# Enhanced ER (EER) Notations

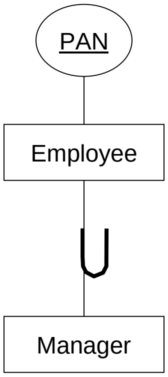
## Subclasses and Inheritance

- An entity class B is said to be subclass of another entity class A, if it shares an *is-a* relationship with A.
  - *Example1: car “is-a” vehicle*
  - *Example2: Monkey “is-a” Primate “is-a” Animal*
  - *Example3: Manager “is-a” Employee*
- An entity of class B is said to be specialization of entities of class A. Conversely, an entities of class A are generalization of class B entities.

## Subclasses and Inheritance

- An Entity cannot exist in database that merely belongs to a subclass. It has to belong to the super class as well.
- Subclasses undergo *type inheritance of the super class. That is, each member of subclass has the same attributes as super class entities and participate in the same relationship types.*

## Subclasses and Inheritance

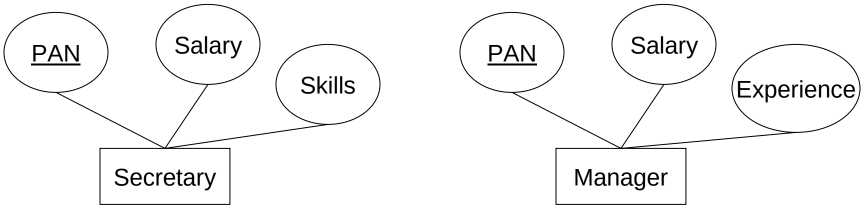


- Depicts the relation Manager *is-a* Employee
- Manager inherits PAN as a key attribute
- Other attribute in addition to PAN may form the key for Manager Entity

# Specialization and Generalization

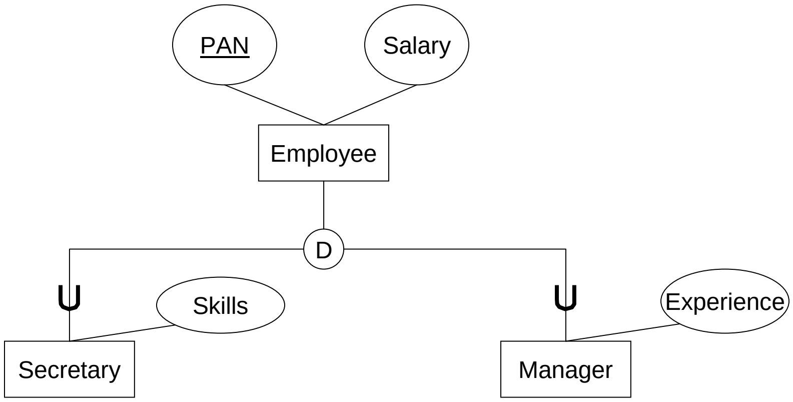
- The processes of creating subclasses out of a given entity type is called *specialization*
- The reverse process of taking two or more entity types and clubbing them under a common super class is called *generalization*

# Generalization



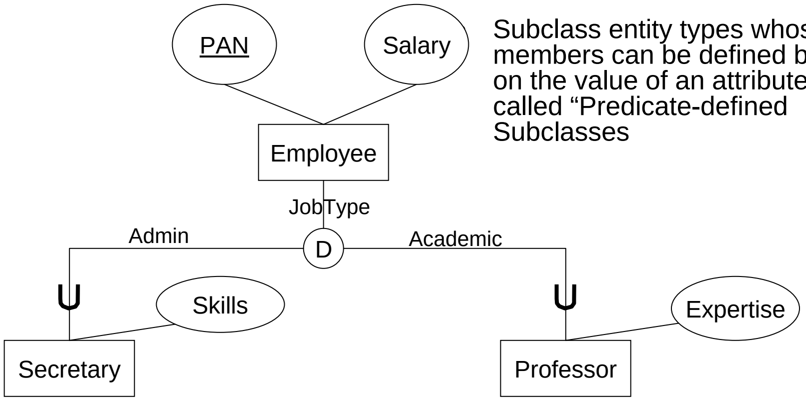
- Consider the above already existing (legacy) Entity type

# Generalization



- Creation of Generalized class

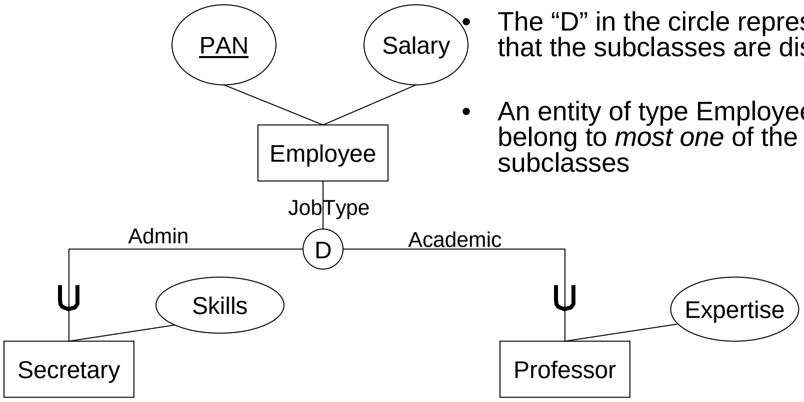
# Predicate-defined Subclasses



Subclass entity types whose members can be defined based on the value of an attribute are called “Predicate-defined Subclasses

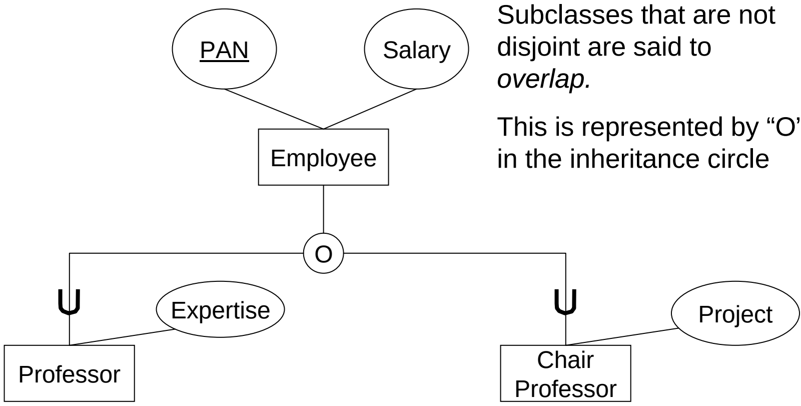
- The attribute “JobType” above, is called the defining predicate

## Disjoint and Overlapping Subclasses



- The “D” in the circle represents that the subclasses are disjoint
- An entity of type Employee may belong to *most one* of the subclasses

## Disjoint and Overlapping Subclasses

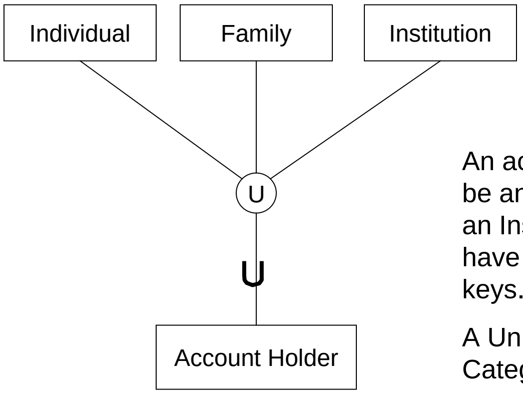


Subclasses that are not disjoint are said to *overlap.*

This is represented by “O” in the inheritance circle

- Disjoint or overlap may be either partial or total

# Union Types or Categories



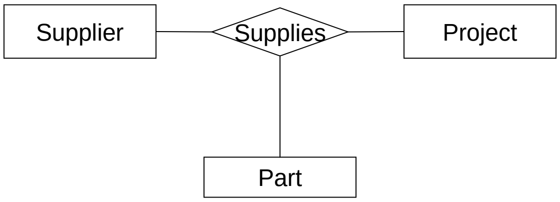
An account holder can either be an Individual, or a Family or an Institution, each of whom have their own attributes and keys.

A Union type is also called a Category

# Union Types or Categories

- In the previous example not every individual, family or institution listed in database may be an account holder.
  - In such cases “Account Holder” is said to be a “partial” union. Usually a constraint is specified that determines which entity participates in the union.

# Higher-order Relationships



Supplier

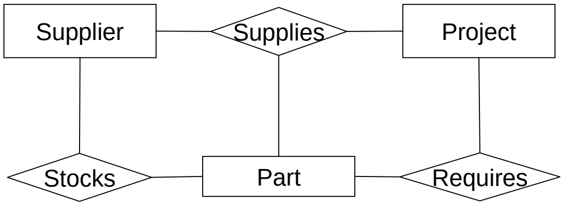
Supplies

Project

Part

- Depicts the semantics that supplier entity supplies part entity that is required by project entity
- Cannot be reduced to three binary relationships without losing meaning

# Higher-order Relationships



Supplier

Supplies

Project

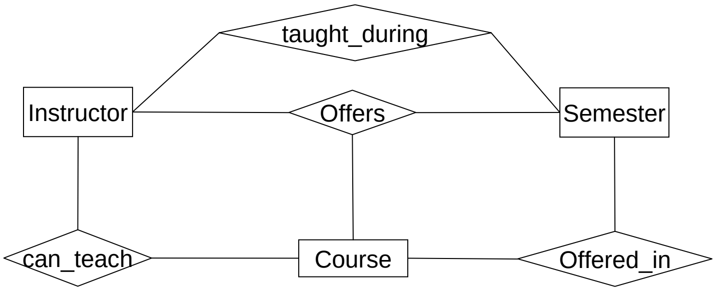
Stocks

Part

Requires

- Closest meaningful binary relationship scheme.
- Note that part is weak entity type and participates fully in the stocks and requires relationships.

# Higher-order Relationships



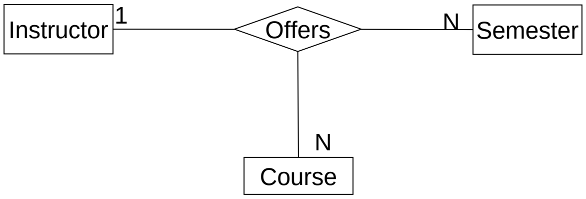
For any given Instance of Instructor (i), Semester (s) and Course (c) note that:

Offers(i,s,c) => taught_during(i,s) && can_teach(i,c) && Offered_in(c,s)

But the Converse need not be true!

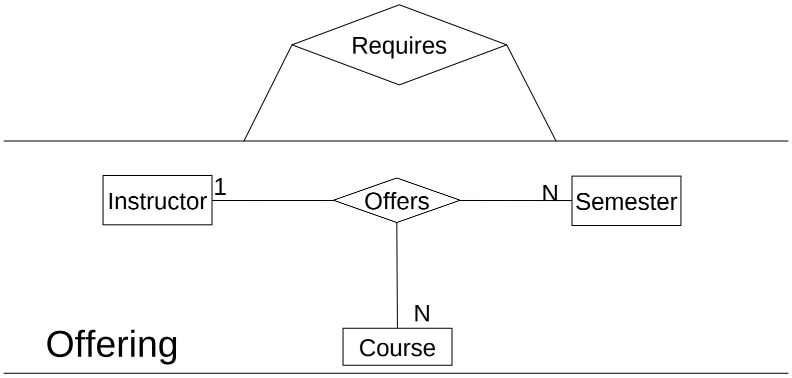
# Higher-order Relationships

Cardinality Constraints



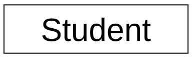
- Above Constraints says that: A given course-semester combination should have only one instructor, however, an instructor may be associated with many course-semester pairs.
- Course-semester pair forms the key for this relationship.

# Aggregation (KM concept)

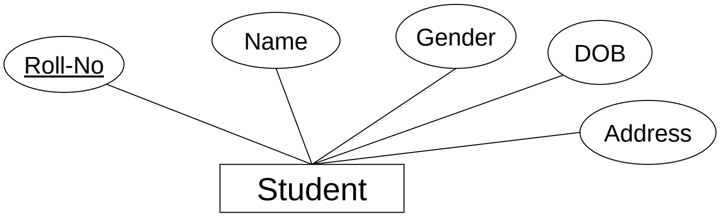


# An Example University Database

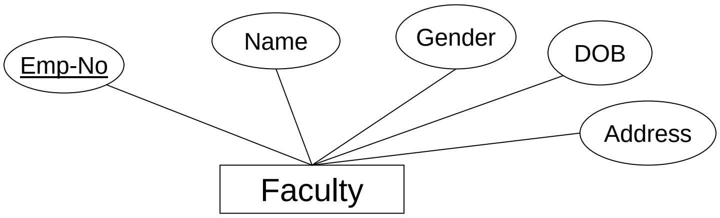
## Some Basic Entity Types



## Some Basic Entity Types



# Some Basic Entity Types



Emp-No

Name

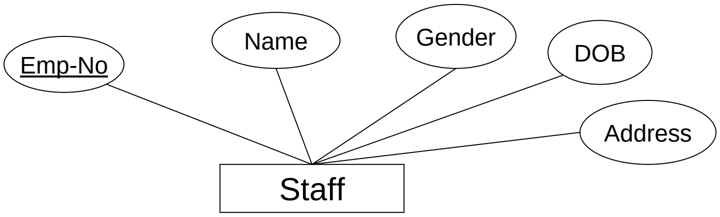
Gender

DOB

Address

Faculty

# Some Basic Entity Types



Emp-No

Name

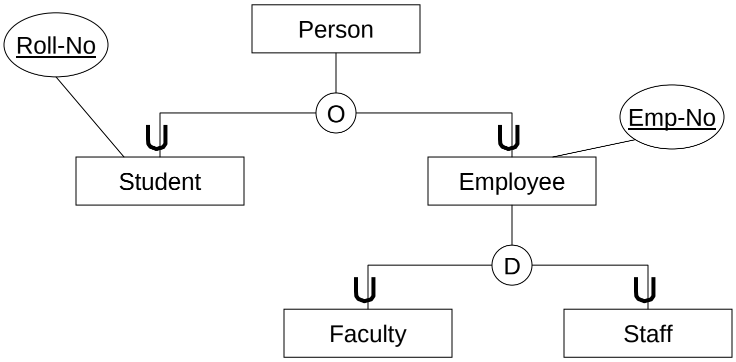
Gender

DOB

Address

Staff

# Generalization



Person

Roll-No

O

Emp-No

Student

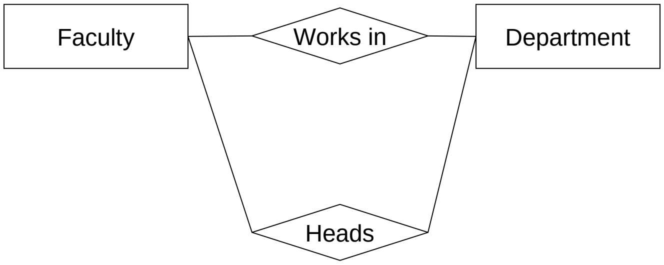
Employee

D

Faculty

Staff

# Associations



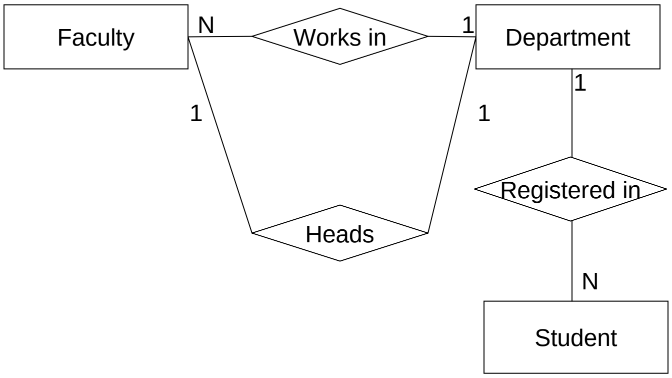
Faculty

Works in

Department

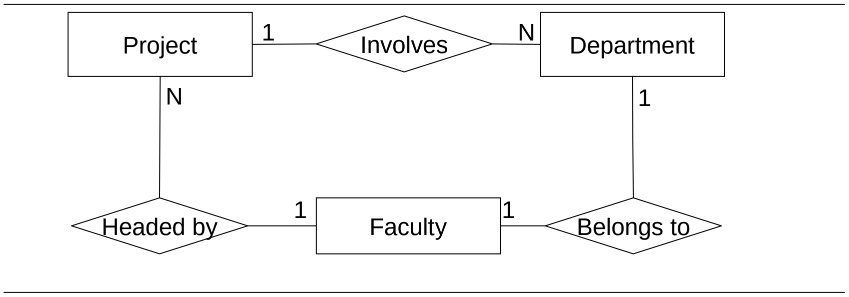
Heads

# Associations

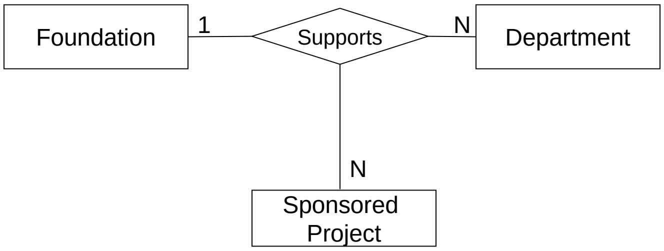

Faculty
N
Works in
1
Department
1
1
1
Registered in
Heads
N
Student

# Aggregations

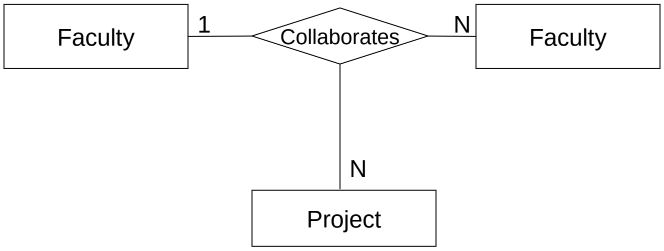
Sponsored
Project


Project
1
Involves
N
Department
N
1
1
1
Headed by
Faculty
Belongs to

# Higher-order Relationships

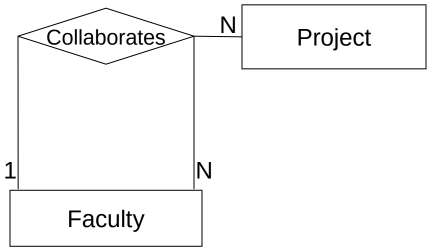


# Higher-order Relationships



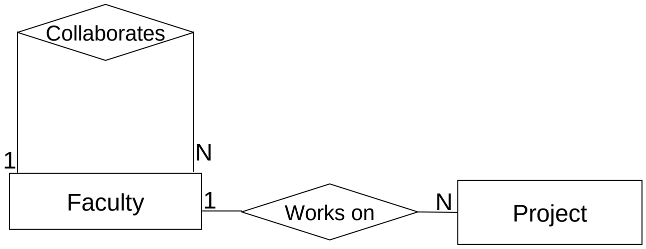
Note the double use of “Faculty” entity type…

# Higher-order Relationships

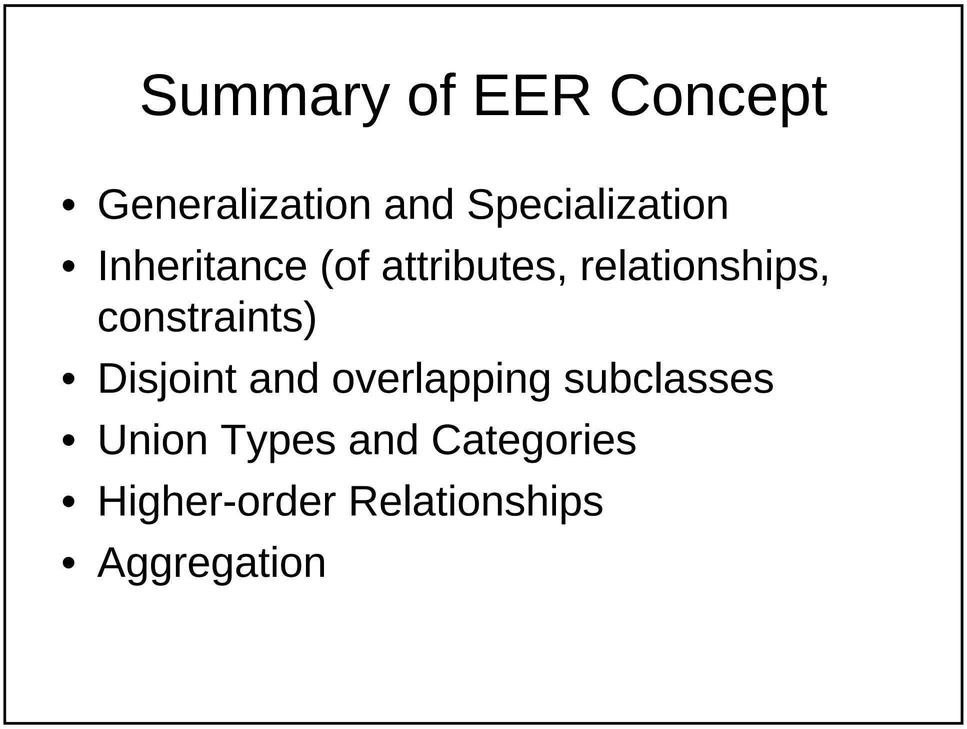


This is equivalent…

# Higher-order Relationships



This is not equivalent…



# Summary of EER Concept

- Generalization and Specialization
- Inheritance (of attributes, relationships, constraints)
- Disjoint and overlapping subclasses
- Union Types and Categories
- Higher-order Relationships
- Aggregation