Normalization

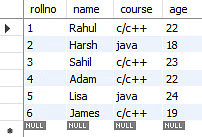
*You often work with enormous amounts of data stored in different tables that are present inside multiple databases. It often becomes strenuous to extract the data if it is not organized correctly. Using Normalization, you can solve the problem of data redundancy and organize the data using different forms.*

* ***What is Normalization in SQL?*** *- Normalization is the process to eliminate data redundancy and enhance data integrity in the table. Normalization also helps to organize the data in the database. It is a multi-step process that sets the data into tabular form and removes the duplicated data from the relational tables.*

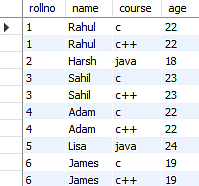
*Normalization organizes the columns and tables of a database to ensure that database integrity constraints properly execute their dependencies. It is a systematic technique of decomposing tables to eliminate data redundancy (repetition) and undesirable characteristics like Insertion, Update, and Deletion anomalies.*

* *First Normal Form - A table is referred to as being in its First Normal Form if atomicity of the table is 1. Here, atomicity states that a single cell cannot hold multiple values. It must hold only a single-valued attribute. The First normal form disallows the multi-valued attribute, composite attribute, and their combinations.*

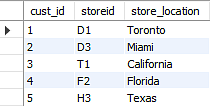
*Ex - Before*

**

*After –*

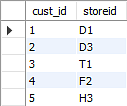


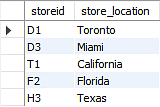
* *Second Normal Form - The first condition for the table to be in Second Normal Form is that the table must be in First Normal Form. The table should not possess partial dependency. The partial dependency here means the proper subset of the candidate key should give a non-prime attribute.*

*Ex. –* 

*The Location table possesses a composite primary key cust\_id, storeid. The non- key attribute is store\_location. In this case, store\_location only depends on storeid, which is a part of the primary key. Hence, this table does not fulfill the second normal form.*

*To bring the table to Second Normal Form, you need to split the table into two parts. This will give you the below tables:*



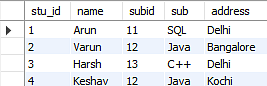


*As you have removed the partial functional dependency from the location table, the column store\_location entirely depends on the primary key of that table, storeid.*

* *Third Normal Form –*

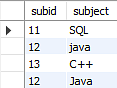
1. *The first condition for the table to be in Third Normal Form is that the table should be in the Second Normal Form.*
2. *The second condition is that there should be no transitive dependency for non-prime attributes, which indicates that non-prime attributes (which are not a part of the candidate key) should not depend on other non-prime attributes in a table. Therefore, a transitive dependency is a functional dependency in which A → C (A determines C) indirectly, because of A → B and B → C (where it is not the case that B → A).*
3. *The third Normal Form ensures the reduction of data duplication. It is also used to achieve data integrity.*

*Ex –*



*After –*

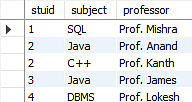




* *Boyce Codd Normal Form - Boyce Codd Normal Form is also known as 3.5 NF.*

*The first condition for the table to be in Boyce Codd Normal Form is that the table should be in the third normal form. Secondly, every Right-Hand Side (RHS) attribute of the functional dependencies should depend on the super key of that particular table.*

*Ex – Before*



*After –*

