GLA UNIVERSITY
ESTD 2010
ऋते ज्ञानान्न मुक्तिः
GLA
UNIVERSITY
MATHURA
Recognised by UGC Under Section 2(f)

DATABASE MANAGEMENT SYSTEM BCSC0003

By

Suman Kumar Bhattacharyya

Assistant Professor

Computer Engineering and Application Department

Today we will discuss

✓Mapping Constraints

✓Key

✓Specialization, Generalization and Aggregation.

Mapping Constraints

- A mapping constraint is a data constraint that expresses the number of entities to which another entity can be related via a relationship set.
- It is most useful in describing the relationship sets that involve more than two entity sets.
- Mapping constraints can be explained in terms of <u>mapping cardinality</u>

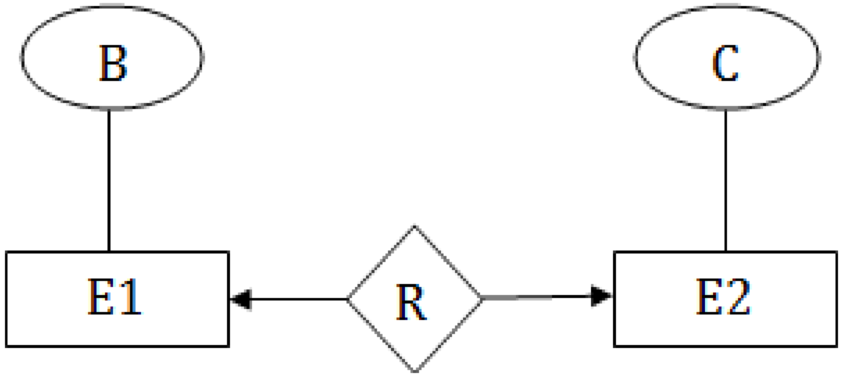
Mapping Cardinalities

For binary relationship set R on an entity set A and B, there are four possible mapping cardinalities. These are as follows:

- ✓ One to one (1:1)
- ✓ One to many (1:M)
- ✓ Many to one (M:1)
- ✓ Many to many (M:M)

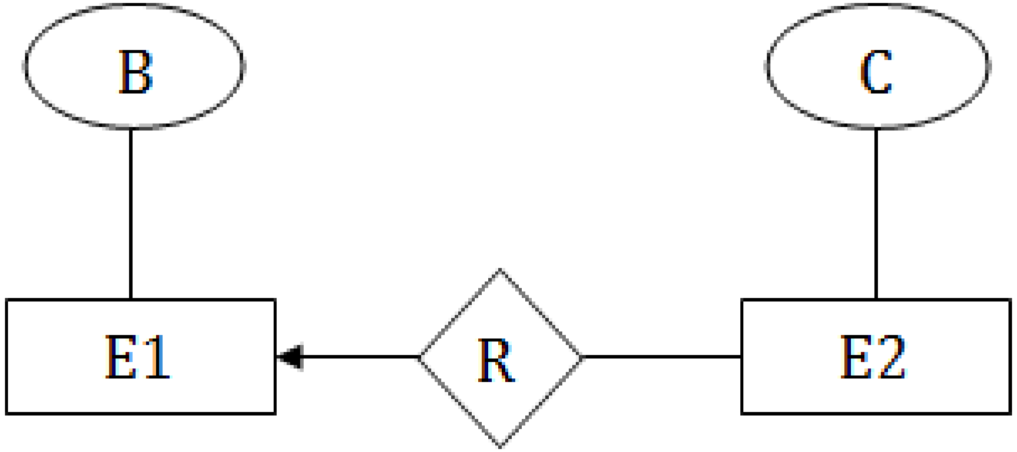
One to one (1:1)

- ✓ In one-to-one mapping, an entity in E1 is associated with at most one entity in E2, and an entity in E2 is associated with at most one entity in E1.



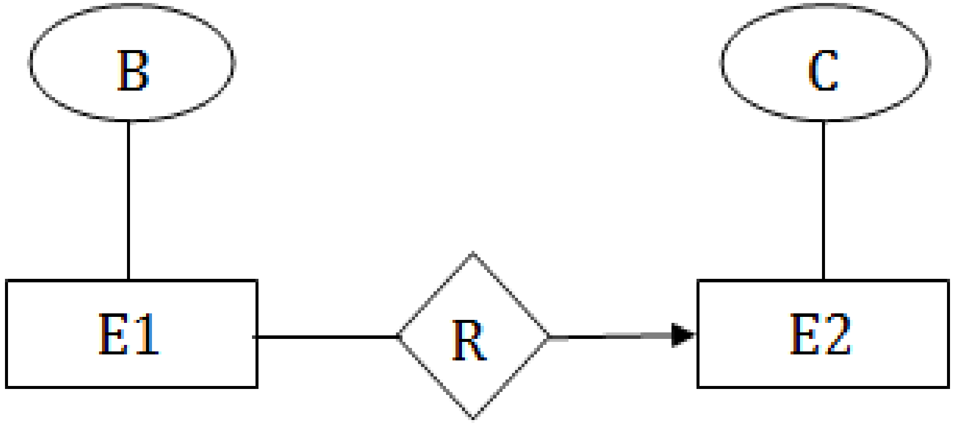
One to many (1:M)

- ✓ In one-to-many mapping, an entity in E1 is associated with any number of entities in E2, and an entity in E2 is associated with at most one entity in E1.



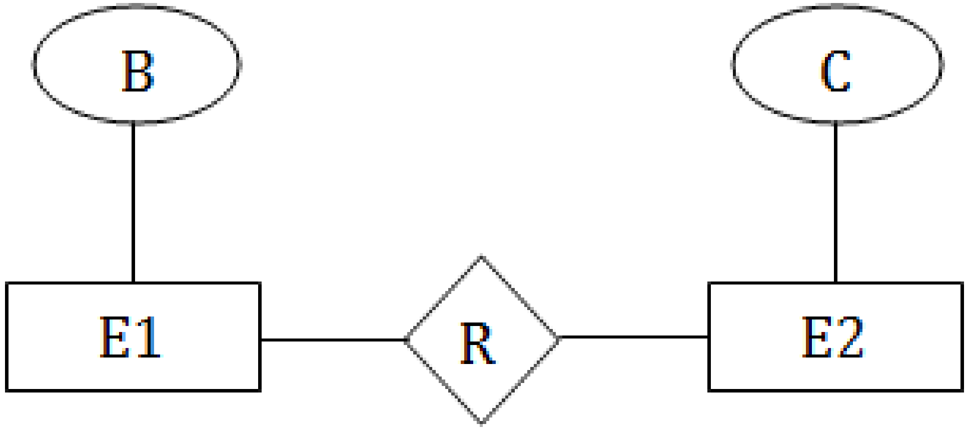
Many to one (M:1)

- ✓ In one-to-many mapping, an entity in E1 is associated with at most one entity in E2, and an entity in E2 is associated with any number of entities in E1.



Many to many (M:N)

- ✓ In many-to-many mapping, an entity in E1 is associated with any number of entities in E2, and an entity in E2 is associated with any number of entities in E1.



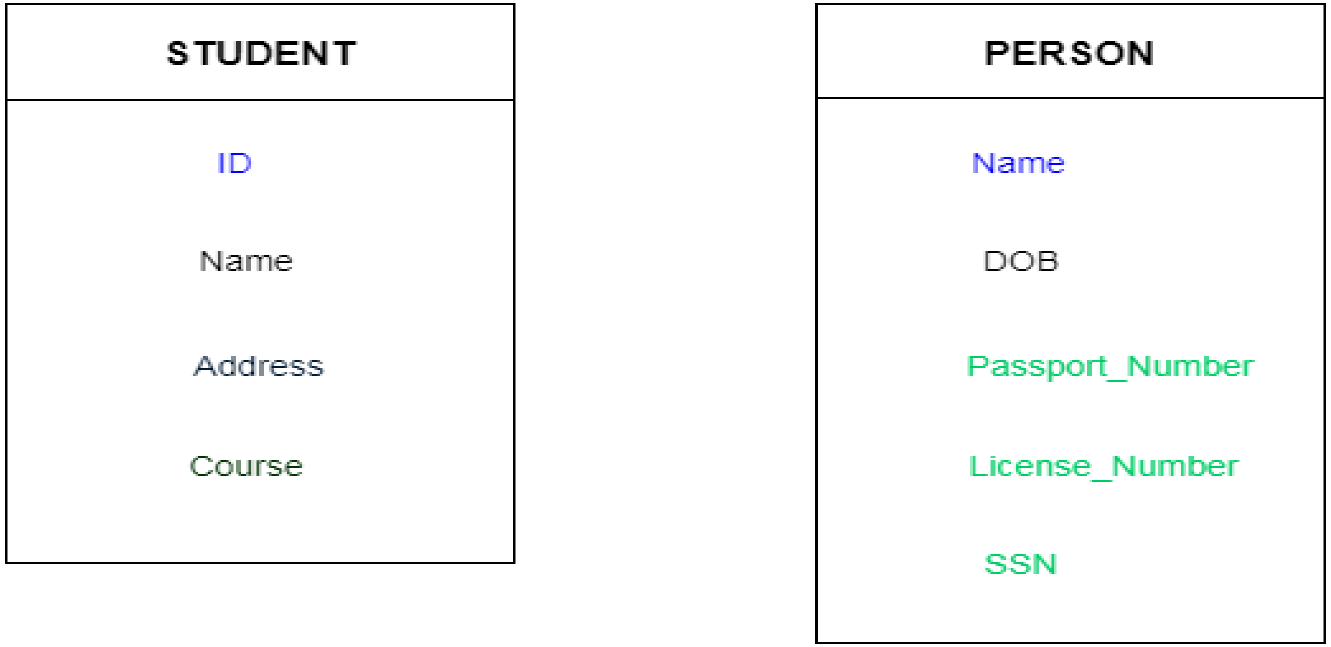
Keys

- Keys play an important role in the relational database.
- It is used to uniquely identify any record or row of data from the table.
- It is also used to establish and identify relationships between tables.

For example: In Student table, ID is used as a key because it is unique for each student.

Keys

- ✓ In PERSON table, passport_number, license_number, SSN are keys since they are unique for each person.

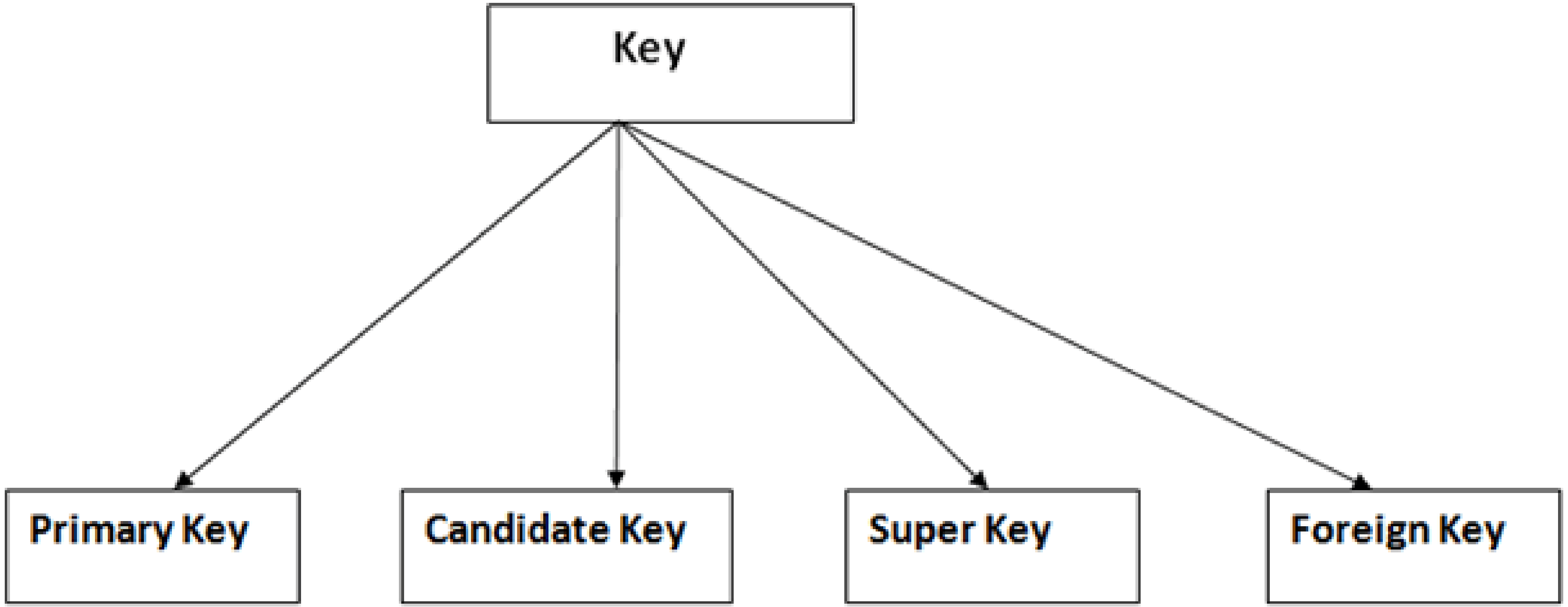


STUDENT
ID
Name
Address
Course

PERSON
Name
DOB
Passport_Number
License_Number
SSN

Keys

- ✓ **Types of Key in the database,**



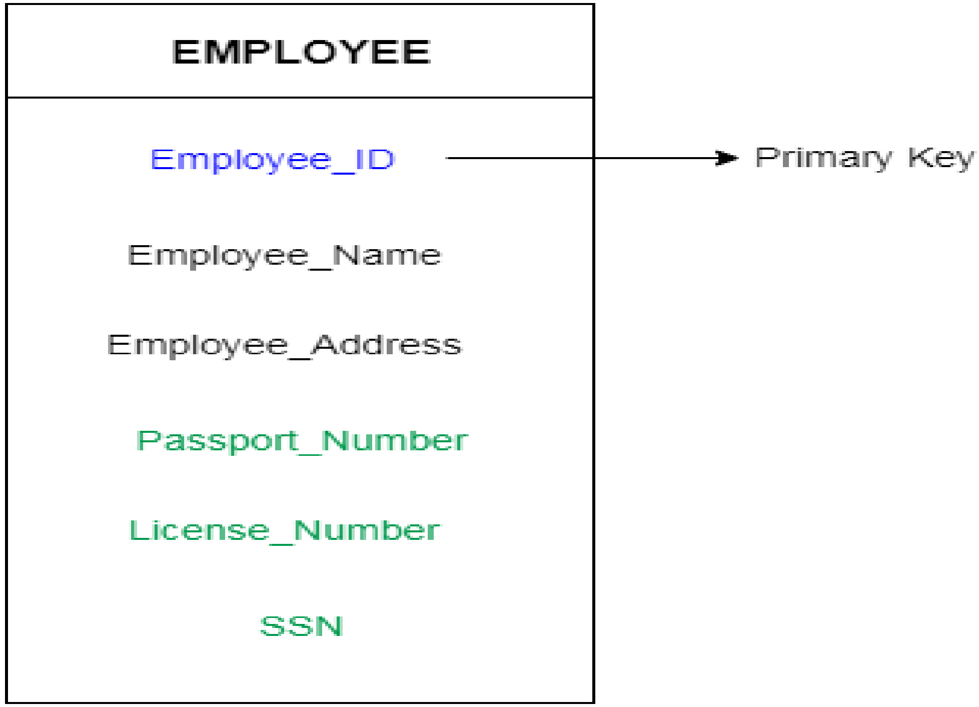
Primary Keys

- ➢ It is the first key which is used to identify one and only one instance of an entity uniquely.

- ✓ In the EMPLOYEE table, ID can be primary key since it is unique for each employee. In the EMPLOYEE table, we can even select License_Number and Passport_Number as primary key since they are also unique.

Primary Keys

- ✓ For each entity, selection of the primary key is based on requirement and developers.



EMPLOYEE

Employee_ID → Primary Key

Employee_Name

Employee_Address

Passport_Number

License_Number

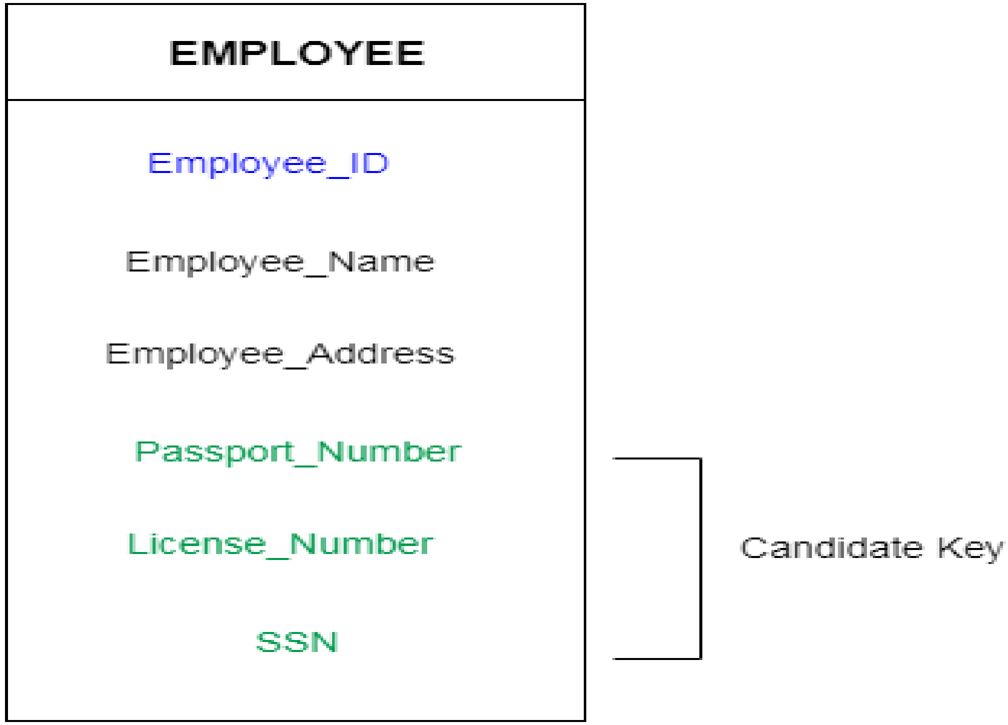
SSN

Candidate key

- A candidate key is an attribute or set of an attribute which can uniquely identify a tuple.
- The remaining attributes except for primary key are considered as a candidate key.
- The candidate keys are as strong as the primary key.

Candidate key

- ✓ **For example:** In the EMPLOYEE table, id is best suited for the primary key. Rest of the attributes like SSN, Passport_Number, and License_Number, etc. are considered as a candidate key.



Super Key

- ➢ Super key is a set of an attribute which can uniquely identify a tuple. Super key is a superset of a candidate key.

✓ **For example:**

In the above EMPLOYEE table, for(EMPLOEE_ID,EMPLOYEE_NAME) the name of two employees can be the same, but their EMPLYEE_ID can't be the same. Hence, this combination can also be a key.

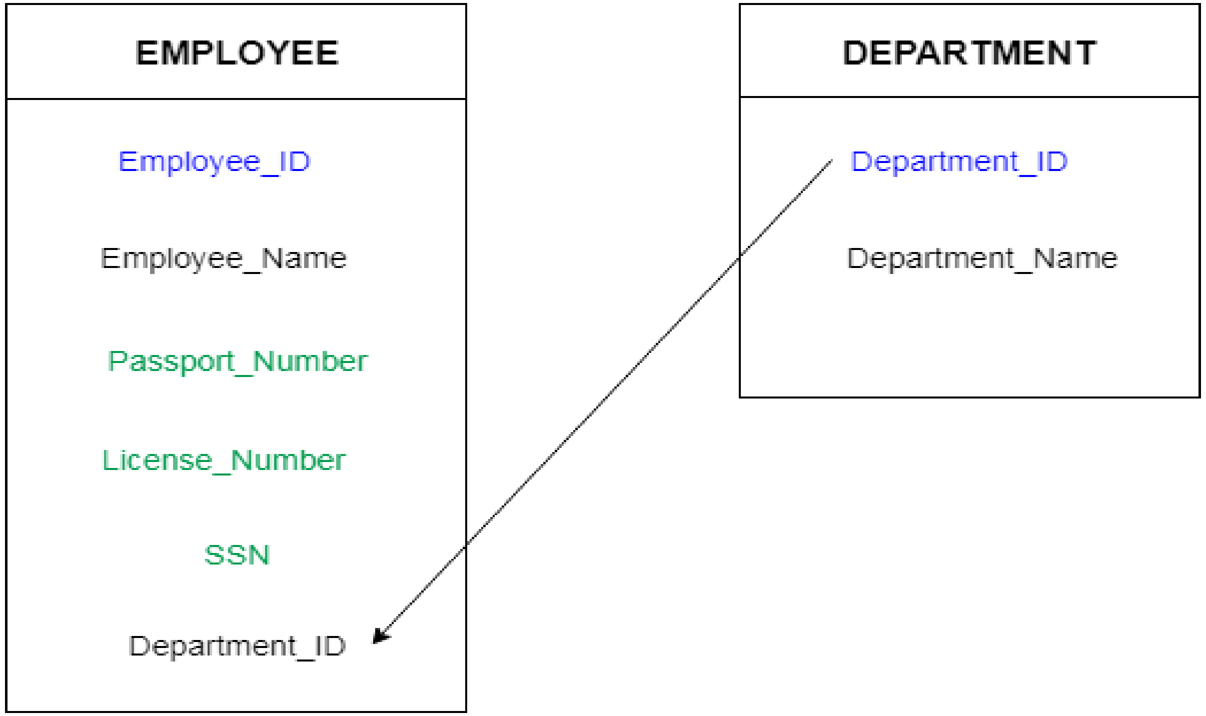
The super key would be EMPLOYEE-ID, (EMPLOYEE_ID, EMPLOYEE-NAME), etc.

Foreign key

- Foreign keys are the column of the table which is used to point to the primary key of another table.
- In a company, every employee works in a specific department, and employee and department are two different entities.
- So we can't store the information of the department in the employee table. That's why we link these two tables through the primary key of one table.
- We add the primary key of the DEPARTMENT table, Department_Id as a new attribute in the EMPLOYEE table.

Foreign key

- ✓ Now in the EMPLOYEE table, Department_Id is the foreign key, and both the tables are related.



Specialization, Generalization, Aggregation

As the complexity of data increased, it became more and more difficult to use the traditional ER Model for database modelling.

Improvements or enhancements were made to the existing ER Model to make it able to handle the complex applications better.

Hence, as part of the **Enhanced ER Model**, along with other improvements, three new concepts were added to the existing ER Model, they were:

- ➢ Generalization
- ➢ Specialization
- ➢ Aggregation

Specialization

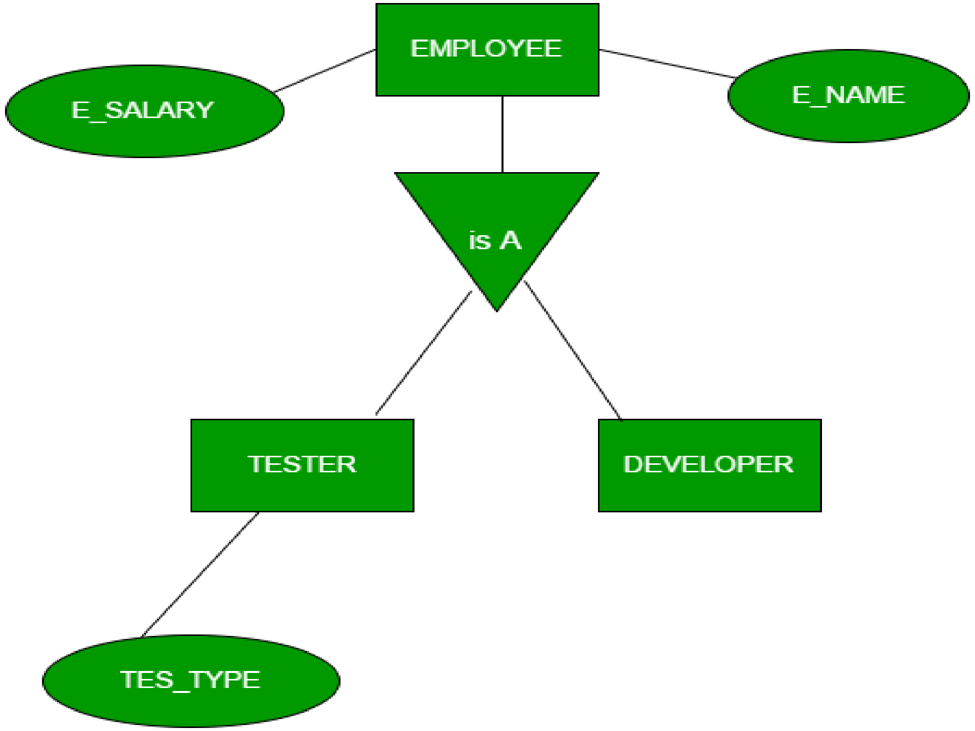
In specialization, an entity is divided into sub-entities based on their characteristics.

It is a top-down approach where higher level entity is specialized into two or more lower level entities.

- ✓ For Example, EMPLOYEE entity in an Employee management system can be specialized into DEVELOPER, TESTER

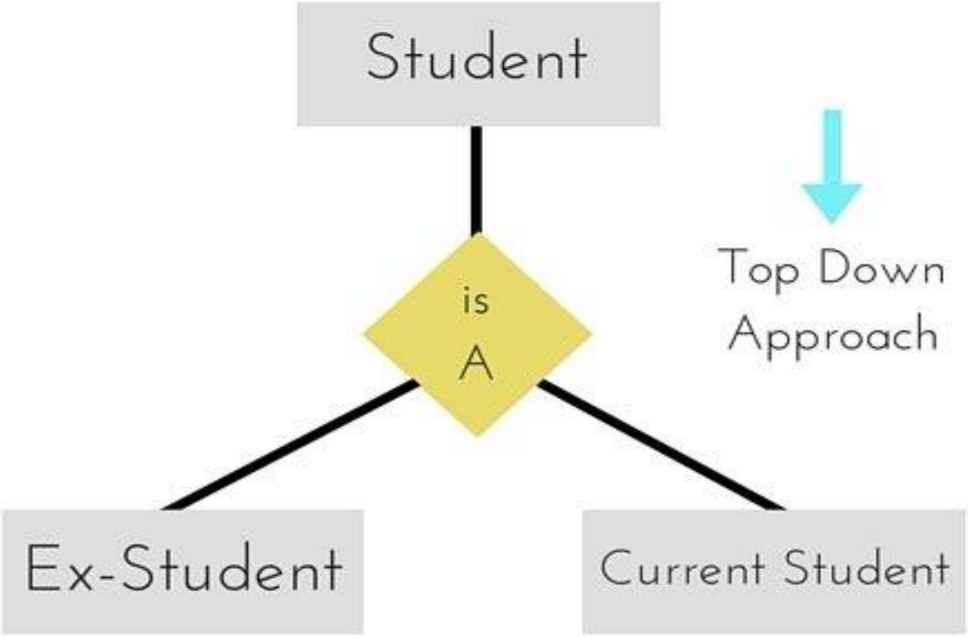
Specialization

In this case, common attributes like E_NAME, E_SAL etc. become part of higher entity (EMPLOYEE) and specialized attributes like TES_TYPE become part of specialized entity (TESTER).



Specialization

Specialization



Generalization

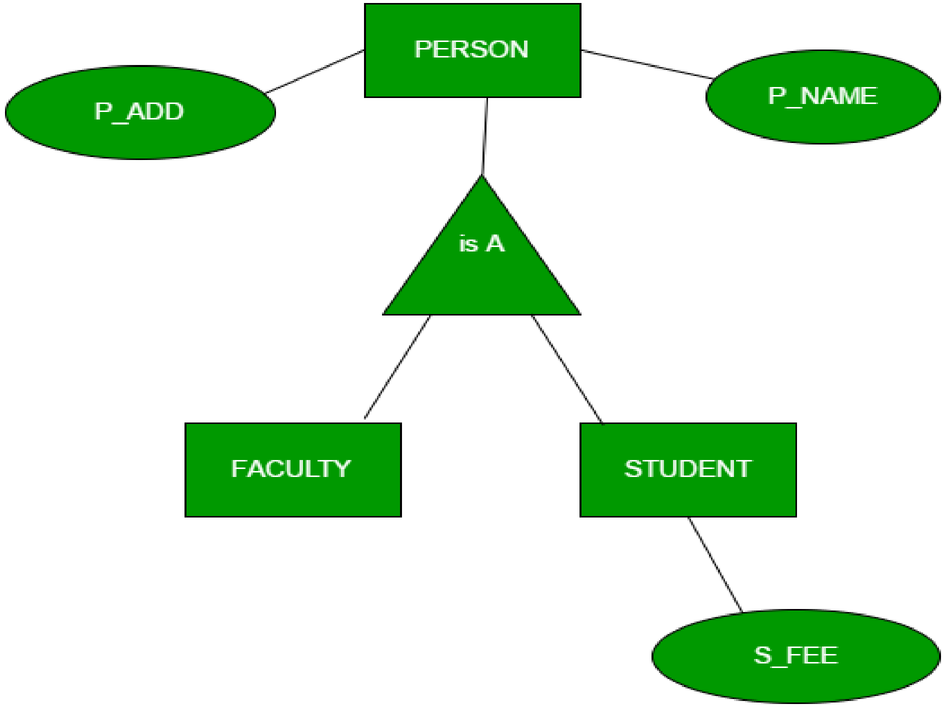
Generalization is the process of extracting common properties from a set of entities and create a generalized entity from it.

It is a bottom-up approach in which two or more entities can be generalized to a higher level entity if they have some attributes in common.

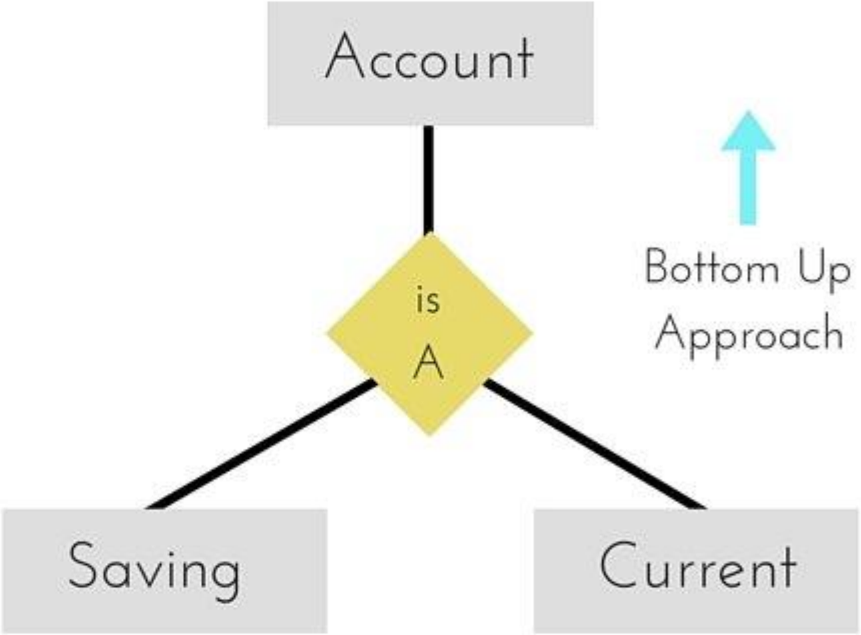
- ✓ For Example, STUDENT and FACULTY can be generalized to a higher level entity called PERSON as shown in Figure 1.

Generalization

In this case, common attributes like P_NAME, P_ADD become part of higher entity (PERSON) and specialized attributes like S_FEE become part of specialized entity (STUDENT).



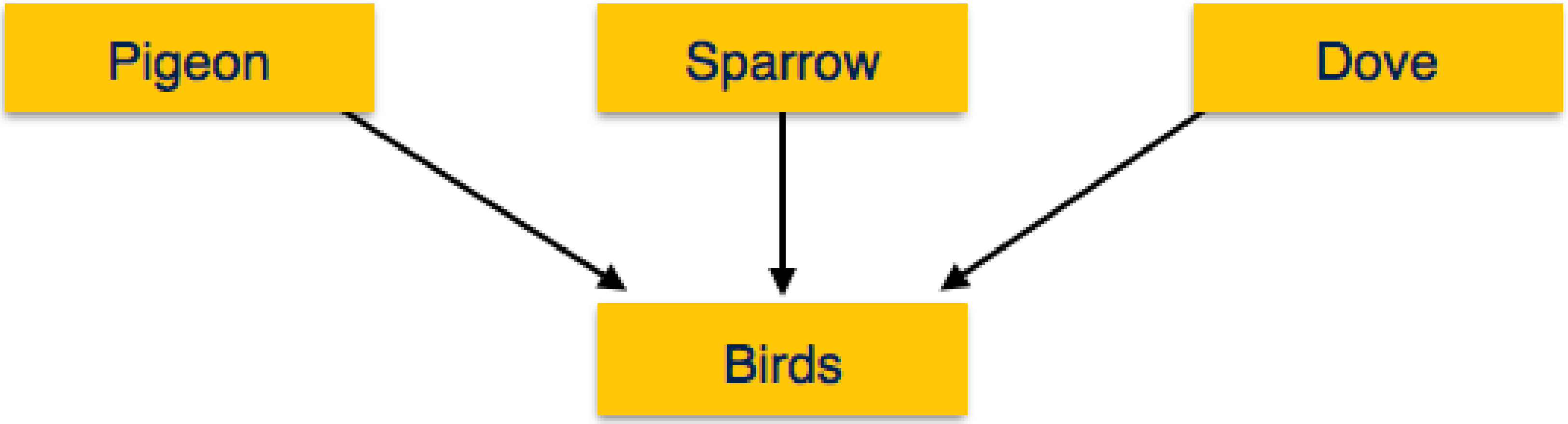
Generalization



Generalization

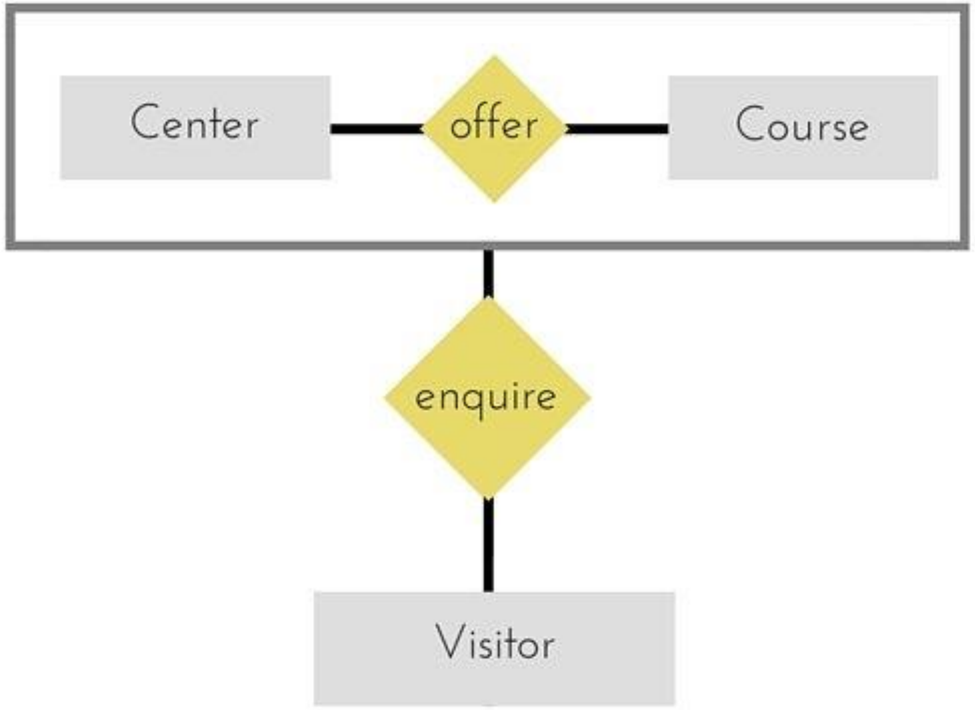
In generalization, a number of entities are brought together into one generalized entity based on their similar characteristics.

- ✓ For example, pigeon, house sparrow, crow and dove can all be generalized as Birds.



Aggregation

- ✓ Aggregation is a process when relation between two entities is treated as a **single entity**.



Aggregation

- ✓ Relationship between **Center** and **Course** together, is acting as an Entity, which is in relationship with another entity **Visitor**.

- ✓ If a Visitor or a Student visits a Coaching Center, he/she will never enquire about the center only or just about the course, rather he/she will ask enquire about both

Thanks