

**Justifications:**

- Each employee works on only one department, but each department has multiple employees.
- Each department is supervised by a single employee who overlooks the department and manages every day in and out activities.
- Each employee is assigned with a specific ID and later have their own details in the form of name, hired date, designation, contact number, email address, address. Required attributes will be added on further during the dynamics of the process.
- Each employee can reside on multiple addresses and have multiple email addresses.

**Assumptions:**

- Each employee can have only one contact number.
- Addition of further attributes like ID, description, and floor to define the department.

**Un-normalized Form (UNF):**

All the list of possible attributes are then assigned under a single table of one belt and then the normalization procedure is carried out.

**Employee** - (Employee\_ID, Employee\_Name, Employee\_Contact\_Number, Employee\_Hired\_Date, Employee\_Designation, {Email\_Address}, {Employee\_Address}, Department\_ID, Department\_Name, Department\_Description, Department\_Floor, Department\_Supervisor)

**First Normal Form (1NF):**

The first normal form defines the removal of repeating groups to separate the repeating data from its corresponding repeating group, hence the remaining tables can be justified by the outlined tables and its respective categories:

**Employee – 1** (Employee\_ID, Employee\_Name, Employee\_Contact\_Number, Employee\_Hired\_Date, Employee\_Designation, Department\_ID, Department\_Name, Department\_Description, Department\_Floor, Department\_Supervisor)

**Employee.Email – 1** (Email\_ID, Email\_Address, Email\_Category, Employee\_ID\*)

**Employee.Address – 1** (Address\_ID, Address, Address\_Category, State, ZIP, Postal Code, Street, Employee\_ID\*)

### **Second Normal Form (2NF):**

Once the repeating group and data are distinguished, the steps of separating partial functional dependencies from its parent are carried out.

The employee table gets credited as the same since it doesn't possess any multiple composite keys. So, it is overwritten as the same table.

**Employee – 2** (Employee\_ID, Employee\_Name, Employee\_Contact\_Number, Employee\_Hired\_Date, Employee\_Designation, Department\_ID, Department\_Name, Department\_Description, Department\_Floor, Department\_Supervisor)

However, the remaining email table consists of composite primary keys and its dependencies need to be evaluated further on.

Email\_ID, Employee\_ID →

Email\_ID → Email\_Address, Email\_Category

Employee\_ID →

Similarly, the following address tables is further checked upon.

Address\_ID, Employee\_ID →

Address\_ID → Address, Address\_Category, State, ZIP, Postal Code, Street

Employee\_ID →

The final tables after 2NF are as follows:

**Employee – 2** (Employee\_ID, Employee\_Name, Employee\_Contact\_Number, Employee\_Hired\_Date, Employee\_Designation, Department\_ID, Department\_Name, Department\_Description, Department\_Floor, Department\_Supervisor)

**Employee.Email – 2** (Email\_ID, Email\_Address, Email\_Category, Employee\_ID\*)

**Employee.Address – 2** (Address\_ID, Address, Address\_Category, State, ZIP, Postal Code, Street, Employee\_ID\*)

### **Third Normal Form (3NF):**

Now, the third normal form defines the removal of transitive dependencies from its actual parent table. For this, each of the table needs to be evaluated based on whether the condition fulfilled by the transitive dependency.

Employee\_ID → Department\_ID → Department\_Name, Department\_Description, Department\_Floor, Department\_Supervisor

Hence, the tables can be separated as employee and department tables.

**Employee – 3** (Employee\_ID, Employee\_Name, Employee\_Contact\_Number, Employee\_Hired\_Date, Employee\_Designation, Department\_ID\*)

**Department – 3** (Department\_ID, Department\_Name, Department\_Description, Department\_Floor, Department\_Supervisor\*)

Since the supervisor of a department is allocated among the employees, it should have a foreign key link up with the employee table. Hence, the department supervisor column is defined as per the employee id.

Like the employee table, transitive dependency examination will be done on the remaining tables too.

Email\_ID →

Address\_ID →

As they do not reciprocate any transitive dependencies, the remaining tables remain the exact same when defined after the 3NF procedure. Hence, the final tables after normalization are as follows:

**Department:** Department\_ID, Department\_Name, Department\_Description,  
Department\_Floor, Department\_Supervisor\*

**Employee:** Employee\_ID, Employee\_Name, Employee\_Contact\_Number,  
Employee\_Hired\_Date, Employee\_Designation, Department\_ID\*

**Email:** Email\_ID, Email\_Address, Email\_Category, Employee\_ID\*

**Address:** Address\_ID, Address, Address\_Category, State, ZIP, Postal Code, Street,  
Employee\_ID\*

### Justifications:

- Employees can vote anyone but themselves.
- Employees can vote anyone from other departments as well.

### Assumptions:

- Each employee can vote to any candidate once a month.
- Addition of further attributes like ID, description, and floor to define the department.

### Un-normalized Form (UNF):

All the list of possible attributes are then assigned under a single table of one belt and then the normalization procedure is carried out.

**Voter** – (Voter\_ID, Voter\_Name {Voting\_Year {Voting\_Month, Candidate\_ID, Candidate\_Name, Candidate\_Department\_ID, Department\_Name, Department\_Description, Department\_Floor, Department\_Supervisor)

### First Normal Form (1NF):

The first normal form specifies the elimination of repeating groups to remove repeating data from its associated repeating group; consequently, the residual tables can be rationalized by the described tables and their related categories.

**Voter – 1** (Voter\_ID, Voter\_Name)

**Voting.Year – 1** (Voting\_Year, Voter\_ID\*)

**Voting.Details – 1** (Detail\_ID, Voter\_ID\*, Voting\_Month, Voting\_Year\*, Candidate\_Name, Candidate\_Department\_ID, Department\_Name, Department\_Description, Department\_Floor, Department\_Supervisor)

## Second Normal Form (2NF):

After distinguishing the repeating group and data, the methods for removing its partial functional dependencies from its parent are executed.

The voter table gets credited as the same since it doesn't possess any multiple composite keys. So, it is overwritten as the same table.

### **Voter – 2** (Voter\_ID, Voter\_Name)

Evaluation for the voting details table for investigation of any existing partial functional dependencies.

Detail\_ID →

Voter\_ID →

Voting\_Year →

Detail\_ID, Voter\_ID →

Detail\_ID, Voting\_Year →

Voter\_ID, Voting\_Year →

Detail\_ID, Voter\_ID, Voting\_Year → Voting\_Month, Candidate\_Name,  
Candidate\_Department\_ID, Department\_Name, Department\_Description,  
Department\_Floor, Department\_Supervisor

Thus, the tables remain as it is after the 2NF evaluation:

**Voting.Details – 2** (Detail\_ID, Voter\_ID\*, Voting\_Month, Voting\_Year\*,  
Candidate\_Name, Candidate\_Department\_ID, Department\_Name,  
Department\_Description, Department\_Floor, Department\_Supervisor)

### Third Normal Form (3NF):

Since, the voter table represents only one non-primary key attribute, it gets shifted as the same table in the third normal form as well.

**Voter – 3** (Voter\_ID, Voter\_Name)

The evaluation of third normal form on the next table can be studied by the following

Detail\_ID, Voter\_ID, Voting\_Year → Candidate\_ID → Candidate\_Name,  
Candidate\_Department\_ID, Department\_Name, Department\_Description,  
Department\_Floor, Department\_Supervisor

Candidate\_Department\_ID → Department\_Name → Department\_Description,  
Department\_Floor, Department\_Supervisor

Hence, the tables organized after the third normal form are as follows:

**Voting.Details – 3** (Detail\_ID, Voter\_ID\*, Voting\_Month, Voting\_Year\*, Candidate\_ID)

**Candidate – 3** (Candidate\_ID, Candidate\_Name, Candidate\_Department\_ID\*)

**Department – 3** (Department\_ID, Department\_Name, Department\_Description,  
Department\_Floor, Department\_Supervisor)

Since, both the candidate and voter table reciprocate as an employee table, they both share the same properties as the employee table. Similar to the prior table description of department, it remains same as above too. The table which represents a bridge entity defining the voting details is outlined:

**Voting.Details:** Detail\_ID, Voter\_ID\*, Voting\_Month, Voting\_Year\*, Candidate\_ID