ERD

1- Describe Entities You need in the system (خليها كلمة واحدة مفرد)

{ Employee , Dept , Contract , Project , Skill , Car , Dependent }

==========

2- Define Attributes for every Entity

-> Employee

-> simple/Single Attributes (SSD , ID , Name , Salary)

-> Multi-Valued Atrributes (Phone)

-> Composite Attributes (Adress : city , Street)

-> Derived Atrributes : Need to be calaulated (Age)

-> Dept

-> Simple Attributes (DNO , DName , DLOC)

-> Contract

-> Simple Attributes (Contract_ID , Type , ST_Date)

-> Project

-> Simple Attributes (PNO , PName)

-> Skill

-> Simple Attributes (SID , SName)

-> Dependent

-> Simple Attributes (Name , Relation)

-> Car

-> Simple Attributes (PlateID , MOD , Color)

==========

3- define Unique Identitfiers Attribute/s

-----Types Of Entities

1. Strong Entity : Has a Unique Identifier Attribute/s

2. Weak Entity : Has NOT a Unique Identifier Attribute/s

->Strong Entities : { Employee(SSD , ID) , Dept(DNO) , Contract(Contarct_ID) , Project(PNO) , Skill(SID) , Car(PlateID) }

->Weak Entities : { Dependent }

==========

4-Define Realtionships between Entities(خليها فعل)

1. Degree Of Realtionships : { Binary Realtionship , Unary Realtionship , Unary Recursive Realtionship }

2. Cardinality Of Realtionships : { One-To-One , One-To-Many , Many-To-Many }

3. Ratio Of Participation : { Partial(May) , Fully(Must) }

Ex01: Employee - (Work) - Dept

1. degree : Binary Relationship

2. Cardinality:

- Employee works in (One) Dept

- Dept Can Have (Many) Employees work in

=> Many-To-One => (Many Employee)-To-(One Dept)

3. Participation:

- Employee (Must) work in a Dept

- Dept (May) have Employees work in (depend on business Case)

=> (Employee Must)-To-(Dept May)(Must-To-May) Participation ()

----> Iportant Notes

1. In Trnary Relationship Cardinality Must Be The Same From All Sides(If Many => (Must - Fully) Participation).

2. Weak Entity Participation is Always (Must - Fully) ,cuz it fully dependent on other Entity

Mapping

1.Mapping Of Regular Entity Type

- Create Table For Each Entity
- Ex: Employee (SSN , ID , Name , Salay , City , Street)
    -> Single/Simple Attributes
    -> Choose Primary Key(SSN)
    -> Composite Attributes Address(City , Street) : Add Compsite Attributes To The Table
    -> Malti-Valued Attribute(Phone) :
        -> Emp-Phone(SSN , Phone)
        -> Make A New Table Emp-Phone for this Malti-Valued Attribute
        -> Add The Primary Key Of Employee As Foreign Key
        -> Use Combination Of (Phone , SSN) as a primary key for Emp-Phone Table
    -> Derived Attribute(Date) : Don't Use If You you don't need it cuz it add calculation headche

==========

2. Mapping Of Weak Entity Type

-> Dependent(SSN , Name , Relation)

-> Add Primary key of Onwer(Parent) Entity As Foreign key in the Weak Entity

-> Use Combination Of (SSN , Name) as a primary key for Dependent Table

==========

3. Mapping Of One-To-Many 1:M (Unary or Binary) Relationship

-> Add Primary Key Of The One Side As Foreign Key At The Many Side

-> Even it a Unary-Recursive do the same but change foreign key name

-> Relationship Attributes if exist (ST_Date) Follow Foreign Key

==========

4. Mapping Of Many-To-Many N:M (Unary or Binary) Relationship

->Ex: Wrok_on(SSN , DNO , Hours)

-> Make A New Table with Relationship Name

-> Add the combination of Primary Keys(SSN , DNO) Of The Two Entities As Foreign Keys At the new table

-> Use Both Foreign Keys To Make Primary Key To The New Table

-> Add Relationship Attributes if exist (Hours)

==========

5. Mapping Of One-To-One 1:1 (Unary or Binary) Relationship

-> 1:1 Relationship Mapping depend on Participation(May-Must , May-May , Must-Must)

1. May-Must 1:1

-> Add Primary Key Of May Side As Foreign Key in Must Side

2. May-May 1:1

-> Add Primary Key Of Any May Side As Foreign Key in The Other May Side

3. Must-Must 1:1

-> Merge The Two Tables in One Of The Two Tables To Make One Table With All Data

==========

6. Mapping Ternary Relationship

-> Make New Table With All Primary Keys From All Three Entities Tables As Foreign Keys

Mapping

Employee(SSN , ID , Name , Salary , Street , City , DNO , MGR-SSN , PlateID , Contract_ID , Type , ST_Date)

Emp_Phone(SSN , Phone)

Dept(DNO , DName , DLOC , DMGR-SSN , ST_Date)

Project(PNO , PName)

Skill(SID , SName)

Dependent (SSN , Name , Relation)

Car (PlateNO , MOD , Color)

~~**Contract**(Contract_ID , Type , ST_Date)~~

Work_On (SSN , PNO , Hours)

Skilled_Use(SID , SSN , PNO)