## Derived Attribute

* An attribute that you do not really need to store in a database, but you can retrieve it via a query and based on other stored attributes.

Examples: GPA, number of enrolled students, number of employees, age, etc.

Screen Clipping

* is shown by

## Weak Entity

* An entity which does not have a key attribute of its own. Regular entities that you have seen so far could be called strong entity too.
* Example: Every employee in COMPANY example has a key attribute and that is SSN, but their dependents do not have such a number while still we may need to store their information for health coverages and benefits.
  + They (dependents) are defined as weak entity.
  + Here the employee related to the dependents is called **owner entity**.
* **Owner entity:** weak entity instances are identified by being related to a strong entity instance, that strong entity is called owner entity (or parent entity).
* **Identifying relationship**: the relationship which relates a weak entity to its owner is called identifying relationship, and we draw it as below:



* **Partial key**: an attribute of a weak entity which uniquely identifies an instance among instances of the same owner entity.
  + In ER, it is **underlined with a dashed line**.
* A weak entity always has a total participation constraint with respect to its identifying relationship.
  + A weak entity can only be identified with regard to its owner entity.

## Relationship Attributes

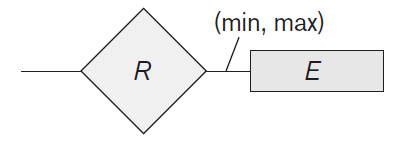
* For relationships with 1:1 and 1:N cardinality ratios, you can move them to one of entities:
  + to the one on the N-side of the relationship.
  + For example: “an EMPLOYEE WORKS\_FOR a DEPARTMENT”.
    - It is cardinality ratio is …. : …..
    - Therefore if the relationship has an attribute such as “Start\_to\_work”, it could be added to ……………………..
* For a M:N relationship, we must keep relationship attributes connected to the relationship diamond.
  + Why? hint M\*N combinations!

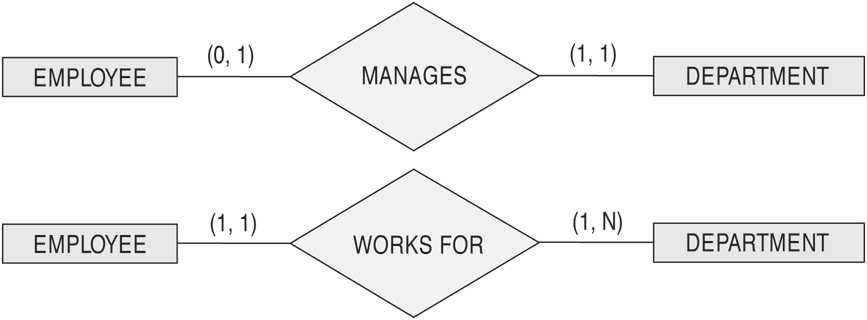
Now look back at the COMPANY ER again and make sure you understand it completely, before moving to the next notation of ER diagrams.

## Min-Max notation for relationship constraints

* Depending on your project description, you may have some constraints like ‘The minimum number of employees work for a department is 4’.

The notation you saw in the previous pages is not capable of demonstrating these kind of constraints. We need to use Min-Max notation for such cases.

* Min-Max notation specifies that each entity instance participates at least **min** and at most **max** times in the relationship.
* Here is Min-Max notation In ER:

Examples:

You can read these ERs as below:

* An employee manages minimum 0 and maximum 1 department.
* A department is managed by minimum 1 and maximum 1 employee (i.e. exactly one employee).
* An employee works for minimum 1 and maximum 1 department (or exactly one department).
* For a department at least 1 and at most any number of employees work.
* How should you change WORKS FOR ER to have the following constraint?
* The minimum number of employees work for a department is 4.