Reduction of E-R Diagram To Tables

For converting Strong Entity Types these are some points that should be keep in Mind

→ Each Entity Type becomes a ~~typ~~ Table.

→ Each Single-Valued attribute become a column

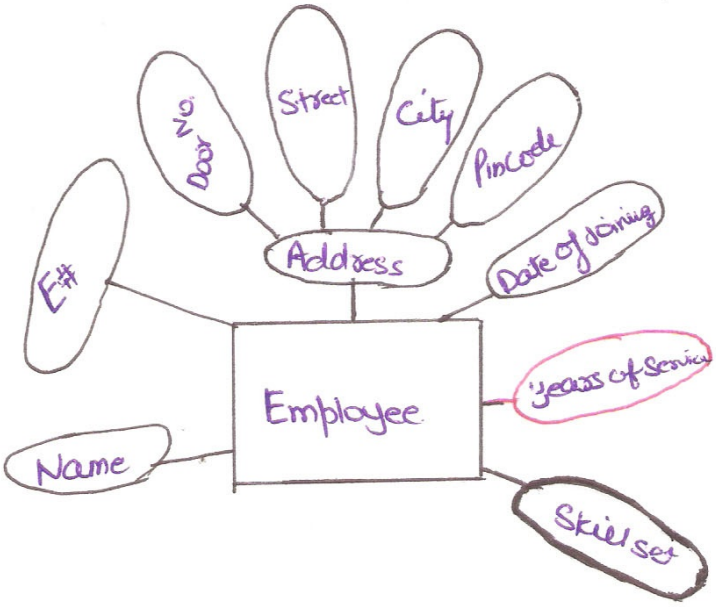
→ Derived attributes are ignored.

Composite attributes are represented by Components

→ Multivalued attributes are represented by a Separate Table

→ Key Attribute of the entity type becomes the primary key of the Table

Example



- Here 'Address' is 'Composite 'attribute'
- Years of Service is Derived attribute
- Skill set is a multi-Valued Attribute → We have to Make Separate Table but Remember there Should be relation blw an Employee Table & Skill set Table

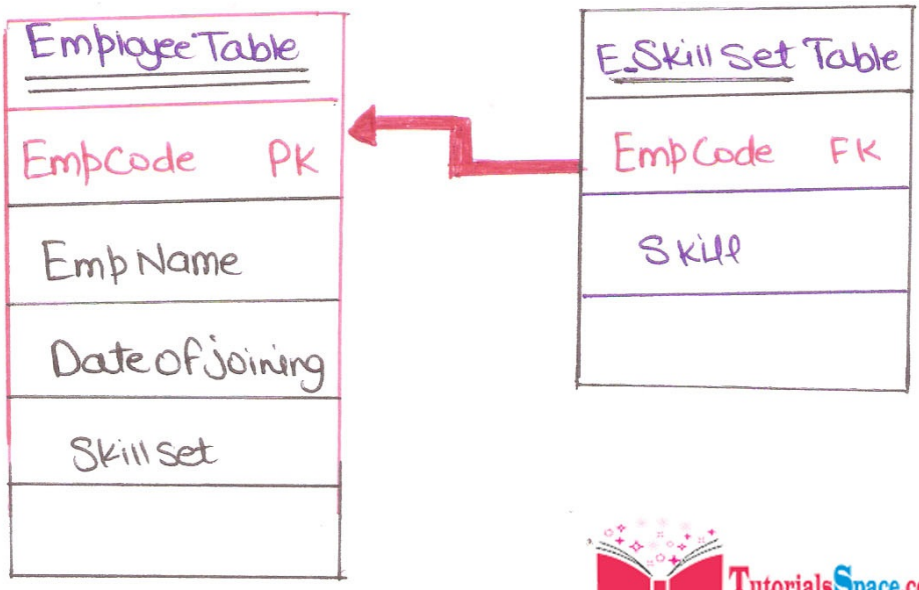
So the Relational Schema & Employee Skill set Table is

Employee (E# , Name , Door_NO , Street, City , Pincode, Date_of_Joining)

Employee Skill set (E# , Skill)

As we saw E# has to be primary key for Employee Table to uniquely identify an entity. & Employee skill set has skill which ~~can~~ come from skill set (Multivalued)

So to Create a Relationship b/w these two table we use E# i.e Employee code in Employee skill set table as a foreign key which implements Referential Integrity.



Converting Weak Entity Types

- Weak Entity types are converted into table of their own, with the primary key of the strong entity acting as a foreign key in the Table.

- This foreign key along with the key of the weak entity form the composite primary key of this Table.
- The Relational Schema

Employee (E# .....)

Dependent (E#, Dependent_ID, Name, Address)