Normalization in DBMS is the process of organizing data in a database efficiently. This involves creating tables and establishing relationships between them to minimize redundancy and dependency. The primary goals of normalization are to:Eliminate data redundancy: Redundant data wastes storage space and can lead to inconsistencies and anomalies.Reduce data dependency: Minimizing dependency ensures that changes in one part of the database do not require changes to be made in multiple places.Improve data integrity: By reducing redundancy and dependency, normalization helps maintain data integrity by ensuring that each piece of information is stored in only one place.Normalization is typically achieved by breaking down large tables into smaller, related tables and organizing them based on certain rules or normal forms. The process involves identifying functional dependencies and organizing data to conform to these normal forms, such as First Normal Form (1NF), Second Normal Form (2NF), Third Normal Form (3NF), and so on.Each normal form has its own set of rules, and achieving higher normal forms involves stricter rules and greater normalization of the data. Normalization helps optimize database design, making it easier to maintain, update, and query the data.**TYPES OF NORMALIZATION:-**

Normalization is a systematic process used in database design to minimize data redundancy and dependency by organizing tables into normal forms. There are several normal forms, each with its own set of rules and requirements. The most commonly used normal forms are:First Normal Form (1NF):Eliminates repeating groups within a table.Ensures that each column contains atomic values.Every table cell should contain a single value, and each column should have a unique name.Second Normal Form (2NF):Meets all the requirements of 1NF.Removes partial dependencies, meaning that non-key attributes are fully functionally dependent on the entire primary key.Involves breaking down composite primary keys and moving attributes that are dependent on only part of the primary key to a separate table.Third Normal Form (3NF):Meets all the requirements of 2NF.Eliminates transitive dependencies, ensuring that non-key attributes are not dependent on other non-key attributes.Involves moving non-key attributes that are dependent on other non-key attributes to separate tables.Boyce-Codd Normal Form (BCNF):A stronger version of 3NF.Every determinant must be a candidate key (no non-trivial dependencies on candidate keys).It removes anomalies related to non-trivial functional dependencies.Fourth Normal Form (4NF):Deals with multi-valued dependencies.Removes the possibility of a table having two or more independent multi-valued facts about an entity.Achieved by creating separate tables for multi-valued attributes.Fifth Normal Form (5NF):Deals with cases where there are overlapping composite keys.Addresses redundancy that arises from overlapping composite keys by creating separate tables for each overlapping set of keys.Each normal form builds upon the previous one, and achieving a higher normal form implies that the database is more normalized and has less redundancy and dependency. However, it's essential to balance normalization with performance considerations and the specific requirements of the application**.**

**ADMIN:-**

To normalize the given table into the first normal form (1NF), we need to ensure that each attribute in the table contains atomic values, meaning there are no repeating groups or arrays within any cell.

The given table already satisfies the requirements of the first normal form. Each attribute contains atomic values, and there are no repeating groups or arrays within any cell. Therefore, no further normalization is required for this table to be in the first normal form.

To bring the provided table into the second normal form (2NF), we first need to ensure it's in the first normal form (1NF), which means each attribute contains atomic values and there are no repeating groups or arrays within any cell.

Given that the table already satisfies the requirements of the first normal form, we can move on to the second normal form.

For a table to be in 2NF, it must meet the following conditions:

It is in 1NF.

All non-prime attributes (attributes not part of any candidate key) are fully functionally dependent on the entire candidate key.

In the given table, the only candidate key is aid. The non-prime attribute is password. Since password is functionally dependent on the entire candidate key (aid), the table is in 2NF.

Therefore, no further normalization is required for this table to be in the second normal form.

To bring the provided table into the third normal form (3NF), we first need to ensure it's in the second normal form (2NF). As we confirmed earlier that the table is in 2NF, we can proceed to achieve 3NF.

For a table to be in 3NF, it must meet the following conditions:

It is in 2NF.

All non-prime attributes (attributes not part of any candidate key) are non-transitively dependent on every candidate key.

In the given table, the only candidate key is aid, and the non-prime attribute is password. Since password is directly dependent on the candidate key (aid), and there are no transitive dependencies, the table already satisfies the conditions for 3NF.

Therefore, the table is already in the third normal form (3NF). No further normalization is required.

To achieve Boyce-Codd Normal Form (BCNF), we need to ensure that for every non-trivial functional dependency

→

X→Y,

X must be a superkey. In simpler terms, every determinant (left side of an arrow) should be a candidate key.

In the provided table, the only candidate key is aid. Let's examine if there are any non-trivial functional dependencies:

aid → password: Here, aid is the candidate key, and password is functionally dependent on it.

Since every determinant is a candidate key, the table already satisfies BCNF.

To achieve Fourth Normal Form (4NF), we need to ensure that there are no multi-valued dependencies within the table.

In the provided table, there are no multi-valued dependencies since each attribute is functionally dependent on the candidate key, and there are no sets of attributes that imply multiple possible values for other attributes.

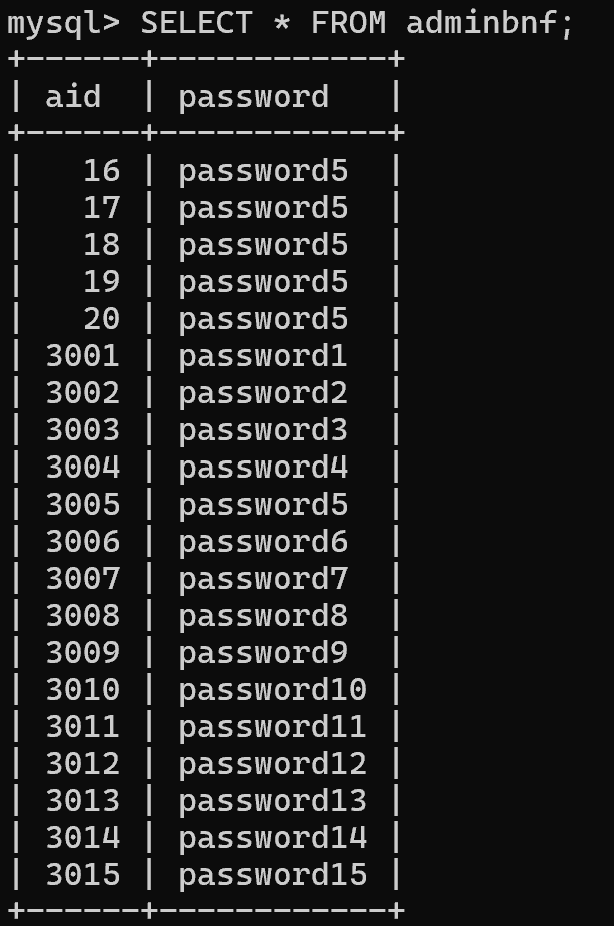
Therefore, the table is already in the Fourth Normal Form (4NF). No further decomposition or normalization is required.

To achieve Fifth Normal Form (5NF), we need to ensure that the table is in Boyce-Codd Normal Form (BCNF), and there are no non-trivial join dependencies.

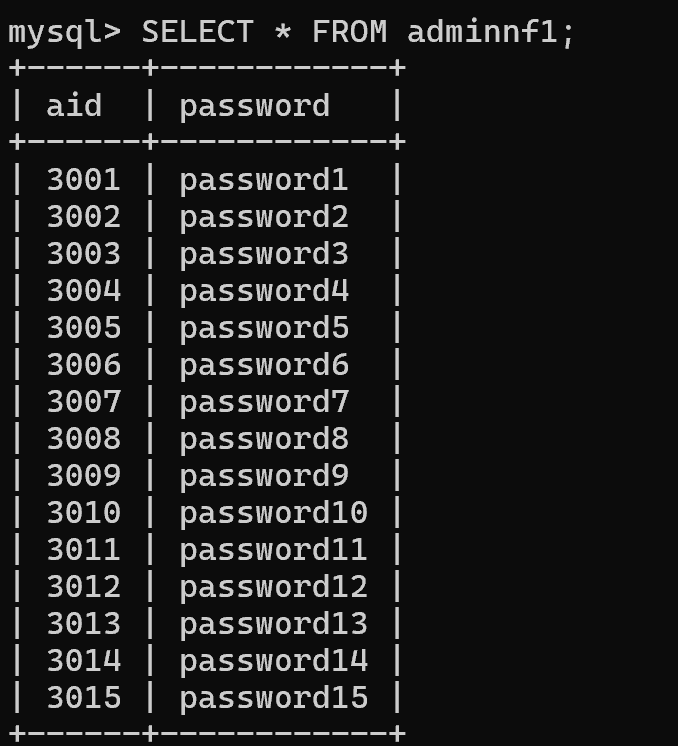
The provided table already satisfies BCNF, as every determinant is a candidate key, and there are no non-trivial join dependencies, as each attribute is functionally dependent on the candidate key.

Therefore, the table is already in the Fifth Normal Form (5NF). No further decomposition or normalization is required.

BEFORE:-



AFTER:-



**CUSTOMER:-**

To convert the provided table into the First Normal Form (1NF), we need to ensure that each attribute contains atomic values, meaning no multi-valued attributes or repeating groups. Looking at the provided table, it is not satisfing the requirements of 1NF, as each cell contains a multiple value

Convet it to such that each cell contains a single value.

To convert the provided table into the Second Normal Form (2NF), we need to ensure that it meets the following conditions:

It is already in 1NF.

All non-prime attributes are fully functionally dependent on the entire candidate key.

Looking at the table, the candidate key seems to be the cid (customer ID) attribute. All other attributes (cname, password, emailid, phone\_number) are functionally dependent on cid. Therefore, the table is already in the Second Normal Form (2NF), as it meets both conditions.

To convert the provided table into the Third Normal Form (3NF), we need to ensure that it meets the following conditions:

It is already in 2NF.

There are no transitive dependencies.

Looking at the table, we observe the following dependencies:

cname → emailid

cid → cname, password, emailid, phone\_number

Here, cname determines emailid, and cid determines all other attributes. There are no transitive dependencies in this table.

Therefore, the table is already in the Third Normal Form (3NF), as it meets both conditions. Each non-prime attribute is determined by the candidate key (cid).

To achieve Boyce-Codd Normal Form (BCNF), we need to ensure that for every non-trivial functional dependency

→

X→Y in the table,

X must be a superkey.

In the provided table, the candidate key is

{

}

{cid}, and all attributes are functionally dependent on the candidate key. Therefore, the table is already in BCNF, as every non-trivial functional dependency satisfies the condition that the determinant (

X) is a superkey.

Thus, no further decomposition is required to achieve BCNF.

To achieve the Fourth Normal Form (4NF), we need to identify and eliminate any multi-valued dependencies (MVDs) in the table.

In the given table, there are no multi-valued dependencies present, as each attribute depends only on the candidate key (cid), and there are no attributes that exhibit non-trivial functional dependencies with other non-key attributes.

Therefore, the table is already in the Fourth Normal Form (4NF), and no further decomposition is required.

The Fifth Normal Form (5NF), also known as Project-Join Normal Form (PJNF), deals with eliminating redundancy that arises from join dependencies.

Given the table:

cname (Customer Name)

cid (Customer ID)

password

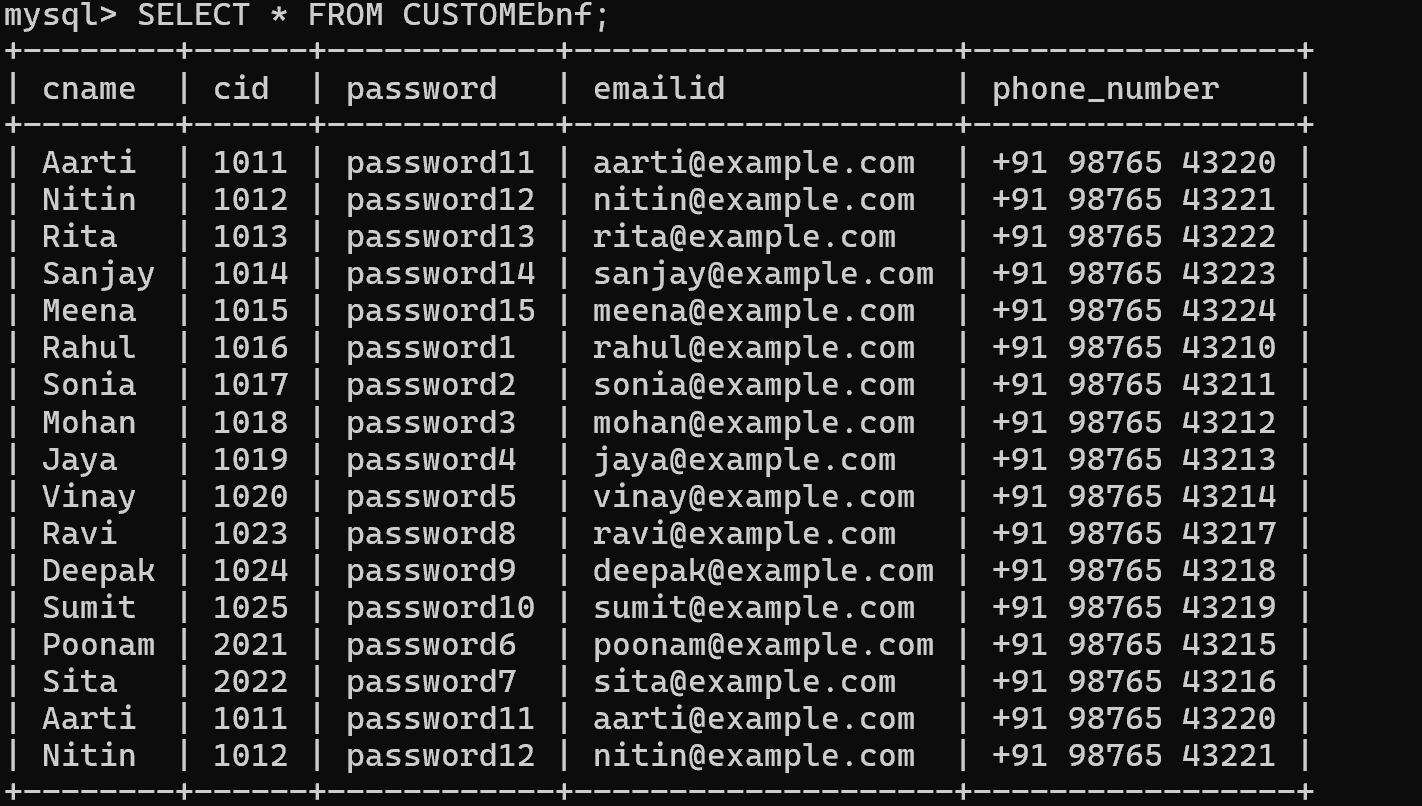
emailid

phone\_numbe

Since there are no join dependencies or further decomposition possibilities that would eliminate redundancy beyond the Fourth Normal Form (4NF), we can conclude that the table is already in the Fifth Normal Form (5NF).

Therefore, no further normalization is needed.

**BEFORE:-**

****

**AFTER:-**

****

**SEAT:-**

The table you provided already satisfies the requirements of the first normal form (1NF). . Looking at the provided table, it is not satisfing the requirements of 1NF, as each cell contains a multiple value

Convet it to such that each cell contains a single value.

To convert the given table to the second normal form (2NF), we need to ensure that it is already in 1NF and then check if any non-prime attributes are dependent on only part of the primary key.

In the given table, there is no composite primary key, and all attributes are fully dependent on the primary key (sid). Therefore, the table is already in the second normal form.

To summarize, the table is in both the first normal form (1NF) and the second normal form (2NF).

To bring the table to the third normal form (3NF), we need to ensure that it is already in the second normal form (2NF) and then remove any transitive dependencies.

Looking at the table:

The primary key is sid.

sname seems to depend only on sid.

number\_of\_seats also seems to depend only on sid.

Since both sname and number\_of\_seats depend directly on the primary key sid, there are no transitive dependencies. Therefore, the table is already in the third normal form (3NF).

To achieve Boyce-Codd Normal Form (BCNF), we need to ensure that for every non-trivial functional dependency X -> Y in the table, X is a superkey. In other words, all determinants must be candidate keys.

Looking at the table:

The primary key is sid.

There are no non-trivial functional dependencies where the determinant (X) is not a superkey.

Each attribute (sid, sname, number\_of\_seats) is functionally determined only by the primary key (sid), and there are no non-trivial dependencies between other attributes.

Since all attributes are functionally determined by the primary key and there are no non-trivial dependencies, the table is already in BCNF.

To determine if the table is in Fourth Normal Form (4NF), we need to check for multivalued dependencies (MVDs). An MVD exists when there is a dependency between sets of attributes regardless of the values in the tuples.

In the given table:

There are no evident multivalued dependencies as each attribute seems to depend only on the sid, the primary key.

The table appears to satisfy the requirements of BCNF, and there are no non-trivial MVDs.

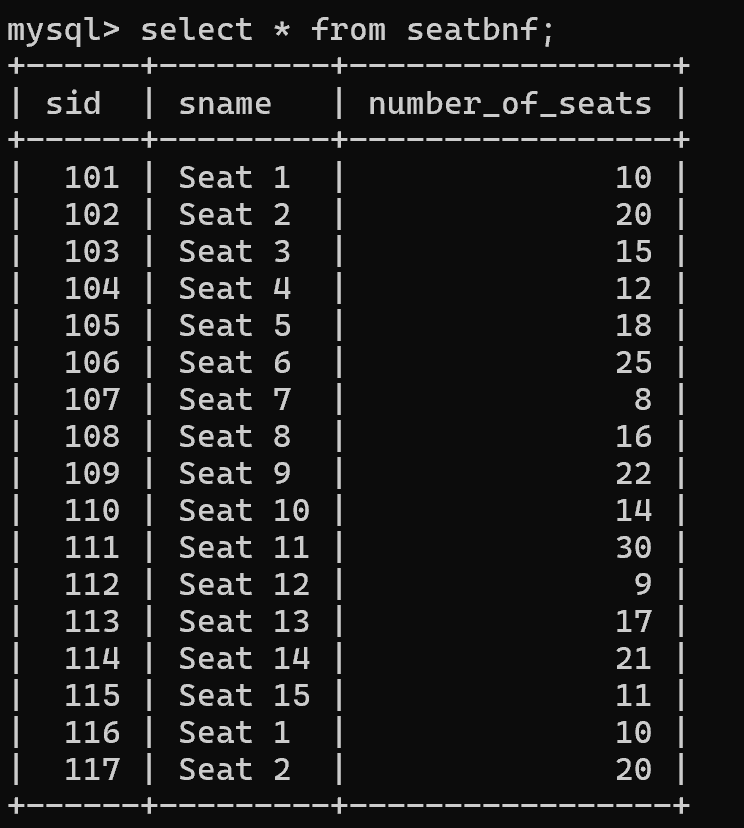
Therefore, the table is already in 4NF.

To achieve Fifth Normal Form (5NF), we need to ensure that there are no non-trivial join dependencies and that the table is in Fourth Normal Form (4NF).

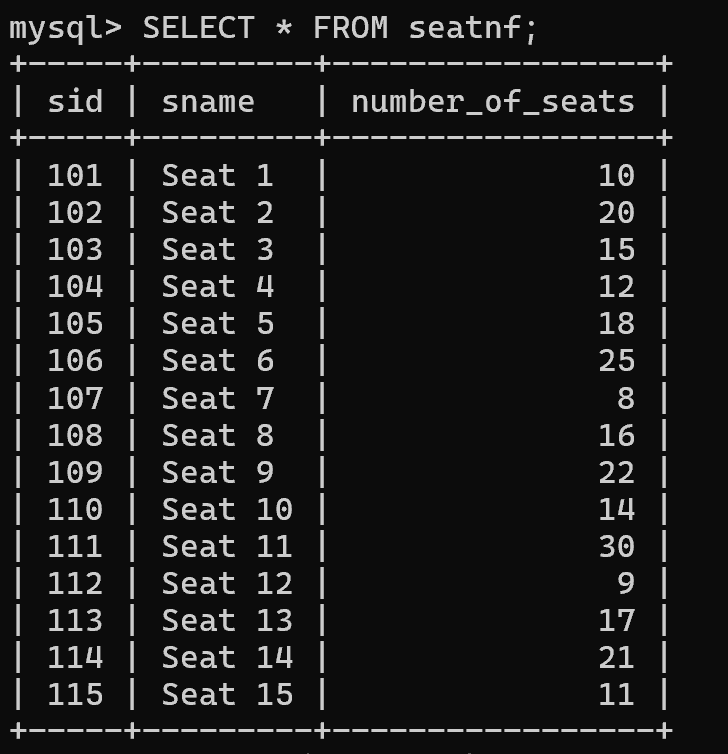
Given that the table is already in 4NF, we need to check for join dependencies. Since there are no evident join dependencies in the table, it satisfies the requirements for 5NF.

Therefore, the table is already in 5NF.

**BEFORE:-**

****

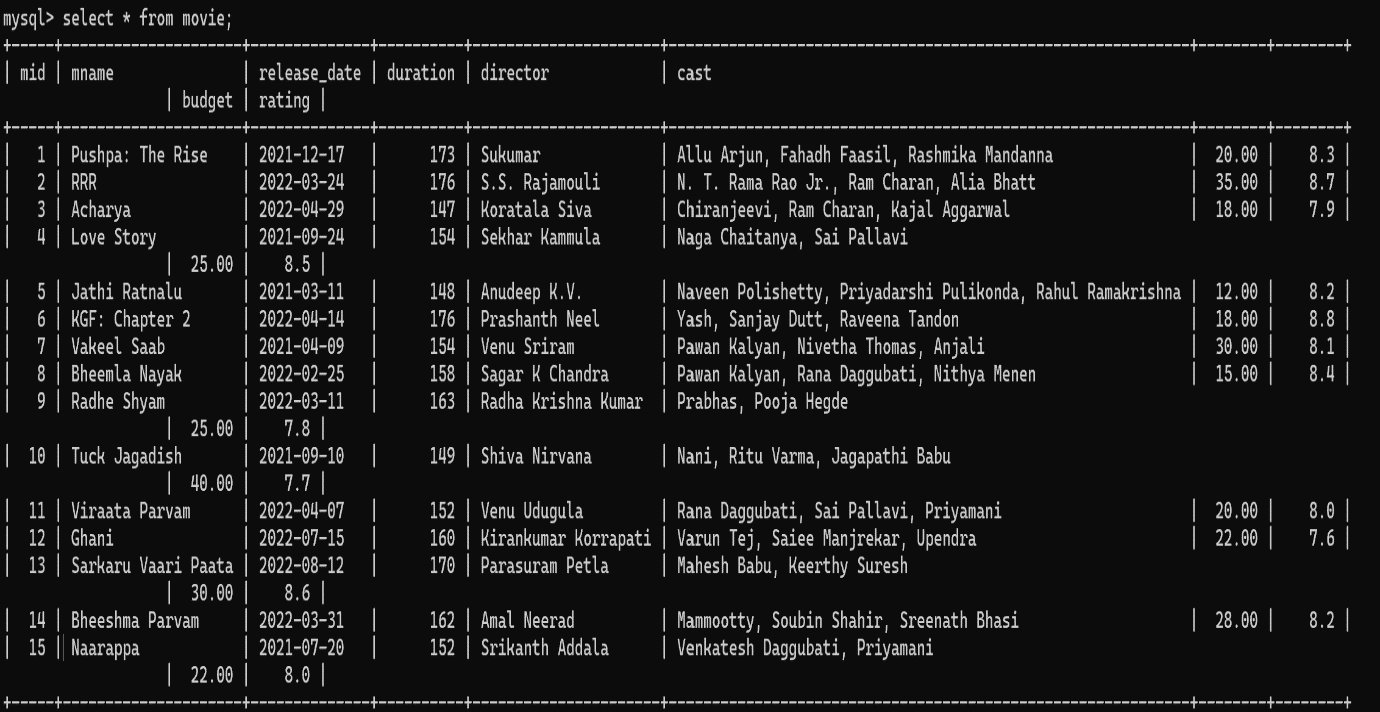
**AFTER:-**

****

**Movie:-**

First Normal Form (1NF):The table appears to already be in 1NF because each cell contains a single value, and there are no repeating groups within the table.Second Normal Form (2NF):To be in 2NF, the table should already be in 1NF and all non-key attributes should be fully functionally dependent on the entire primary key.In this case, mid seems to be a primary key. The director and cast columns might potentially have multiple values, but they seem to be attributes of a movie rather than standalone entities. So, the table is likely already in 2NF.Third Normal Form (3NF):For 3NF, the table should already be in 2NF, and no transitive dependencies should exist.In this table, there don't seem to be any transitive dependencies. All non-key attributes are directly dependent on the primary key (mid), so the table appears to be in 3NF.Boyce-Codd Normal Form (BCNF):BCNF is a higher level of normalization that ensures that every determinant is a candidate key.Since mid is the primary key, and each non-key attribute is dependent on the primary key, the table appears to satisfy BCNF as well.Fourth Normal Form (4NF) and Fifth Normal Form (5NF):These higher normal forms deal with more complex dependencies like multi-valued dependencies and overlapping composite keys. Typically, achieving these forms involves breaking down the table further into smaller, more specialized tables to remove such dependencies.In the provided table, there doesn't seem to be any indication of multi-valued dependencies or overlapping composite keys, so 4NF and 5NF are not relevant in this context.In summary, the movie table appears to be in at least 3NF and possibly BCNF, given the provided schema and data.

OUTPUT:-

****

**THEATRE:-**

Normal Form 1 (NF1) ensures that each cell in the table holds exactly one value. Looking at the provided table, it appears to already satisfy NF1. Each row contains atomic values in each column, without any repeating groups or composite values. Therefore, the table is already in Normal Form 1.

o achieve Normal Form 2 (NF2), we need to ensure that the table is in 1NF and that it does not contain any partial dependencies. In other words, all non-prime attributes (attributes not part of any candidate key) must be fully functionally dependent on all candidate keys.We see that tid is a candidate key (unique identifier) for this table. Both tname and location are fully dependent on tid, which means there are no partial dependencies.

Therefore, the table is already in Normal Form 2.

To achieve Normal Form 3 (NF3), we need to ensure that the table is in 2NF and that it does not contain any transitive dependencies. In other words, all non-prime attributes (attributes not part of any candidate key) must be fully functionally dependent on the candidate keys, and there should be no dependencies between non-prime attributes.

We observe that tid is the candidate key for this table. There are no attributes that are transitively dependent on the candidate key because both tname and location are directly dependent on tid.

Therefore, the table is already in Normal Form 3.

To achieve Boyce-Codd Normal Form (BCNF), we need to ensure that for every non-trivial functional dependency

→

X→Y in the table,

X must be a superkey. A superkey is a set of attributes that uniquely determines each tuple in the table.

In the provided table:

tid

→

tname

tid

→

location

tid

tid

​

→tname

→location

​

Both functional dependencies have the same determinant, tid, which is also the candidate key. Therefore, the table is already in BCNF because all functional dependencies satisfy the requirement that the determinant is a superkey.

To achieve Fourth Normal Form (4NF), we need to ensure that the table is already in BCNF and there are no multi-valued dependencies (MVDs) present.

Looking at the table:

There are no multi-valued dependencies present.

The table is already in BCNF.

Since the table meets the requirements of BCNF and there are no MVDs present, it automatically satisfies the conditions of 4NF. Therefore, the table is already in 4NF.

To reach Fifth Normal Form (5NF), we must ensure that the table is already in Fourth Normal Form (4NF) and that no join dependencies exist beyond the primary key dependencies.

Since the table is already in 4NF, we need to check for any join dependencies beyond the primary key dependencies.

In the table:

The primary key is (tid).

There are no join dependencies beyond the primary key dependencies.

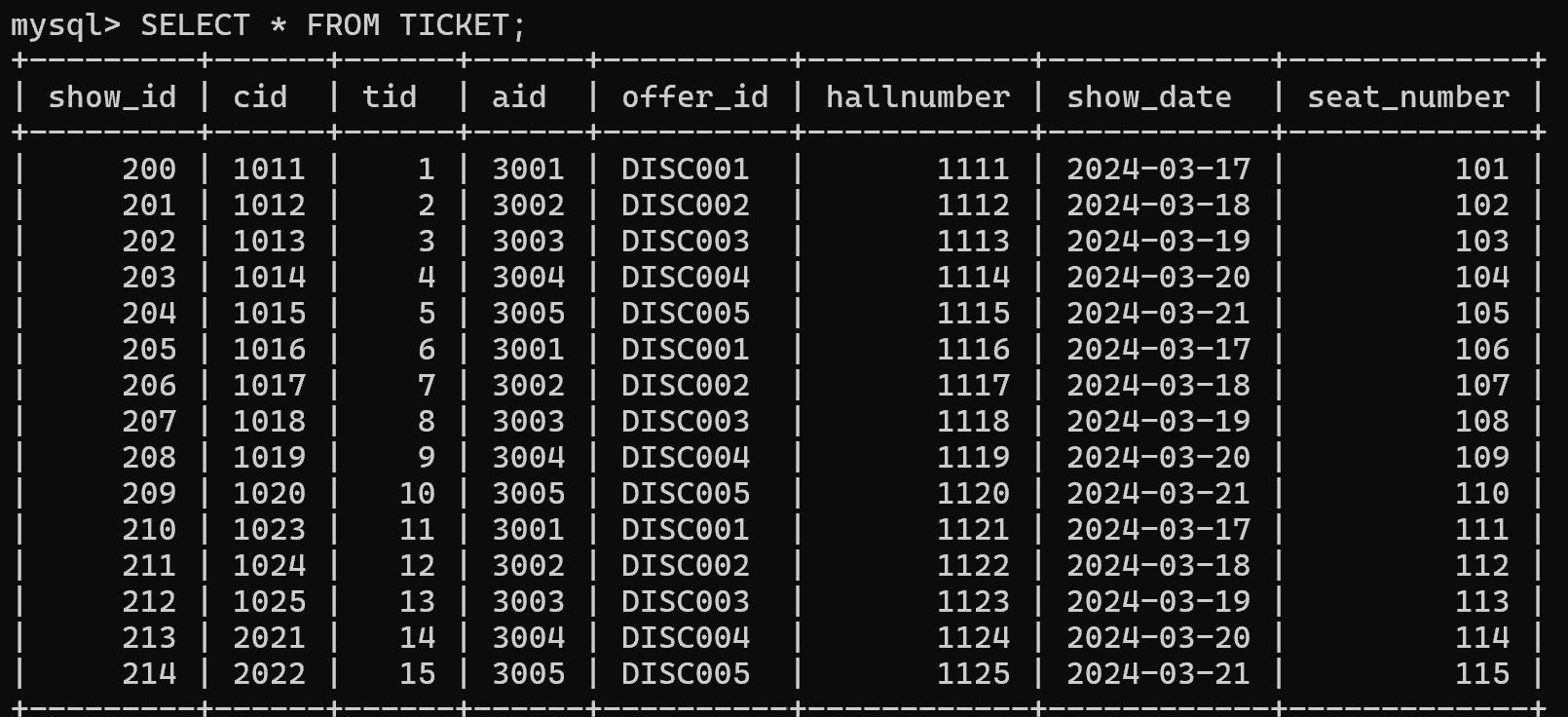
Therefore, the table is already in 5NF as it meets the requirements of being in 4NF and having no join dependencies beyond the primary key dependencies.

****

**TICKET:-**

First Normal Form (1NF):To be in 1NF, each cell of the table must hold a single value, and there should be no repeating groups or arrays.The provided table seems to meet the requirements of 1NF. Each cell appears to contain a single value, and there are no repeating groups.Second Normal Form (2NF):To be in 2NF, the table must first be in 1NF, and all non-prime attributes (attributes not part of any candidate key) should be fully functionally dependent on the entire primary key.In the provided table:show\_id appears to be the primary key.All other attributes (cid, tid, aid, offer\_id, hallnumber, show\_date, seat\_number) are functionally dependent on the primary key.Therefore, the table appears to be in 2NF.Third Normal Form (3NF):To be in 3NF, the table must first be in 2NF, and no transitive dependencies should exist.In the provided table, there don't appear to be any transitive dependencies. All attributes seem to be directly dependent on the primary key (show\_id). Therefore, the table seems to be in 3NF.Boyce-Codd Normal Form (BCNF):BCNF is a higher level of normalization where every determinant must be a candidate key.Since the primary key (show\_id) seems to uniquely determine all other attributes, the table likely satisfies BCNF.Fourth Normal Form (4NF) and Fifth Normal Form (5NF):These normal forms deal with more complex dependencies like multi-valued dependencies and overlapping composite keys. Typically, achieving these forms involves breaking down the table further into smaller, more specialized tables to remove such dependencies.In the provided table, there don't seem to be any indications of multi-valued dependencies or overlapping composite keys, so 4NF and 5NF may not be relevant in this context.In summary, based on the given information, the table appears to be at least in 3NF and possibly in BCNF.

**OUTPUT:-**

****