WEEK 6

INTRODUCTION TO EER DIAGRAMS – ENHANCED ENTITY-RELATIONSHIP DIAGRAMS

STUDENT OBJECTIVES

- Upon completion of this video, you should be able to:
  - Determine when an entity type could be further categorized as a set of subclasses and what their superclass would look like and represent this in an EER diagram
  - Determine when several entity types could be grouped together and generalized to form a superclass and represent this in an EER diagram
  - Determine if the subclasses overlap or are disjoint and represent this in an EER diagram
  - Determine if the instances of superclass must have partial or total participation with the subclasses and represent this in an EER diagram

ENTITIES AS OBJECTS

- If we think of entities as objects, then the entity type is an object class.
  - Eg. Entity STUDENT in an ER Diagram could also be the *class* STUDENT in Java.
- Then we can introduce *subclasses* (subtypes) to enhance our data modeling.
- Just as with OO programming, **subclasses inherit** *attribute, relation.* properties from their **superclass**
- In the database case, they inherit all the **attributes** and **relationships.**
- NOTE: an entity cannot exist in the database by JUST being a member of the subclass, it must also be a member of the superclass.
- The **Enhanced** in Enhanced ER Diagrams (EER Diagrams) means we have ER diagrams with inheritance.

SPECIALIZATION

- Process of finding a set of subclass (subtypes) of an entity.

- Each subclass will have some distinguishing characteristics of the superclass

- Example: laptop and desktop are subclasses of computer *(entity)*

- Remember the ISA rule → Only use this if you can say the subclass IS A superclass.

    - Example:

        - Laptop IS A Computer – YES (so laptop can be a subclass of Computer)

        - Country IS A Continent – NO (so Country should NOT be a subclass of Continent)
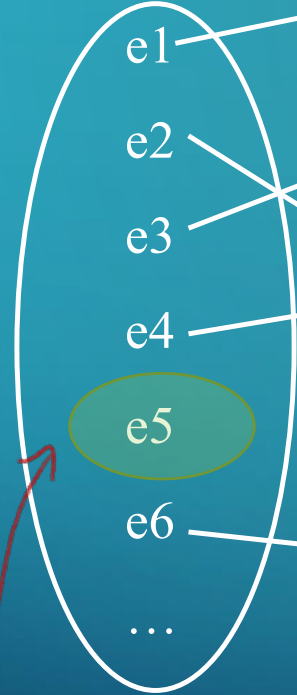
EXAMPLE:
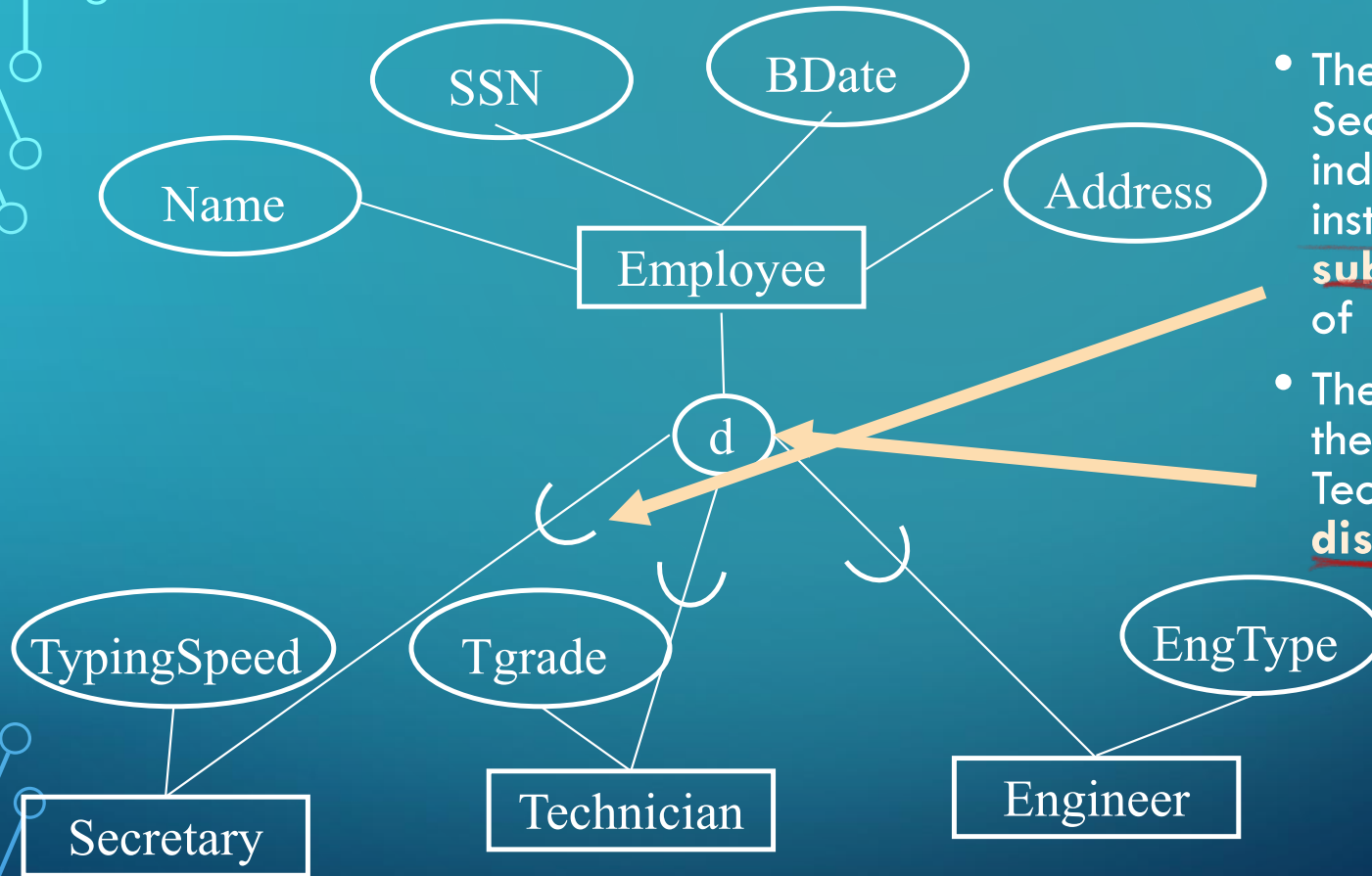
- The **U** on the line from Secretary to the d node indicates that the set of all instances of Secretary is a ~~subset~~ of the set of instances of Employee.

- The **d** in the node indicates that these three subsets (Secretary, Technician and Engineer) are **disjoint**

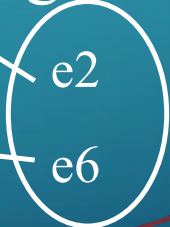i.e. an emp is either Sec, Tech, or Eng.

10/15/2023

Example

Secretary

e1
e3

Employee
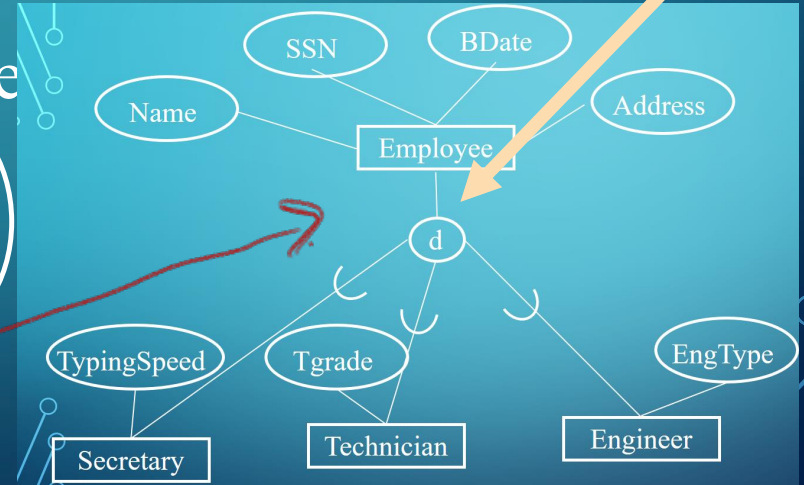
e1
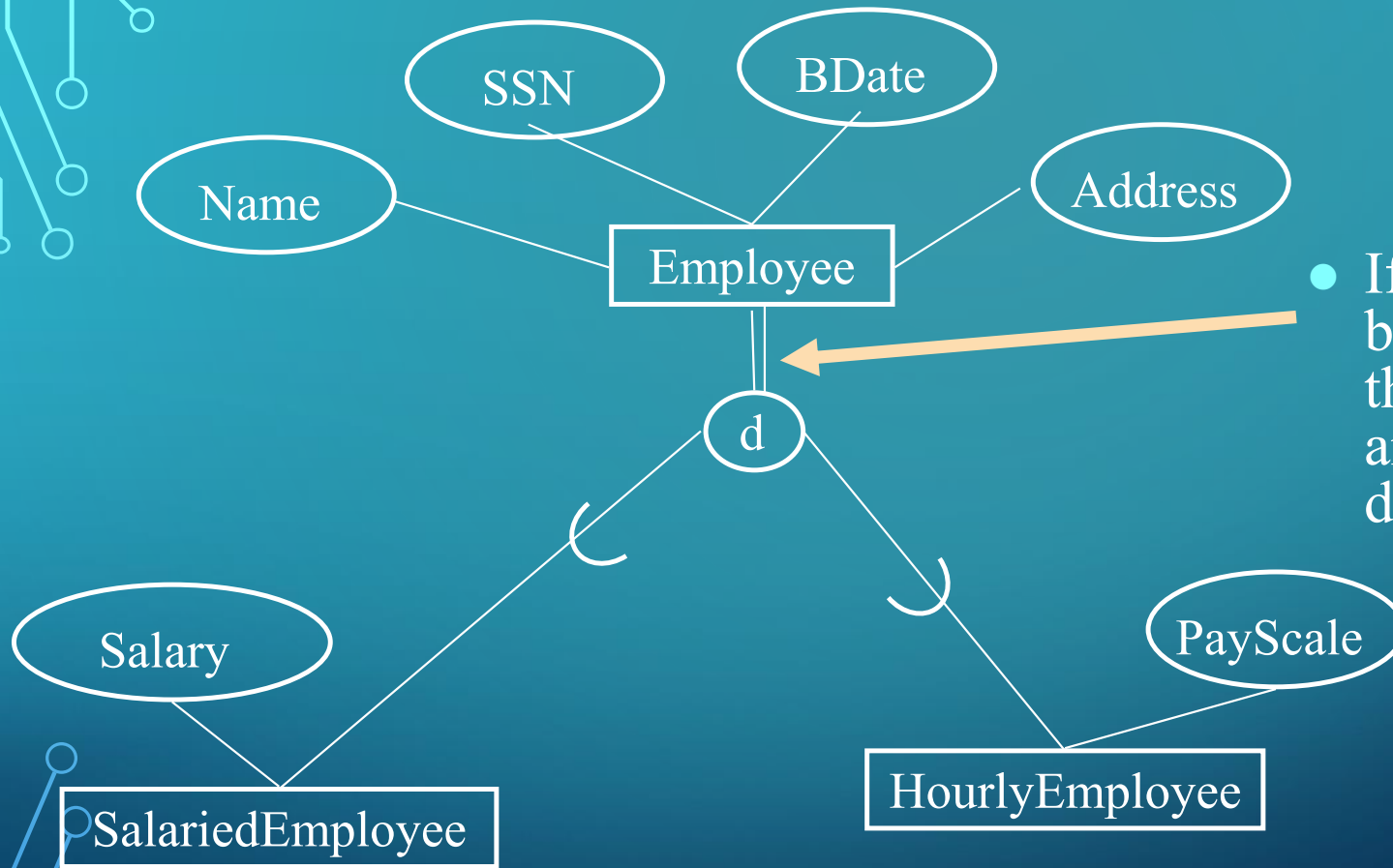e2
e3
e4
e5
e6
…

Technician

e4

Engineer

e2
e6

- This shows a few instances and we can see *partial* participation in the inheritance, i.e. only some employees belong to one of the subclasses (e.g. e5 doesn't).
- Indicate partial with a single line

e5 is just an emp.

SSN   BDate

Name   Address

Employee

d

TypingSpeed   Tgrade   EngType
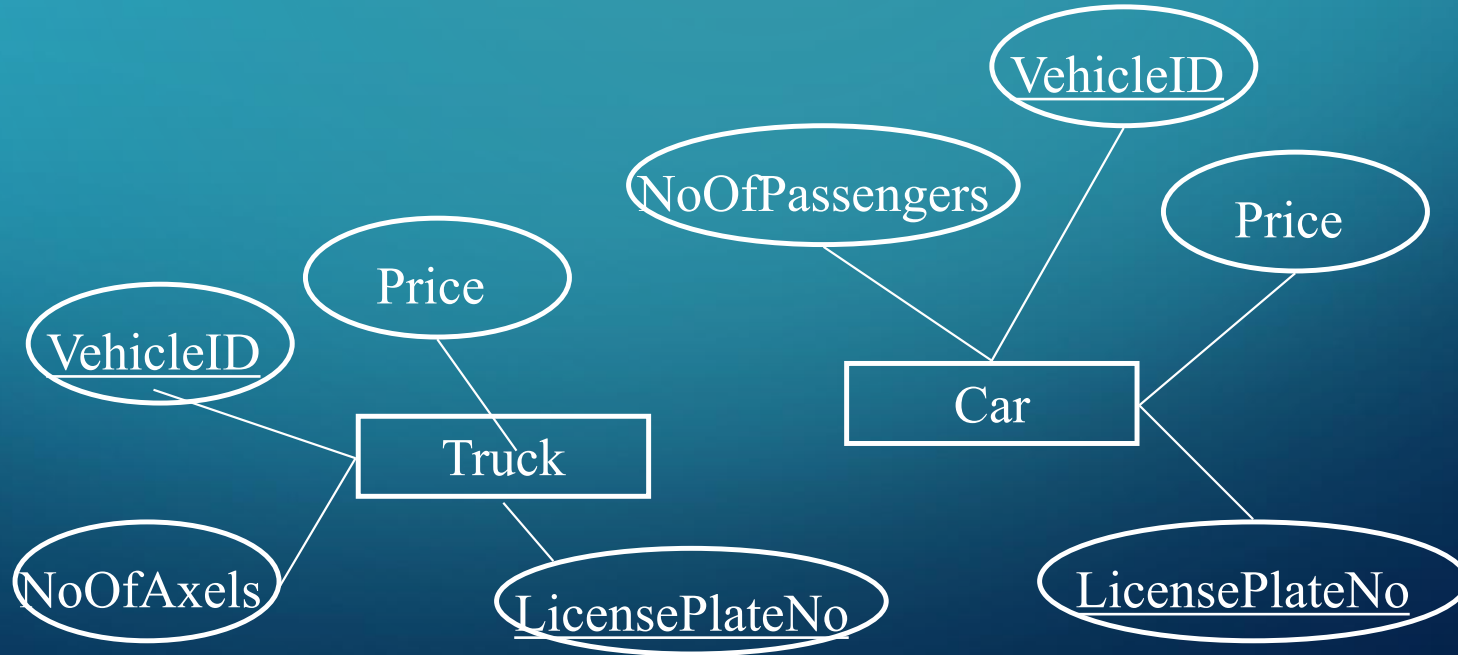
Secretary   Technician   Engineer

10/15/2023

- If all Employees must belong to a subclass, then it is called *total*, and is indicated by a double line
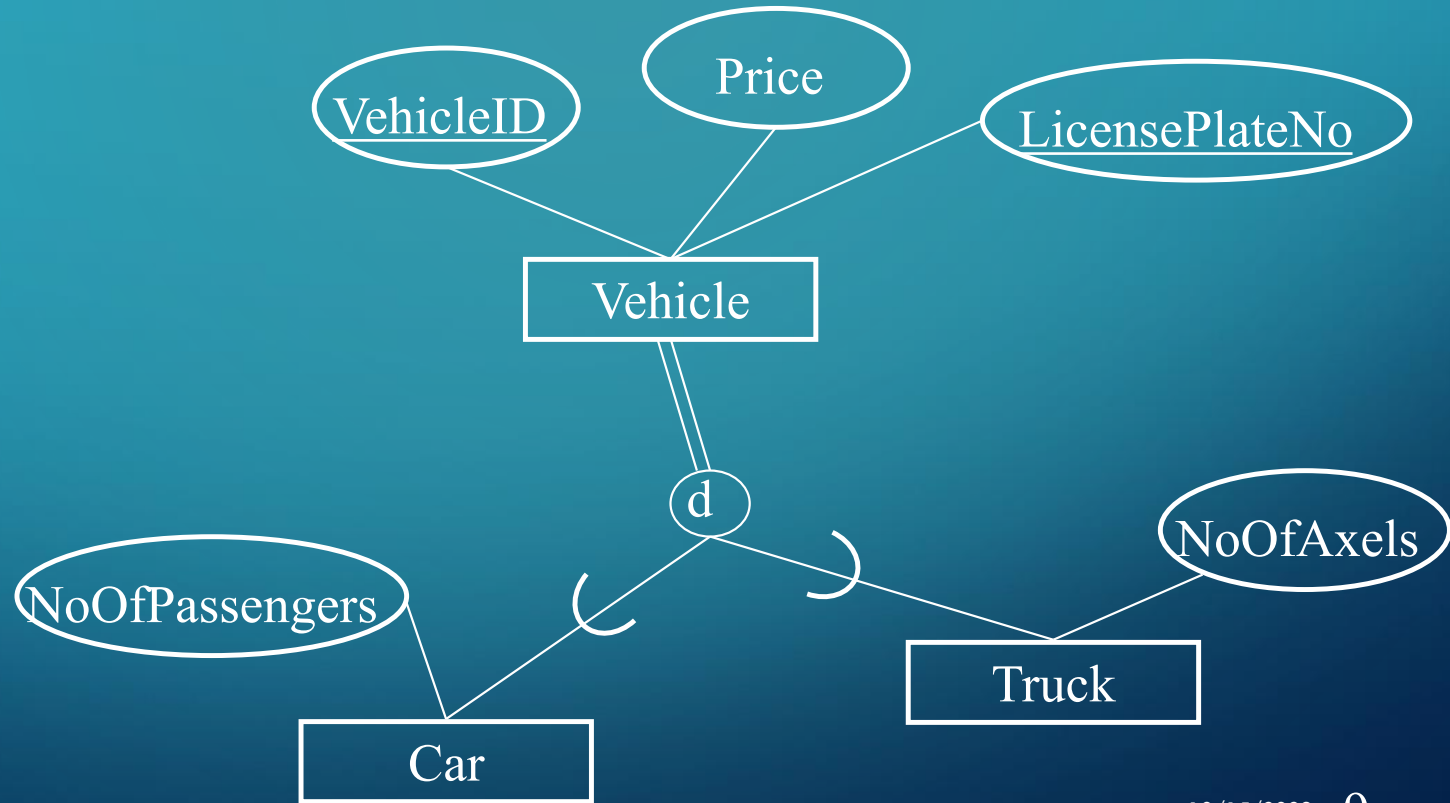
10/15/2023

GENERALIZATION

- The opposite of Specialization is *Generalization*
- Identify common features and generalized them into a single superclass.

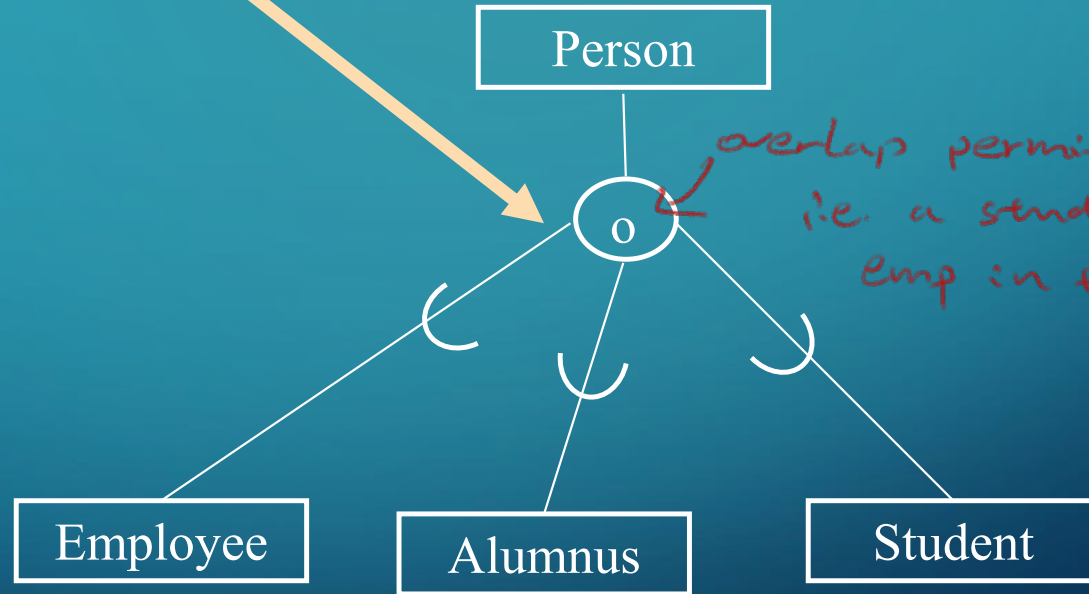Example: Suppose you are modeling vehicles:

- And you realize that it makes sense to have a superclass containing the common attributes:

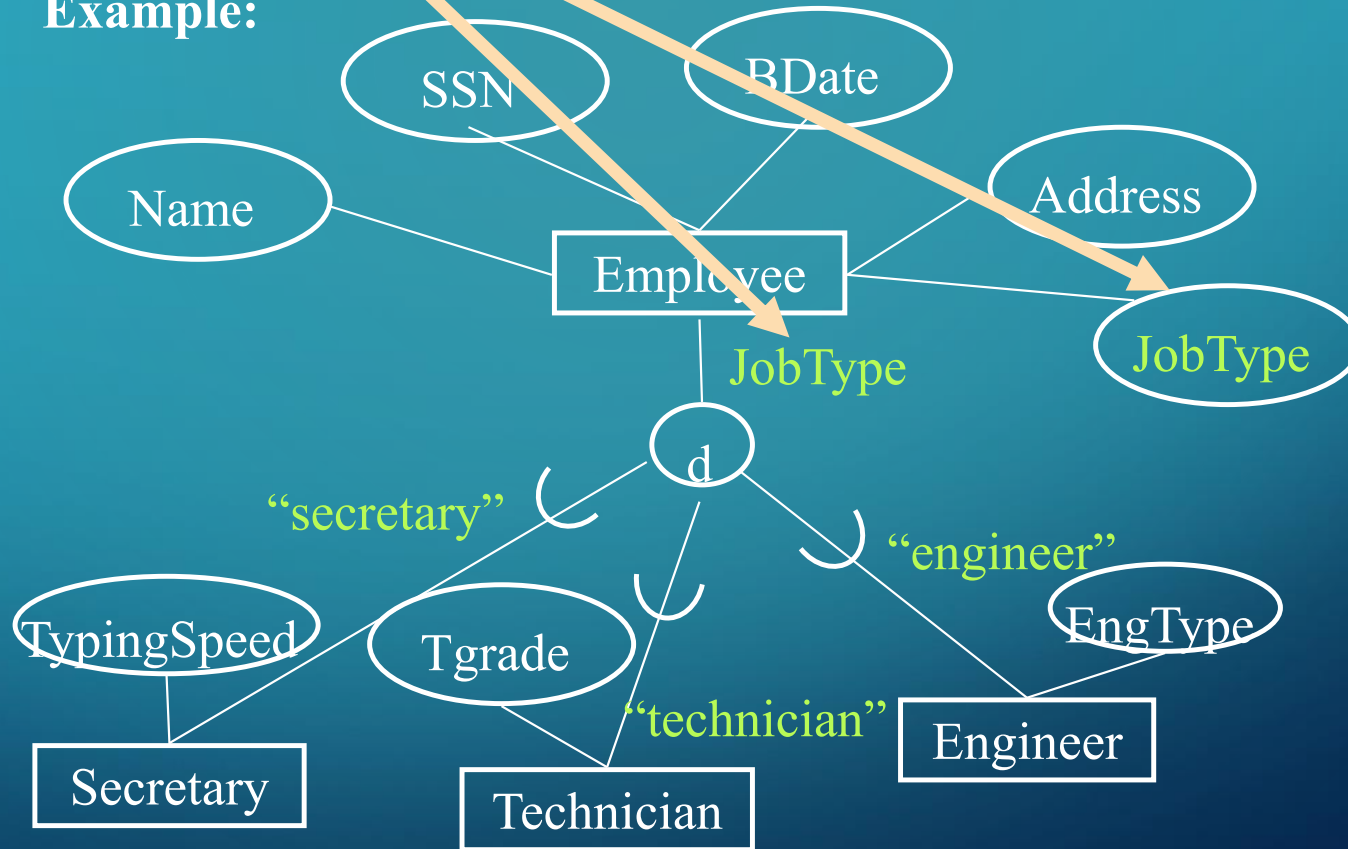Sometimes the instances of the subclasses overlap – indicated with an o
in the circle

Example:

Person

o

overlap permit.
i.e. a student could be an
emp in this case

Employee   Alumnus   Student

Can use a *category* to indicate which subclass an instance belongs to

Example:

In this case, a trigger could insert a new employee into the appropriate subclass based on the value of JobType.

NOTES

- Help to make the conceptual model more accurate (but can cause clutter)

- If subclass has few attributes and no relationships, might want to merge into superclass (specific attributes would be null for entities not a member of the subclass)

  i.e. leave it blank .

- The default for disjoint/overlapping and total/partial is **overlapped/partial** (only put disjoint and/or total if it makes sense)

QUESTION

- Consider the following entity sets and attributes. Place a checkmark ✓ in ONE column in each row to indicate the relationship.

Entity Set	Has a relationship with	Has an attribute that is	Is a specialization of	Is a generalization of	Entity Set or Attribute
Mother			✓		Person
Daughter	✓				Mother
Student			✓		Person
Student		✓			StudentNumber
School	✓				Student
Animal				✓	Horse
Horse		✓			Age
Furniture				✓	Chair

Hint: use the "IS A" Rule to help decide!