

# Basic Elements of Relational Databases

## Keys

### Primary key:

- A specially designated candidate key.
- The primary key for a table cannot contain null values.
- Candidate key that is chosen by the database designer as the principal means of identifying tuples within a relation.
- Is the candidate key whose values are used to *identify* tuples in the relation.

### Foreign key:

A column or combination of columns in which the values must match those of a candidate key.

Include among its attributes the primary key of another relation

### Null value:

- A special value that represents the absence of an actual value.
- A null value can mean that the actual value is unknown or does not apply to the given row.

# Database Normalization

Normalization is the process of organizing data in a database. This includes creating tables and establishing relationships between those tables according to rules designed both to protect the data and to make the database more flexible by eliminating redundancy and inconsistent dependency.

It is used to limit a table to one purpose you reduce the number of duplicate data that is contained within your database

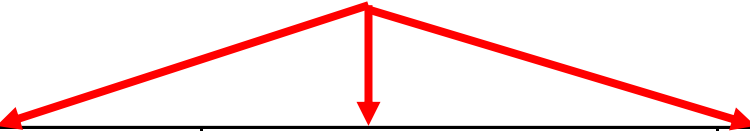
## Reasons for Normalization

1. To minimize duplicate data
2. To minimize or avoid data modification issues,
3. To simplify queries.

<https://www.guru99.com/database-normalization.html>

# Forms of Normalization

## 1NF No Repeating Columns



LastName	FirstName	Address	Phone	Book1	Book2	Book3
Jones	Janet	First St. #4	(514)256-5878	Fifty Shades of Grey	Harry Potter and the Sorcerer's Stone	Angels & Demons
Phil	Robert	Third St. #34	(438)547-9896	Dracula	Steve Jobs	
Clark	Stan	Fifth St. #4	(514)547-9565	The Lord of the Rings		
Coburn	James	32th ST #14	(514)647-98-74	Gone with the Wind	On the Edge	

1. From the previous table, we create 2 tables. One with a Primary Key and another with a foreign Key

**Table: Personal**

UserCode	LastName	FirstName	Address	Phone
101	Jones	Janet	First St. #4	(514)256-5878
102	Phil	Robert	Third St. #34	(438)547-9896
103	Clark	Stan	Fifth St. #4	(514)547-9565
104	Coburn	James	32th ST #14	(514)647-98-74

**Table: Codes**

UserCode	Bookcode	Book
101	ISBN303	Fifty Shades of Grey
101	ISBN304	Harry Potter and the Sorcerer's Stone
101	ISBN305	Angels & Demons
102	ISBN451	Dracula
102	ISBN452	Steve Jobs
103	ISBN361	The Lord of the Rings
104	ISBN121	Gone with the Wind
104	ISBN122	On the Edge

# Forms of Normalization

## 2NF All attributes Dependent on Full Primary Key

Example:

The company IT Best is planning to develop special projects in which employees will participate. Here there is the list of employees and projects to work

Jones Janet, Networking and Database

Phil Robert, Website and Networking

Clark Stan, Database

Coburn James, Networking and Database

Create and normalise a database, according to this information

LastName	FirstName	ProjectName
Jones	Janet	Networking
Jones	Janet	Database
Phil	Robert	Website
Phil	Robert	Networking
Clark	Stan	Database
Coburn	James	Networking
Coburn	James	Website

Table: Employee		
EmployeeId	LastName	FirstName
101	Jones	Janet
102	Phil	Robert
103	Clark	Stan
104	Coburn	James

Table: Project	
ProjectId	ProjectName
P001	Database
P002	Networking
P003	Website

Table: ProjectAssignment	
EmployeeId	ProjectId
101	P001
101	P002
102	P002
102	P003
103	P001
104	P002
104	P003

# Entity-Relationship Diagram (ERD)

An entity-relationship diagram (ERD) is a data modeling technique that graphically illustrates an information system's entities and the relationships between those entities.

The elements of an ERD are:

- Entities
- Relationships
- Attributes

Steps involved in creating an ERD include:

- Identifying and defining the entities
- Determining all interactions between the entities
- Analyzing the nature of interactions/determining the cardinality of the relationships
- Creating the ERD

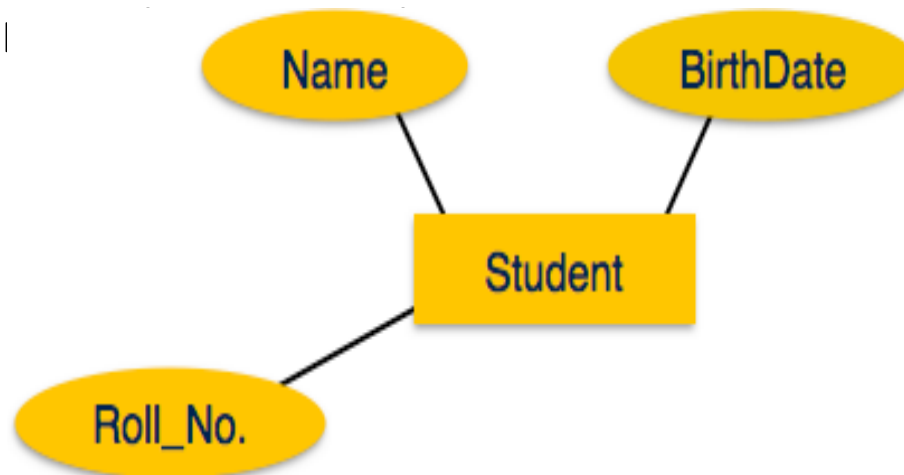
# Entity

Entities are represented by means of rectangles. Rectangles are named with the entity set they represent.



## Attributes

- Attributes are the properties of entities.
- Attributes are represented by means of ellipses.
- Every ellipse represents one attribute and is directly connected to its entity.

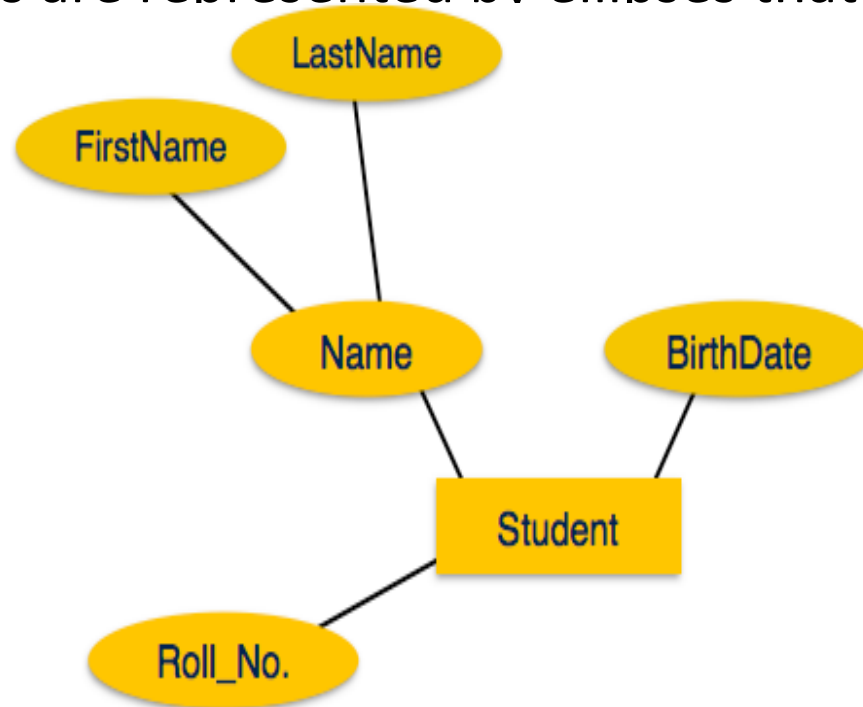


## Attributes

If the attributes are composite, they are further divided in a tree like structure.

Every node is then connected to its attribute.

That is, composite attributes are represented by ellipses that are connected with an ellipse.



## Relationship

Relationships are represented by diamond-shaped box.

Name of the relationship is written inside the diamond-box.

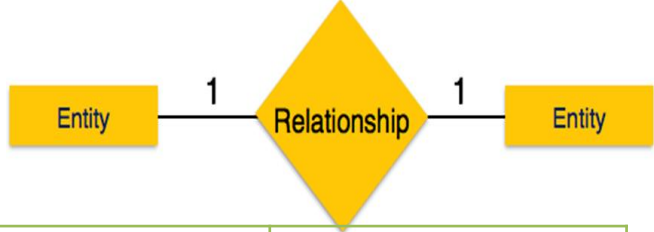
All the entities (rectangles) participating in a relationship, are connected to it by a line.

**One-to-one Relationship.** When only one instance of an entity is associated with the relationship, it is marked as '1:1'. The following image reflects that only one instance of each entity should be associated with the relationship. It depicts one-to-one relationship

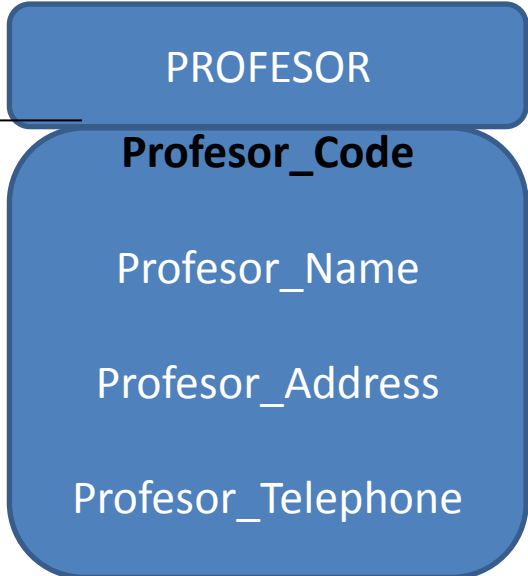
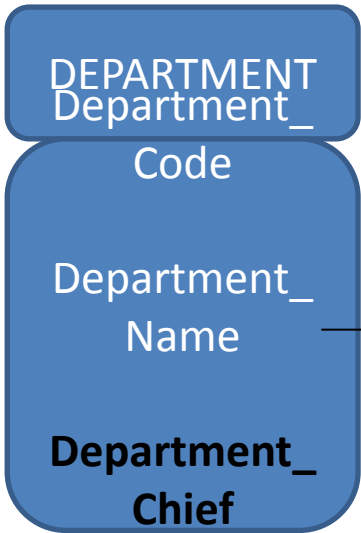
One A is associated with one B:



Department_C ode	Tecno
Department_ Name	Technology
Department_C hief	030



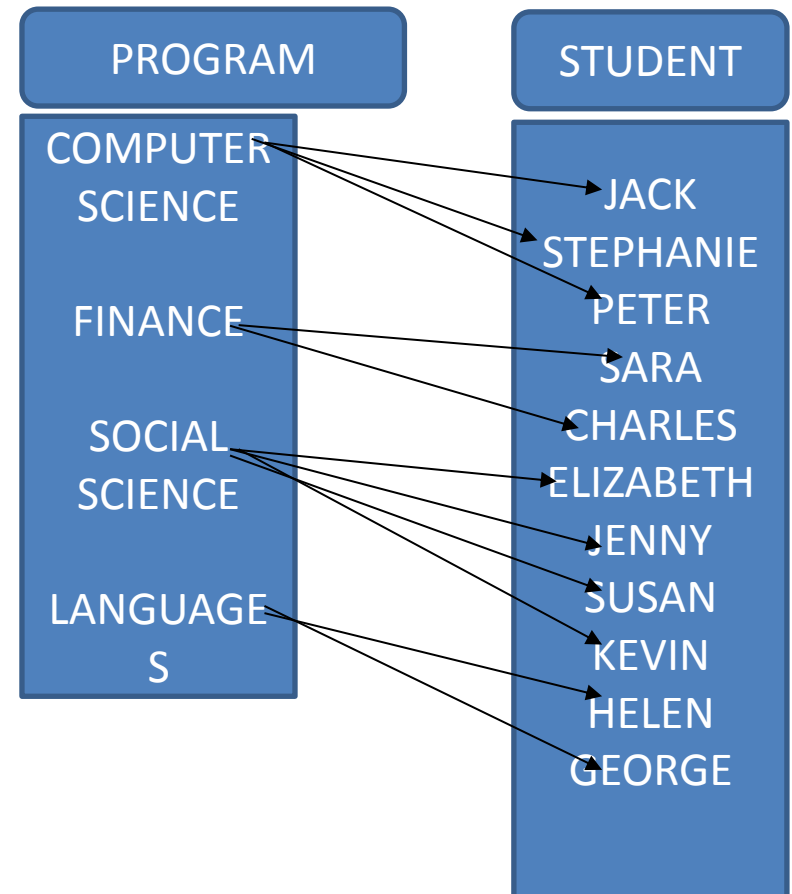
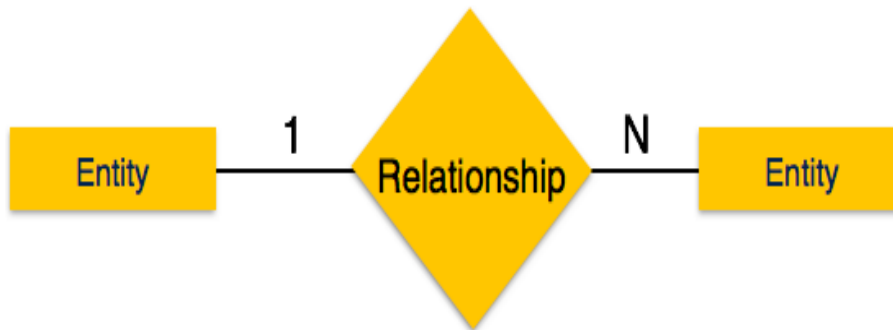
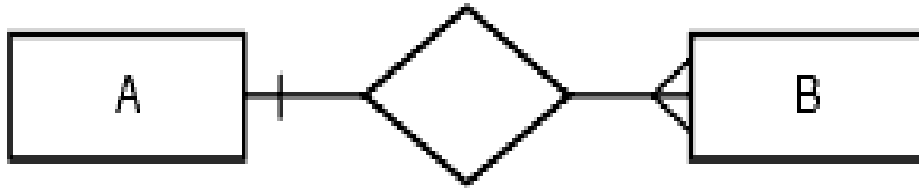
Profesor_Code	030
Profesor_Name	James Taylor
Profesor_Addres s	Sherbrooke st.
Profesor_Teleph one	(514) 578-9536



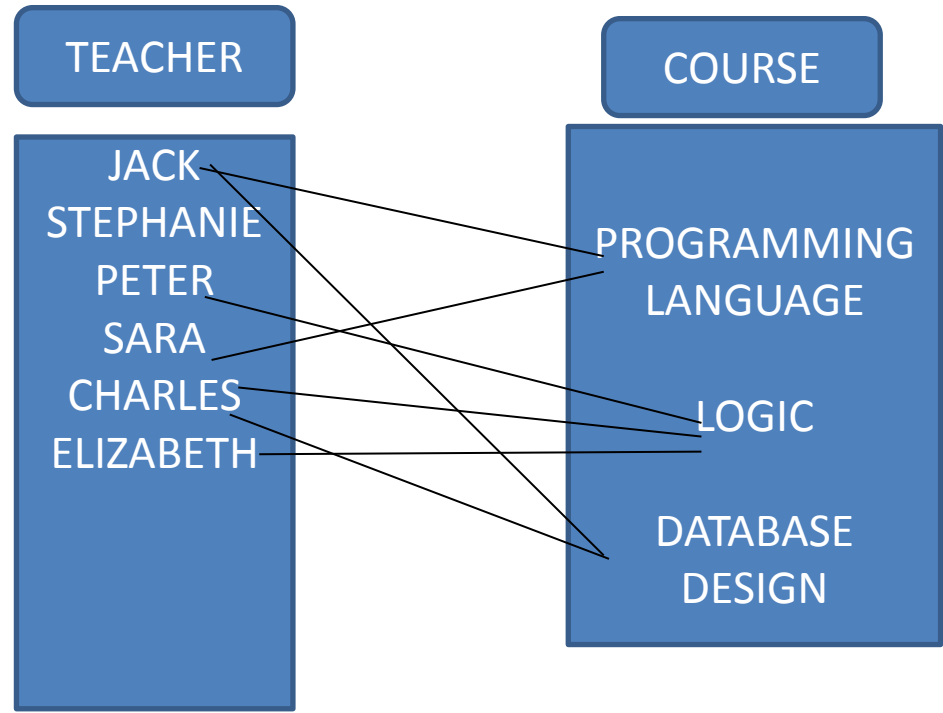
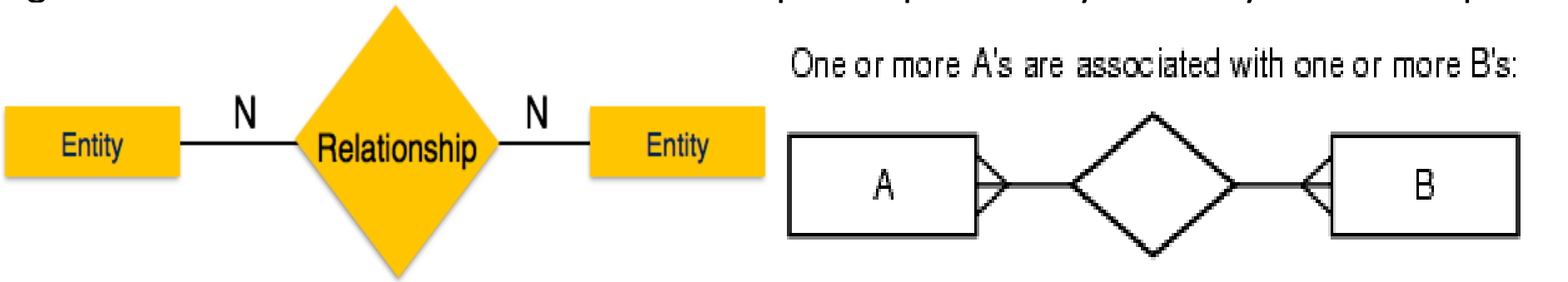


**One-to-many Relationship.** When more than one instance of an entity is associated with a relationship, it is marked as '1:N'. The following image reflects that only one instance of entity on the left and more than one instance of an entity on the right can be associated with the relationship. It depicts one-to-many relationship.

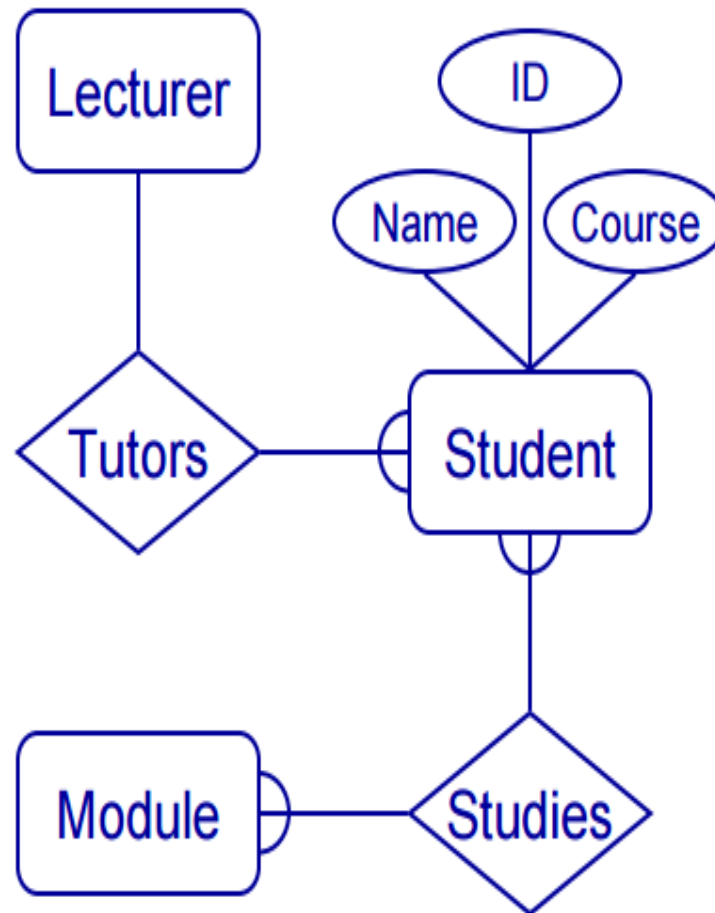
One A is associated with one or more B's:



**Many-to-many Relationship.** The following image reflects that more than one instance of an entity on the left and more than one instance of an entity on the right can be associated with the relationship. It depicts many-to-many relationship.

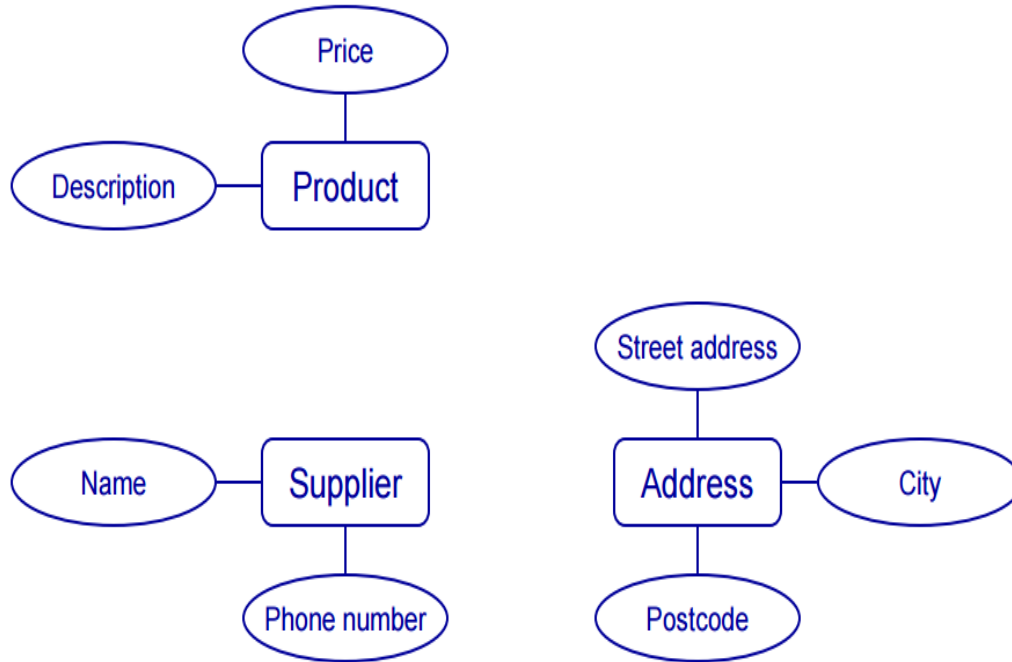


In a University database we might have entities for Students, Modules and Lecturers. Students might have attributes such as their ID, Name, and Course, and could have relationships with Modules (enrolment) and Lecturers (tutor/tutee)



We want to represent information about products in a database. Each product has a description, a price and a supplier. Suppliers have addresses, phone numbers, and names. Each address is made up of a street address, a city, and a postcode.

## Example - E/R Diagram

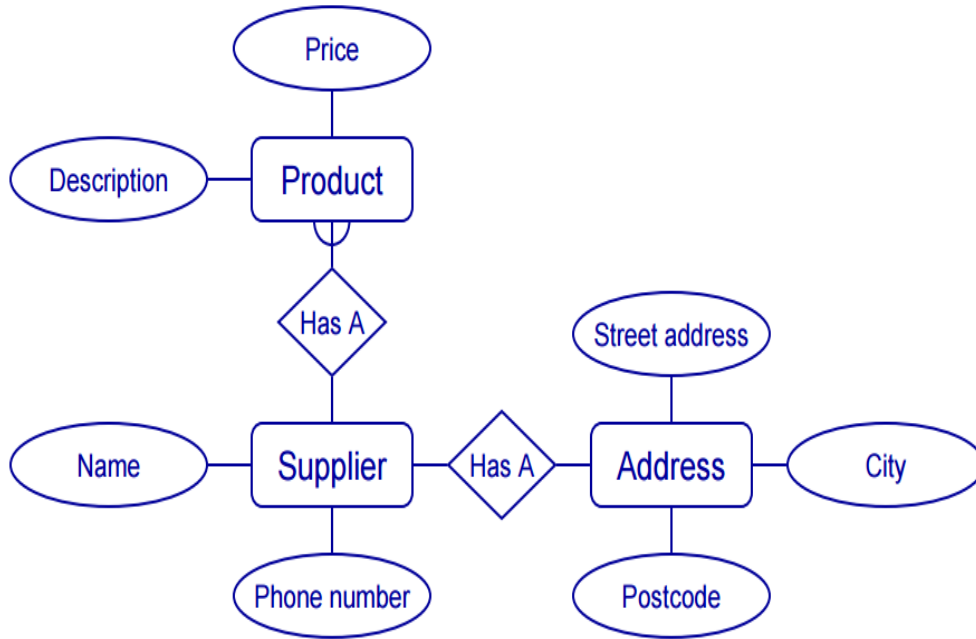


Entity Relationship Modelling

## Example - Relationships

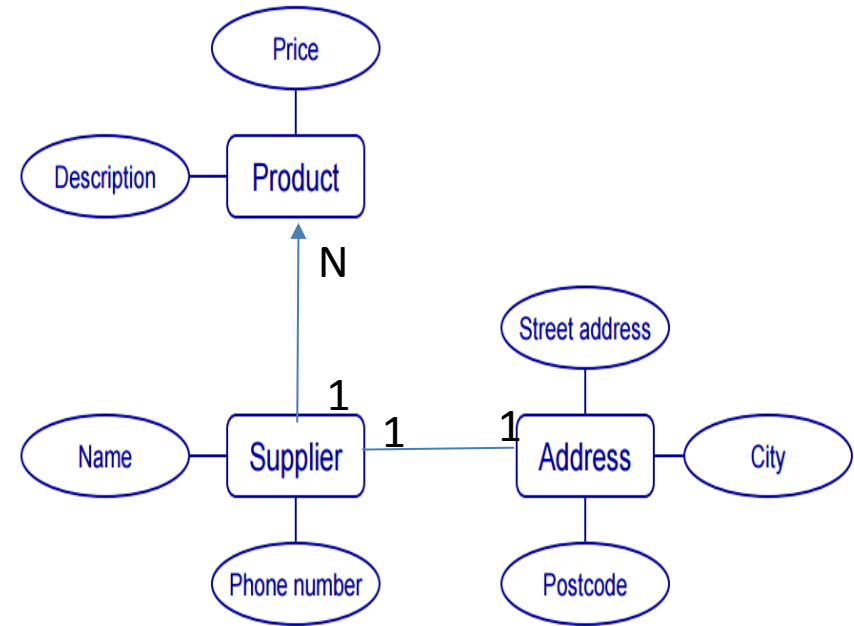
- Each product has a supplier
  - Each product has a single supplier but there is nothing to stop a supplier supplying many products
  - A many to one relationship
- Each supplier has an address
  - A supplier has a single address
  - It does not seem sensible for two different suppliers to have the same address
  - A one to one relationship

# Example - E/R Diagram



Entity Relationship Modelling

# Example - E/R Diagram

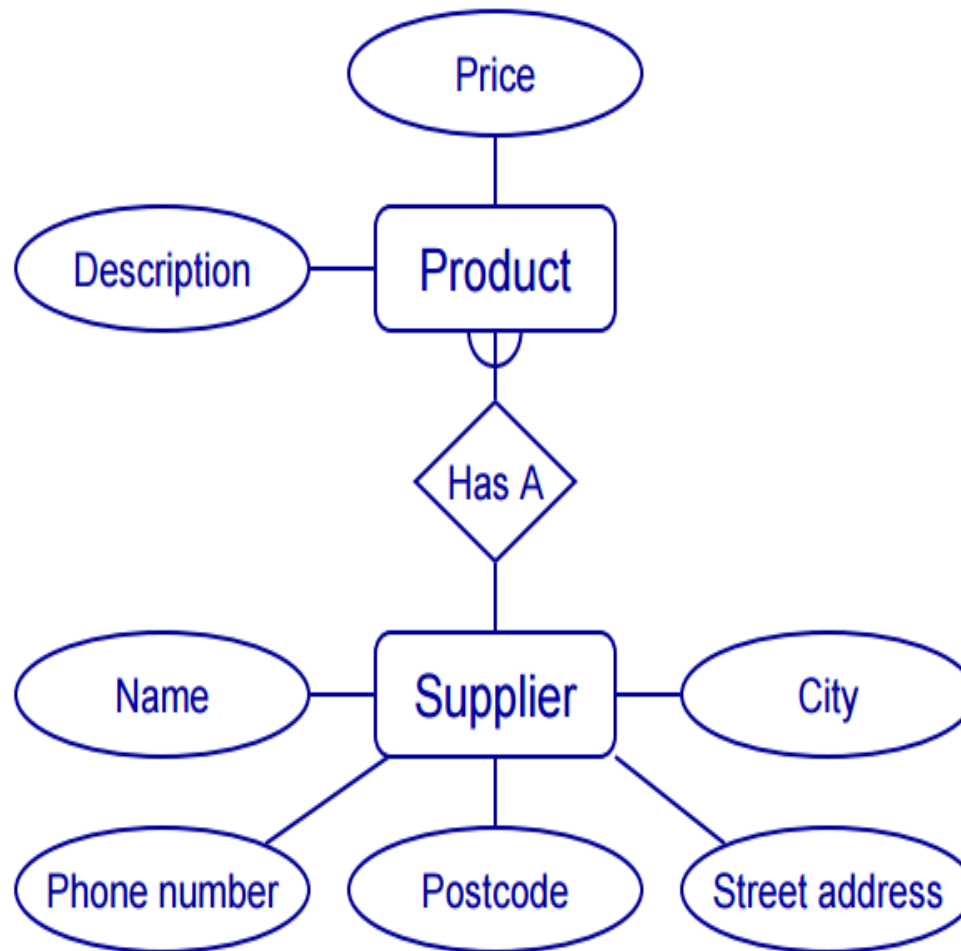


Entity Relationship Modelling

## Example - Relationships

- Each product has a supplier
  - Each product has a single supplier but there is nothing to stop a supplier supplying many products
  - A many to one relationship
- Each supplier has an address
  - A supplier has a single address
  - It does not seem sensible for two different suppliers to have the same address
  - A one to one relationship

# Example - E/R Diagram



# VIDEO RENTAL

A small video rental store offers its services to clients in its neighborhood. Its entities are Customer, Rental and Video. The rental entity has three attributes: dateDue, dateRented, and cost. These are the relationship sentences for the diagram:

- A customer may have many rentals.
- A video may have no more than one rental.

Draw a Entity-relation diagram to describe this process