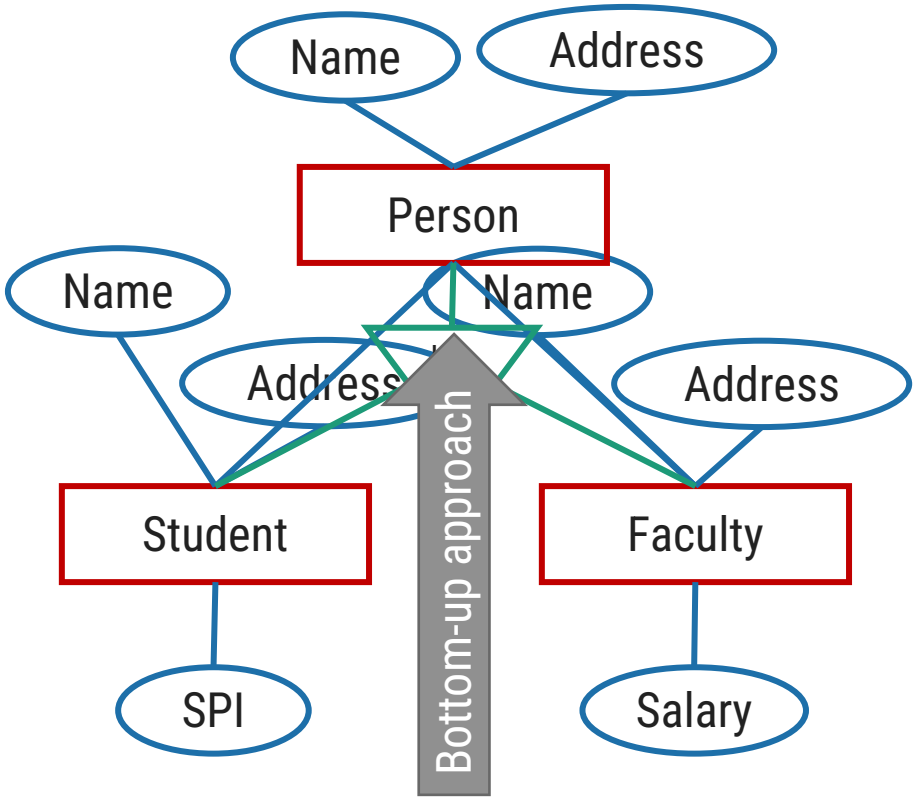
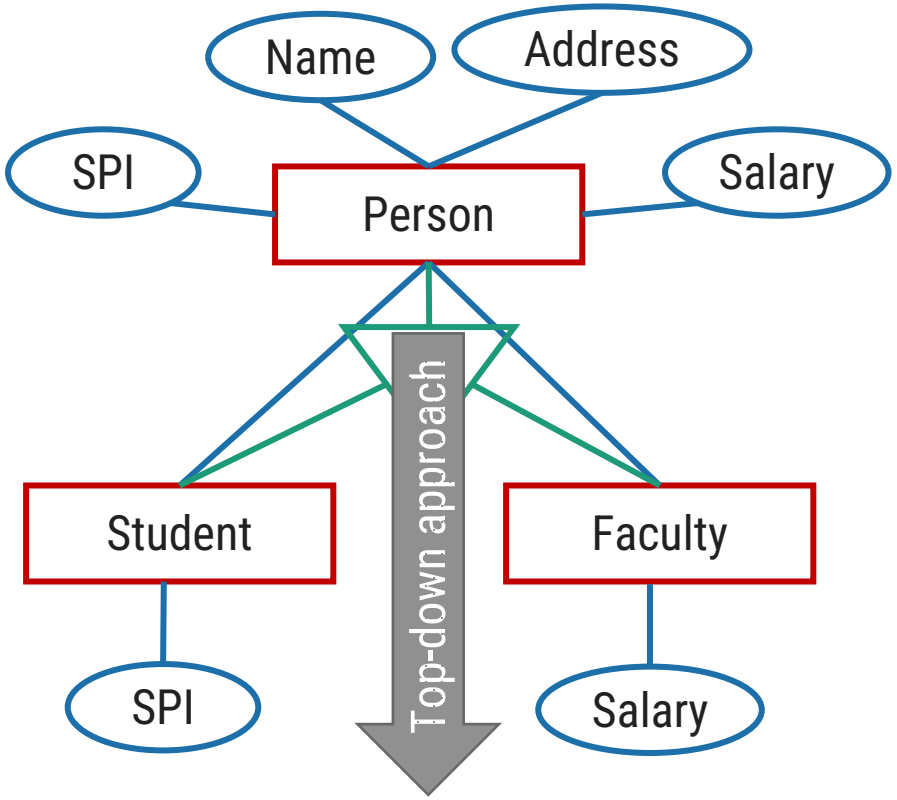
Solution


address
model
driver-id
name
license
year
person
owns
car
location
report-number
date
driver
participated
accident
damage-amount

E-R diagram of Hospital Management System



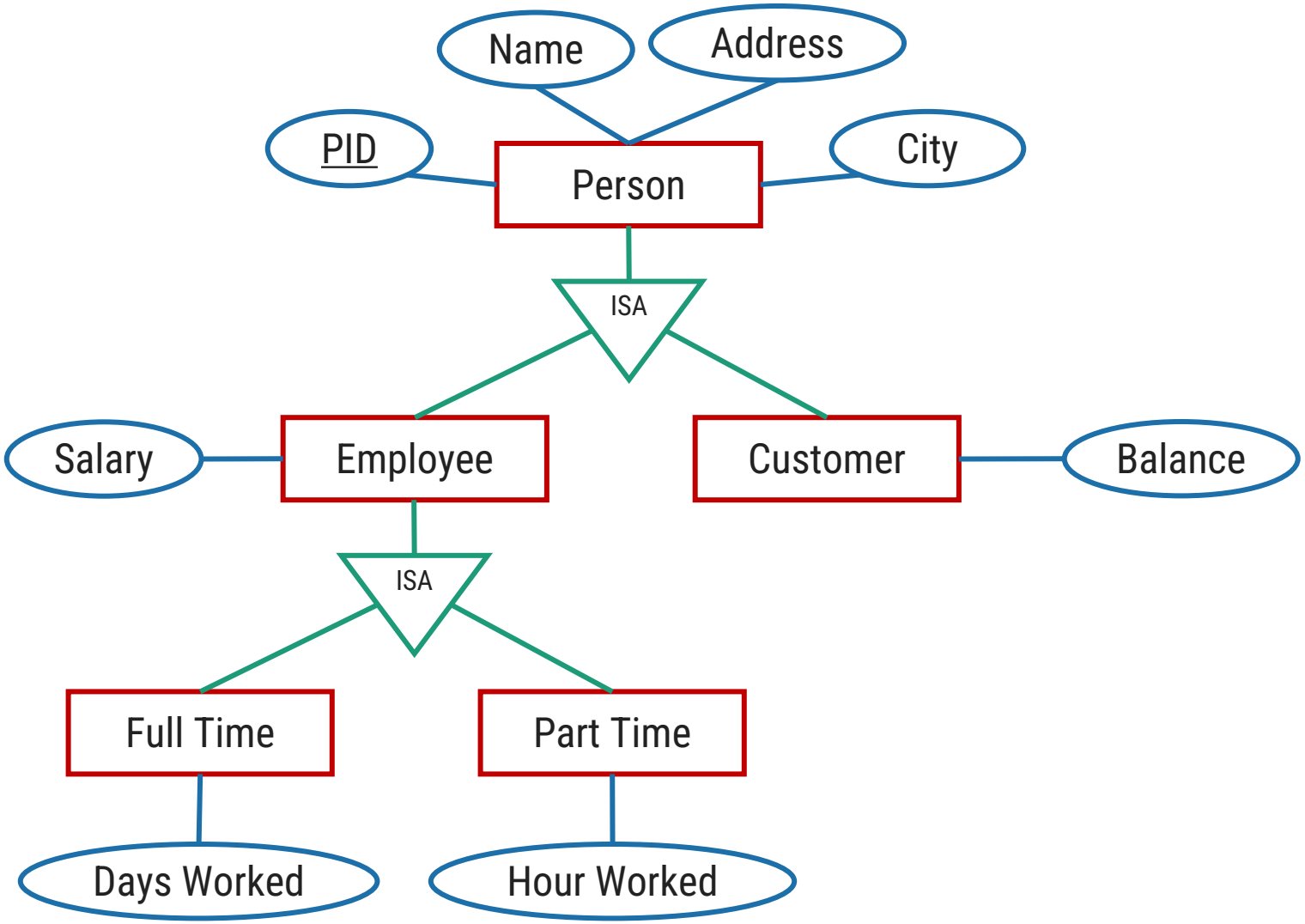
Generalization v/s Specialization

Generalization	Specialization
It **extracts the common features** of **multiple entities** to **form a new entity**.	It **splits an entity to form multiple new entities** that **inherit some feature of the splitting entity**.
	

Generalization v/s Specialization

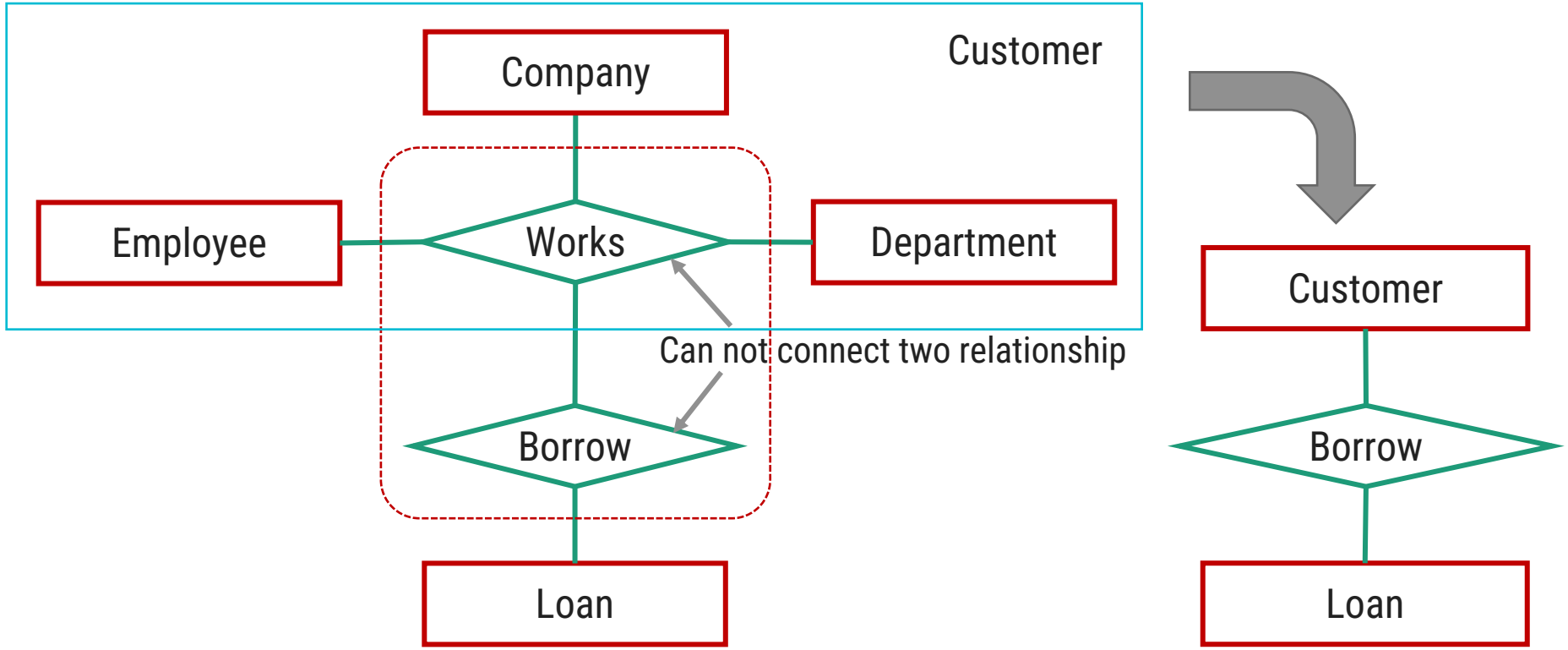
Generalization	Specialization
The process of **creation of group from various entities** is called generalization.	The process of **creation of sub-groups within an entity** is called specialization.
It is **Bottom-up** approach.	It is **Top-down** approach.
The process of taking the **union of two or more lower level entity** sets to produce a higher level entity set.	The process of taking a **sub set of higher level entity set** to form a lower level entity set.
It starts from the number of entity sets and creates high level entity set using some common features.	It starts from a single entity set and creates different low level entity sets using some different features.

Generalization & Specialization example



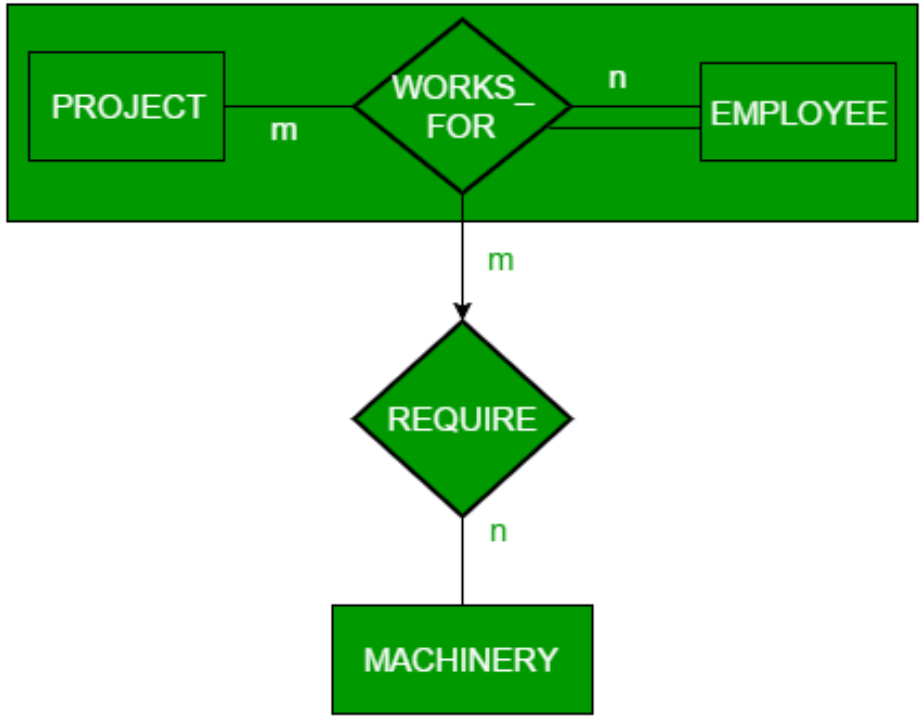
Aggregation

Aggregation is a process in which a single entity alone is not able to make sense in a relationship so the relationship of two entities acts as one entity.



Process of creating an entity by combining various components of E-R diagram is called aggregation.

- Employee working for a project may require some machinery. So, REQUIRE relationship is needed between relationship WORKS_FOR and entity MACHINERY. Using aggregation, WORKS_FOR relationship with its entities EMPLOYEE and PROJECT is aggregated into single entity and relationship REQUIRE is created between aggregated entity and MACHINERY.



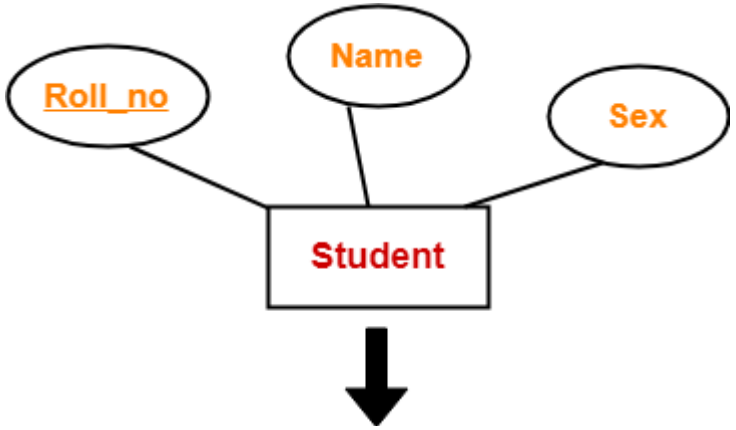
Aggregation

Conversion of ER model to Relational model

Rule-01: For Strong Entity Set With Only Simple Attributes-

A strong entity set with only simple attributes will require only one table in relational model.

- Attributes of the table will be the attributes of the entity set.
- The primary key of the table will be the key attribute of the entity set.

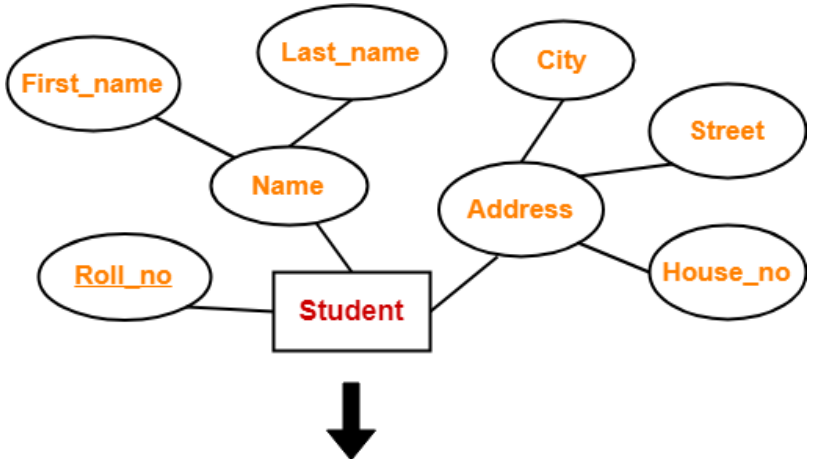


Roll_no	Name	Sex

Schema : Student (Roll_no , Name , Sex)

Rule-02: For Strong Entity Set With Composite Attributes-

- A strong entity set with any number of composite attributes will require only one table in relational model.
- We will create column for components of composite attribute only.



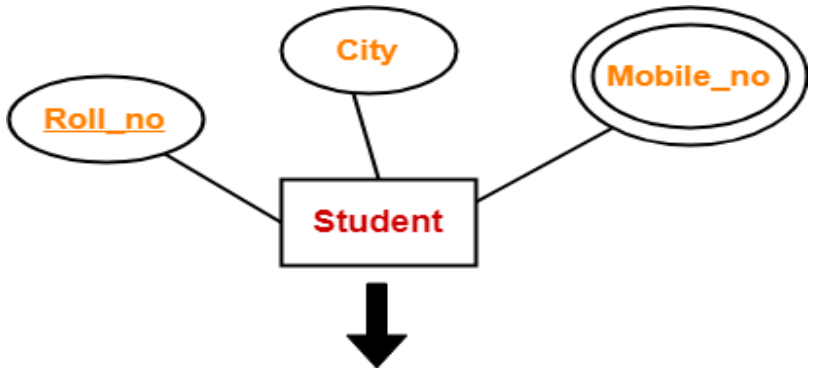
Roll_no	First_name	Last_name	House_no	Street	City

Schema : Student (Roll_no , First_name , Last_name , House_no , Street , City)

Rule-03: For Strong Entity Set With Multi Valued Attributes-

A strong entity set with any number of multi valued attributes will require two tables in relational model.

- One table will contain all the simple attributes with the primary key.
- Other table will contain the primary key and all the multi valued attributes.



Roll_no	City

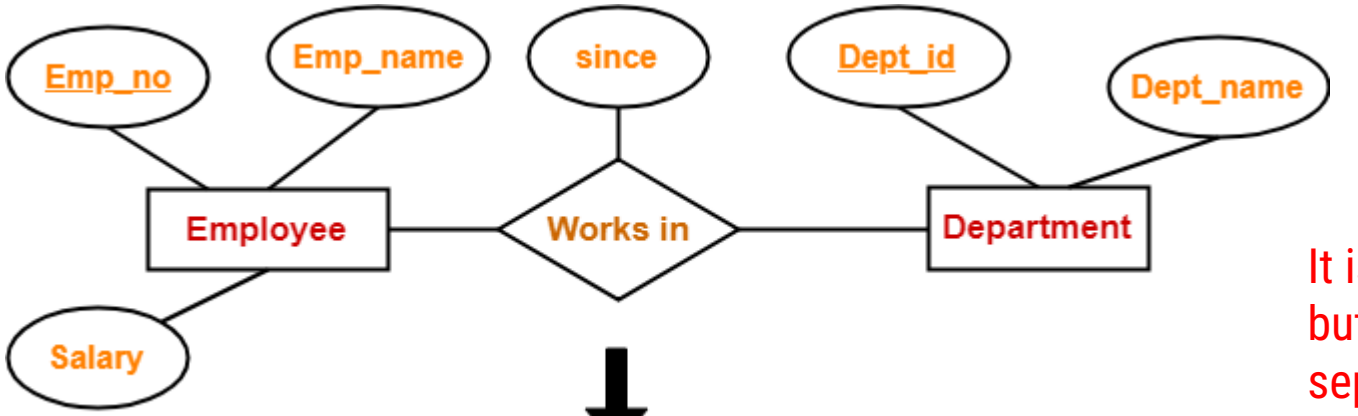
Roll_no	Mobile_no

Rule-04: Translating Relationship Set into a Table-

A relationship set will require one table in the relational model.
Attributes of the table are-

- **Primary key attributes of the participating entity sets**
- Its own descriptive attributes if any.

Set of non-descriptive attributes will be the primary key.



Emp_no	Dept_id	since

Schema : Works in (Emp_no , Dept_id , since)

It is not mandatory to create table for relationship, but in case of M:N relationship we must need a separate table for relationship

If we consider the overall ER diagram, three tables will be required in relational model-

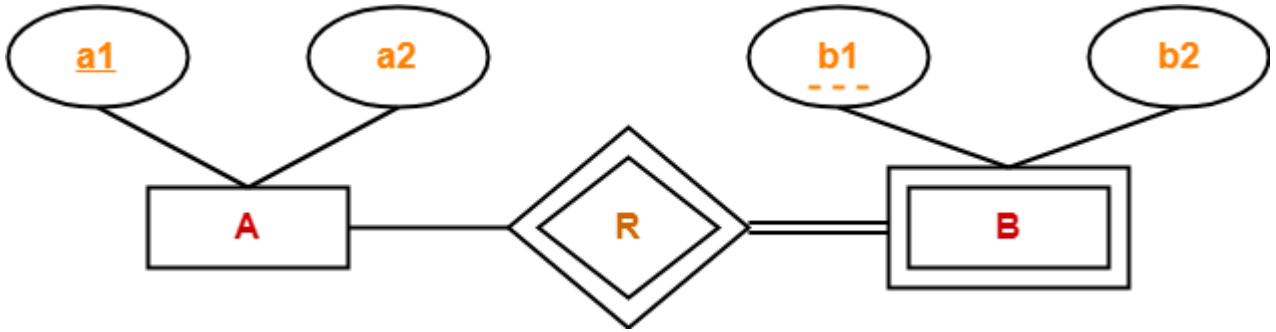
- One table for the entity set “Employee”
- One table for the entity set “Department”
- One table for the relationship set “Works in”

Rule-04.5: For Weak Entity Set :-

Weak entity are converted into tables of their own with primary key of strong entity acting as foreign key in the table.

- Composite prime attribute of the table= FK + attribute of WE

Weak entity set always appears in association with identifying relationship with total participation constraint.



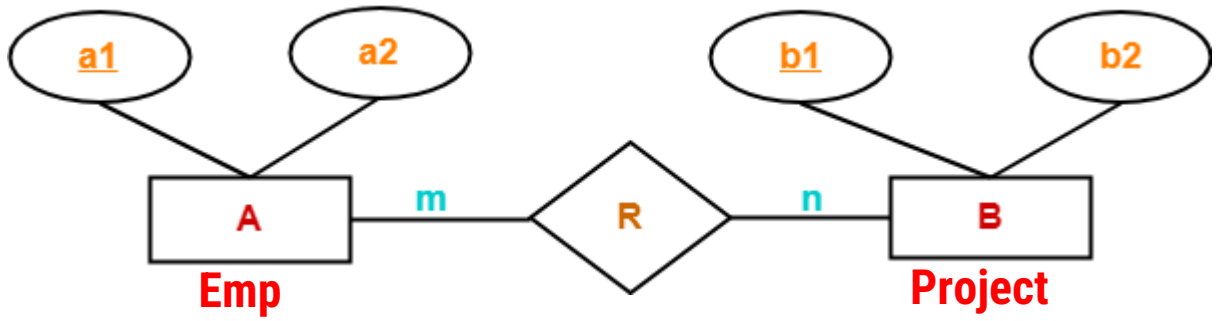
Here, two tables will be required-
1.A (a1 , a2)
2.B (a1 , b1 , b2)
Composite key attribute(primary key)=(a1+b1)

Rule-05: For Binary Relationships With Cardinality Ratios-

The following four cases are possible-

Case-01: Binary relationship with cardinality ratio m:n
Case-02: Binary relationship with cardinality ratio 1:n
Case-03: Binary relationship with cardinality ratio m:1
Case-04: Binary relationship with cardinality ratio 1:1

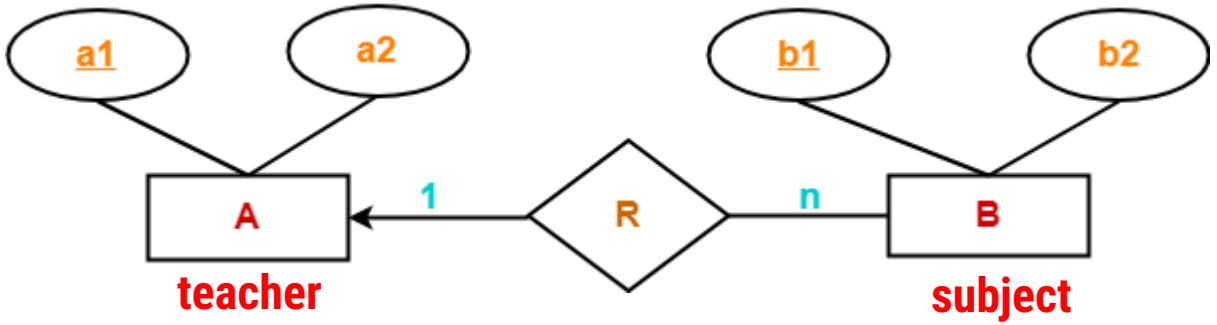
Case-01: For Binary Relationship With Cardinality Ratio m:n



Here, three tables will be required-
1.A (a1 , a2)
2.R (a1 , b1)
3.B (b1 , b2)

❑ **If relationship attribute(start_date) Will be in the relationship table**

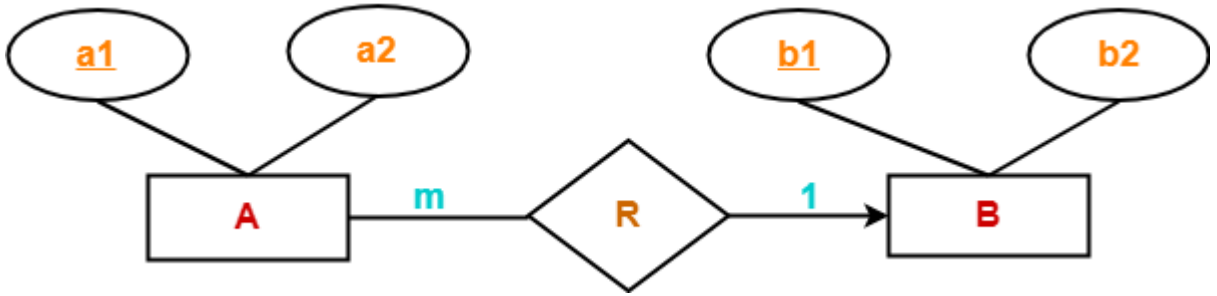
Case-02: For Binary Relationship With Cardinality Ratio 1:n



Here, two tables will be required-
1.A (a1 , a2)
2.BR (a1 , b1 , b2)

❑ **If relationship attribute(start_date) Shift towards N side**

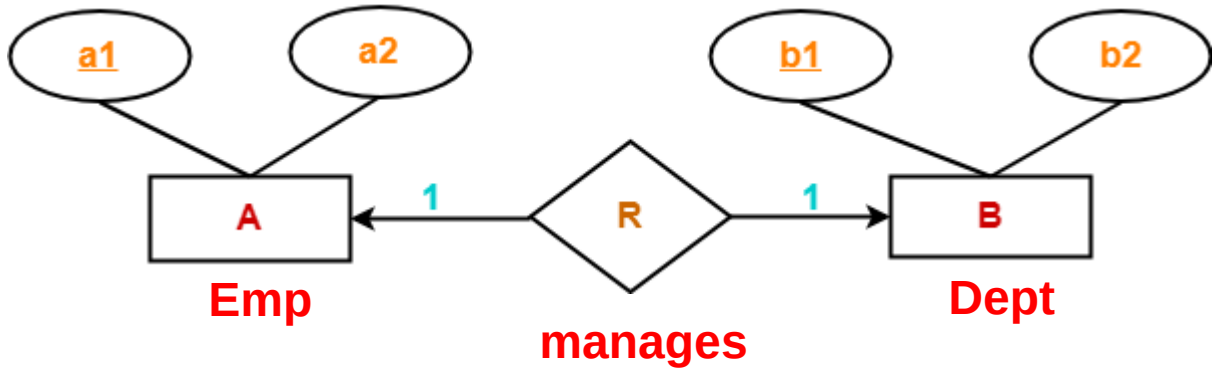
Case-03: For Binary Relationship With Cardinality Ratio m:1



Here, two tables will be required-
1.AR (a1 , a2 , b1)
2.B (b1 , b2)

❑ **If relationship attribute(start_date) Shift towards N side**

Case-04: For Binary Relationship With Cardinality Ratio 1:1



Here, two tables will be required. Either combine 'R' with 'A' or 'B'

Way-01:

- AR (a1 , a2 , b1)
- B (b1 , b2)

Way-02:

- A (a1 , a2)
- BR (a1 , b1 , b2)

Way-03:

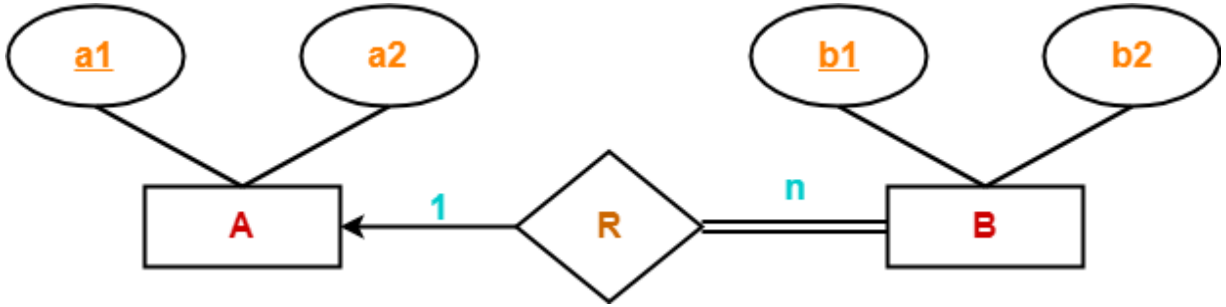
- A(a1 , a2)
- B (b1 , b2)
- R (a1, b1) primary key(a1 or b1)

❑ **If relationship attribute(start_date) Shift towards total side side**

Rule-06: For Binary Relationship With Both Cardinality Constraints and Participation Constraints-

- Cardinality constraints will be implemented as discussed in Rule-05.
- Because of the total participation constraint, foreign key acquires **NOT NULL** constraint i.e. now foreign key can not be null.

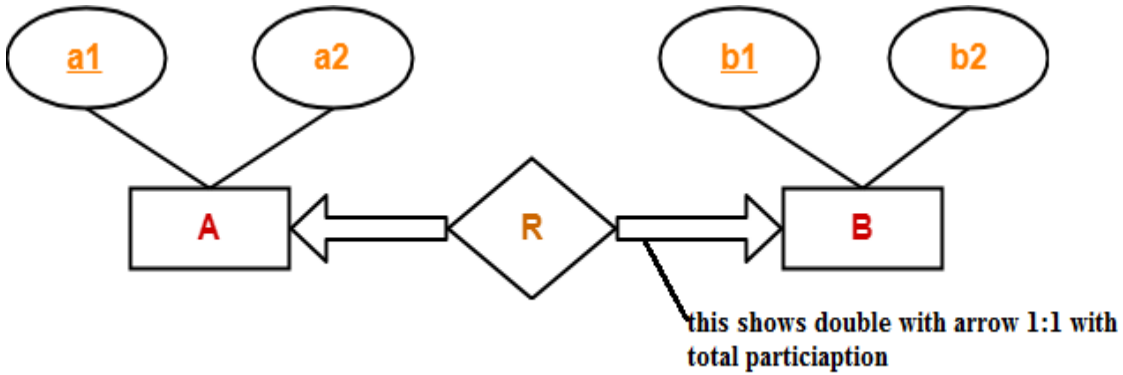
Case-01: For Binary Relationship With Cardinality Constraint and Total Participation Constraint From One Side-



Because cardinality ratio = 1 : n , so we will combine the entity set B and relationship set R.
Then, two tables will be required-
1.A (a1 , a2)
2.BR (a1 , b1 , b2)
Because of total participation, foreign key a1 has acquired NOT NULL constraint, so it can't be null now.

Case-02: For Binary Relationship With Cardinality Constraint and Total Participation Constraint From Both Sides-

If there is a key constraint from both the sides of an entity set with total participation, then that binary relationship is represented using only single table.



Here, Only one table is required.

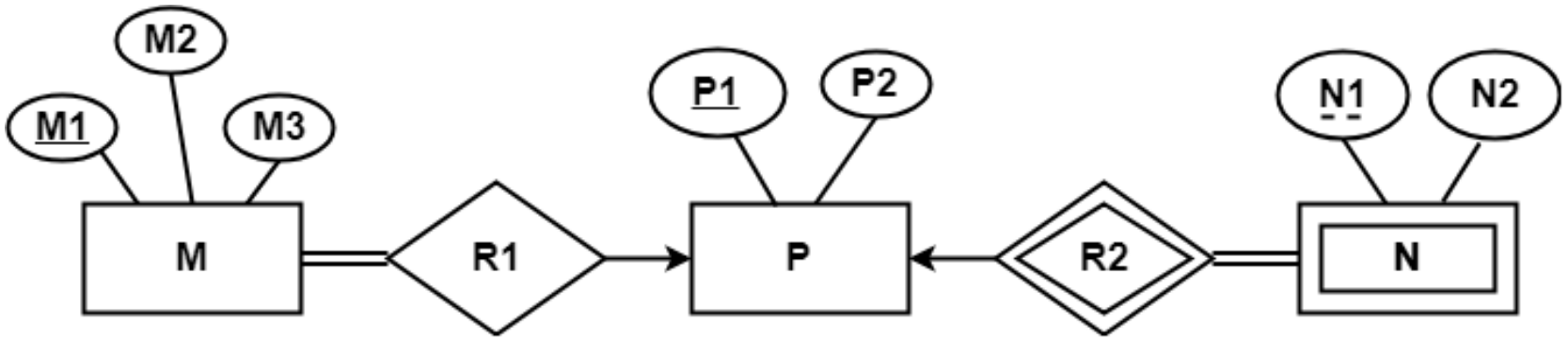
- ARB (a1 , a2 , b1 , b2)

❑ **If relationship attribute will be there in ARB**

Exercise

Problem-01:

Find the minimum number of tables required for the following ER diagram in relational model-

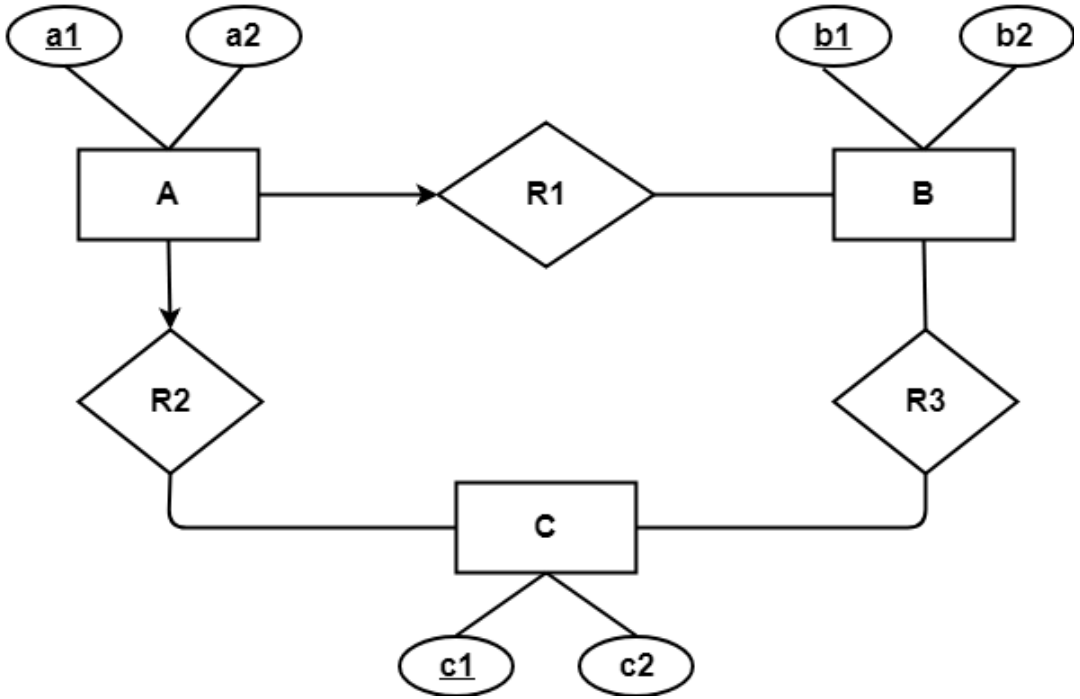


Applying the rules, minimum 3 tables will be required-

- MR1 (M1 , M2 , M3 , P1)
- P (P1 , P2)
- NR2 (P1 , N1 , N2)

Problem-02:

Find the minimum number of tables required to represent the given ER diagram in relational model-



R3 M:N

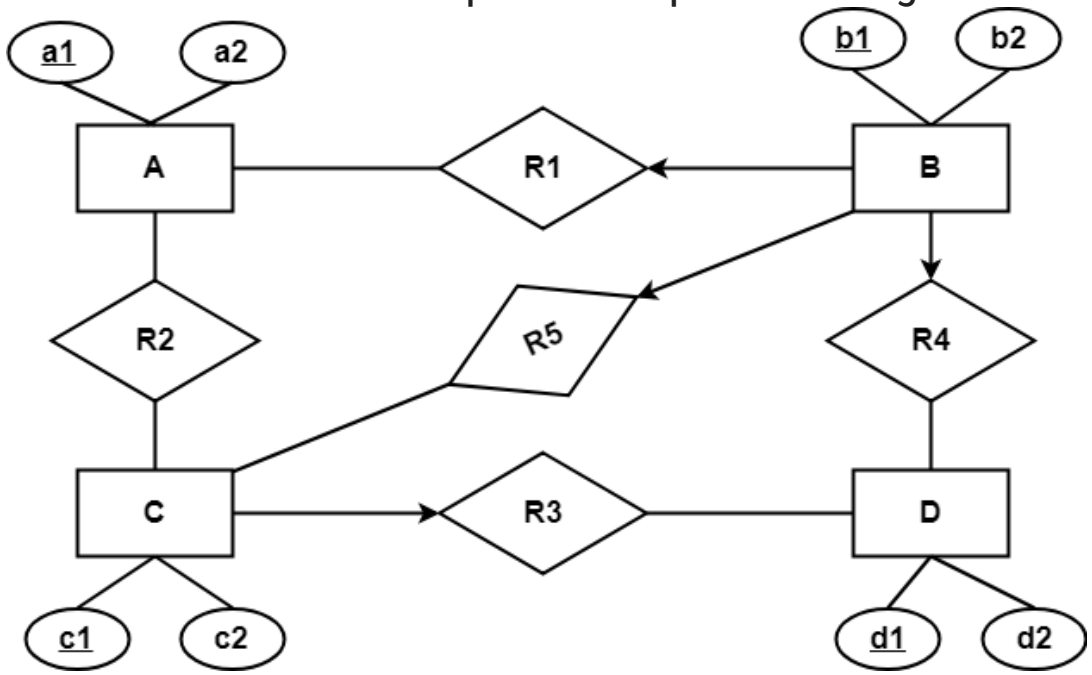
Applying the rules, minimum 4 tables will be required-

- AR1R2 (a1 , a2 , b1 , c1)
- B (b1 , b2)
- C (c1 , c2)
- R3 (b1 , c1)

Consider the A as employee, B as Department, C as project, Many employee working in one department and many employee working on one project

Problem-03:

Find the minimum number of tables required to represent the given ER diagram in relational model-

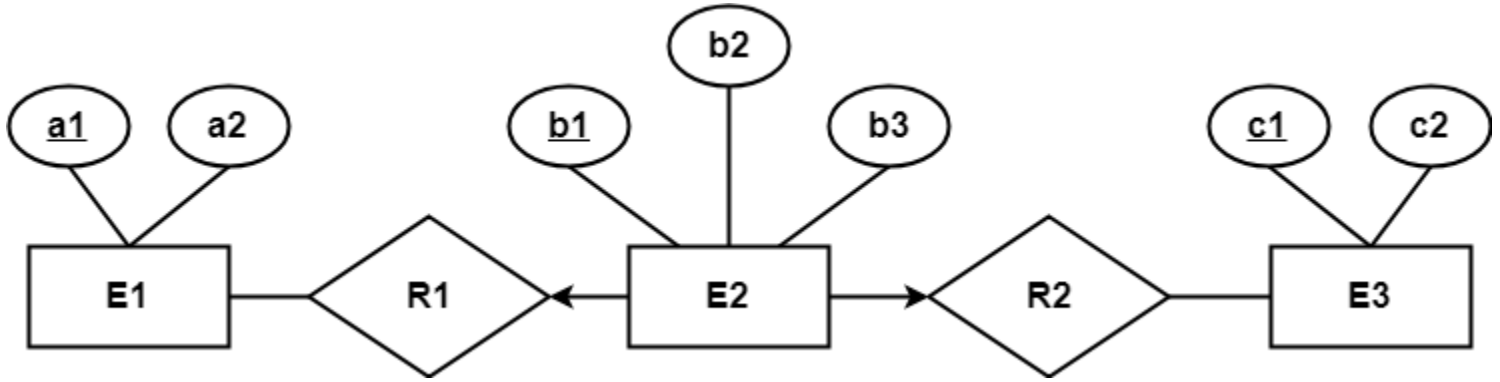


Applying the rules, minimum 5 tables will be required-

- BR1R4R5 (b1 , b2 , a1 , c1 , d1)
- A (a1 , a2)
- R2 (a1 , c1) **M:N**
- CR3 (c1 , c2 , d1)
- D (d1 , d2)

Problem-04:

Find the minimum number of tables required to represent the given ER diagram in relational model-

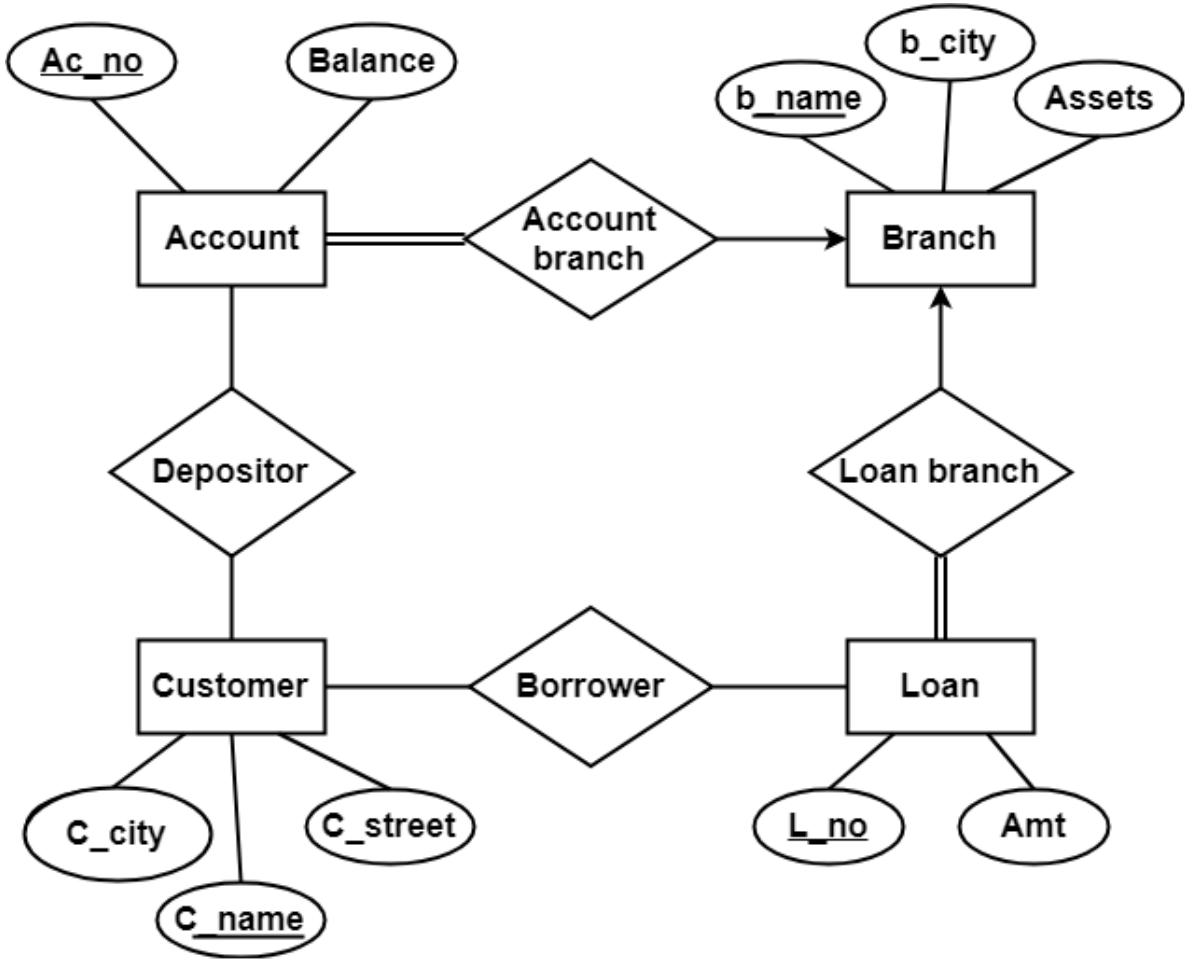


Applying the rules, minimum 3 tables will be required-

- E1 (a1 , a2)
- E2R1R2 (b1 , b2 , a1 , c1 , b3)
- E3 (c1 , c2)

Problem-05:

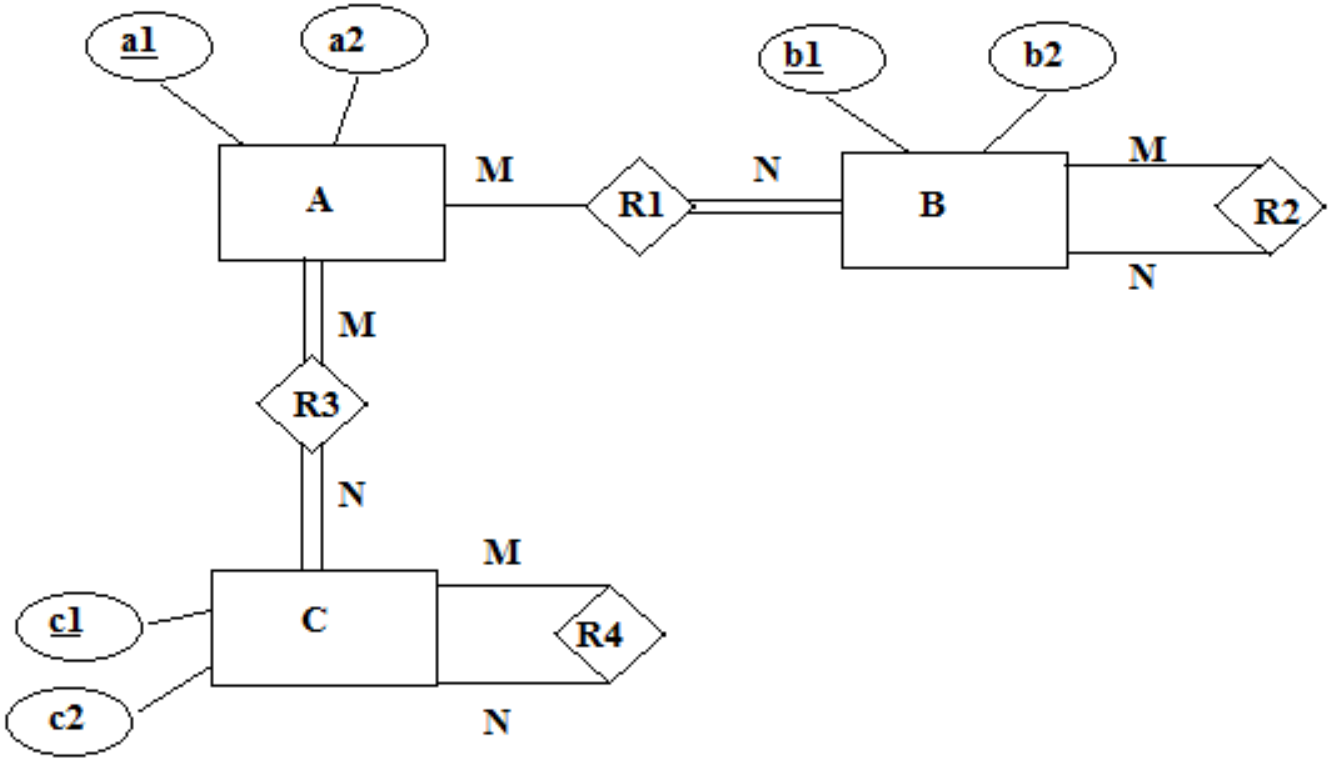
Find the minimum number of tables required to represent the given ER diagram in relational model-



Applying the rules that we have learnt, minimum 6 tables will be required-

- Account (Ac_no , Balance , b_name)
- Branch (b_name , b_city , Assets)
- Loan (L_no , Amt , b_name)
- Borrower (C_name , L_no) **M:N**
- Customer (C_name , C_street , C_city)
- Depositor (C_name , Ac_no) **M:N**

Problem 6



Summery of Symbols used in E-R diagram

Entity

Attribute

Relationship

Primary Key Attribute

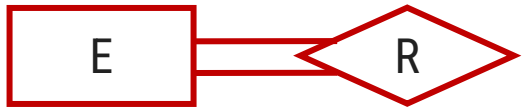
Derived Attribute

Multi Valued Attribute

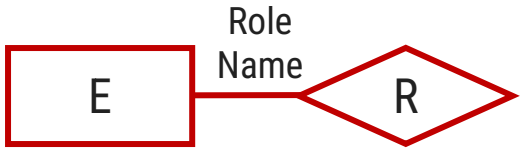
Weak Entity

Discriminating Attribute

Weak Entity Relationship



Total Participation



Role Indicator

Specialization/ Generalization