# Homework 3

Your Name: Naga Satya Silpa Annadevara

Student ID: A20517818

---

**Q1. Look at the data table below. Convert it to BCNF. Show the steps for 1NF, 2NF, 3NF, BCNF. In each step, explain why it is in or not in a specific normal form, and discuss your solutions to transform them into the corresponding normal form [20 points]**

**TABLE 4-6 Parking Tickets at Millennium College**

Parking Ticket Table

| St ID | L Name | F Name | Phone No | St Lic | Lic No | Ticket # | Date | Code | Fine |
|---|---|---|---|---|---|---|---|---|---|
| 38249 | Brown | Thomas | 111-7804 | FL | BRY 123 | 15634 | 10/17/2018 | 2 | $25 |
| | | | | | | 16017 | 11/13/2018 | 1 | $15 |
| 82453 | Green | Sally | 391-1689 | AL | TRE 141 | 14987 | 10/05/2018 | 3 | $100 |
| | | | | | | 16293 | 11/18/2018 | 1 | $15 |
| | | | | | | 17892 | 12/13/2018 | 2 | $25 |

**ANSWER:**

**1NF Requirement:**

1. There should be no multi-valued attributes. Each column should contain single values only.
2. Primary key should be identified.

**Solution:**
The table above is not in 1NF because the attributes (Ticket#, Date, Code, Fine) have multi-values in them. So, create a new table by adding some more rows by eleminating multi-values in one row and set Ticket# as a primary key.

**Hence the new design that is in 1NF is as follows:**

| St ID | L Name | F Name | Phone No | St Lic | Lic No | **Ticket # (PK)** | Date | Code | Fine |
|---|---|---|---|---|---|---|---|---|---|
| 38249 | Brown | Thomas | 111-7804 | FL | BRY 123 | 15634 | 10/17/2018 | 2 | $25 |
| 38249 | Brown | Thomas | 111-7804 | FL | BRY 123 | 16017 | 11/13/2018 | 1 | $15 |
| 82453 | Green | Sally | 391-1689 | AL | TRE 141 | 14987 | 10/05/2018 | 3 | $100 |
| 82453 | Green | Sally | 391-1689 | AL | TRE 141 | 16293 | 11/18/2018 | 1 | $15 |
| 82453 | Green | Sally | 391-1689 | AL | TRE 141 | 17892 | 12/13/2018 | 2 | $25 |

**2NF Requirement:**

- The entity must be in 1NF first.
- 2. Second, there should be no Partial dependency. ( X -> Y (X determines Y (or) Y is partially dependent on X) and X is only part of the candidate key, not the entire candidate key.

**Solution:**

- Create a new table with X and Y(Y can a single attribute or a combination of attributes).
- Set X as PK in new table, and as FK in old table.
- Remove Y from the old table.

The table above is in 1NF but not in 2NF because if we set (St ID and Ticket #) together as a candidate key, (L name, F name, Phone No, St Lic, Lic No (Y)) are partially dependent on St Id(X). To resolve this, follow the steps below

- Create a new table with all these attributes, X and Y
- Set St Id(X) as an FK in original table and PK in the new table.
- Remove (Y) (L name, F name, Phone No, St Lic, Lic No) in the original table.
- All the attributes are dependent on Ticket #.
- Ticket #, Code, Date, Fine, St Id(FK) will stay in original table.

**Hence the new design that is in 2NF is as follows:**

| Ticket# (PK) | Date | Code | Fine | StID (FK) |
|---|---|---|---|---|
| 15634 | 10/17/2018 | 2 | 25$ | 38249 |
| 16017 | 11/13/2018 | 1 | $15 | 38249 |
| 14987 | 10/05/2018 | 3 | $100 | 82453 |
| 16293 | 11/18/2018 | 1 | $15 | 82453 |
| 17892 | 12/13/2018 | 2 | $25 | 82453 |

| StID (PK) | Lname | Fname | Phone | St Lic | Lic no |
|---|---|---|---|---|---|
| 38249 | Brown | Thomas | 111-7804 | FL | BRY123 |
| 82453 | Green | Sally | 391-01689 | AL | TRE 141 |

**3 NF Requirement:**

- The table must be in 2NF.
- 2. There should be no Transitive dependency in the table. (If X -> B ->Y (X determines B and B determines Y), and B is a non-key attribute, then Y is transitively dependent on X via B.)

**Solution:**

- Create a new table with B and Y (Y can a single attribute or a combination of attributes)

- Remove Y from the old table.
- Set B as PK in new Table and as a FK in old table.

The design above is in 2NF but not in 3NF because there is Transitive dependency in table 1 where Fine (Y) is dependent on Code(B) and Code is dependent on Ticket # (X).
To resolve this, follow the steps below:

- Create a new table with Code(B) and Fine(Y)
- Remove Fine(Y) from the old table.
- Set Code(B) as PK in new table and as a FK in old table.

There is Transitive dependency in table 2 as well, where St Lic(Y) is dependent on Lic No(B) and Lic No is dependent on St ID(X).
To Resolve this, follow the steps below:

- Create a new table with Lic No(B) and St Lic(Y)
- Remove St Lic(Y) from the old table
- Set Lic No(B) as PK in new table and as a FK in old table

**Hence the new design that is in 3NF is as follows:**

| Ticket# (PK) | Date | Code (FK) | St ID (FK) |
|---|---|---|---|
| 15634 | 10/17/2018 | 2 | 38249 |
| 16017 | 11/13/2018 | 1 | 38249 |
| 14987 | 10/05/2018 | 3 | 82453 |
| 16293 | 11/18/2018 | 1 | 82453 |
| 17892 | 12/13/2018 | 2 | 82453 |

| Code (PK) | Fine |
|---|---|
| 1 | $15 |
| 2 | $25 |
| 3 | $100 |

| StID (PK) | Lname | Fname | Phone | Lic No (FK) |
|---|---|---|---|---|
| 38249 | Brown | Thomas | 111-7804 | BRY123 |
| 82453 | Green | Sally | 391-01689 | TRE 141 |

| St Lic | Lic No (PK) |
|---|---|
| FL | BRY123 |
| AL | TRE 141 |

**BCNF Requirement:**

- The table must be in 3NF
- Every determinant must be a candidate key ( in other words only full dependency is allowed)

the table must meet one of the following:

- Each table must have 1 candidate key (single attribute or a combination of attributes)
- Each table must have multiple candidate keys but each candidate key must have only 1 attribute.

**Solution:**
the table must meet one of the following:

- It must be in 3NF
- Each table must have 1 candidate key (single attribute or a combination of attributes)
- Each table must have multiple candidate keys but each candidate key must have only 1 attribute.

The design above is in 3NF but in BCNF, Because in table 1 there are multiple candidate keys ( Ticket# is already a CK and Code + St ID together forms as a CK which is BCNF rule violation)
So, create separate table.
Break down the tables even more into smaller ones.

**Hence the BCNF design is as follows:**

| Ticket# (PK) | Date | Code (FK) |
|---|---|---|
| 15634 | 10/17/2018 | 2 |
| 16017 | 11/13/2018 | 1 |
| 14987 | 10/05/2018 | 3 |
| 16293 | 11/18/2018 | 1 |
| 17892 | 12/13/2018 | 2 |

| Ticket# (PK) | Date | StID (FK) |
|---|---|---|
| 15634 | 10/17/2018 | 38249 |
| 16017 | 11/13/2018 | 38249 |
| 14987 | 10/05/2018 | 82453 |
| 16293 | 11/18/2018 | 82453 |
| 17892 | 12/13/2018 | 82453 |

| Code (PK) | Fine |
|---|---|
| 1 | $15 |
| 2 | $25 |
| 3 | $100 |

| StID (PK) | Lname | Fname | Lic No (FK) |
|---|---|---|---|
| 38249 | Brown | Thomas | BRY123 |
| 82453 | Green | Sally | TRE 141 |

| St Lic | Lic No (PK) |
|---|---|
| FL | BRY123 |
| AL | TRE 141 |

| StID (PK) | Phone |
|---|---|
| 38249 | 111-7804 |
| 82453 | 391-01689 |

**Q2. Given the tables below, figure out super keys, candidate keys, primary keys, secondary keys and foreign keys [10 points]**

| EmpID | Name | BirthYear | Gender | Address | SSN | DeptID | Email |
|---|---|---|---|---|---|---|---|
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |

| DeptID | Name | Address | DeptHead |
|---|---|---|---|
| | | | |
| | | | |
| | | | |
| | | | |

**ANSWER:**

**Table 1 (Employee table):**

**Super key:** EmpID , SSN, Email

- EmpID with any other attributes.
- SSN with any other attributes.
- Email with any other attributes.
- All the 3 super keys together
- All the attributes together

**Candidate key:** EmpID , SSN, Email

**Primary key:** EmpID

**Secondary key:** SSN, Email

**Foreign key:** DeptID

**Table 2 (Department table):**

**Super key:** DeptID with any other attribute

**Candidate key:** DeptID

**Primary key:** DeptID

**Secondary key:** None

**Foreign key:** None

**Q3. Below is the diagram for a university dinning service [20 points]**

**In this scenario, we assume that the work schedule is defined for each specific event, and one staff may work for different positions on different events.**

**a). identify the relationships for STAFF-STAFF and explain why.**

**ANSWER:**

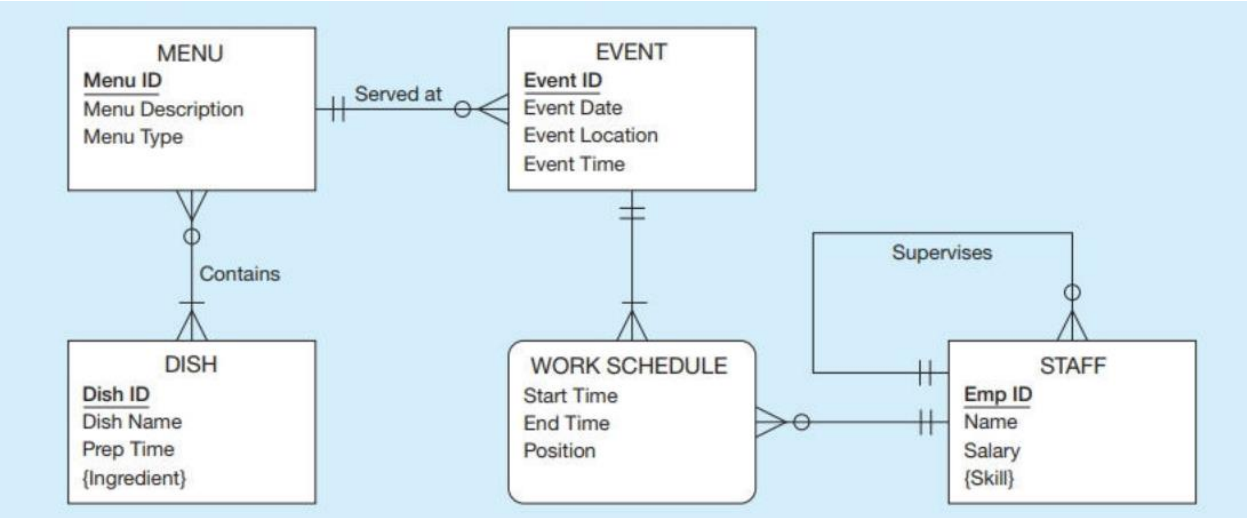
The relation for STAFF-STAFF is "Recursive relationship" is one in which a relationship can exist between occurrences of the same entity set.

It is because, In STAFF entity, there might be a MANAGER or SUPERVISOR who is also an Employee/Staff with EmpID that SUPERVISES other employees. So, here the MANAGER or SUPERVISOR (employee/Staff) details will also be held in staff entity.

An Employee/Staff(MANAGER or SUPERVISOR) may supervise many employees/Staff, and each Employee/Staff is managed by at least 1 employee/staff(MANAGER/SUPERVISOR).

So, The relation is 1(mandatory) : M(optional) (unary relationship)

**b). determine the normal form for each entity, if it is not in 3NF, convert it to 3NF.**




**ANSWER:**

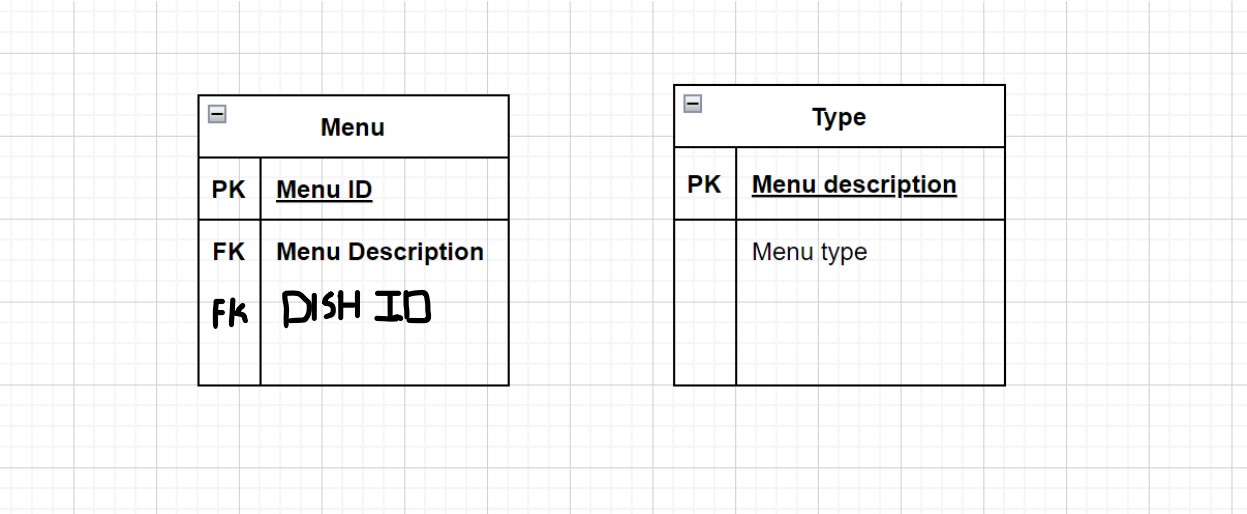
MENU:

- it's in 1NF (because there are no multi-valued attributes)
- It's in 2NF (because there is no partial dependency)
- But it's not in 3NF because there is transitive dependency where Menu Type (Y) is dependent on Menu Description(B) and Menu Description(B) is dependent on Menu ID(X).

Solution:

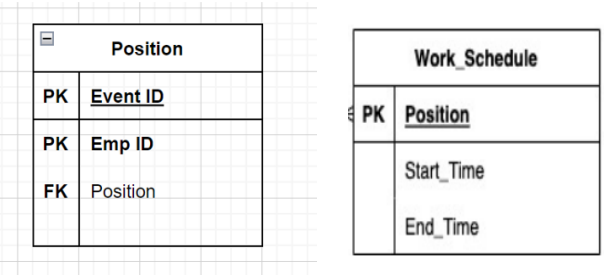
- So, create a new table called Type with Menu Description (B) and Menu Type (Y)
- Set Menu Description (B) as PK in new table and as a FK in old table.(Menu table)
- Remove Menu Type (Y) from the old table. (Menu Table)

**Hence 3NF Design for Menu table is as follows:**




EVENT: it is already in 3NF.

WORK SCHEDULE: it is not in 3NF. Because of transitive dependency. So the below table is introduced.




DISH: its not in 1NF, 2NF, and in 3NF

- The ingredient is a multivalued attribute. So, it's not 1NF.

  Solution:
  - Remove multi-values by adding extra rows.

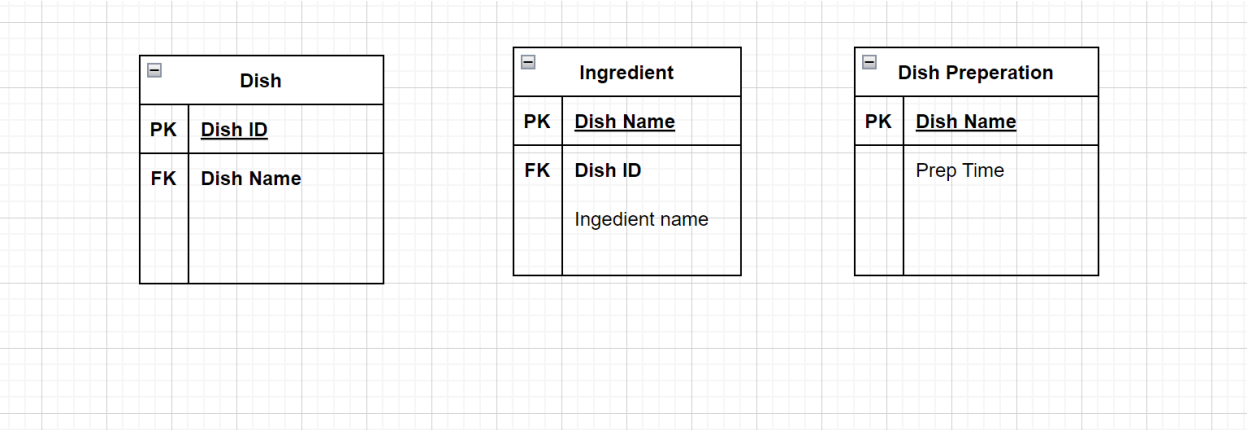
- There is Transitive dependency where, Ingredient(Y) depends on Dish Name(B) and Dish Name(B) is dependent on Dish ID(X). So, it's not in 3NF.

  Solution:
  - Create new table called INGREDIENT with Dish Name(B) and Ingredient(Y)
  - Set Dish Name(B) as PK in new table and as a FK in old table.
  - Remove Ingredient(Y) from the old table.

- There is partial dependency where prep Time(Y) is partially dependent on Dish Name.(X) So, it's not in 2NF.

  Solution:
  - Create a new table called Dish Preparation with Dish Name(X) and Prep Time(Y)
  - Set Dish Name as PK in new table and as a FK in old table.
  - Remove Prep Time (Y) from the old table.

**Hence 3NF Design for Dish table is as follows:**



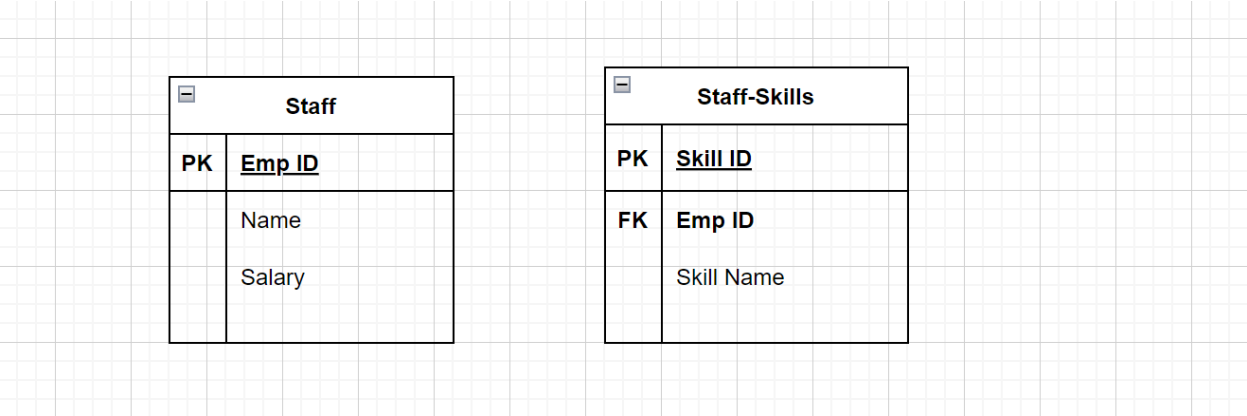

STAFF:

- it's not in 1NF, because The Skills is a multivalued attribute.

  Solution:
  - Remove multi-values by adding extra rows. (or)
  - Create a separate table called (staff-skills table) with attributes Skill ID, Skill Name and Emp ID.

- assigning Skill ID as a PK and Emp ID as a FK in the new table (staff-skills table).
- Remove skills column in the original table. (Staff table)

**Hence New Design for Staff table is as follows:**

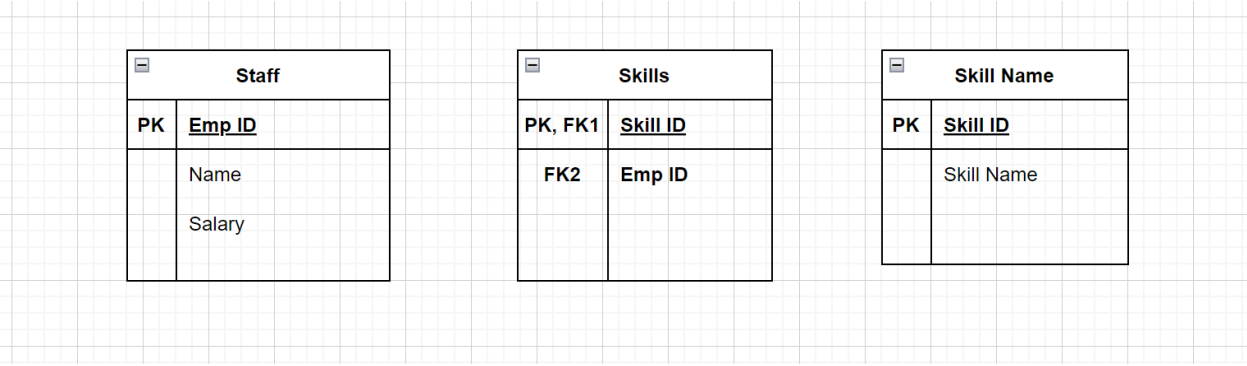


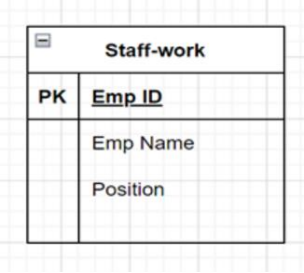

- The Skills table is not in 2NF because there is a partial dependency.
- Because if Skills ID + Emp ID together forms as a candidate key, Skill name(Y) is partially dependent on Skill ID(X) Which is a part of Candidate key.

Solution:

- So crate a new table called skills with Skill ID(X) and Skill name(Y)
- Set Skill ID as a PK in new table(skill table) and as a FK in old table.(staff-skills table)
- Remove Skill Name(Y) from the old table.(staff-skills table)

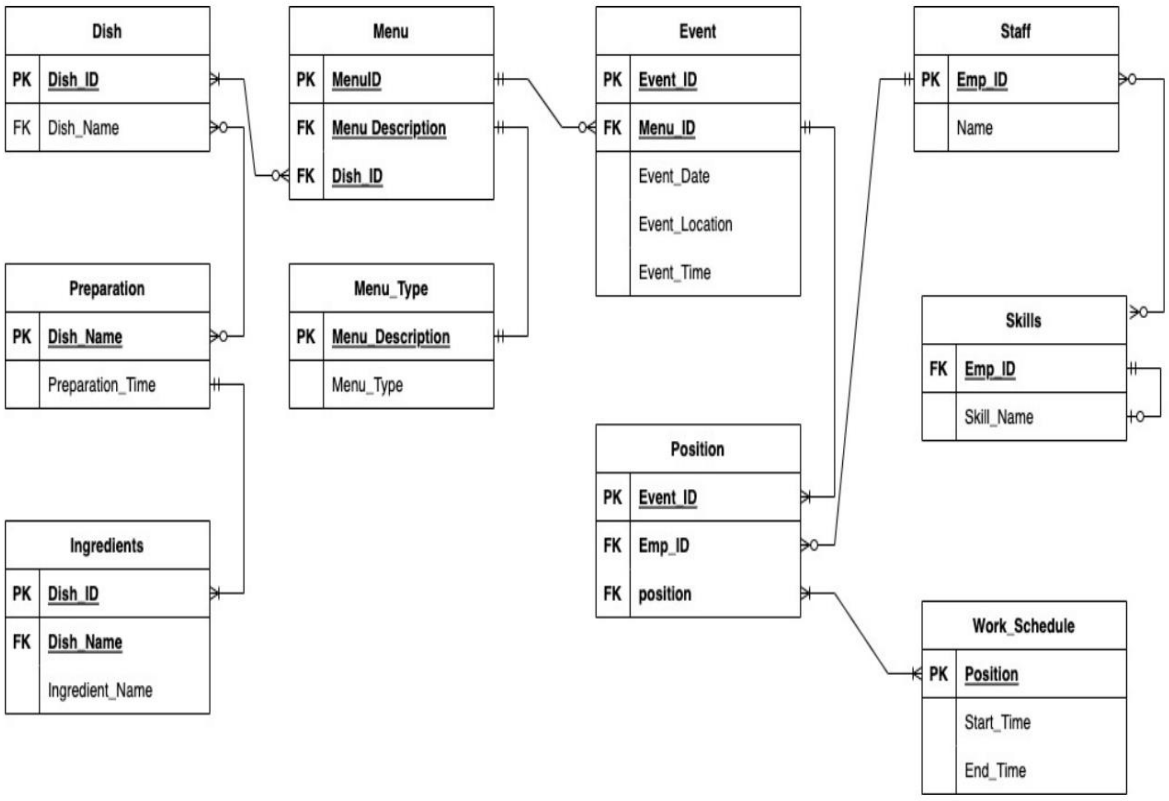
**Hence New Design for Staff table is as follows:**



- Staff -work table is introduced to make sure each staffer is assigned a role or to check position of each staff. Because there is also a supervisor who is also a staff.

**The final 3NF ERD is as follows:**




**Q4. DB Design [50 points]**

System Descriptions

- Build a DB for hotel bookings
- Each hotel only has two types of rooms: 1-double bedroom, and two-twin bedroom
- Each hotel may have multiple rooms
- Clients can place orders. In one order, they can book multiple rooms from a same hotel

Draw ERD to design a DB in 3NF

**Note: you can directly give the ERD in physical DB design, but you must make sure it is in 3NF.**

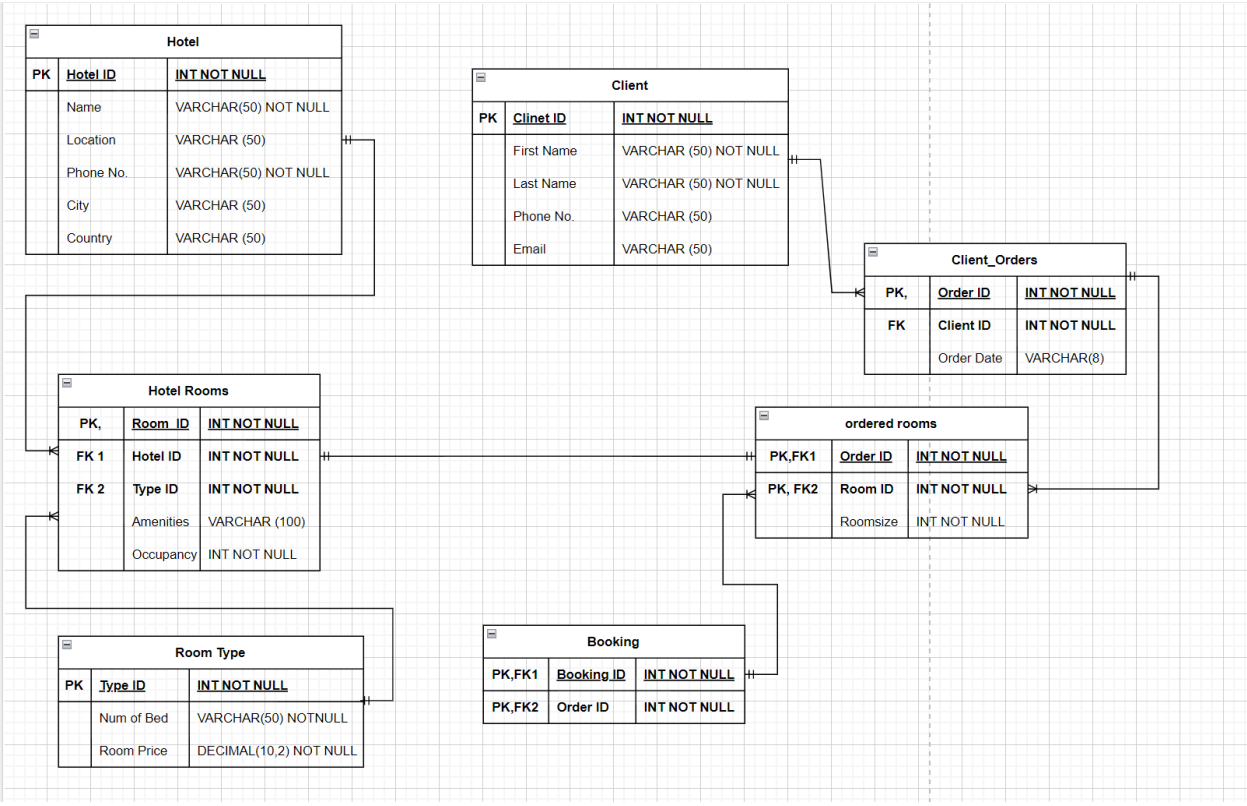
**ANSWER**:

The Main entities we have with attributes and PK's:

1) **Hotel** (HotelID(pk), Name, Location, PhNo.,City,Country)
2) **Client** (Client ID(pk), F_Name, L_Name, PhNo., Email)

**Additional entities that need to be added in the process of 3NF Normalization:**

3) **Hotel_Rooms** (Room ID-pk&fk , Hotel ID – fk , Type ID – fk ,Amenities, Occupancy)
4) **Room Type** (Type ID – pk, Num of Bed, Room price)
5) **Client_Orders** ( Order ID -pk, Client ID – pk, Order_date)
6) **Ordered_Rooms** (Oder ID – pk&fk, Room ID – pk&fk, Room size)
7) **Booking** (Booking ID – pk&fk , Oder ID – pk&fk)




**FEEDBACK & SCORE : 99**

HW3

Q4. -1 Relationship between hotel rooms and ordered rooms must be 1:M not 1:1