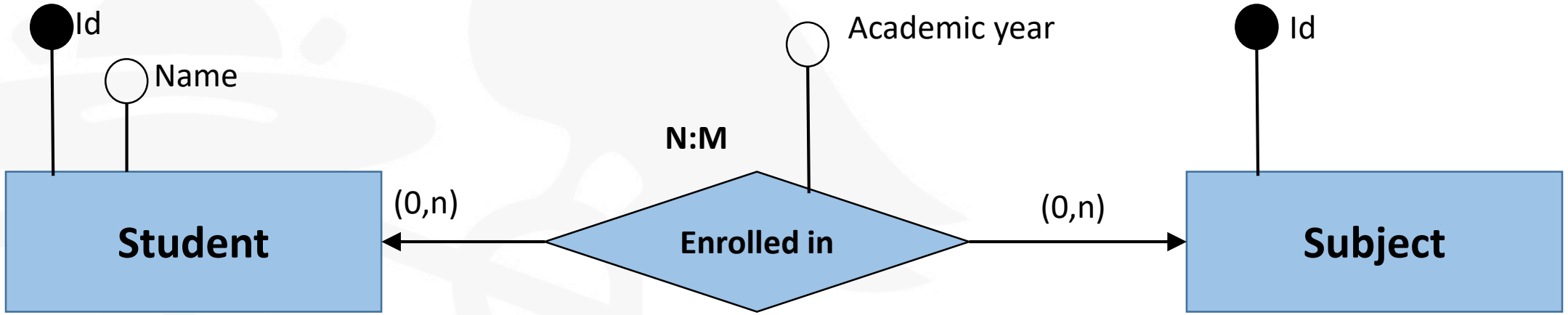
# Database Laboratory
# Lesson 2

Entity Relationship Model

# Introduction



# Entities

# Entities

- An entity is a representation of a "thing", "object" or "concept" of the real world with independent existence.
  - A person. (He/she differs from any other person).
  - A car. (Even if 2 cars has the same brand, the same model, ..., they will have different attributes, for example, the chassis number).
  - A house. (Although it might be exactly the same as another, they will have different addresses).

# Types of Entities

- Regular


  - A **regular** entity is one that can exist without any problem

- Strong


  - A **strong** entity is one that can exist and on which other entities depend

- Weak


  - A **weak** entity is an entity whose existence depends on the value of another entity

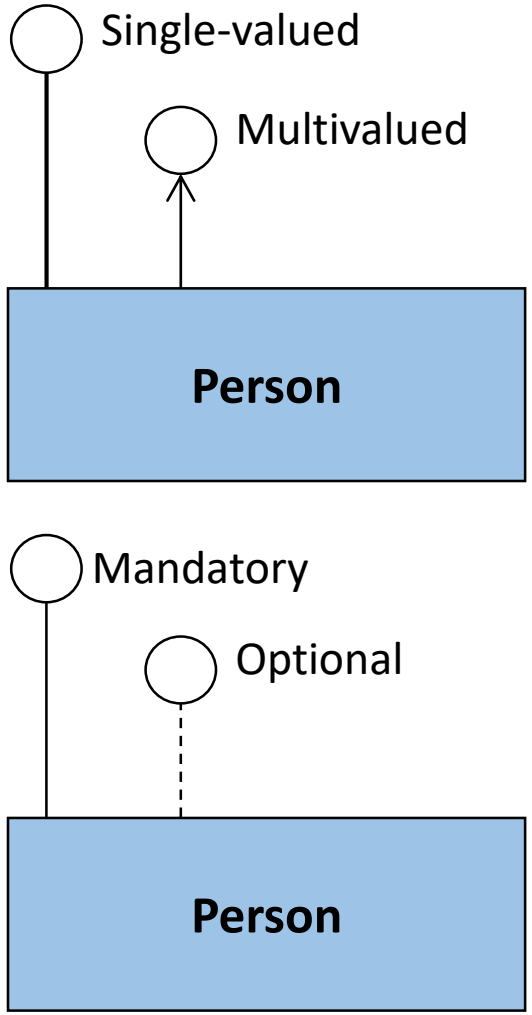
# Attributes

# Attributes

- Attributes are the properties that define an entity
- They take their values from a certain domain, which is the set of values allowed in an attribute
- For example:
  - Domain of dayOfTheWeek = {Monday, Tuesday, …, Sunday}
  - Domain of trafficLightColor = {red, green, orange}

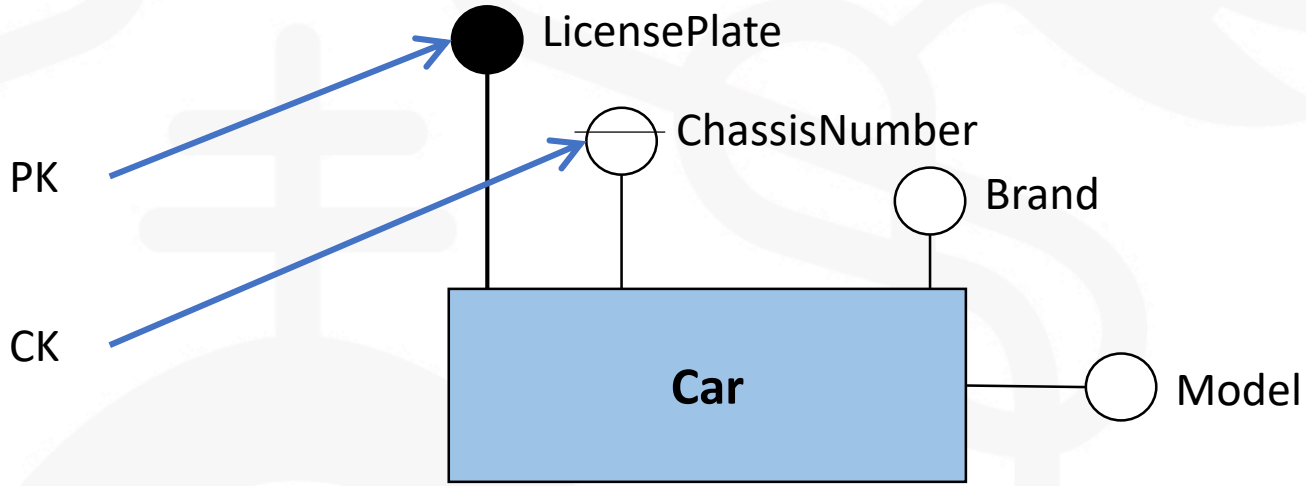
# Attributes - Types

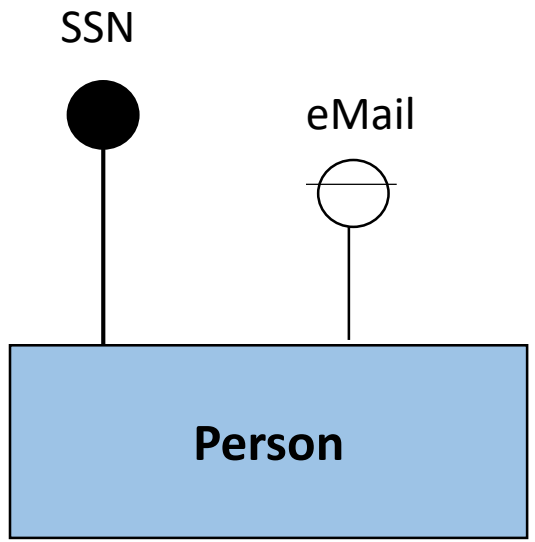
- Single-valued Vs. Multivalued
  - Single-valued: they only have a value
  - Multivalued: they have several values
- Mandatory Vs. Optional
  - Mandatory: they must have a value
  - Optional: they can be empty (null)



# Attributes - Keys

- Key
  - Primary Key (PK). Main identifier of the entity
  - Candidate Key (CK). Additional identifiers

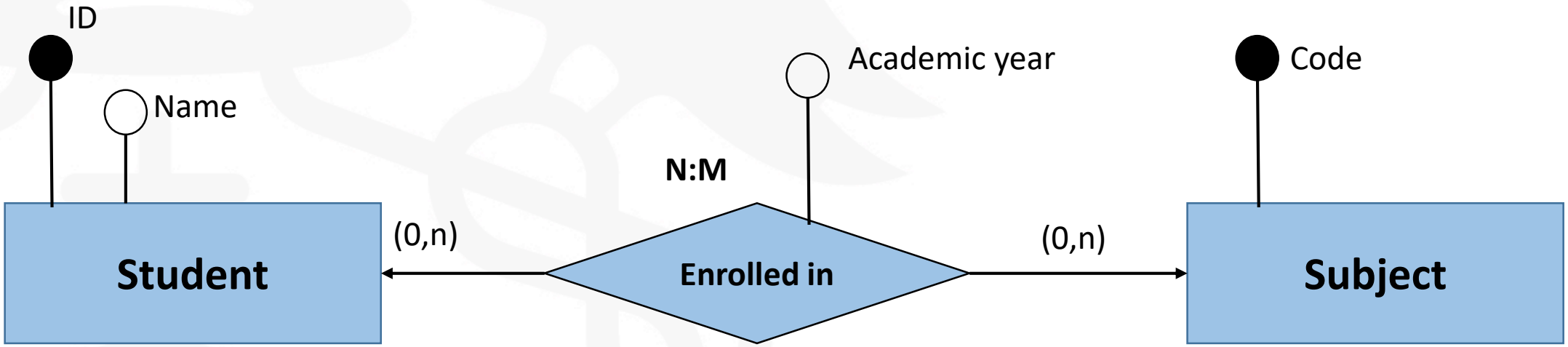


# Relationships

# Relationships

- Relationships are associations between entities
  - They can have attributes
  - They have a cardinality



# Relationships- Elements

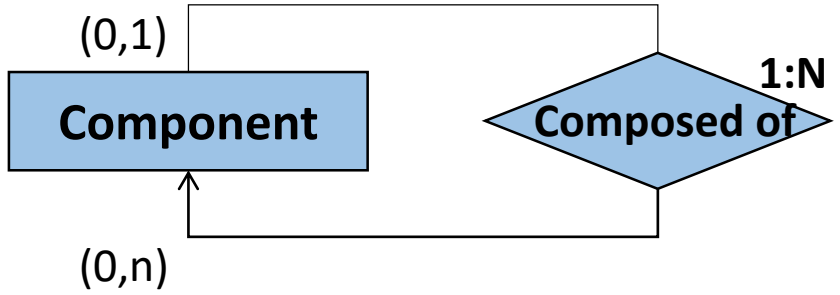
- Name
- Grade
  - Number of associated entities: one (reflexive), two (binary), three (ternary), …
- Cardinality
  - For binary relationships: 1:1, 1:N, N:M
- Role
  - Role played by each entity
  - It is normally something implicit. It is typically used in reflexive relationships

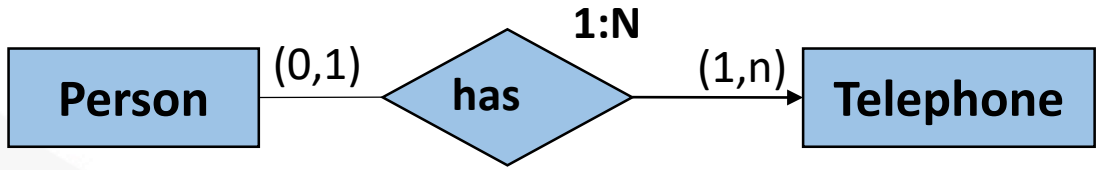
# Relationships - Cardinality

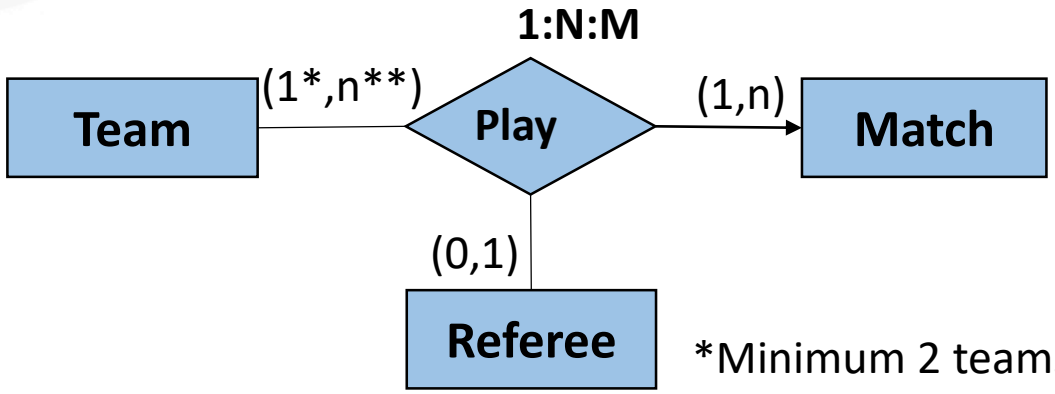
- Minimum values: 0,1
- Maximum values: 1,n
  - Combinations: (0,1), (1,1), (0,n), (1,n)
- Cardinality of the relationship (binary): 1:1, 1:N, N:M
- In case of minimum / maximum values different from the previous ones, we have to write footnote:
  - Minimum: 11= 1*
  - Maximum: 2= n*
- Relationships with a degree greater than 2 are quite difficult to handle
  - Sometimes, they can be transformed into several binary relationships. Other times it is not possible

# Relationships - Types

- Reflexive
  - It relates one entity
- Binary
  - It relates 2 entities
- Ternary
  - It relates 3 entities



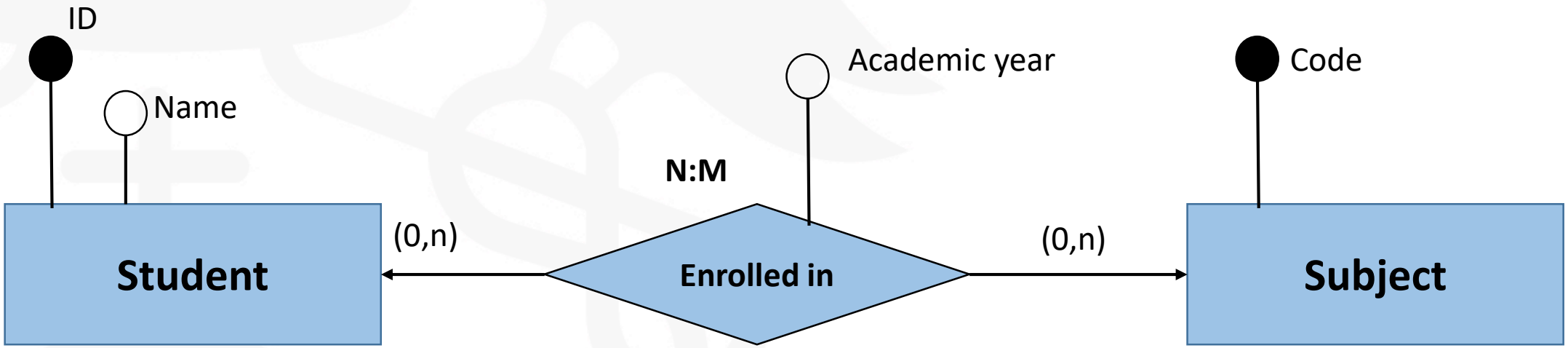




*Minimum 2 teams
**Maximum 2 teams

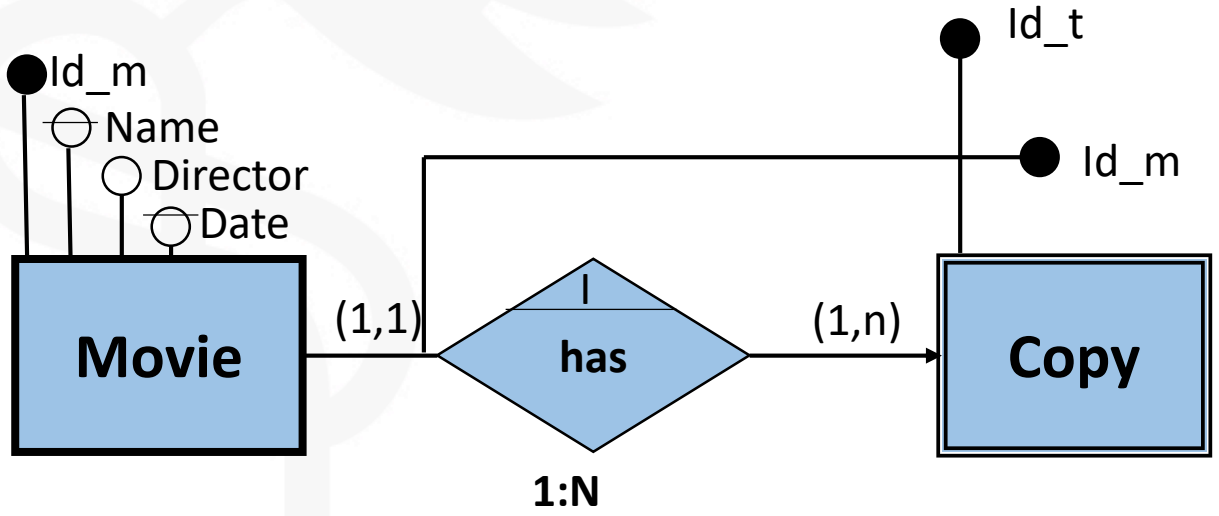
# Relationships - Types

- **Regular**
  - **It associates regular entities**
- Identifying



# Relationships - Types

- Regular
- **Identifying**
  - **There is a weak entity that cannot be uniquely identified by its attributes but needs to add the key of the strong entity on which it depends**



# Exercises

# Exercise 1

- Design a relational database for a company that has to manage employees, departments and projects.
  - The company is organized into departments. Each department has a number and name that are unique and there is a certain employee who manage it. We are interested in the date on which that employee began to manage the department. In addition a department can be distributed in several places of the company.
  - Each department controls a certain number of projects. Each project has a unique name and number, and is carried out in a single place.
  - The name, social security number, address, salary, gender and date of birth will be stored for each employee. Every employee is assigned to a department, but can work on several projects, which will not necessarily be controlled by the same department. We are interested in the number of hours per week an employee works on each project and we are also interested in who is the direct supervisor of each employee.
  - We also want to keep data of the family members of each employee in order to manage their insurance. The name, gender, date of birth and family relationship will be stored for each family member.

# Exercise 2

- A video club has decided to use a database to store information regarding the films it offers for rent. The information is as follows:
  - A film is characterized by its title, nationality, producer, date and role played by the main actors.
  - Several actors can participate in a movie (name, nationality, gender), some of them as main actors.
  - A film is directed by a director (name, nationality).
  - Each film has one or several copies identified by a copy number and characterized by their state of preservation.
  - A copy can be found rented to a client (ID, name, address, telephone). We want to store the rental start date and the return date.
  - A client has to be endorsed by another client who responds to him in case of problems in the rental.

# Exercise 3

- Design a database with a schema that captures all the information that art galleries need to maintain.
  - Galleries keep information about artists, their names (which are unique), birthplaces, age, and style of art.
  - For each piece of artwork, the artist, the year it was made, its unique title, its type of art (e.g., painting, lithograph, sculpture, photograph), and its price must be stored. Pieces of artwork are also classified into groups of various kinds, for example, portraits, still lifes, works by Picasso, or works of the 19th century; a given piece may belong to more than one group. Each group is identified by a name (like those just given) that describes the group.
  - Finally, galleries keep information about customers. For each customer, galleries keep that person's unique name, address, total amount of dollars spent in the gallery (very important!), and the artists and groups of art that the customer tends to like. Draw the ER diagram for the database.