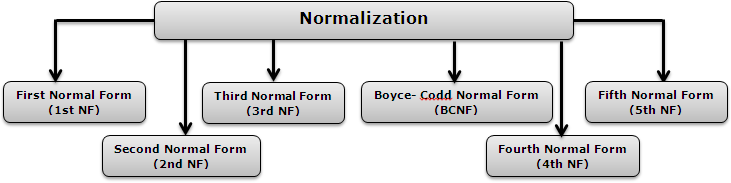
**Normalization**

**what is normalization in database?**

* Normalization is the process of organizing the data in the database.
* It is used to minimize the redundancy from a relation or set of relations.
* It is used to eliminate undesirable characteristics like insertion, update, and Deletion Anomalies.
* Normalization divides the large table into smaller and links them using relationships.
* The normal form is used to reduce redundancy from the database table.

**When to Normalize the database?**

* Arranging data into logical groupings such that each group describes a small part of the whole.
* Minimizing the amount of duplicate data, called redundancy, stored in a database.
* Organizing the data such that, when you modify it, you make the changes only in one place.
* Building a database in which you can access and manipulate the data quickly and efficiently without compromising the integrity of the data in storage.



**Types of Normal forms**

1NF

2NF

3NF

4NF

5NF

BCNF

**Levels of Normalization**

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

1NF is the simplest level of normalization. It ensures that each cell in a table contains only one atomic value (indivisible value).

It eliminates repeating groups.

**Example:**

Unnormalized Table (Not in 1NF):

|  |  |  |  |
| --- | --- | --- | --- |
| StudentID | Name | Course | Grade |
| 101 | Deepam | Math, Science | A, B |
| 102 | Aditya | History, English | C, D |

Normalized Table (in 1NF):

|  |  |  |  |
| --- | --- | --- | --- |
| StudentID | Name | Course | Grade |
| 101 | Deepam | Math | A |
| 101 | Deepam | Science | B |
| 102 | Aditya | History | C |
| 102 | Aditya | English | D |

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

2NF is another level where a certain relation is in 1NF and all non-key attributes are fully functional dependent on the primary key.

It eliminates partial functional dependency.

**Example:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| OrderID | ProductID | ProductName | CustomerID | CustomerName |
| 101 | P1 | Laptop | C1 | Alice |
| 101 | P2 | Phone | C1 | Alice |
| 102 | P3 | Tablet | C2 | Bob |

2NF Normalize Table:

Orders table:

|  |  |
| --- | --- |
| OrderID | CustomerID |
| 101 | C1 |
| 102 | C2 |

Customers Table

|  |  |
| --- | --- |
| CustomerID | CustomerName |
| C1 | Alice |
| C2 | Bob |

Order details Table

|  |  |  |
| --- | --- | --- |
| OrderID | ProductID | ProductName |
| 101 | P1 | Laptop |
| 101 | P2 | Phone |
| 102 | P3 | Tablet |

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

3NF is a level where a relation is in 2NF and there are no transitive dependency exists.

It eliminate transitive dependency.

**Example:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| OrderID | ProductID | ProductName | CustomerID | CustomerName | City |
| 101 | P1 | Laptop | C1 | Alice | New York |
| 101 | P2 | Phone | C1 | Alice | New York |
| 102 | P3 | Tablet | C2 | Bob | Los Angeles |

3NF Normalize table:

Orders Table:

|  |  |
| --- | --- |
| OrderID | CustomerID |
| 101 | C1 |
| 102 | C2 |

Customers Table:

|  |  |  |
| --- | --- | --- |
| CustomerID | CustomerName | City |
| C1 | Alice | New York |
| C2 | Bob | Los Angeles |

Products Table:

|  |  |
| --- | --- |
| ProductID | ProductName |
| P1 | Laptop |
| P2 | Phone |
| P3 | Tablet |

Order Details Table:

|  |  |
| --- | --- |
| OrderID | ProductID |
| 101 | P1 |
| 101 | P2 |
| 102 | P3 |

**BCNF (Boyce Codd’s Normal Form):**

BCNF is a stricter or stronger form of normalization than 3NF.

It is also called as 3.5NF.

For every functional dependency, the right-hand side (RHS) attribute depends on the super key of the table.

For every non-trivial functional dependency, the left-hand side (LHS) is a super key or candidate key.

**Example:**

|  |  |  |  |
| --- | --- | --- | --- |
| EmployeeId | Department | DepartmentHead | Salary |
| 101 | IT | Jhon | 50000 |
| 102 | IT | Jhon | 45000 |
| 103 | HR | Jane | 40000 |

BCNF Normalized Table:

Employee table:

|  |  |  |
| --- | --- | --- |
| EmployeeId | Department | Salary |
| 101 | IT | 50000 |
| 102 | IT | 45000 |
| 103 | HR | 40000 |

Department table:

|  |  |
| --- | --- |
| Department | DepartmentHead |
| IT | Jhon |
| IT | Jhon |
| HR | Jane |

**4NF (Fourth Normal Form):**

A relation will be in 4NF if it is in Boyce Codd’s Normal Form (BCNF) and has no multi-valued dependency.

Eliminate multi-values Dependency.

**Example:**

|  |  |  |
| --- | --- | --- |
| StudentID | Course | Hobby |
| 101 | Math | Swimming |
| 101 | Science | Reading |
| 102 | History | Gaming |
| 102 | Geography | Painting |

4NF Normalized table:

Student Table:

|  |
| --- |
| StudentID |
| 101 |
| 102 |

StudentCourse Table:

|  |  |
| --- | --- |
| StudentID | Course |
| 101 | Math |
| 101 | Science |
| 102 | History |
| 102 | Geography |

StudentHobby Table:

|  |  |
| --- | --- |
| StudentID | Hobby |
| 101 | Swimming |
| 101 | Reading |
| 102 | Gaming |
| 102 | Painting |

**5NF (Fifth Normal Form):**

5NF is also called as Project-Join Normal Form (PJNF) if it is in 4NF and does not contain any join dependencies.

Joining should be lossless.

Eliminate Join Dependency.

**Example:**

|  |  |  |
| --- | --- | --- |
| Product | Supplier | City |
| Laptop | Dell | Texas |
| Laptop | HP | California |
| Phone | Apple | California |
| Phone | Samsung | Korea |

5NF Normalize Form:

ProductSupplier Table:

|  |  |
| --- | --- |
| Product | Supplier |
| Laptop | Dell |
| Laptop | HP |
| Phone | Apple |
| Phone | Samsung |

SupplierCity Table:

|  |  |
| --- | --- |
| Supplier | City |
| Dell | Texas |
| HP | California |
| Apple | California |
| Samsung | Korea |