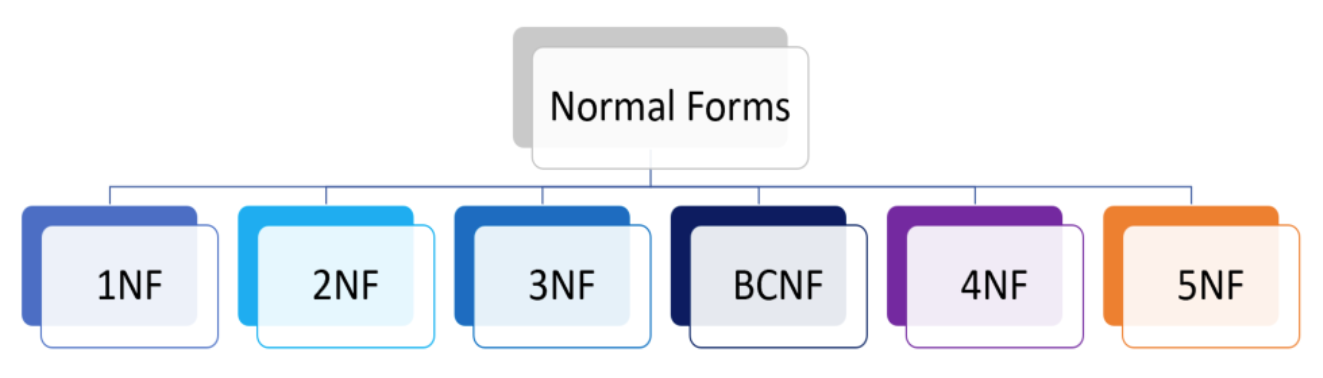
Introduction to Normalization :-

Database normalization is a crucial concept in the world of database management. It is a process that optimizes database structure by reducing data redundancy and improving data integrity. Normalization is a set of rules and guidelines that help organize data efficiently and prevent common data anomalies like update anomalies, insertion anomalies, and deletion anomalies.In this document, we will delve into the fundamentals of database normalization, the various normal forms, and provide practical examples to illustrate each level of normalization.



Why Normalize a Database?

Before we dive into the details of database normalization, it’s essential to understand why it’s necessary. Normalization offers several advantages:Data Integrity:

Normalization helps maintain data accuracy and consistency by reducing redundancy. When data is stored in a non-repetitive manner, it is less prone to errors.

Efficient Storage:

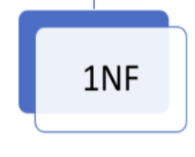
Normalized databases tend to occupy less storage space as duplicate data is minimized. This reduces the overall cost of storage.

Query Optimization:

Queries become more efficient in normalized databases because they need to access smaller, well-structured tables instead of large, denormalized ones.

Flexibility:

Normalized databases are more flexible when it comes to accommodating changes in data requirements or business rules.



The First Normal Form (1NF)

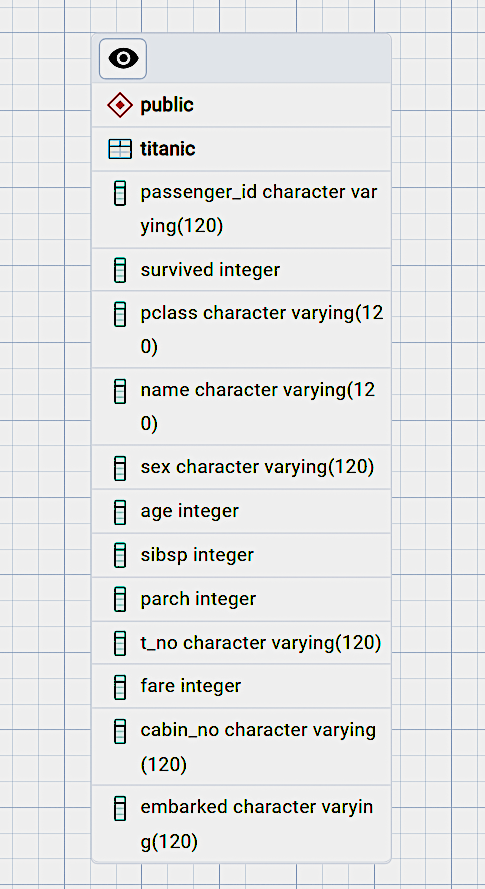
First normal form is a property of a relation in a relational database. A relation is in first normal form if and only if no attribute domain has relations as elements. Or more informally, that no table column can have tables as values.

Rules For 1NF:

* Every column/attribute need to have a single value.
* Each row should be unique: either though a single or multiple column.
* Not mandatory have a primary key.

Note :

In ‘Titanic’ Database All the column already have a single value; so, there is no need to do a query for 1NF on this data.





The second Normal Form (2NF)

The second Normal Form applies to relations with composite keys, that is, relations with a primary key composed of two or more attributes. A relation with a single-attribute primary key is automatically in at least 2NF. A relation that is not in 2NF may suffer from the update anomalies. To be in the second normal form, a relation must be in the first normal form and the relation must not contain any partial dependency. A relation is in 2NF if it has No Partial Dependency, i.e., no non-prime attribute (attributes that are not part of any candidate key) is dependent on any proper subset of any candidate key of the table.

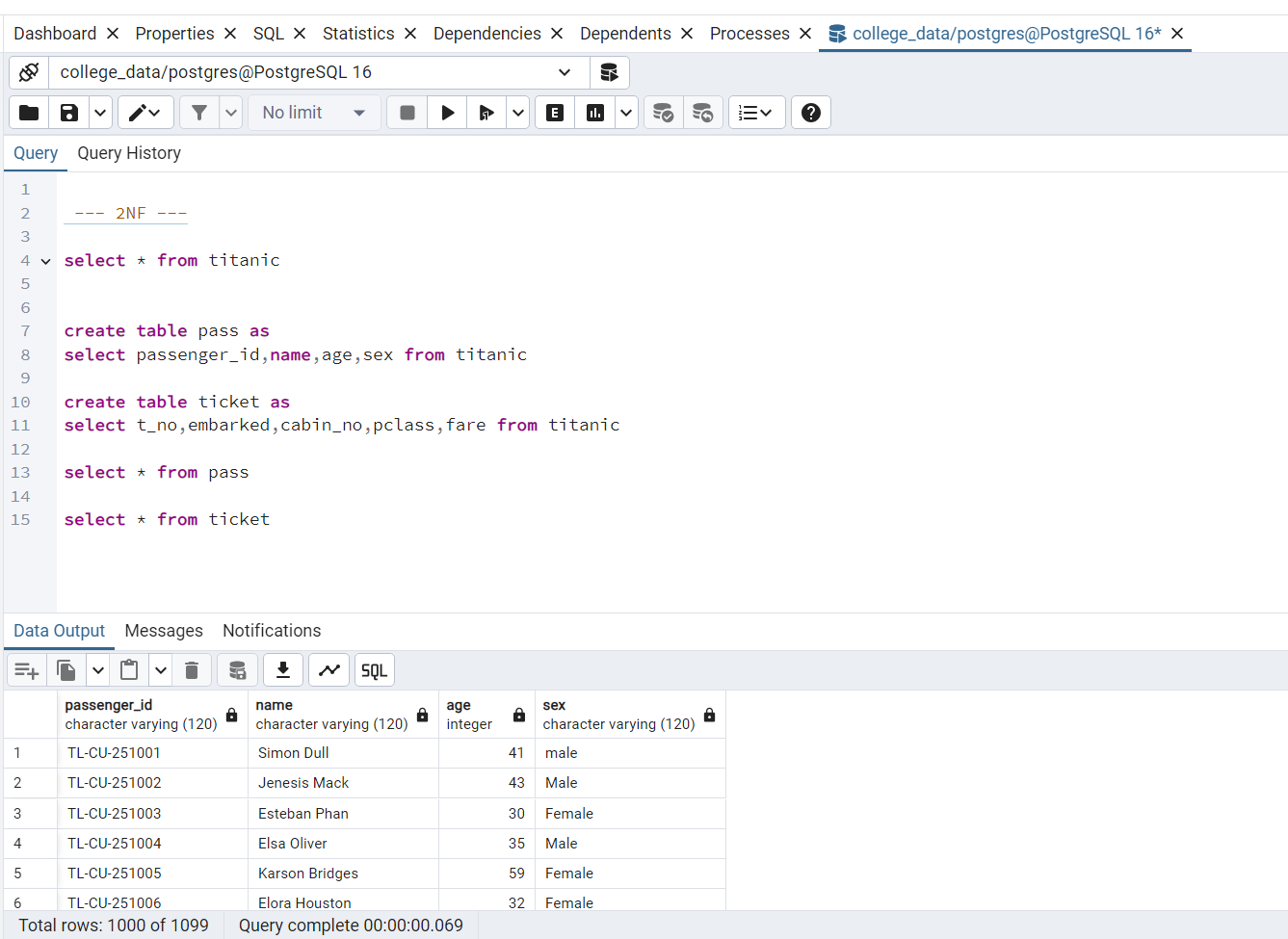
Rules For 2NF:

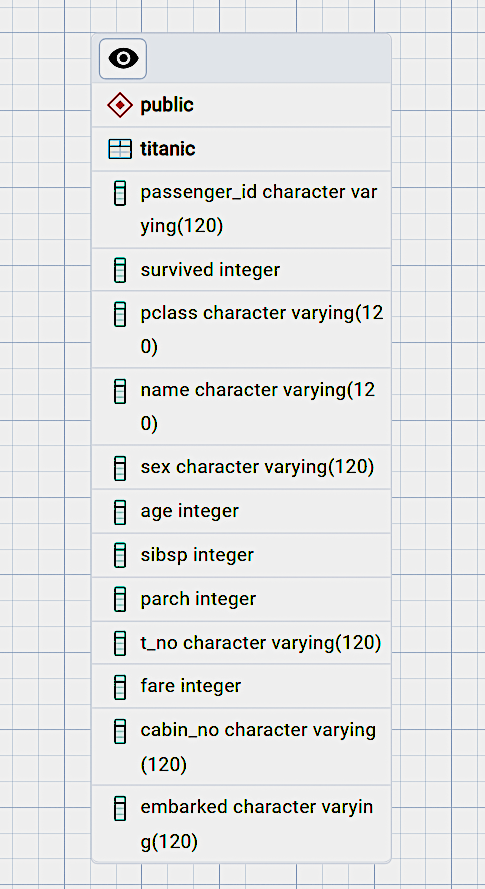
* Must be in 1nf.
* All non - key attributes must be fully dependent on composite key;

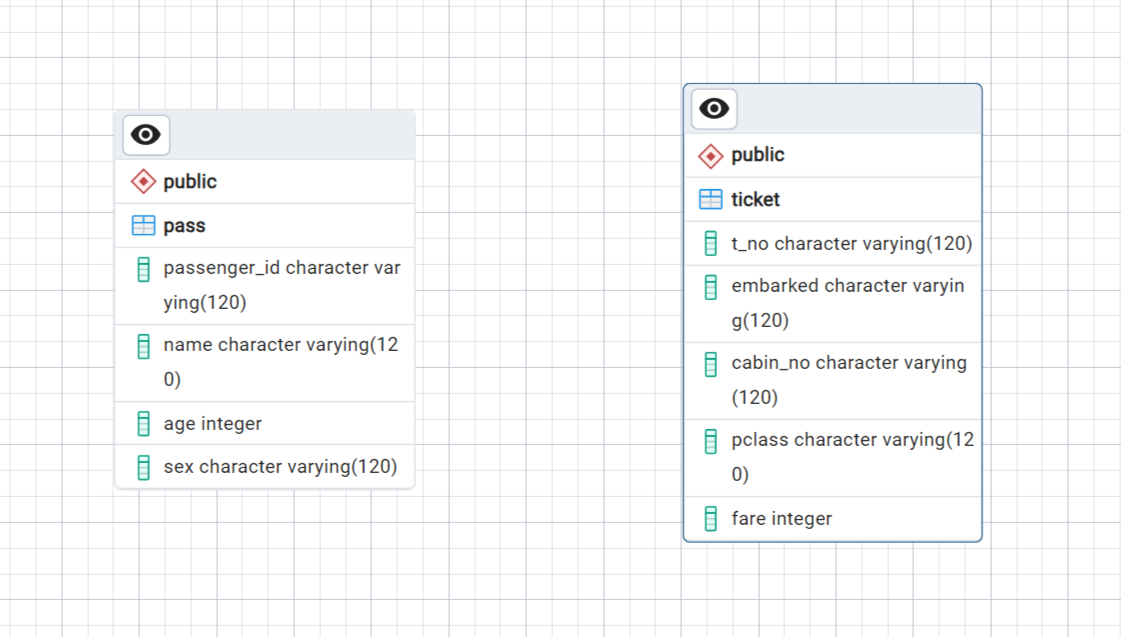
i.e. If a non-key column is partially dependent on composite key then split them into separate tables.

* Every table should have primary key and relationship between the table should be formed using foreign key.

Query for 2NF:









The Third Normal Form (3NF)

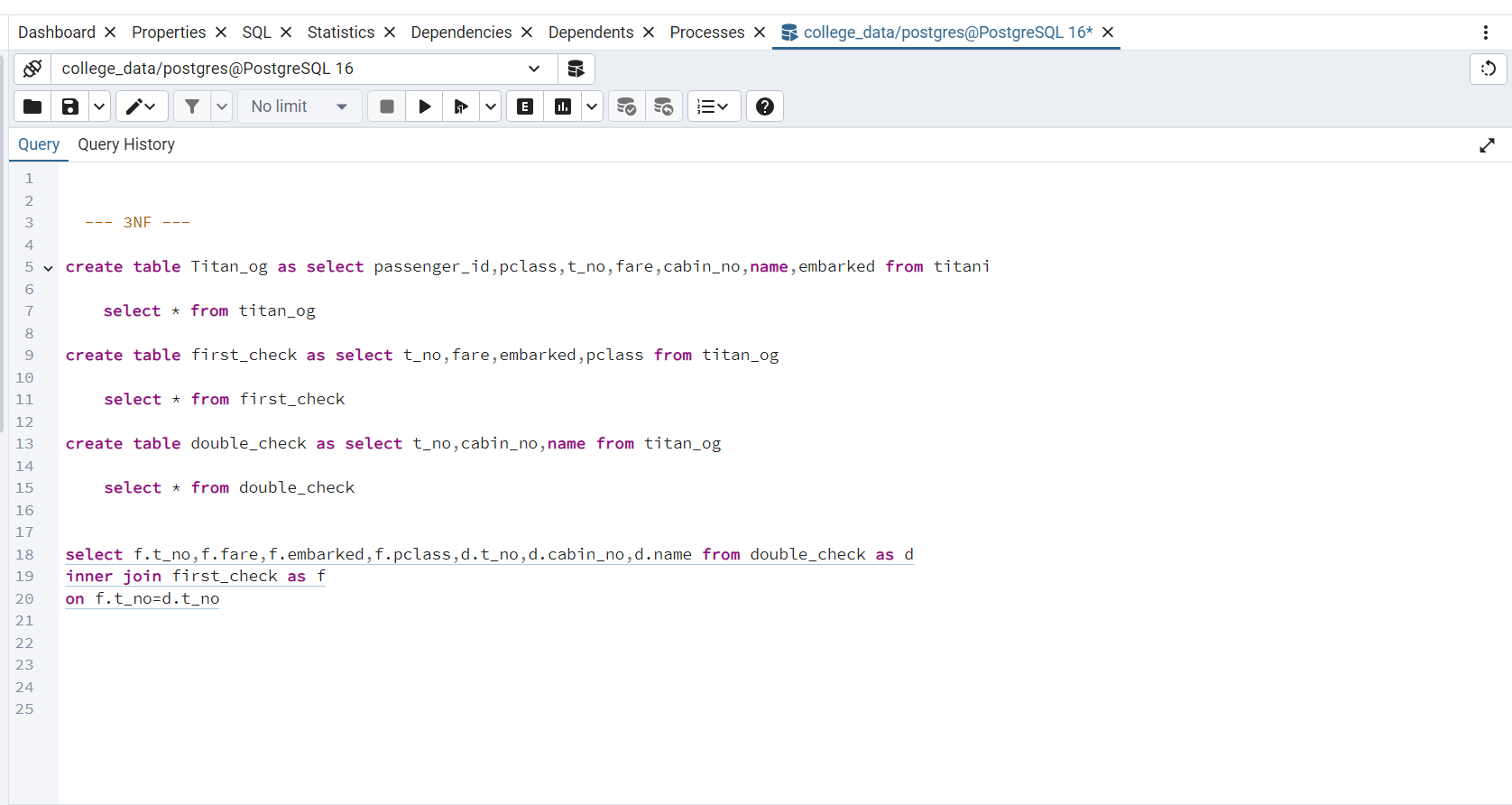
A relation is in the third normal form, if there is no transitive dependency for non-prime attributes as well as it is in the second normal form. A relation is in 3NF if at least one of the following conditions holds in every non-trivial function dependency X -> Y.

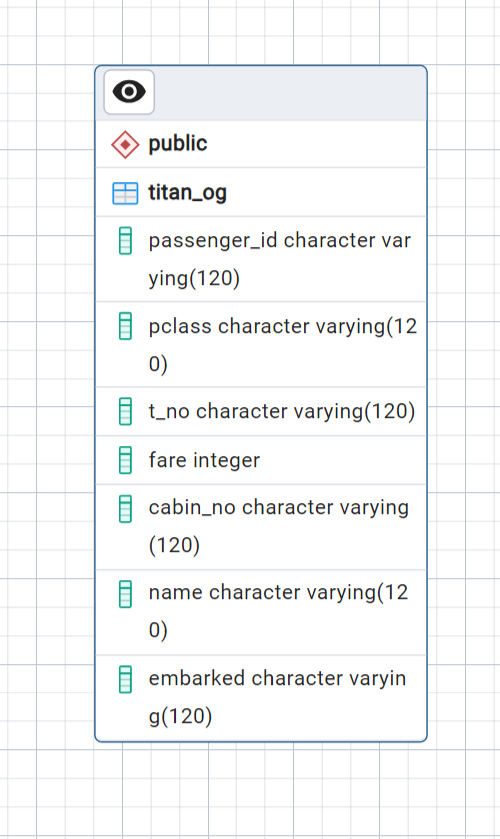
* X is a super key.
* Y is a prime attribute (each element of Y is part of some candidate key).

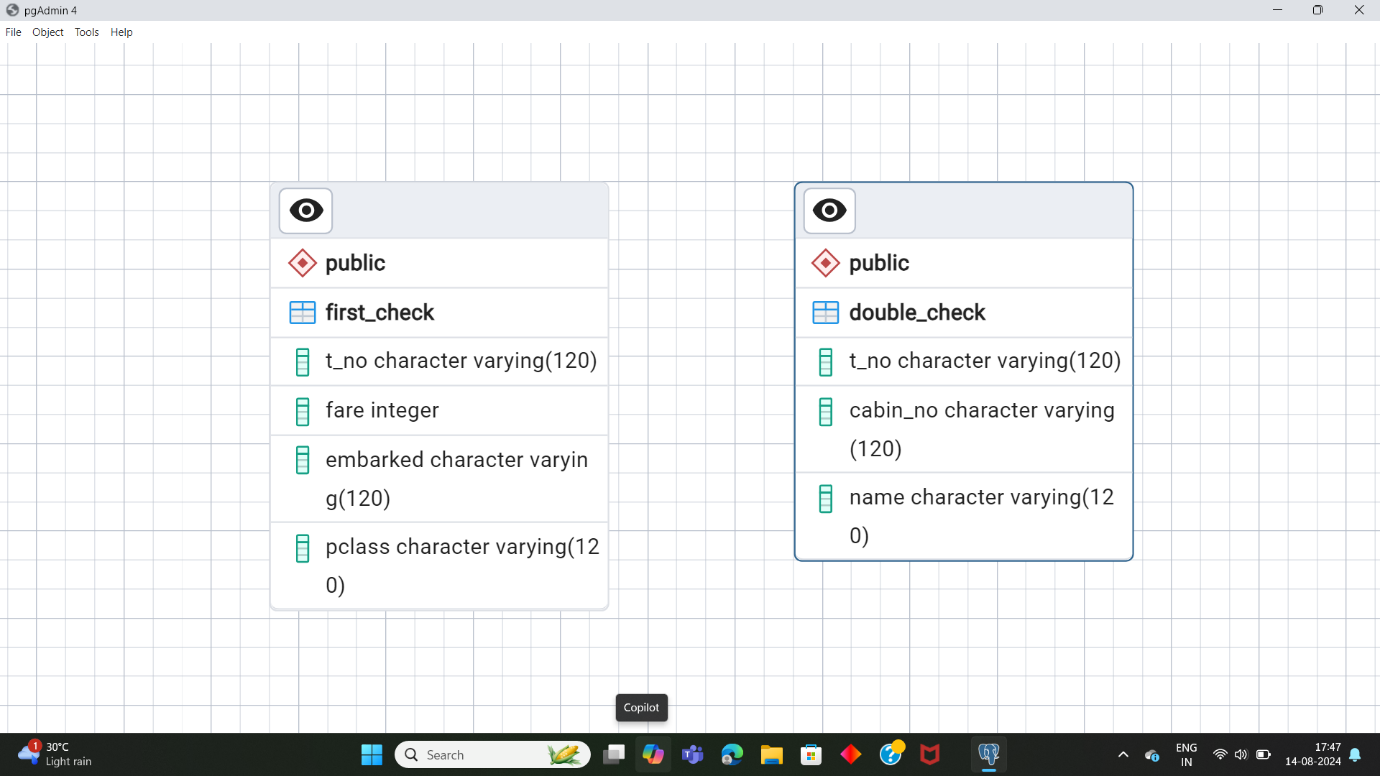
Rules For 3NF:

* Must be in 2NF.
* Avoid transitive dependencies.

Query for 3NF:







Conclusion:

Database normalization is a critical process in database design, aimed at optimizing data storage, improving data integrity, and reducing data anomalies. By organizing data into normalized tables, It enhances the efficiency and maintainability of your database system.When designing a database, it’s essential to strike a balance between normalization and practicality. In many cases, achieving 3NF is sufficient to ensure data integrity while maintaining good query performance.Normalization is basically depends on the practicality of the data or how we normalize the database to understand the data easily.