Normalization

Normalization is the process of organizing the data in the database.

Normalization is used to minimize the redundancy of data.

Normalization divides the larger table into the smaller table.

1NF

A relation will be in 1NF if it contains an atomic value in a cell of the table.

|  |  |  |
| --- | --- | --- |
| id | name | Dept. |
| 1 | Gautam | Developer, Business |
| 2 | Ravi | Testing |

|  |  |  |
| --- | --- | --- |
| id | name | Dept. |
| 1 | Gautam | Developer |
| 2 | Gautam | Business |
| 2 | Ravi | Testing |

2NF

In the 2NF, relational must be in 1NF. There must be a primary key in the table.

|  |  |  |
| --- | --- | --- |
| id | name | Dept. |
| 1 | Gautam | Developer |
| 2 | Gautam | Business |
| 2 | Ravi | Testing |

|  |  |
| --- | --- |
| id | name |
| 1 | Gautam |
| 2 | Ravi |

|  |  |
| --- | --- |
| Dept\_id | Dept\_name |
| 1 | Developer |
| 2 | Business |
| 3 | Testing |

3NF

A relation will be in 3NF if it is in 2NF and not contain any transitive partial dependency.

For each functional dependency X ⟶ A in R,

at least one of the following conditions are met:

X is superkey in R

A is a prime attribute in R

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| emp\_id | emp\_name | emp\_zip | emp\_state | emp\_city |
| 1001 | John | 282005 | UP | Agra |
| 1002 | Ajeet | 222008 | TN | Chennai |
| 1006 | Lora | 282007 | TN | Chennai |
| 1101 | Lilly | 292008 | UK | Pauri |

Here, emp\_state, emp\_city dependent on emp\_zip. And, emp\_zip is dependent on emp\_id that makes non-prime attributes (emp\_state, emp\_city) transitively dependent on super key (emp\_id). This violates the rule of 3NF.

|  |  |  |
| --- | --- | --- |
| emp\_id | emp\_name | emp\_zip |
| 1001 | John | 282005 |
| 1002 | Ajeet | 222008 |
| 1006 | Lora | 282007 |
| 1101 | Lilly | 292008 |

|  |  |  |
| --- | --- | --- |
| emp\_zip | emp\_state | emp\_city |
| 282005 | UP | Agra |
| 222008 | TN | Chennai |
| 282007 | TN | Chennai |
| 292008 | UK | Pauri |

BCNF

A relation R is in BCNF if it is in 3NF and for each functional dependency X ⟶ A in R, X is a key

or superkey in R. In other words, the only difference between 3NF and BCNF is that in BCNF it is

not present the second condition of the 3NF. This makes BCNF stricter than 3NF as any relation

that is in BCNF will be in 3NF but not necessarily every relation that is in 3NF will be in BCNF

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| emp\_id | emp\_nationality | emp\_nationality | emp\_dept | dept\_type | dept\_no\_of\_emp |
| 1001 | Austrian | Austrian | Production | D001 | 200 |
| 1001 | Austrian | Austrian | stores | D001 | 250 |
| 1002 | American | American | design | D134 | 100 |
| 1002 | American | American | Purchasing | D134 | 600 |

|  |  |
| --- | --- |
| emp\_id | emp\_nationality |
| 1001 | Austrian |
| 1002 | American |

|  |  |  |
| --- | --- | --- |
| emp\_dept | dept\_type | dept\_no\_of\_emp |
| Production | D001 | 200 |
| stores | D001 | 250 |
| design | D134 | 100 |
| Purchasing | D134 | 600 |

|  |  |
| --- | --- |
| emp\_id | emp\_dept |
| 1001 | Production |
| 1001 | stores |
| 1002 | design |
| 1002 | Purchasing |

4NF

Tables cannot have multi-valued dependencies on a Primary Key.