

# Lecture 8: The E/R Model

## جلسه هشتم: مدل موجودیت-رابطه

Copyright: These slides are the modified version of the slides used in CS145 Introduction to Databases course at Stanford by Dr. Peter Bailis

# Today's Lecture

1. Advanced E/R Concepts ( مفاهیم پیشرفته‌ی موجودیت-رابطه )
  - ACTIVITY: E/R Translation

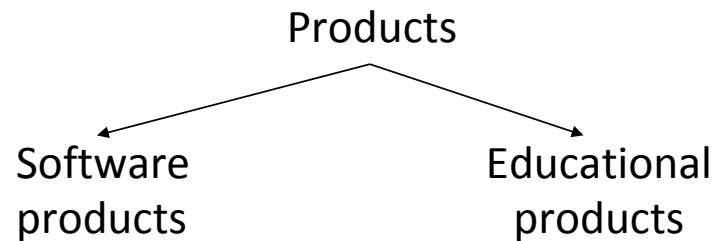
# 1. Advanced E/R Concepts

# What you will learn about in this section

1. Subclasses & connection to OO
2. Constraints
3. Weak entity sets
4. ACTIVITY: Crayon Time! Drawing E/R diagrams Pt. III

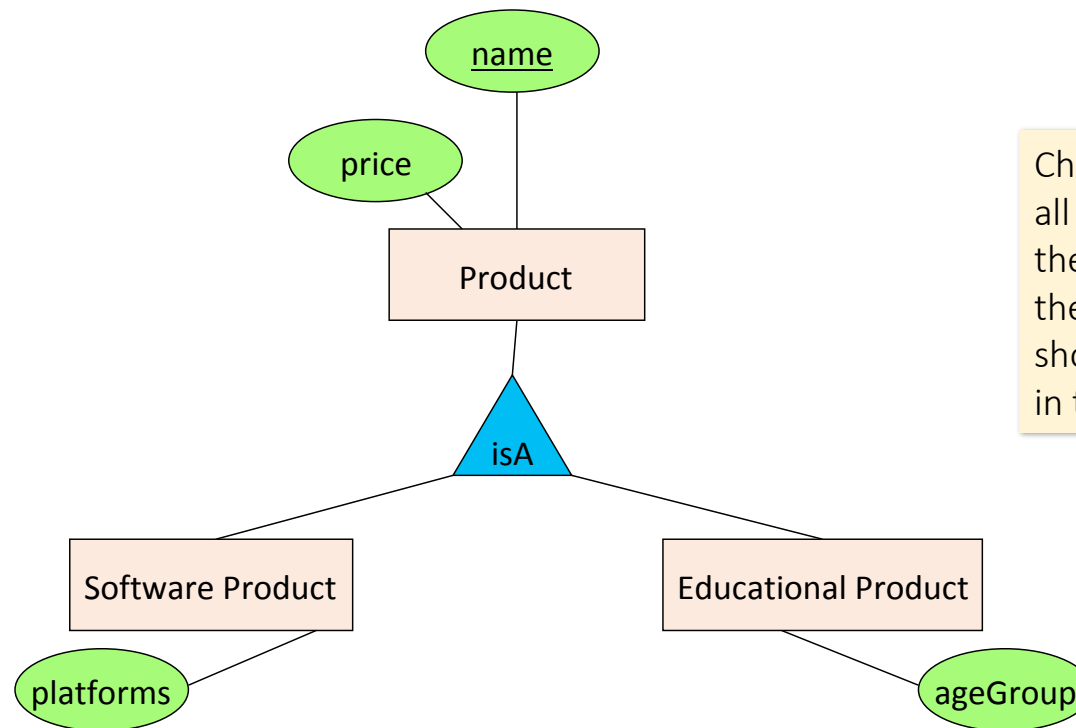
# Modeling Subclasses

- Some objects in a class may be special, i.e. worthy of their own class
  - Define a new class?
    - *But what if we want to maintain connection to current class?*
  - Better: define a *subclass*
    - *Ex:*



We can define **subclasses** in E/R!

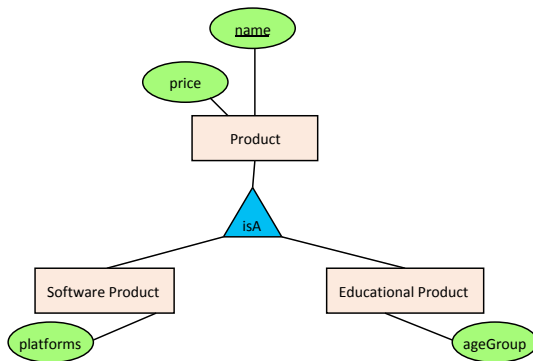
# Modeling Subclasses



Child subclasses contain all the attributes of *all* of their parent classes **plus** the new attributes shown attached to them in the E/R diagram

# Understanding Subclasses

- Think in terms of records; ex:



- **Product**

name
price

- **SoftwareProduct**

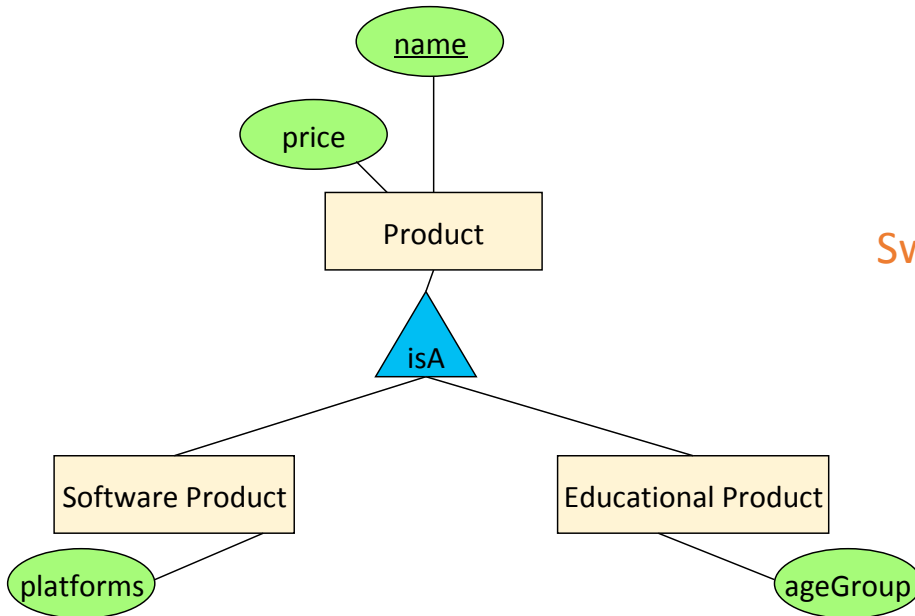
name
price
platforms

- **EducationalProduct**

name
price
ageGroup

Child subclasses contain all the attributes of *all* of their parent classes **plus** the new attributes shown attached to them in the E/R diagram

Think like tables...



Product

<u>name</u>	price	category
Gizmo	99	gadget
Camera	49	photo
Toy	39	gadget

Sw.Product

<u>name</u>	platforms
Gizmo	unix

Ed.Product

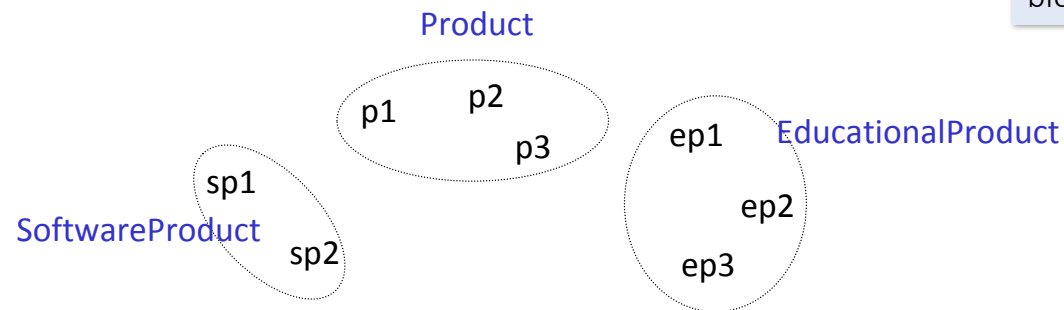
<u>name</u>	ageGroup
Gizmo	toddler
Toy	retired



# Difference between OO and E/R inheritance

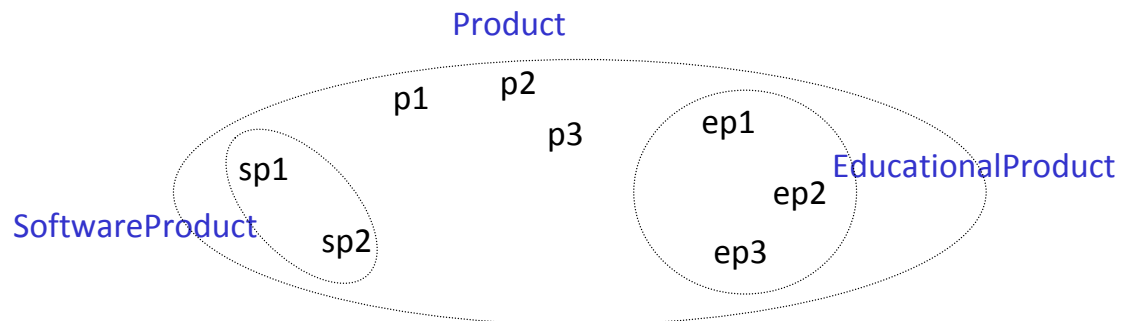
- OO: Classes are disjoint (same for Java, C++)

OO = Object Oriented.  
E.g. classes as  
fundamental building  
block, etc...



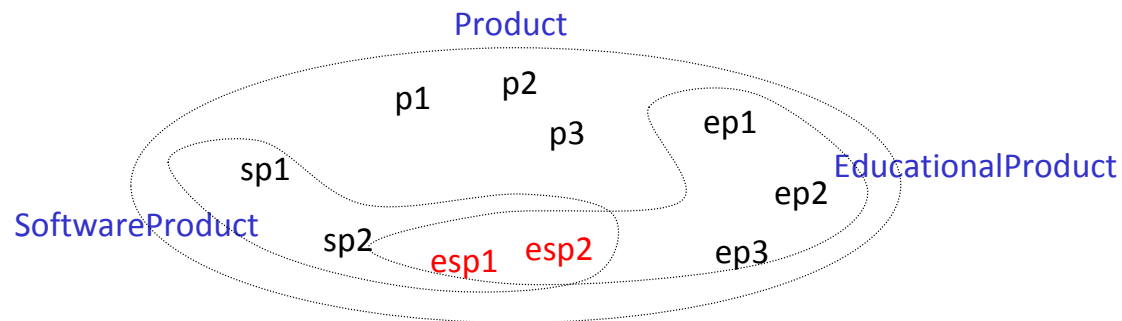
# Difference between OO and E/R inheritance

- E/R: entity sets overlap



# Difference between OO and E/R inheritance

We have three entity sets, but four different kinds of objects

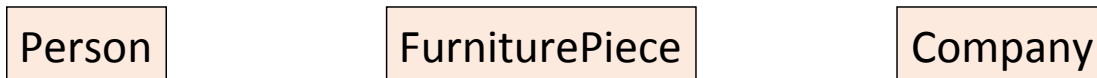


No need for multiple inheritance in E/R

## IsA Review

- If we declare **A IsA B** then every **A** is a **B**
- We use IsA to
  - Add descriptive attributes to a subclass
  - To identify entities that participate in a relationship
- **No need for multiple inheritance**

# Modeling UnionTypes With Subclasses

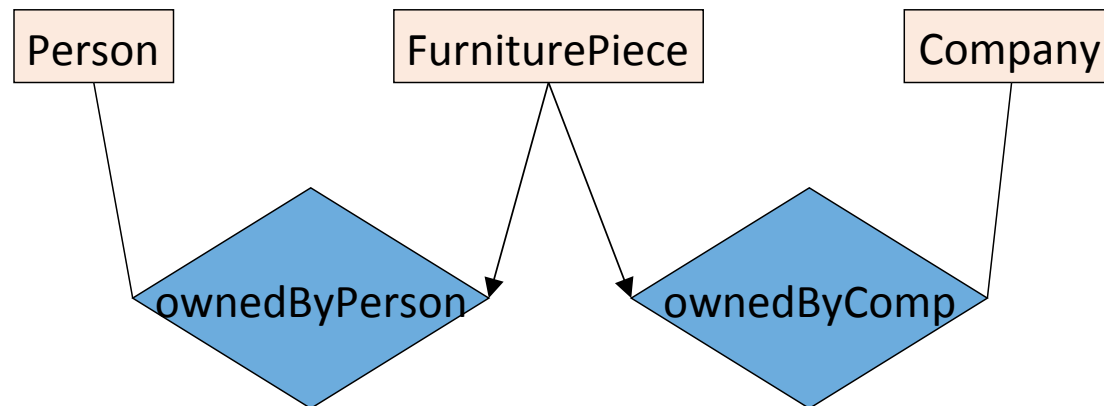


Suppose each piece of furniture is owned either by a person, or by a company. *How do we represent this?*

# Modeling Union Types with Subclasses

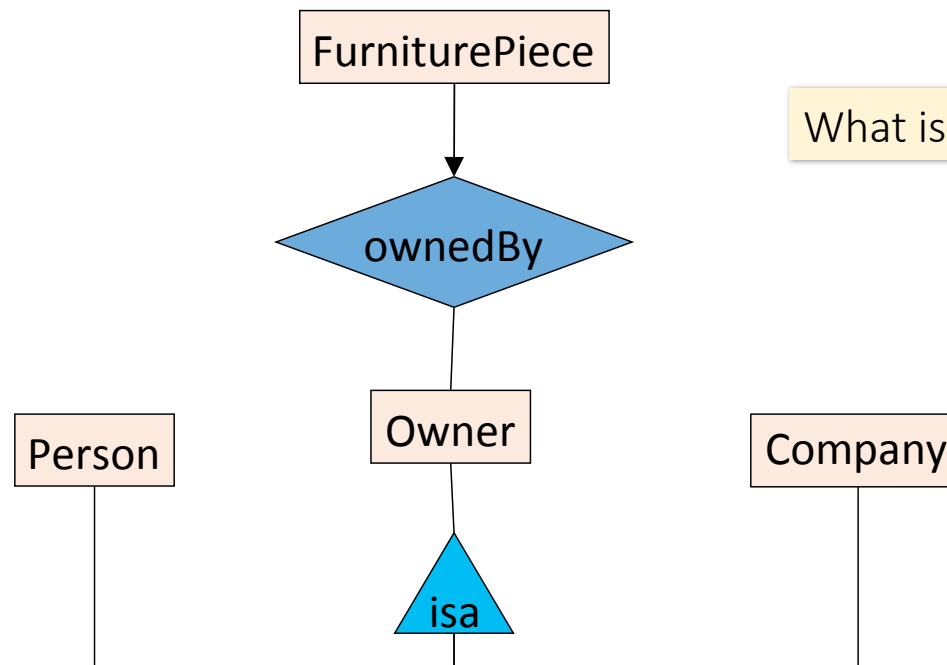
Say: each piece of furniture is owned either by a person, or by a company

Solution 1. Acceptable, but imperfect (What's wrong ?)



# Modeling Union Types with Subclasses

Solution 2: better (though more laborious)



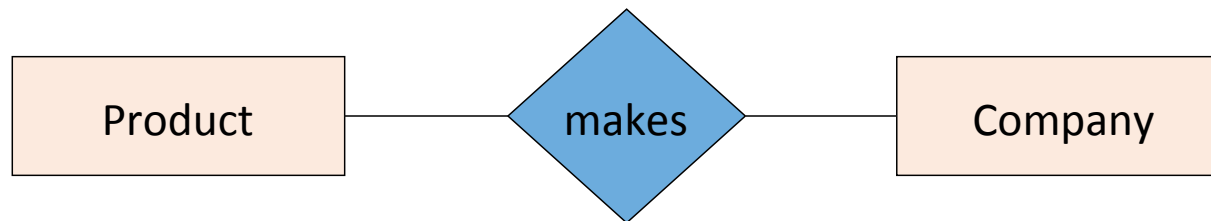
# Constraints in E/R Diagrams

- Finding constraints is part of the E/R modeling process. Commonly used constraints are:
  - Keys: Implicit constraints on uniqueness of entities
    - *Ex: An SSN (کد ملی) uniquely identifies a person*
  - Single-value constraints:
    - *Ex: a person can have only one father*
  - Referential integrity constraints: Referenced entities must exist
    - *Ex: if you work for a company, it must exist in the database*
  - Other constraints:
    - *Ex: peoples' ages are between 0 and 150*

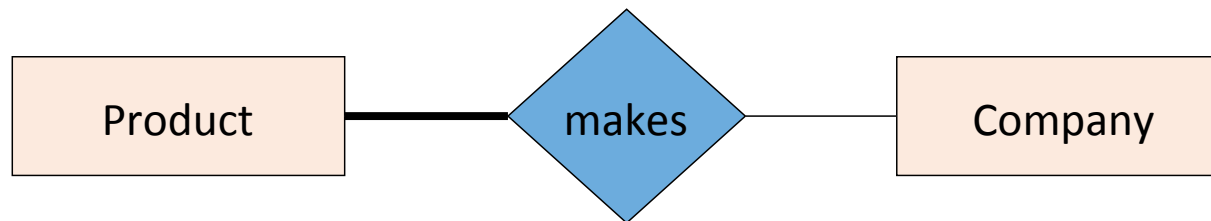
Recall  
FOREIGN  
KEYs!



## Participation Constraints: Partial v. Total



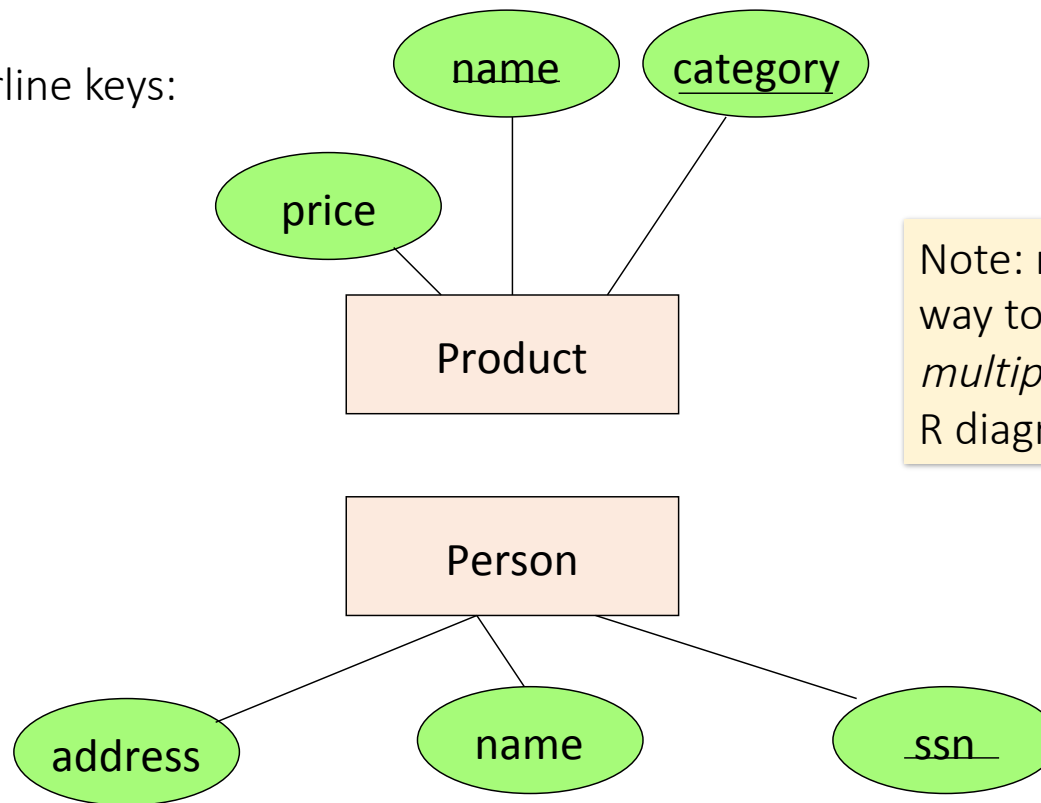
Are there products made by no company?  
Companies that don't make a product?



Bold line indicates total participation (i.e. here: all products are made by a company)

# Keys in E/R Diagrams

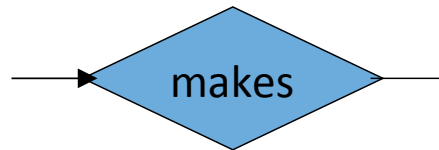
Underline keys:



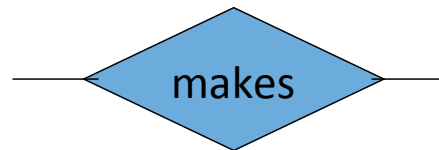
Note: no formal way to specify *multiple* keys in E/R diagrams...

# Single Value Constraints

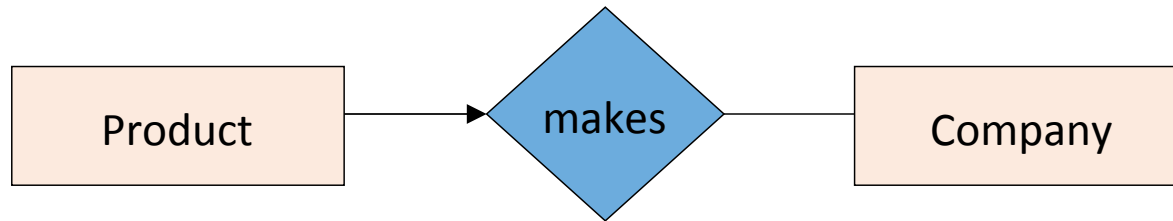
See previous section!



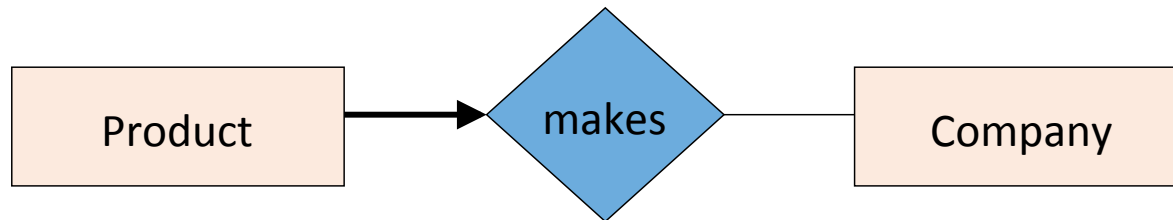
v. s.



# Referential Integrity Constraints



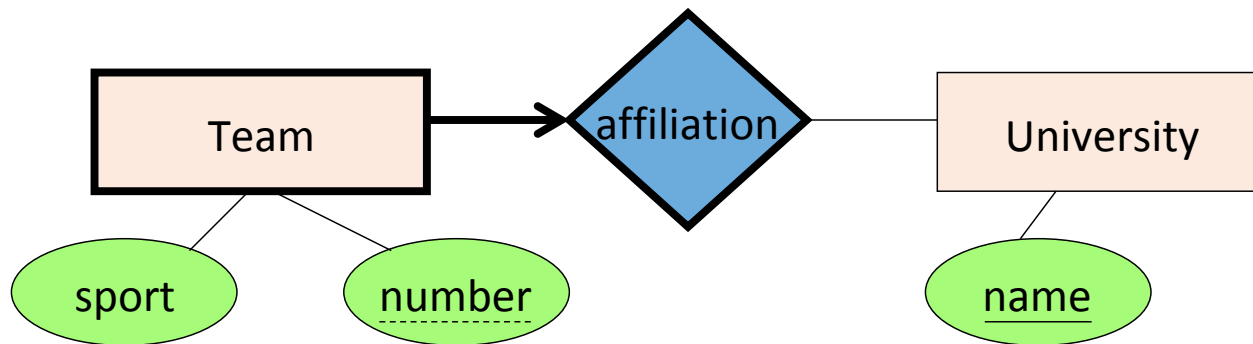
Each product made by at most one company.  
Some products made by no company?



Each product made by exactly one company.

# Weak Entity Sets

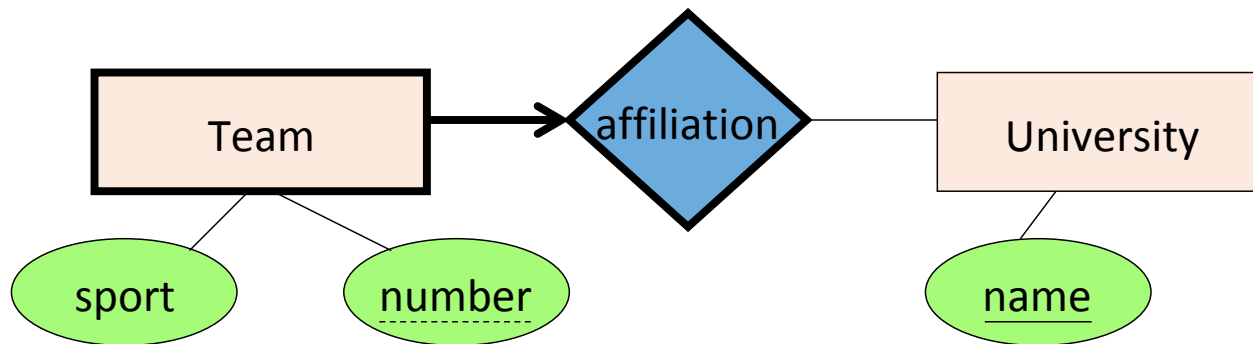
Entity sets are weak when their key comes from other entity sets to which they are related.



“Football team” v. “*The Yazd Uni* Football team” (e.g., *Tehran Uni* has a football team too)

# Weak Entity Sets

Entity sets are weak when their key comes from other classes to which they are related.



- number is a partial key. (denote with dashed underline).
- University is called the identifying owner.
- Participation in affiliation must be total. Why?

## E/R Summary

- E/R diagrams are a visual syntax that allows technical and non-technical people to talk
  - For conceptual design
- Basic constructs: **entity**, **relationship**, and **attributes**
- A good design is faithful to the constraints of the application, but not overzealous

# Extra Activity



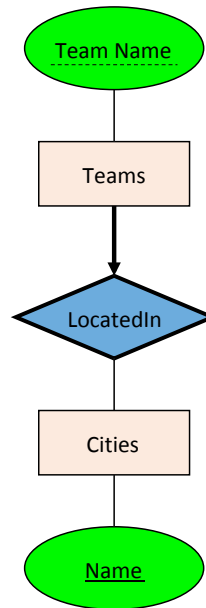
# Add in: Subclasses, constraints, and weak entity sets

Concepts to include / model:

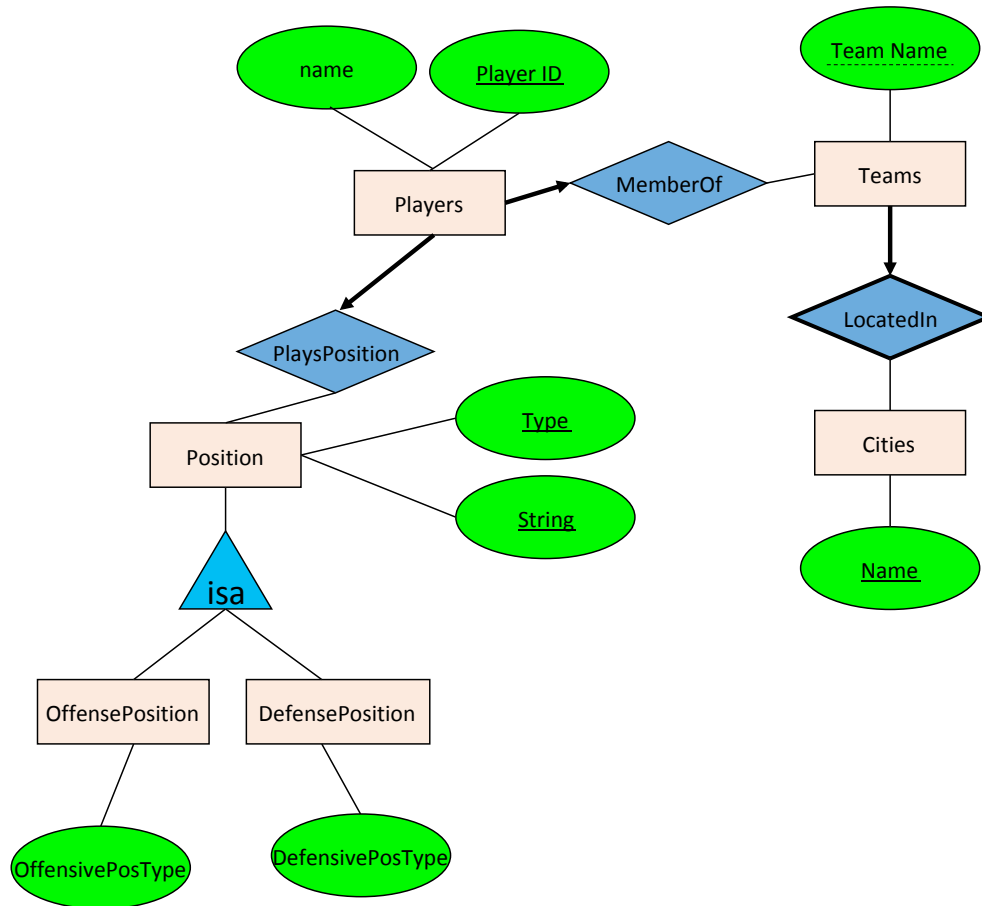
Teams belong to cities- model as *weak entity sets*

Players are either on Offense or Defense, and are of types (QB, RB, WR, TE)

Make sure you have designated keys for all our concepts!



Teams belong to cities- model as *weak entity sets*



Players are either on Offense or Defense, and are of types (QB, RB, WR, TE)