What is Data?

Data is a collection of facts , such as values and measurements.

Types of data

Qualitative

Quantitative

Discrete

Continuous

Different options for storing data

Spread sheet

Text file

Pdf

Etc…. the problem in using this is

**FIR** 🡪

Flexibility

Inconsistency

Redundancy

**DBMS**

It is a software that acts as an interface between the actual files containing data and the application program uses the data.

**Advantages of RDBMS 🡪 I2 S2 FE**

Data independence

Integrity

Sharing

Security

Flexibility

Efficient

**Database design methodology**

ERD 🡪 Entity relationship diagram

Normalization

Entity relationship diagram

What is entity ?

Being existence

Thing about which descriptive information is to be stored

Which is capable of independent existence

Can be uniquely identified

Example :

An object like : horse a student or a car etc…

An activity like : an event

An event is a meaningfully described by its ATTRIBUTES

What is relationship ?

Connection , Association

A relationship is named association between two or more entity types

Candidate takes tests

It is defined by the cardinality :

1:1

1:M

N:M

Entity 🡪

Attributes 🡪

Relationship 🡪

Cardinality

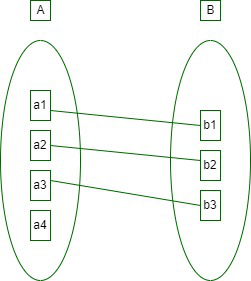
Types of cardinality in between tables are:

* One-to-One
* One-to-Many
* Many-to-One
* Many-to-Many

**One-to-One**

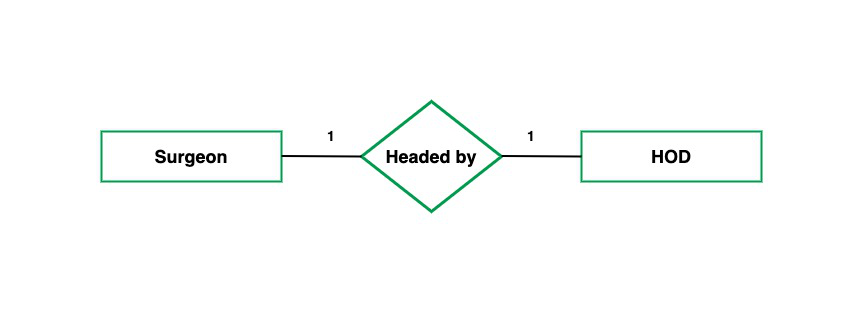
In this type of cardinality mapping, an entity in A is connected to at most one entity in B. Or we can say that a unit or item in B is connected to at most one unit or item in A.

In one on one cardinality any one of the table primary key can work as foreign key in another table.



**Example:**

In a particular hospital, the surgeon department has one head of department. They both serve one-to-one relationships.



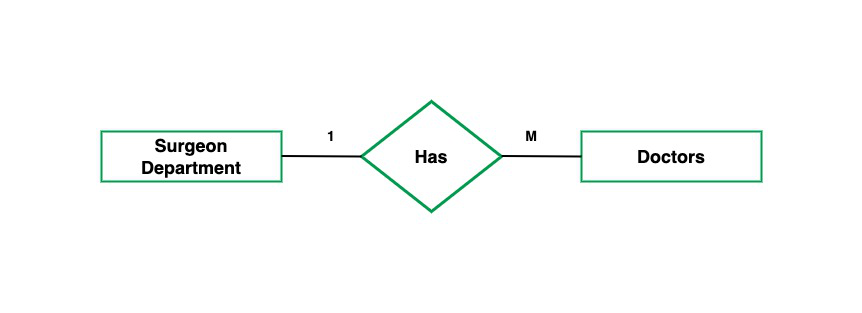
**One-to-Many**

In this type of cardinality mapping, an [entity](https://www.geeksforgeeks.org/dbms/difference-between-entity-entity-set-and-entity-type/) in A is associated with any number of entities in B. Or we can say that one unit or item in B can be connected to at most one unit or item in A.

In case of one to many cardinality single entity primary key works as foreign key in another table.

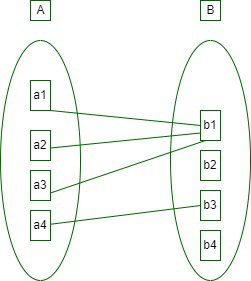
**Example:**

In a particular hospital, the surgeon department has multiple doctors. They serve one-to-many relationships.



**Many-to-One**

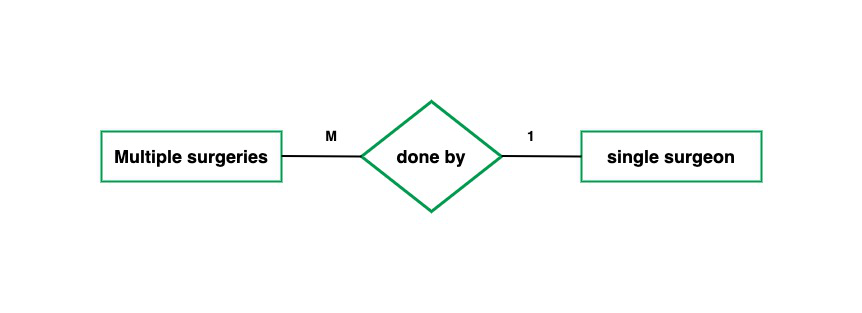
In this type of cardinality mapping, an entity in A is connected to at most one entity in B. Or we can say a unit or item in B can be associated with any number (zero or more) of entities or items in A.



Many to One

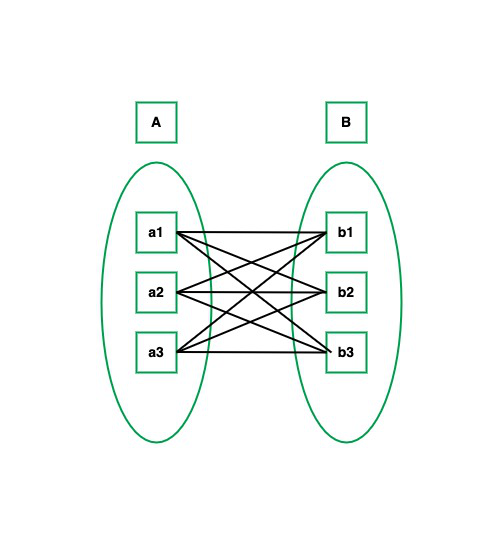
**Example:**

In a particular hospital, multiple surgeries are done by a single surgeon. Such a type of relationship is known as a many-to-one relationship.



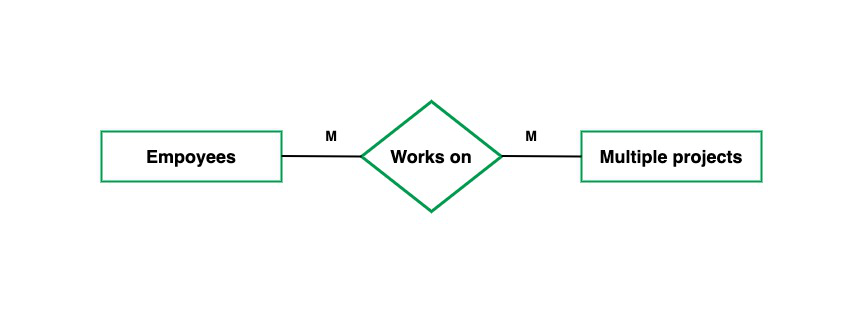
**Many-to-Many**

In this type of cardinality mapping, an entity in A is associated with any number of entities in B, and an entity in B is associated with any number of entities in A.



**Example:**

In a particular company, multiple people work on multiple projects. They serve many-to-many relationships.



The appropriate mapping cardinality for a particular relation set obviously depends on the real-world situation in which the relation set is modeled.

* If we have cardinality one-to-many or many to one then, we can mix relational tables with many involved tables.
* If the cardinality is many-to-many we cant mix any two tables.
* If we have a one-to-one relation and we have total participation of one entity then we can mix that entity with a relation table and if we have total participation of both entities then we can make one table by mixing two entities and their relation.

Keys in RDBMS

1. **Primary Key**  
   Uniquely identifies each record in a table; cannot be null or duplicated.
2. **Foreign Key**  
   Links one table to another by referencing the primary key of another table.
3. **Candidate Keys**  
   A field or combination of fields that can qualify as a primary key.
4. **Composite Key**  
   A key made up of two or more columns to uniquely identify a record.
5. **Alternate Key**  
   Candidate keys not chosen as the primary key.
6. **Super Key**  
   Any combination of fields that uniquely identifies a record (includes candidate keys).
7. **Surrogate Key**  
   A system-generated key (like an auto-incremented number) used as a substitute for a natural key.
8. **Natural Key**  
   A key derived from real-world data that uniquely identifies a record.

**Case study – 1**

An university has many departments

Each department has multiple instructors ; one among them is the head of the department

An instructor belongs to only one department

Each department offers multiple courses, each of them is taught by a single instructor

A student may enroll in different courses offered by different departments

Many courses taught by one instructor

Steps:

Identify the entities

Identify relevant attributes for every entity

Identify the key attributes for every entity

Find relationships between

Complete E-R diagram

Case study – 2

Create an E-R diagram to store cognizant candidates registration data

Following Registration details are captured

Candidate personal details - Name, DOB, Gender, email-id, phone-number

Parents (both mother & Father) details - Name, contact number

Address - Add line 1, Line 2, City, State, Zipcode

Qualification details (for 10th, 12th, Degree) - School/college name, Board / University, Year of Passing, Marks %