

## Movie Rentals Tables Normalization Explanation:

### 1. Movies Table:

Column	Data Type	Description
movie_id	SERIAL (PK)	Unique identifier for each movie
title	VARCHAR(255)	Title of the movie
release_year	INT	Year the movie was released
genre	VARCHAR(50)	Genre of the movie
director	VARCHAR(255)	Director of the movie

#### Normal Form Analysis:

- **1NF:** All columns are atomic (e.g., title, release\_year, genre, director are single, indivisible values), and each row is unique because of the movie\_id primary key.
- **2NF:** Since movie\_id is the primary key, all other columns (title, release\_year, genre, director) depend directly on movie\_id. There are no partial dependencies.
- **3NF:** There are no transitive dependencies—each non-key column describes only the movie\_id (the primary key). For example, title is directly related to movie\_id and does not depend on release\_year, genre, or director.

The Movies table is in 3NF because it meets the requirements of 1NF, 2NF, and 3NF.

### 2. Customers Table:

Column	Data Type	Description
customer_id	SERIAL (PK)	Unique identifier for each customer
first_name	VARCHAR(50)	First name of the customer
last_name	VARCHAR(50)	Last name of the customer
email	VARCHAR(255)	Email address (unique)
phone	TEXT	Phone number of the customer

#### Normal Form Analysis:

- **1NF:** All columns have atomic values, with no arrays or repeating groups (e.g., first\_name, last\_name, email, and phone are single values).
- **2NF:** All columns depend fully on the primary key customer\_id. There are no partial dependencies.
- **3NF:** There are no transitive dependencies in this table. Each column directly relates to customer\_id and does not depend on any other non-key column.

The Customers table is in 3NF.

### 3. Rentals Table:

Column	Data Type	Description
rental_id	SERIAL (PK)	Unique identifier for each rental
customer_id	INT (FK)	Foreign key referencing Customers
movie_id	INT (FK)	Foreign key referencing Movies
rental_date	DATE	Date when the movie was rented
return_date	DATE	Date when the movie was returned (or expected)

#### Normal Form Analysis:

- **1NF**: All columns contain atomic values (e.g., rental\_date and return\_date are individual date values).
- **2NF**: The primary key rental\_id uniquely identifies each rental record, and all other columns depend fully on this primary key. There are no partial dependencies because rental\_id is a single-column primary key.
- **3NF**: There are no transitive dependencies. Each non-key column is directly related to rental\_id and not to any other non-primary key columns.

The Rentals table is in 3NF.

### Summary:

Each table in the movie rental system database (Movies, Customers, and Rentals) is in **3NF** because:

1. They each satisfy **1NF** by storing atomic values in each column.
2. They each satisfy **2NF** because every non-primary key column depends entirely on the primary key.
3. They each satisfy **3NF** by avoiding transitive dependencies, ensuring that each non-primary key column only depends directly on the primary key.