## Normalization

- Normalization is a database design technique that reduces data redundancy and eliminates undesirable characteristics like Insertion, Update and Deletion Anomalies.
- Normalization rules divides larger tables into smaller tables and links them using relationships.
- The purpose of Normalization in SQL is to eliminate redundant (repetitive) data and ensure data is stored logically.

**Normal forms** are used to eliminate or reduce redundancy in database tables.

## First Normal Form (1NF)

- A relation will be in 1NF if it contains an atomic value
- It states that an attribute of a table cannot hold multiple values. It must hold only single valued attribute.
- First normal form disallows the multi valued attribute, composite attribute, and their combinations.

### Example

Relation EMPLOYEE is not in 1NF because of multi-valued attribute EMP_PHONE.

| EMP_ID | EMP_NAME | EMP_PHONE | STATE |
|---|---|---|---|
| 2925 | RAJESH | 9500407832,9088170884 | TAMILNADU |
| 6796 | KISHORE | 9510508668 | KARNATAKA |
| 4592 | ALOK | 8455661732 | PUNJAB |
| 3565 | HEMANT | 9759105522,9414525254 | GUJRAT |

The decomposition of the EMPLOYEE table into 1NF has been shown below

| EMP_ID | EMP_NAME | EMP_PHONE | STATE |
|---|---|---|---|
| 2925 | RAJESH | 9500407832 | TAMILNADU |
| 2925 | RAJESH | 9088170884 | TAMILNADU |
| 6796 | KISHORE | 9510508668 | KARNATAKA |
| 4592 | ALOK | 8455661732 | PUNJAB |
| 3565 | HEMANT | 9759105522 | GUJRAT |
| 3565 | HEMANT | 9414525254 | GUJRAT |

## Second Normal Form (2NF)

- In the 2NF, relational must be in 1 NF

- In the second normal form, all non-key attributes are fully functionally dependent on the primary key.

**Example**

In a college, a teacher can teach more than one subject.

TEACHER Table:

| TEACHER_ID | SUBJECT | TEACHER_AGE |
|---|---|---|
| 2529 | Information Technology | 30 |
| 2529 | Soft Computing | 30 |
| 6535 | Data Communication | 45 |
| 3578 | Soft Computing | 35 |
| 3578 | Digital Image Processing | 35 |

In the given table, non-prime attribute TEACHER_AGE is dependent on TEACHER_ID. That's why it violates the rule for 2NF. To convert the given table into 2NF, we decompose it into two tables.

TEACHER_DETAIL

| TEACHER_ID | TEACHER_AGE |
|---|---|
| 2529 | 30 |
| 6535 | 45 |
| 3578 | 35 |

TEACHER_SUBJECT

| TEACHER_ID | SUBJECT |
|---|---|
| 2529 | Information Technology |
| 2529 | Soft Computing |
| 6535 | Data Communication |
| 3578 | Soft Computing |
| 3578 | Digital Image Processing |

## Third Normal Form (3NF)

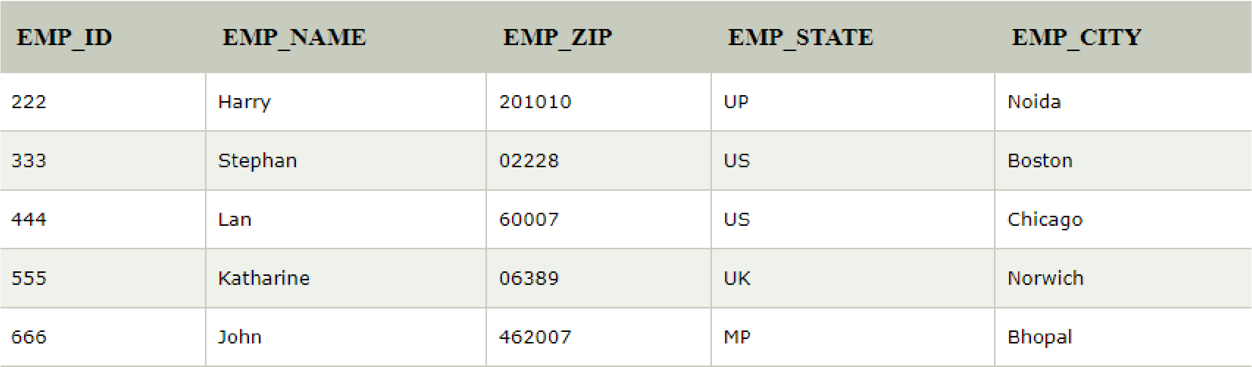
- A relation will be in 3NF if it is in 2NF and not contain any transitive partial dependency.
- 3NF is used to reduce the data duplication It is also used to achieve the data integrity.
- If there is no transitive dependency for non-prime attributes, then the relation must be in third normal form.

A relation is in third normal form if it holds at least one of the following conditions for every non-trivial function dependency X-> Y

1. X is a super key.

2. Y is a prime attribute, i.e., each element of Y is part of some candidate key.

EMPLOYEE_DETAIL table



| EMP_ID | EMP_NAME | EMP_ZIP | EMP_STATE | EMP_CITY |
|---|---|---|---|---|
| 222 | Harry | 201010 | UP | Noida |
| 333 | Stephan | 02228 | US | Boston |
| 444 | Lan | 60007 | US | Chicago |
| 555 | Katharine | 06389 | UK | Norwich |
| 666 | John | 462007 | MP | Bhopal |

Super key: {EMP_ID}, {EMP_ID, EMP_NAME},{EMP_ ID, EMP_NAME, EMP_ZIP} so on.

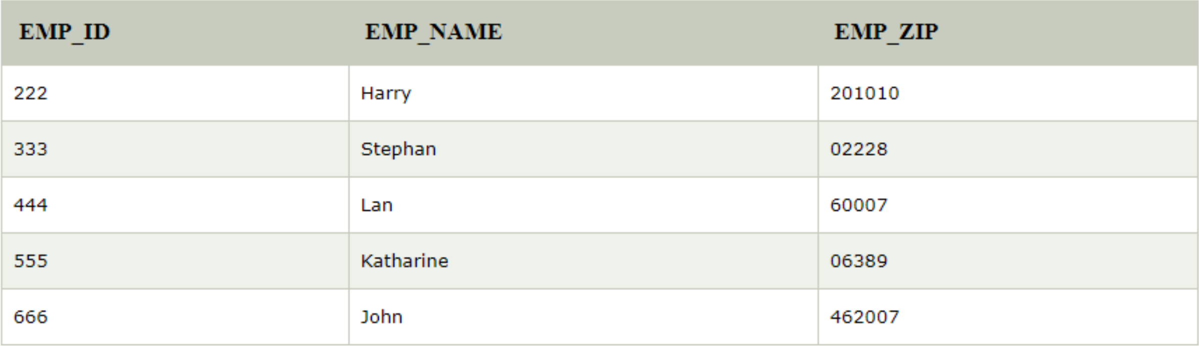
Candidate key: {EMP_ID}

Non-prime attributes: All attributes except EMP_ID

Here EMP_STATE and EMP_CITY dependent on EMP_ZIP and EMP_ZIP dependent on EMP_ID. The non-prime attributes (EMP_ STATE, EMP_CITY) transitively dependent on super key (EMP_ID). It violates the rule of third normal form.

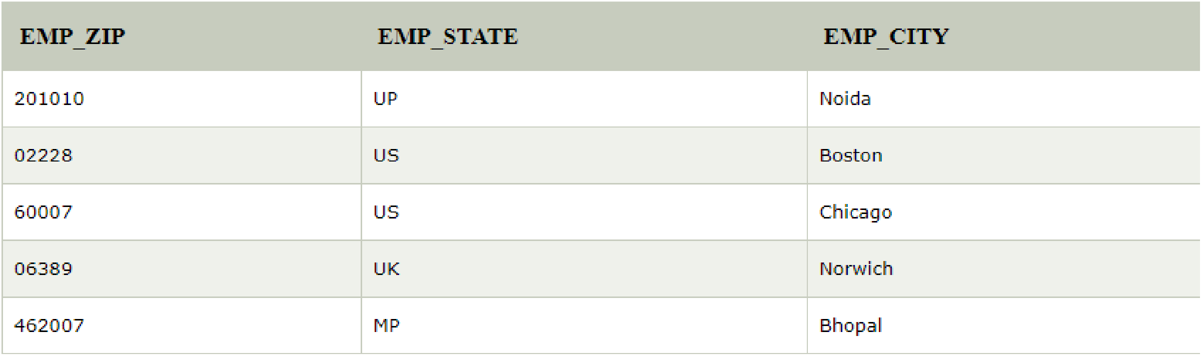
We need to move the EMP_CITY and EMP_STATE to the new EMPLOYEE_ZIP table, with EMP_ZIP as a Primary key.

EMPLOYEE Table:



| EMP_ID | EMP_NAME | EMP_ZIP |
|---|---|---|
| 222 | Harry | 201010 |
| 333 | Stephan | 02228 |
| 444 | Lan | 60007 |
| 555 | Katharine | 06389 |
| 666 | John | 462007 |

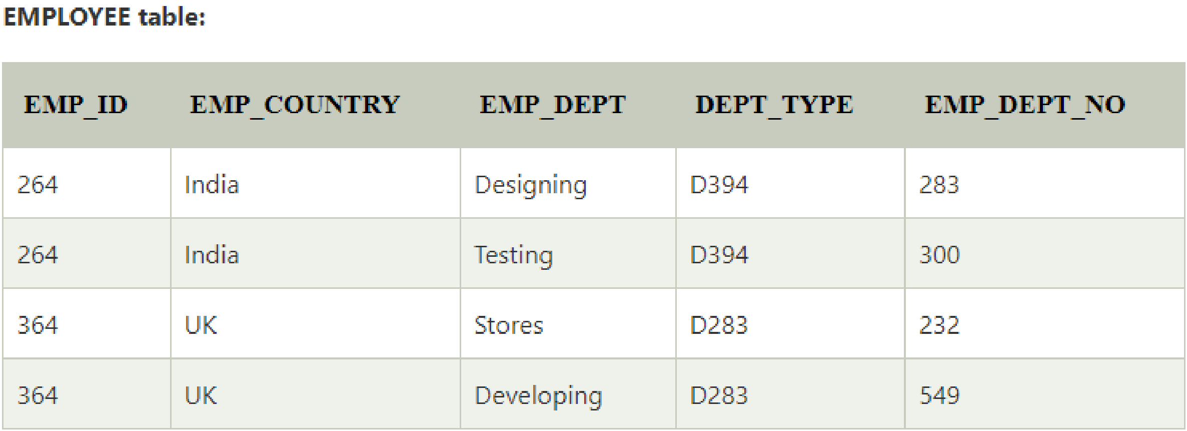
EMPLOYEE_ZIP Table:



| EMP_ZIP | EMP_STATE | EMP_CITY |
|---|---|---|
| 201010 | UP | Noida |
| 02228 | US | Boston |
| 60007 | US | Chicago |
| 06389 | UK | Norwich |
| 462007 | MP | Bhopal |

## Boyce Codd Normal Form (BCNF)

BCNF is the advance version of 3NF. It is stricter than 3NF. A table is in BCNF if for every functional dependency X -> Y, X is the super key of the table. For BCNF, the table should be in 3NF, and for every FD, LHS is super key.



**EMPLOYEE table:**

| EMP_ID | EMP_COUNTRY | EMP_DEPT | DEPT_TYPE | EMP_DEPT_NO |
|---|---|---|---|---|
| 264 | India | Designing | D394 | 283 |
| 264 | India | Testing | D394 | 300 |
| 364 | UK | Stores | D283 | 232 |
| 364 | UK | Developing | D283 | 549 |

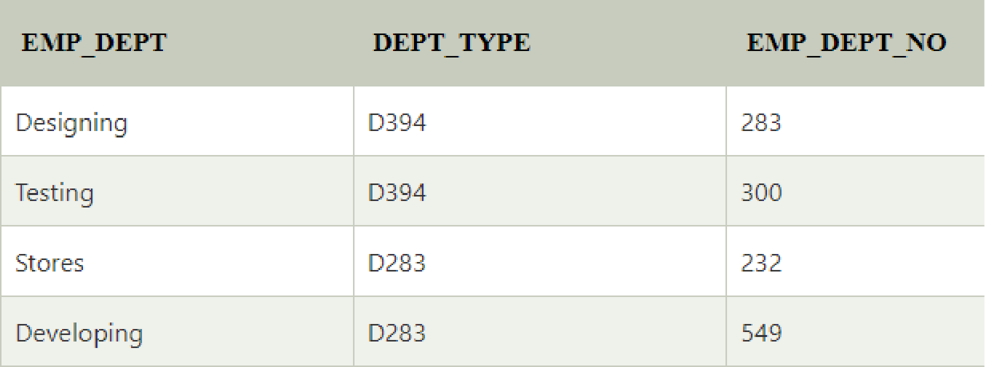
In the above table Functional dependencies are as follows:

EMP_ID -> EMP_COUNTRY

EMP_DEPT ->{DEPT_TYPE, EMP_DEPT_NO}

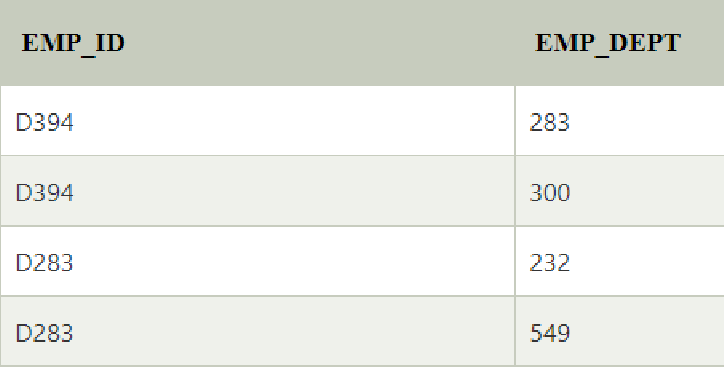
| EMP_ID | EMP_COUNTRY |
|---|---|
| 264 | India |
| 364 | UK |

**EMP_DEPT table:**



| EMP_DEPT | DEPT_TYPE | EMP_DEPT_NO |
|---|---|---|
| Designing | D394 | 283 |
| Testing | D394 | 300 |
| Stores | D283 | 232 |
| Developing | D283 | 549 |

**EMP_DEPT_MAPPING table:**



| EMP_ID | EMP_DEPT |
|---|---|
| D394 | 283 |
| D394 | 300 |
| D283 | 232 |
| D283 | 549 |