Modelling Databases: Part 1

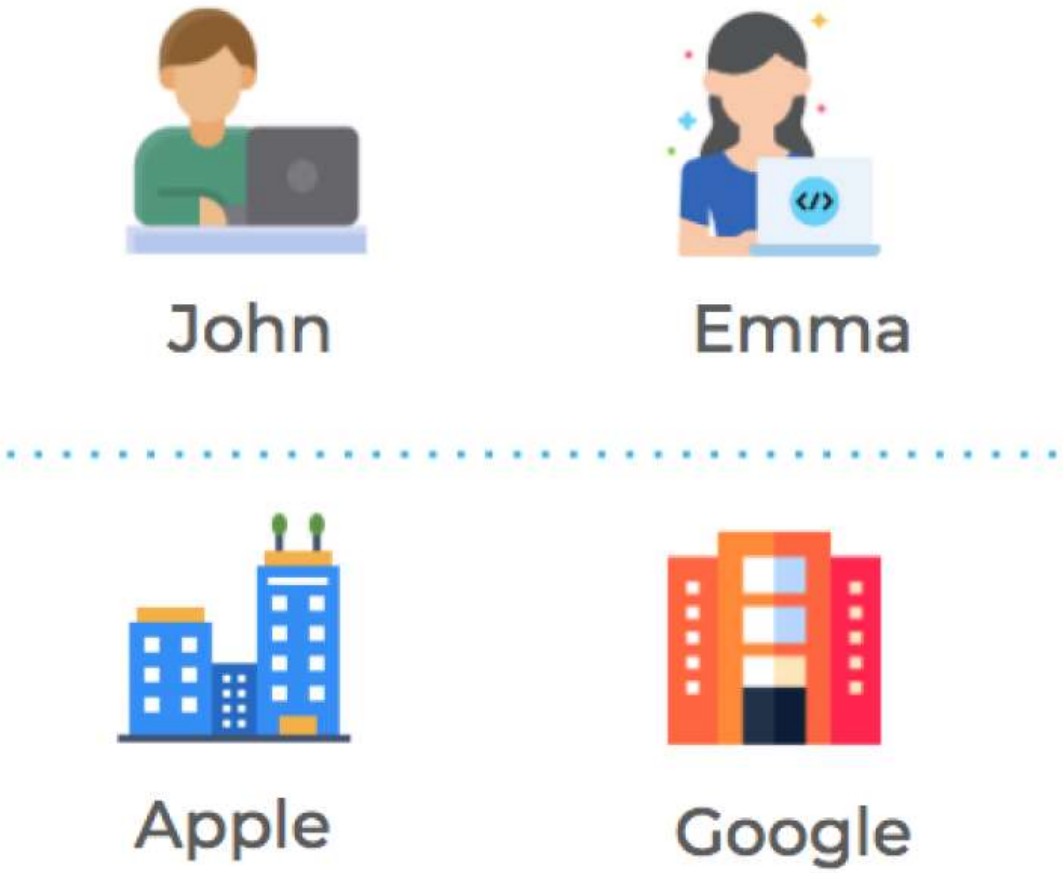
To model a database, we have to first understand the business requirements at conceptual level, which is later translated into a relational database.

For understanding the business requirements at a conceptual level, we use

`Entity Relationship Model (ER Model)` .

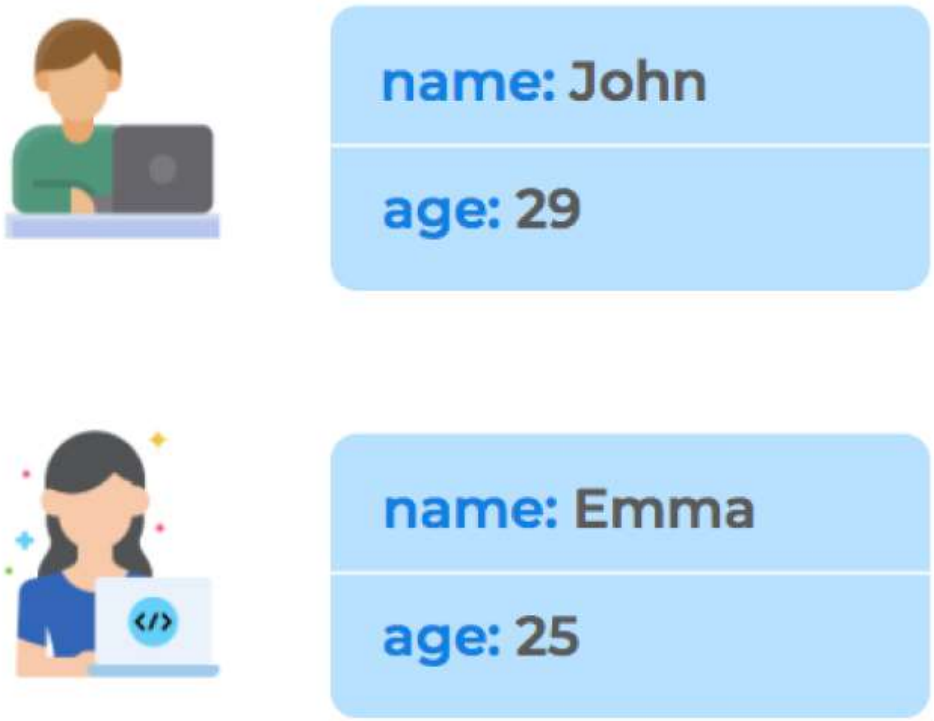
Core Concepts in ER Model

Entity



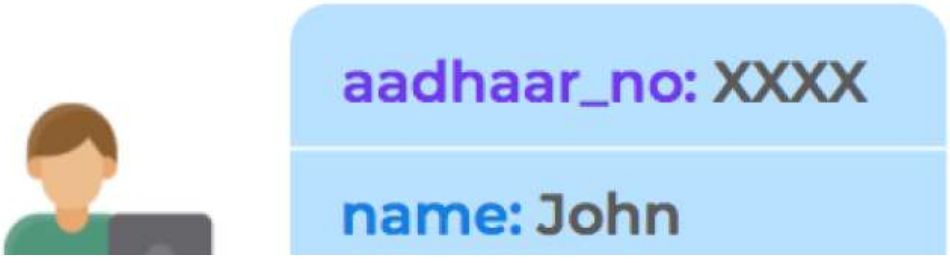
Real world objects/concepts are called **entities** in ER Model.

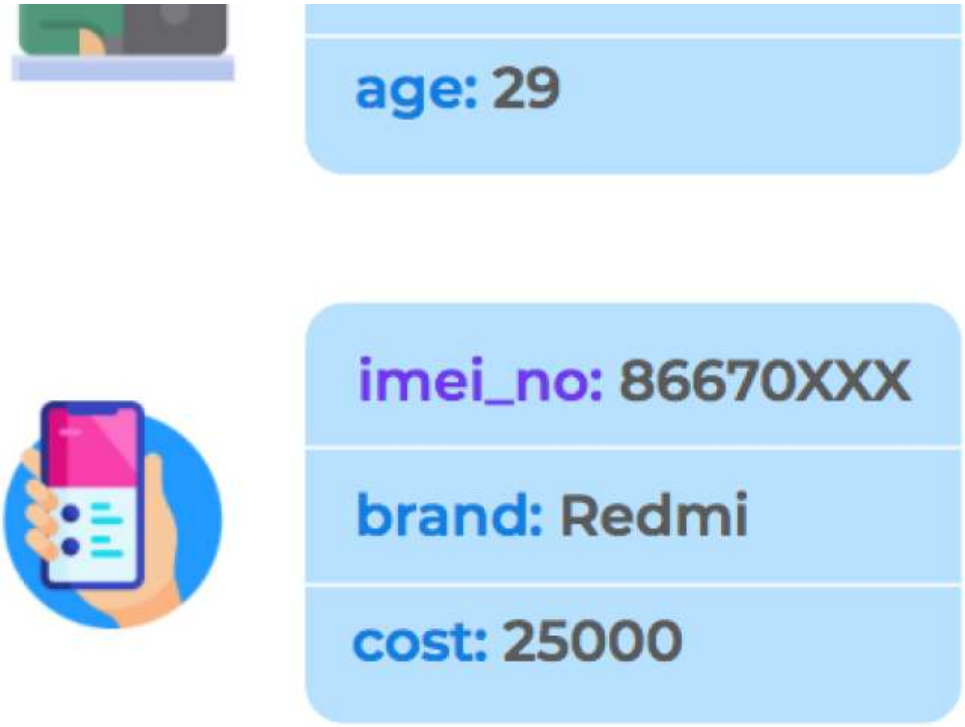
Attributes of an Entity



Properties of real world objects/concepts are represented as **attributes** of an entity in ER model.

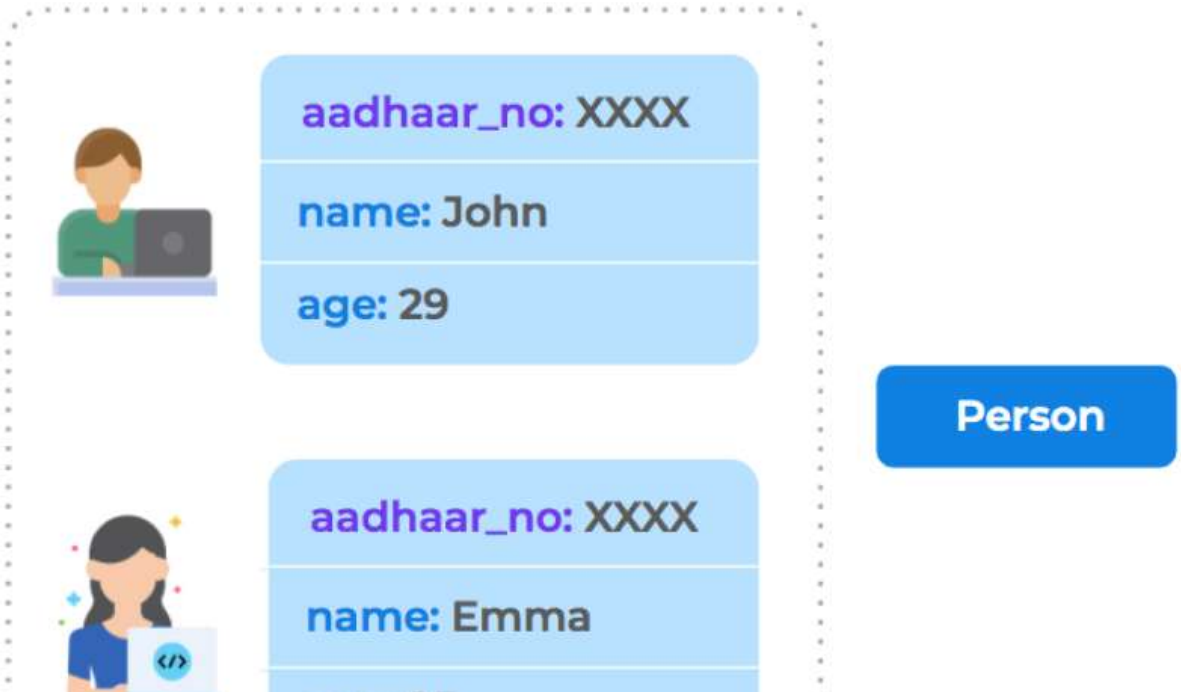
Key Attribute





The attribute that uniquely identifies each entity is called **key attribute**.

Entity Type



Entity Type is a **collection of entities** that have the same attributes (not values).

Relationships

Association among the entities is called a **relationship**.

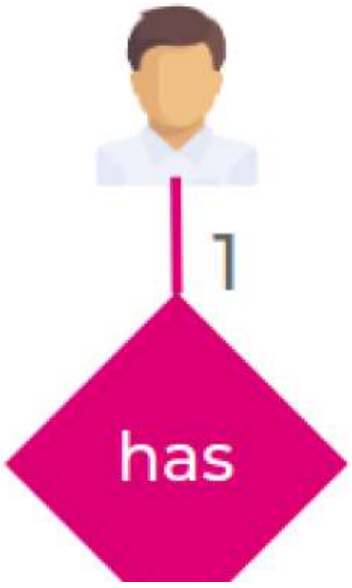
Example:

- Person **has a** passport.
- Person can **have many** cars.
- Each student can **register for many** courses, and a course can **have many** students.

Types of relationships

- One-to-One Relationship
- One-to-Many or Many-to-One Relationship
- Many-to-Many Relationship

One-to-One Relationship

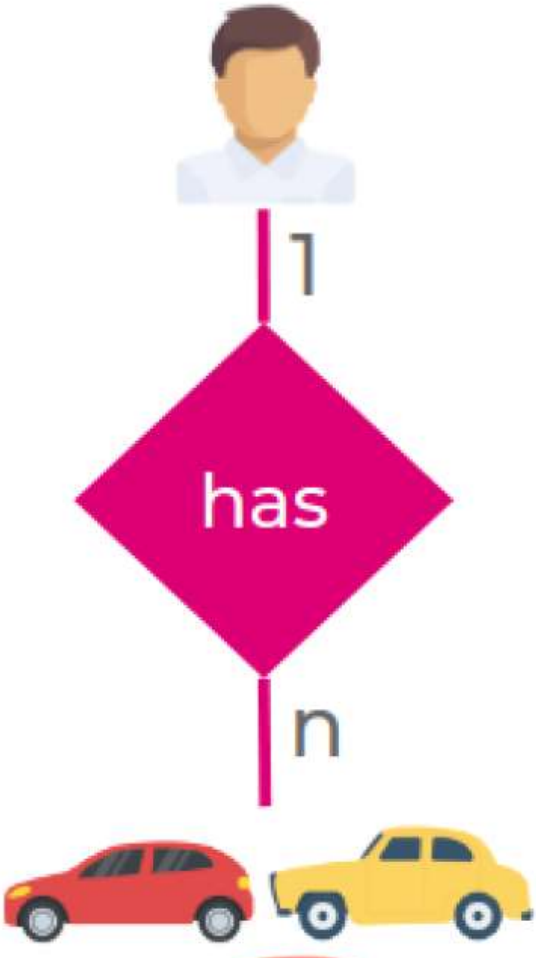


An entity is related to **only one entity**, and vice versa.

Example

- A person can have **only one** passport.
- similarly, a passport belongs to **one and only one** person.

One-to-Many Relationship

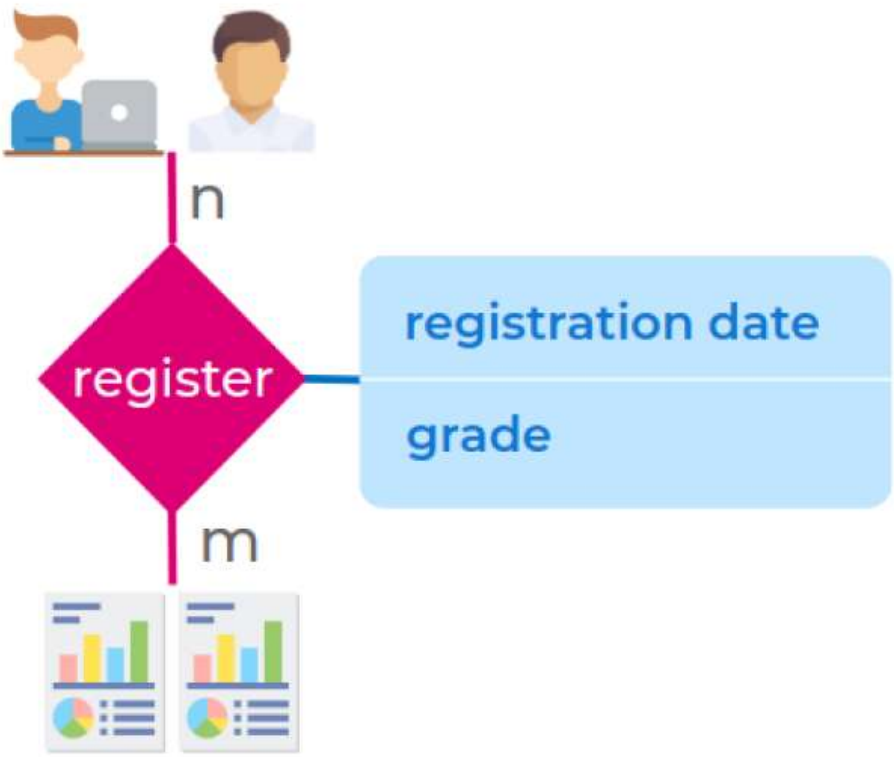


An entity is related to **many other** entities.

Example

- A person **can have many** cars. But a car belongs to **only one** person.

Many-to-Many Relationship



Multiple entities are related to multiple entities.

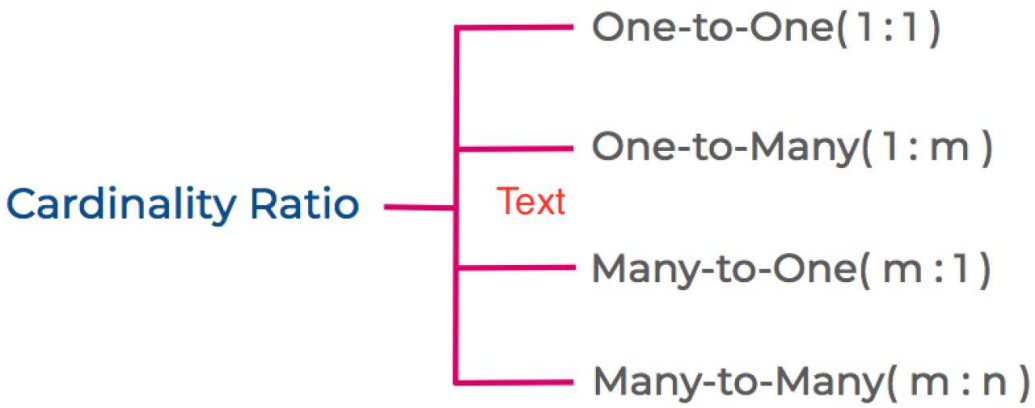
Example

- Each student can register to multiple courses.
- similarly each course is taken by multiple students

- Similarly each course is taken by multiple students.

Cardinality Ratio

Cardinality in DBMS defines the maximum number of times an instance in one entity can relate to instances of another entity.



MARK AS COMPLETED

Submit Feedback