

A decorative graphic on the left side of the slide, consisting of a network of white lines and circles on a blue gradient background, resembling a circuit board or a neural network.

# WEEK 1

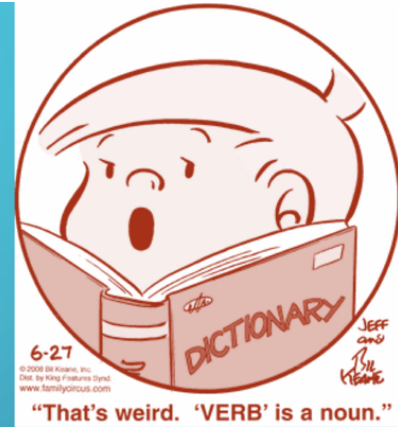
## ENTITY RELATIONSHIP DIAGRAMS – PART 2

CS3319

# STUDENT OBJECTIVES

- Upon completion of this video, you should be able to:
  - Given a description of a mini-world, identify which nouns should be Entities and which nouns should be Attributes
  - Define the following terms: *Entity*, *Key*, *Attribute*, *Multi-valued Attribute*, *Composite Attribute* and *Derived Attribute* and select the correct shape to represent these in an ER Diagram
  - Determine if an attribute is allowed to have null values and if so, decide if the null value is not applicable OR not known
  - Distinguish between an attribute and an attribute's value (its *domain*)

# WHEN DESIGNING A DATABASE → NOUNS ARE KING!



- Look at your requirements, your specs, your forms, your reports and identify the nouns.

- Usually the nouns turn into:

*data* <

- Entities
- Attributes

- Let's look at our case study and identify some nouns

# CASE STUDY – CREATING AN ER DIAGRAM

- Suppose we plan to model a company which is organized into <sup>nonn</sup> departments.
- Each department has a unique name, number and employee who manages it (we want to keep track of when the employee started managing the department)
- A department may have several locations
- A department controls a bunch of projects, each project has a unique number, name and a single location
- Each employee has a name, ssnumber, address, salary, sex and birthdate
- An employee is assigned to only one department but may work on several projects which are not necessarily from the same department
- Keep track of the number of hours each employee works on each project.
- Keep track of the direct supervisor of each employee
- Keep track of the dependents of each employee (name, sex, birthdate and relation)

# WHAT'S WHAT?

- What is different about the noun *employee* vs. the noun *salary*?
- If you can see the different between them, you understand what an ENTITY is compared to an ATTRIBUTE!

*is used to  
describe one  
thing*

*i.e. salary is used  
to describe employee.*

*a thing that exist  
employee an exist  
thing.*

# E-R MODEL CONCEPTS AND KEY TERMS

- **Entities and Attributes:**

- **Entity** - A single "THING" that exists, has independent existence → Employee

- **Attribute** - describes a "thing" → Age, SSN, Sex, Name

- **Value** - taken on by an attribute → 25, 456-876-788, Female, Bart Simpson

- **Composite Attributes vs. Atomic or Simple Attributes** → Bart Simpson vs. 45  
*this one would not be broken down*

- **Single-Valued Attributes vs. Multivalued Attributes** → Age vs. CollegeDegrees  
*a person would have only one age, but it could be derived from birthday. probably multiple degrees.*

- **Derived Attributes vs. Stored Attributes** → Age vs. Birthdate

- **ENTITY** → any object in our mini-world that we want to model and store information about.  
E.g. Student, Professor, Classroom

- **ATTRIBUTE** → defines the information about the entity that needs to be stored. An entity will have 0 or more attributes. An attribute has a DOMAIN. E.g. student number

- **DOMAIN** → the type of values that an attribute can take. E.g. String, integer, real, date  
*range.*

- **Null Values** - Why and when would we use this?

- Apartment Number *somebody live in house, which does not have AN.*
- Phone Number *← don't know the phone number.*

**QUESTION: What is the difference between attributes with null values that are not applicable vs. not known?**

# MORE TERMINOLOGY

- Entity Types, Value Sets (DOMAIN) and Key Attributes
  - Entity Type - defines a set of entities that have the same set of attributes
  - Entity – an instance of an entity type
  - Entity Set, Collections - group of Entities
  - Key, Uniqueness
  - Combination to create key *key + entity.*
  - Value Sets (Domains) *i.e 0 ~ 100*

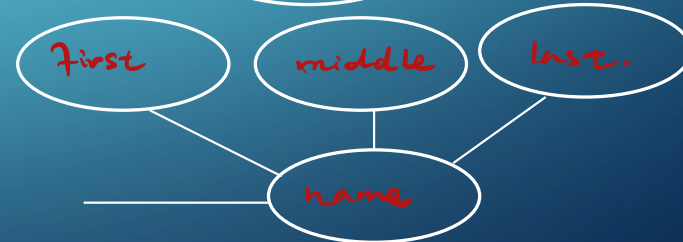
**QUESTION: What is Age's domain, what is Name's domain, what is Cost's domain?**

*real number.*      *int > 0*      *String*



# E-R DIAGRAM NOTATION SO FAR:

- *instance.*  
Entity Type
- Attribute
- Key Attribute
- Multi-valued Attribute
- Composite Attribute
- Derived Attribute



9/17/19

9

← not necessarily to show

# QUESTION: WHAT ARE THE ENTITIES FROM OUR EXAMPLE?

- 
- 
- 
-

# CASE STUDY – CREATING AN ER DIAGRAM

*entity, a thing.*

- Suppose we plan to model a company which is organized into **departments**.
- Each department has a **unique** **name**, **number** and employee who manages it (we want to keep track of when the employee started managing the department)
- A department may have **several** **locations**
- A department controls a bunch of projects, each project has a **unique** number, name and a single location
- Each employee has a name, ssnnumber, address, salary, sex and birthdate *attributes.*
- An employee is assigned to only one department but may work on several projects which are not necessarily from the same department
- Keep track of the number of hours each employee works on each project.
- Keep track of the direct supervisor of each employee
- Keep track of the dependents of each employee (name, sex, birthdate and relation)

QUESTION: WHAT IS OUR DIAGRAM  
SO FAR? (IT IS STARTED BELOW)

Let's use draw.io to finish the diagram.

*For most of times an entity,  
does not have more than one  
unique entity.*

