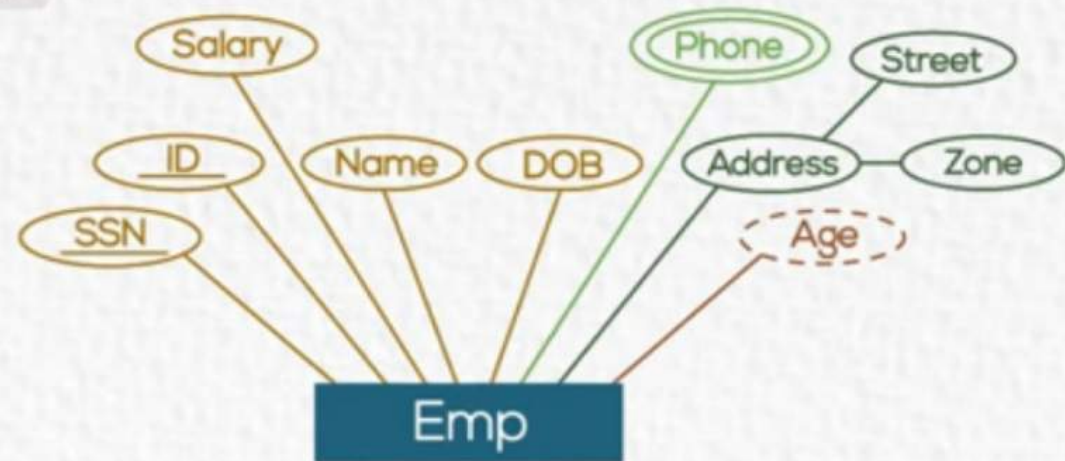# Step 1: Mapping of regular entity types

Choose one of key attributes to be the primary key

Emp (ID, SSN, Salary, Name, DOB, Street, Zone)

Emp - Phone (SSN, Phone)

SSN	Phone
123	0245678
123	0159875

Emp (ID, SSN, Salary, Name, DOB, Street, Zone)

Emp - Phone (SSN, Phone) ← multi-valued

Dept (DNO, D Name, LOC)

Project (PNO, P Name)

Dependent (SSN, Name, Relation)

Car (Plate_NO, Model, Color)

Contract (Contract_ID, Type, Start_date)

Skill (Skill_id, Skill_name)

N

1

Emp (ID, SSN, Salary, Name, DOB, Street, Zone, DNO, Sup-SSN
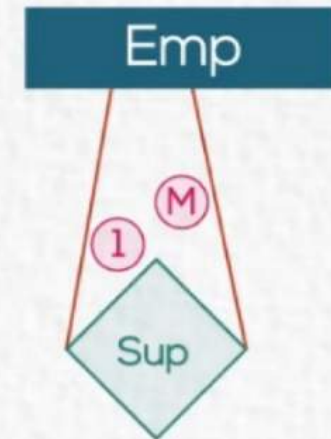
Emp - Phone (SSN, Phone)

Dept (DNO, D Name, LOC)

Project (PNO, P Name)

Dependent (SSN, Name, Relation)

Car (Plate_NO, Model, Color)

Contract (Contract_ID, Type, Start_date)

Skill (Skill_id, Skill_name

Emp

1 M

Sup

Add FK to N-side table

N

Emp (ID, SSN, Salary, Name, DOB, Street, Zone DNO
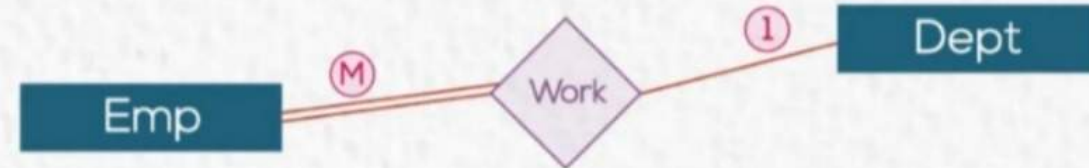
Emp - Phone (SSN, Phone)

Dept (DNO, D Name, LOC)   1

Project (PNO, P Name)

Dependent (SSN, Name, Relation)

Car (Plate_NO, Model, Color)

Contract (Contract_ID, Type, Start_date)

Skill (Skill_id, Skill_name

Emp —(M)— Work —(1)— Dept

Add FKs to the new table for both parent tables

Emp (ID, SSN, Salary, Name, DOB, Street, Zone, DNO, Sup-SSN

Project (PNO, P Name)

Emp —N— Work on —M— Project

Hours

new table

Work_On (SSN, PNO, Hours)

Step 5: Mapping of relationship types **Binary / Unary 1:1**

Emp (ID, SSN, Salary, Name, DOB, Street, Zone, DNO, Sup-SSN

Emp - Phone (SSN, Phone)

Dept (DNO, D Name, LOC, MGR _SSN ST. Date

Project (PNO, P Name)

Dependent (SSN, Name, Relation)

Car (Plate_NO, Model, Color)

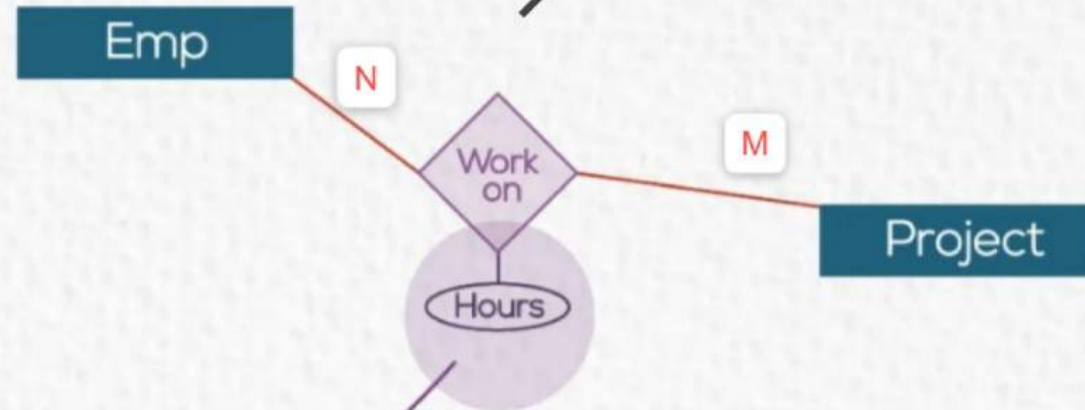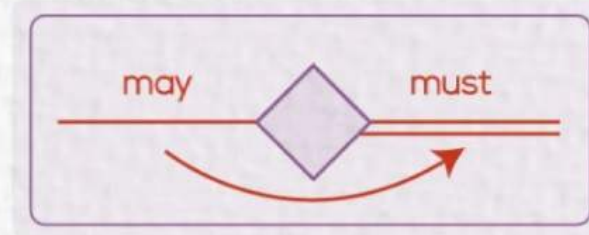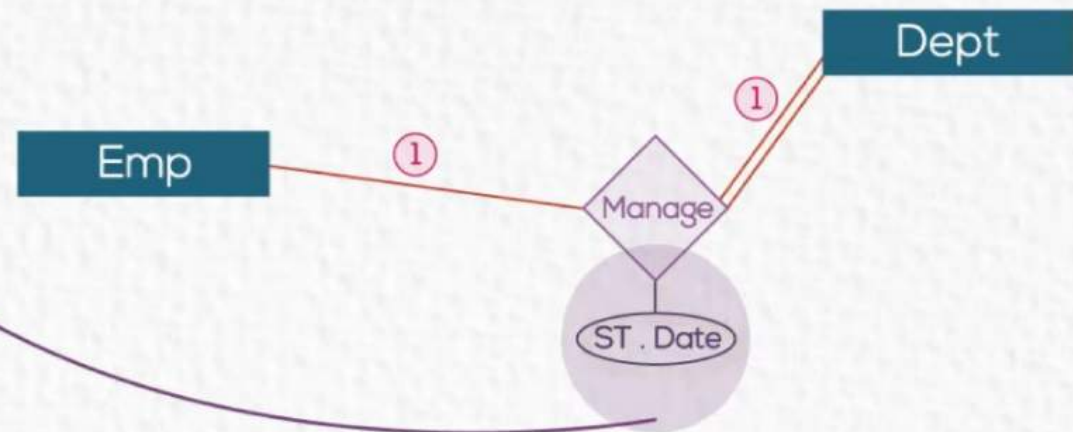Contract (Contract_ID, Type, Start_date)

Skill (Skill_id, Skill_name

Work_On (SSN, PNO, Hours)

SSN still can be used as the PK

Emp (ID, SSN, Salary, Name, DOB, Street, Zone, DNO, Sup-SSN, ⊕ Contract_ID, Type, Start_date

Emp - Phone (SSN, Phone)

Dept (DNO, D Name, LOC, MGR _SSN, ST. Date

Project (PNO, P Name)

Dependent (SSN, Name, Relation)

Car (Plate_NO, Model, Color

~~Contract (Contract_ID, Type, Start_date)~~ ⊗
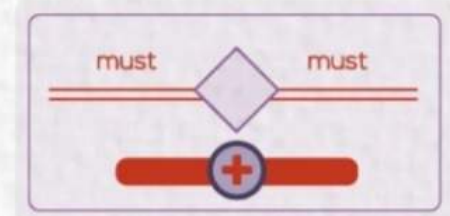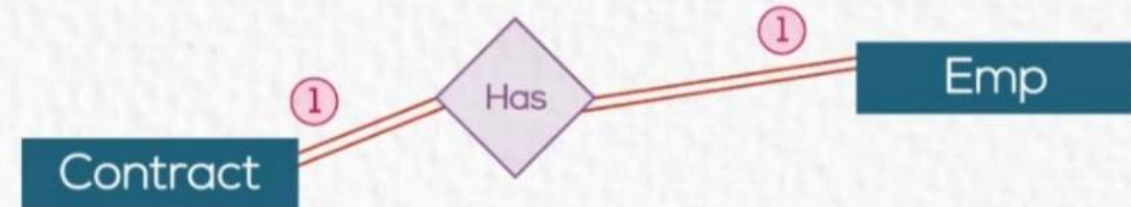
Skill (Skill_id, Skill_name

Work_On (SSN, PNO, Hours)

Contract ——①—— Has ——①—— Emp

Step 6: Mapping of of **ternary relationship** types

Add FKs to the new table for all parent tables

Emp- Contract (ID, SSN, Salary, Name, DOB, Street, Zone, DNO, Sup-SSN, Plate_NO, Contract_ID, Type, Start_date

Emp - Phone (SSN, Phone)

Dept (DNO, D Name, LOC, MGR _SSN, ST. Date

Project (PNO, P Name)

Dependent (SSN, Name, Relation)

Car (Plate_NO, Model, Color

Skill (Skill_id, Skill_name

Work_On (SSN, PNO, Hours)

Skills Used (SSN, PNO, Skill_id)

Step 1: Mapping of regular entity types

Step 2: Mapping of weak entity types

Step 3: Mapping of Binary / Unary 1:M relationship types

Add FK to N-side table

Step 4: Mapping of Binary / Unary M:N relationship types

Add FKs to the new table for
both parent tables

Step 5: Mapping of Binary / Unary 1:1 relationship types

may   must   may   may   must   must
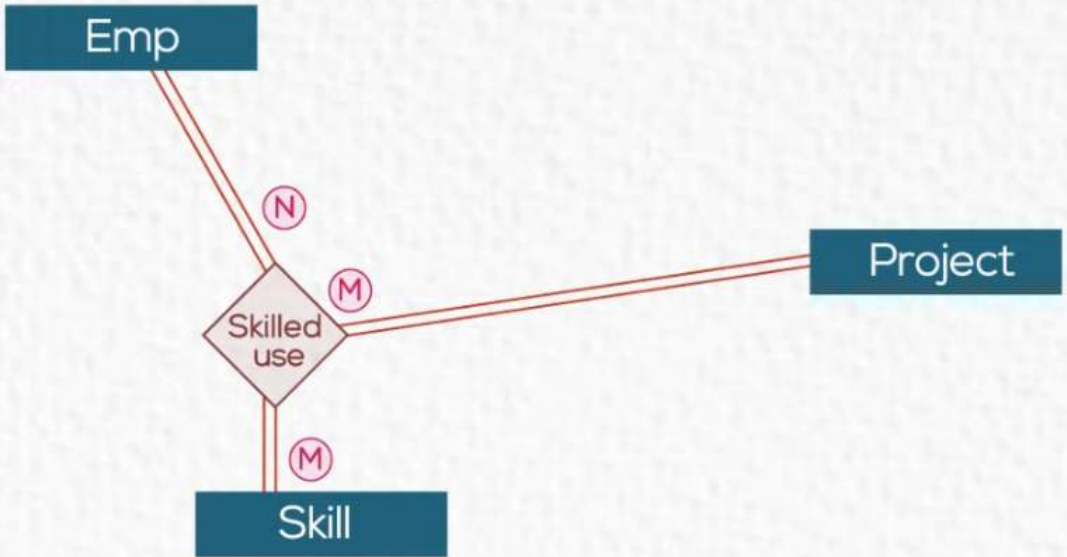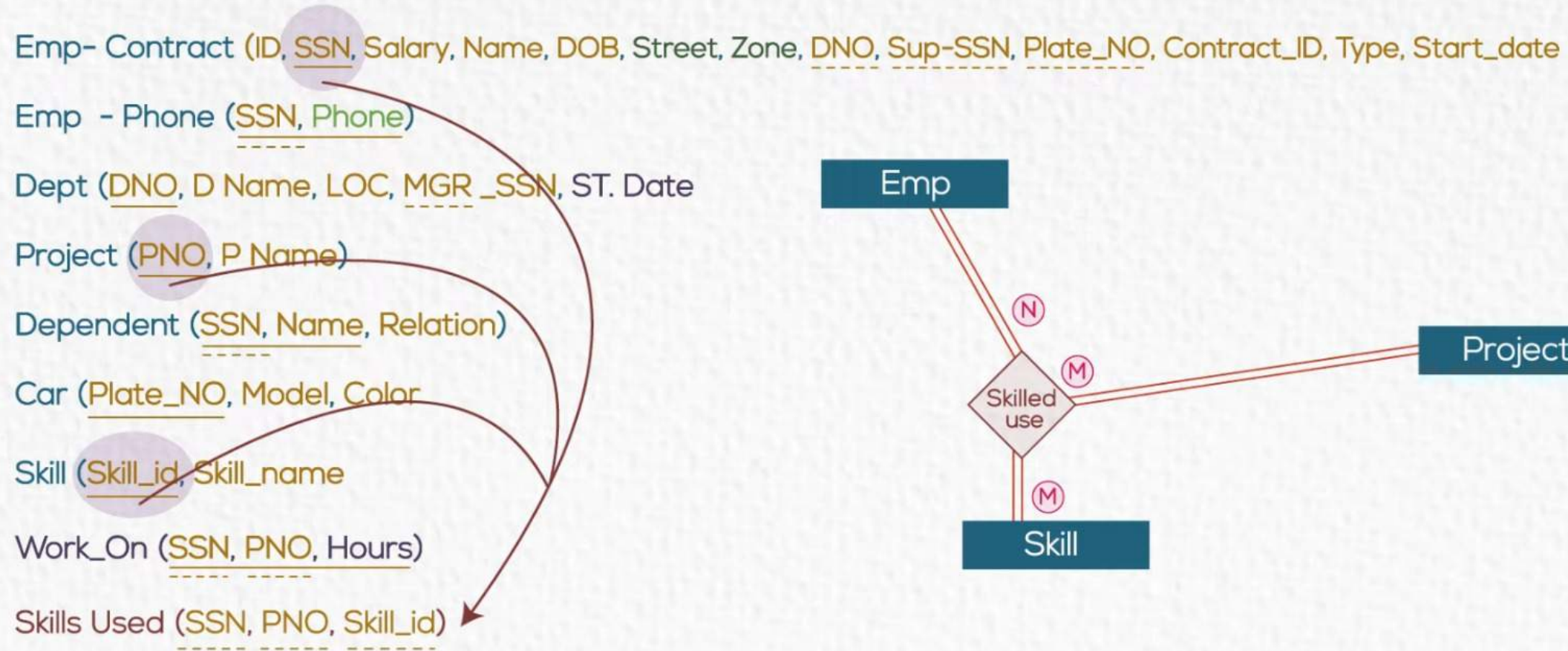
PK of any side
as a FK into the other

Step 6: Mapping of ternary relationship types

Emp- Contract (ID, SSN, Salary, Name, DOB, Street, Zone, DNO, Sup-SSN, Plate_NO, Contract_ID, Type, Start_date

Emp - Phone (SSN, Phone)

Dept (DNO, D Name, LOC, MGR _SSN, ST. Date

Project (PNO, P Name)

Dependent (SSN, Name, Relation)
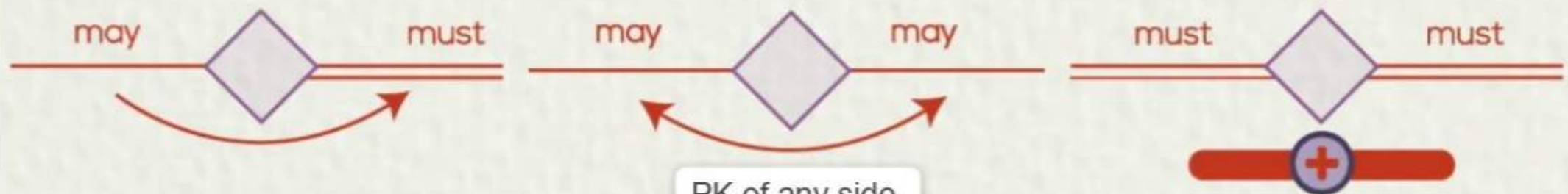
Car (Plate_NO, Model, Color

Skill (Skill_id, Skill_name

Work_On (SSN, PNO, Hours)

Skills Used (SSN, PNO, Skill_id)

attributes on relations follow the FK

Summary

Step 1: Mapping of regular entity types

Step 2: Mapping of weak entity types

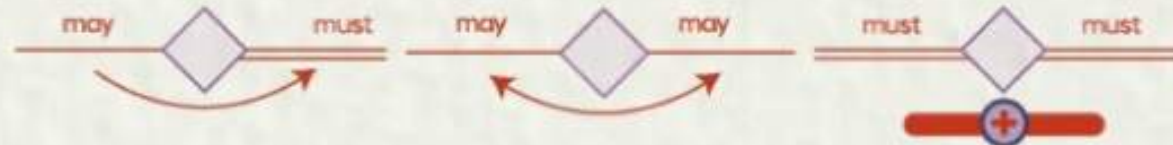**Step 3: Mapping of Binary / Unary 1:M relationship types**

Add FK to N-side table

Step 4: Mapping of Binary / Unary M:N relationship types

Add FKs to the new table for both parent tables

Step 5: Mapping of Binary / Unary 1:1 relationship types

may ◇ must   may ◇ may   must ◇ must

Step 6: Mapping of ternary relationship types

Add FKs to the new table for all parent tables

MOOCA

Step 1: Mapping of **regular entity types**

Step 2: Mapping of **weak entity types**

Step 3: Mapping of **Binary / Unary 1:M** relationship types

> Add FK to N-side table

Step 4: Mapping of **Binary / Unary M:N** relationship types

> Add FKs to the new table for both parent tables

Step 5: Mapping of **Binary / Unary 1:1** relationship types

may ◇ must   may ◇ may   must ◇ must

Step 6: Mapping of **ternary relationship types**

> Add FKs to the new table for all parent tables