

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. Ігоря Сікорського»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

# Кафедра системного програмування та спеціалізованих комп’ютерних систем

**Лабораторна робота № 1**

з дисципліни

**«Бази даних і засоби управління»**

**Форматування згідно вимог**

**SQL текст**

Виконав: студент III курсу

ФПМ групи КВ-02

Костюков Сергій Васильович

Перевірив: Павловський В. І.

Київ-2022

***Проектування бази даних та ознайомлення з базовими операціями СУБД PostgreSQL***

*Метою роботи* є здобуття вмінь проектування бази даних та практичних навичок створення реляційних баз даних за допомогою PostgreSQL.

*Завдання* роботи полягає у наступному:

1. Розробити модель «сутність-зв’язок» предметної галузі, обраної студентом самостійно, відповідно до пункту «Вимоги до ER-моделі»;
2. Перетворити розроблену модель у схему бази даних (таблиці) PostgreSQL;
3. Виконати нормалізацію схеми бази даних до третьої нормальної форми (3НФ);
4. Ознайомитись із інструментарієм PostgreSQL та pgAdmin 4 та внести декілька рядків даних у кожну з таблиць засобами pgAdmin 4.

*Вимоги до ER-моделі*

1. Сутності моделі предметної галузі мають містити зв’язки типу 1:N або N:M;
2. Кількість сутностей у моделі – 3-4. Кількість атрибутів у кожній сутності: від двох до п’яти;
3. Передбачити наявність зв’язку з атрибутом;
4. Для побудови ER-діаграм використовувати одну із нотацій: Чена, “Пташиної лапки (Crow’s foot)”, UML.

Посилання на гіт: https://github.com/SergioKostyukov/Database

**Етап 1** - Розробка моделі “сутність-зв’язок” для предметної галузі  “Сервіс продажу квитків кіно” (нотація Crow’s foot)

**Перелік сутностей з описом їх призначення:**

В концептуальній моделі предметної області "Продаж квитків кіно" (рис.1) виділяються наступні сутності та зв'язки між ними.

Сутність “Client” призначено для визначення клієнтів (ім’я, номер телефону, статус).

Сутність “Film” призначено для визначення фільму (назва, рік прем’єри, тривалість, продюсер).

Сутність “Session” призначено для визначення сеансу (час початку, місце, фільм).

Сутність “Seat” призначено для визначення місця (номер, ряд, поверх, ТЦ).

Сеанс може відбуватись без клієнтів, а клієнт може не відвідувати жодного сеансу, тому між сутностями “Session” та “Client” зв’язок (N:M) з додатковим атрибутом “book time”.

Місце може бути використано для декількох сеансів, і сеанс може відбуватись для багатьох місць, отже між сутностями “Session” та “Seat” зв’язок (N:M).

Фільм може показуватись на багатьох сеансах, але на одному сеансі може бути лише один фільм, тому між сутностями “Session” та “Film” зв’язок (1:N).

**А яке місце у клієнта незрозуміло! Але залишимо так**

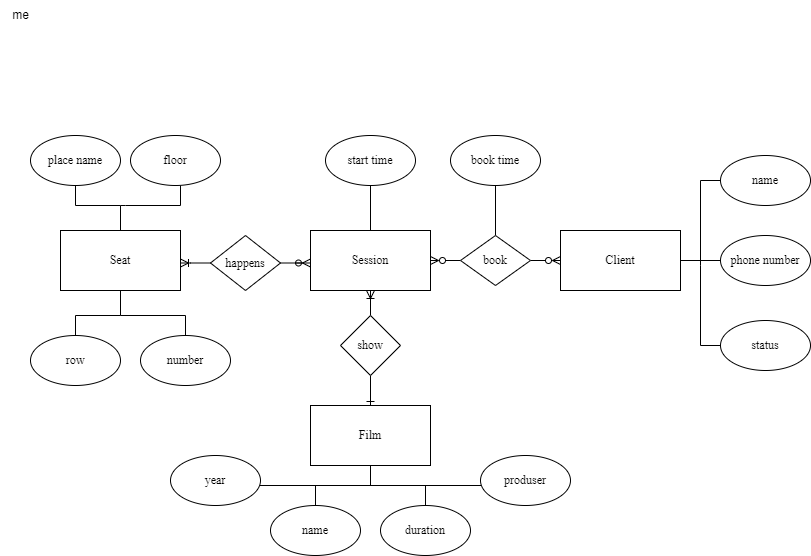


рис.1 - Концептуальна модель предметної області “Продаж квитків кіно”

**Етап 2 -** Перетворення розробленої моделі у схему бази даних PostgreSQL

**Опис процесу перетворення:**

Сутність “Client” перетворена в таблицю “Client”.

Сутність “Session” була перетворена в таблицю “Session”.

Сутність “Film” була перетворена в таблицю “Film”.

Сутність “Seat” була перетворена в таблицю “Seat”.

Зв’язок “book” (M:N) зумовив створення перехідної таблиці “Client\_Session”.

Зв’язок “happens” (M:N) зумовив створення перехідної таблиці “Seat\_ Session”.

**Схема бази даних у графічному вигляді:**

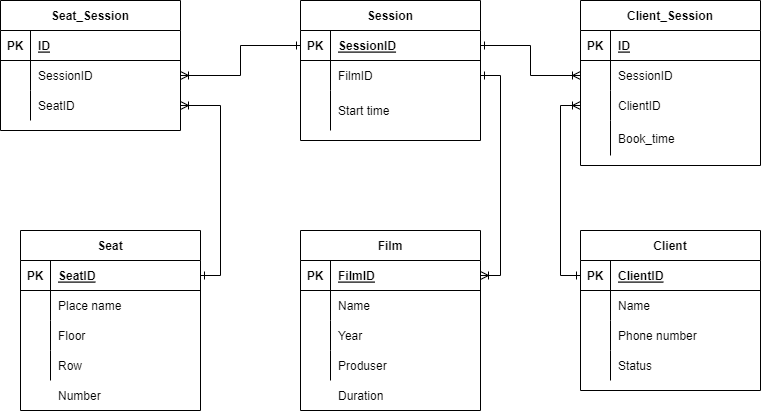


Рис.2 Схема бази даних у графічному вигляді

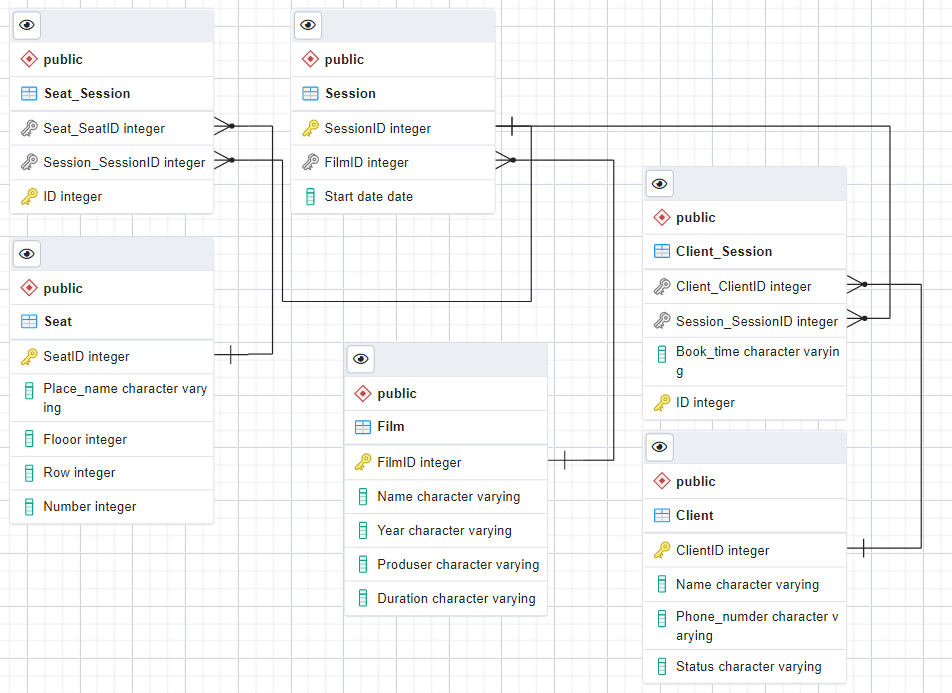


Рис.3 Схема бази даних у pgAdmin 4

**Етап 3** - Нормалізація схеми бази даних до третьої нормальної форми

*Функціональні залежності:*

Session (SessionID, FilmID, Start date)

SessionID → FilmID

SessionID → Start date

SessionID → FilmID, Start date

Client (ClientID, Name, Phone\_number, Status)

ClientID → Name

ClientID → Phone\_number

ClientID → Status

ClientID → Name, Phone\_number, Status

Film (FilmID, Name, Year, Producer, Duration)

FilmID → Name

FilmID → Year

FilmID → Producer

FilmID → Duration

FilmID → Name, Year, Producer, Duration

Seat (SeatID, Place\_name, Floor, Row, Number)

SeatID → Place\_name

SeatID → Floor

SeatID → Row

SeatID → Number

SeatID → Place\_name, Floor, Row, Number

Seat\_Session (ID, Seat\_SeatID, Session\_SessionID)

ID → Seat\_SeatID

ID → Session\_SessionID

ID → Seat\_SeatID, Session\_SessionID

Client\_Session (ID, Client\_ClientID, Session\_SessionID)

ID → Client\_ClientID

ID → Session\_SessionID

ID → Client\_ClientID, Session\_SessionID

**Пояснення щодо відповідності схеми бази даних нормальним формам:**

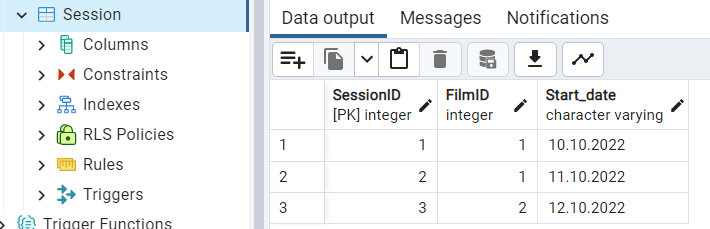
Схема бази даних відповідає 1НФ тому, що всі атрибути таблиці є атомарними, кожна таблиця має primary key та мінімальний набір атрибутів.

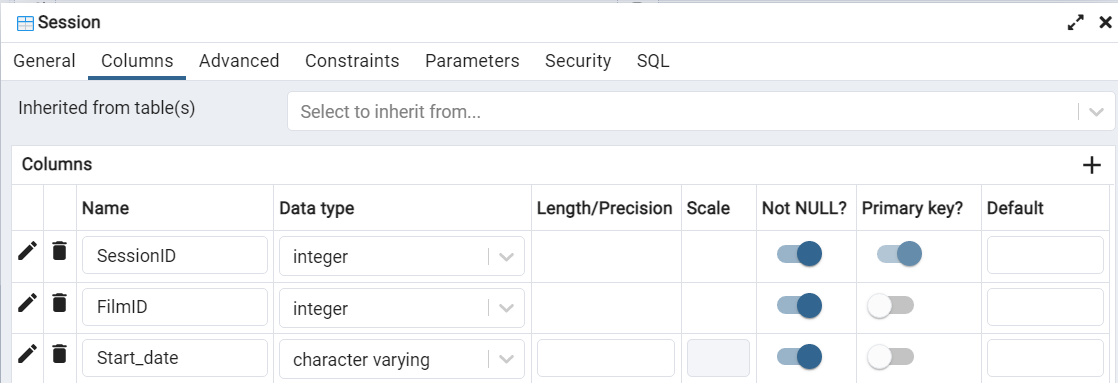
Схема бази даних відповідає 2НФ тому, що відповідає 1НФ, та не має часткової залежності, тобто кожен неключовий атрибут функціонально залежить від цілого ключа.

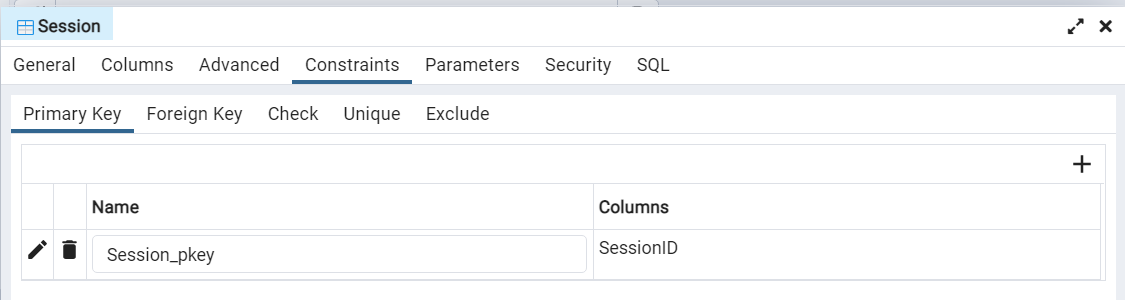
Схема бази даних відповідає 3НФ тому, що відповідає 2НФ, та відсутні транзитивні функціональні залежності неключових атрибутів від ключових.

**Етап 4** - наведення скріншотів роботи з PostgreSQL та pgAdmin 4

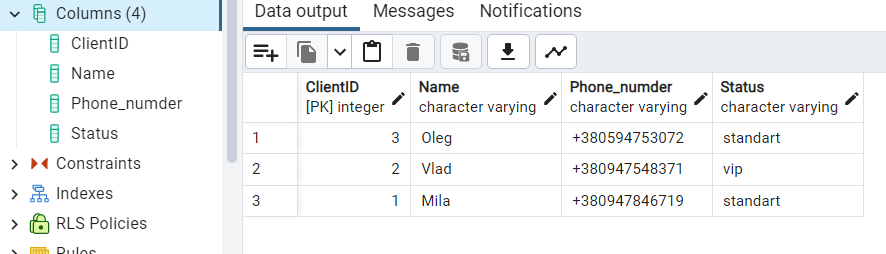
Session:

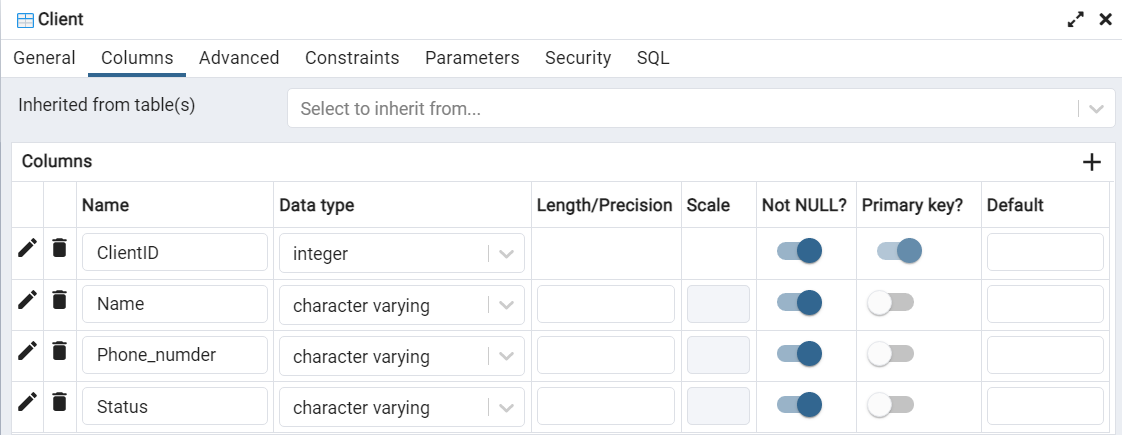


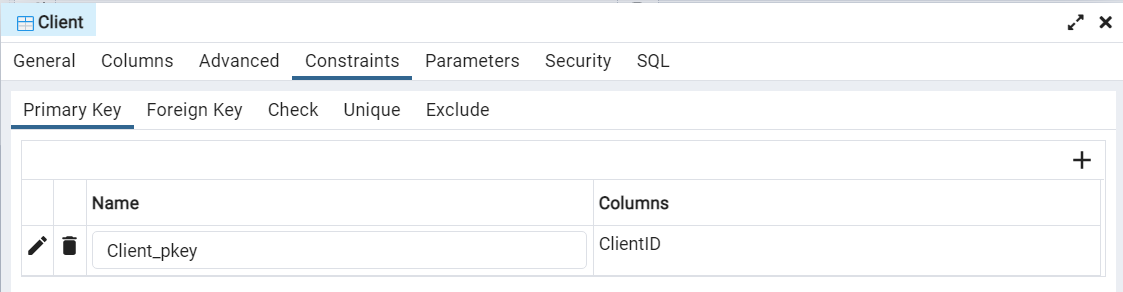




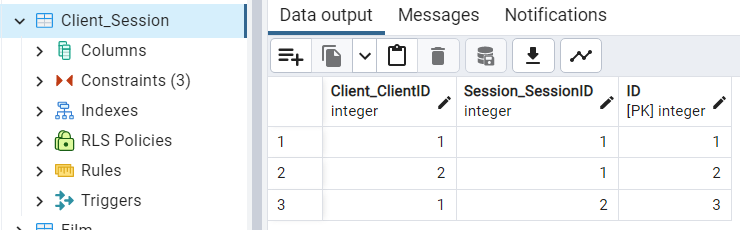
Client:

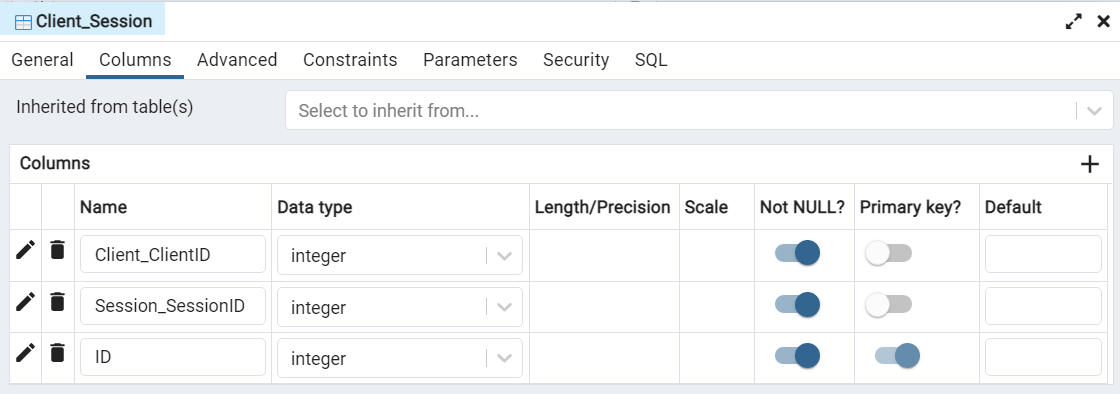


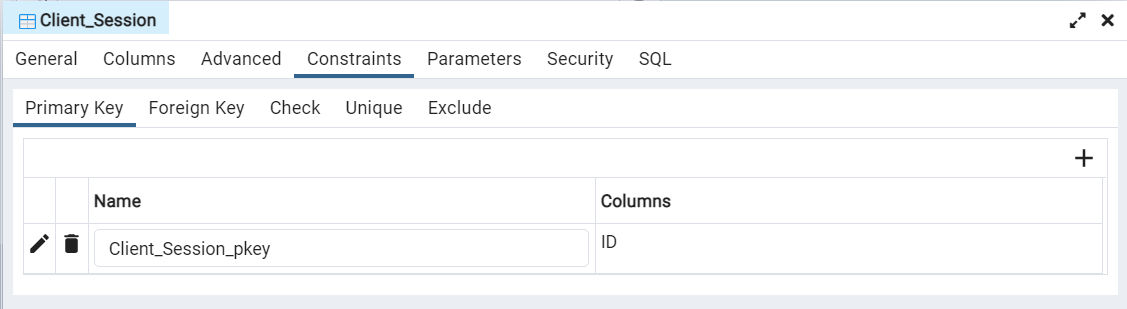




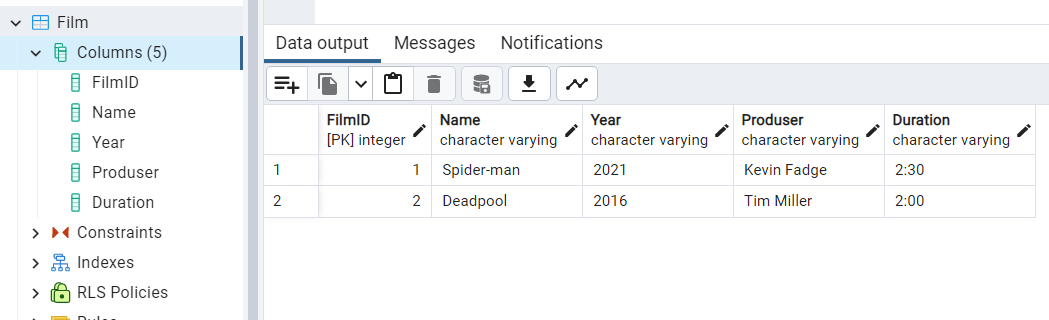
Client\_ Session:

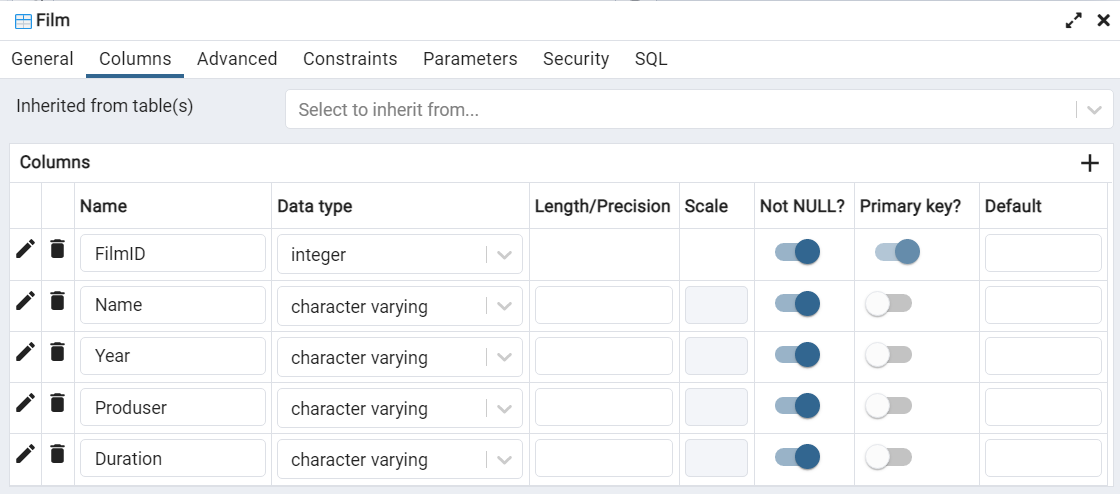


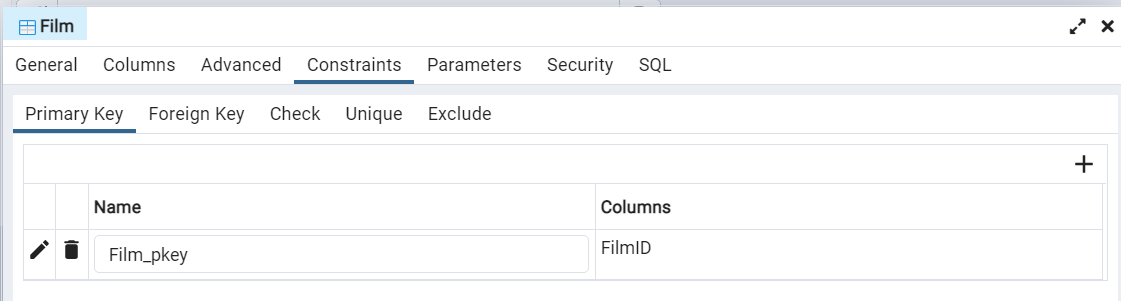




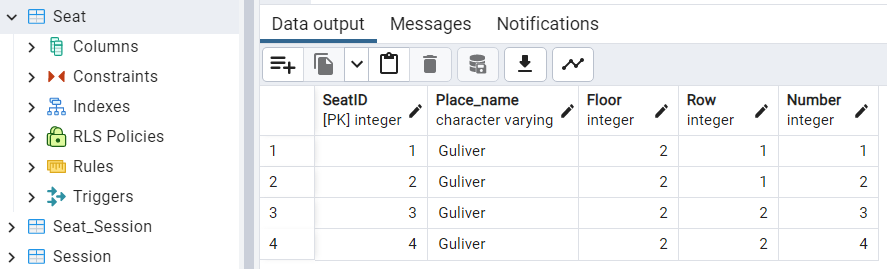
Film:

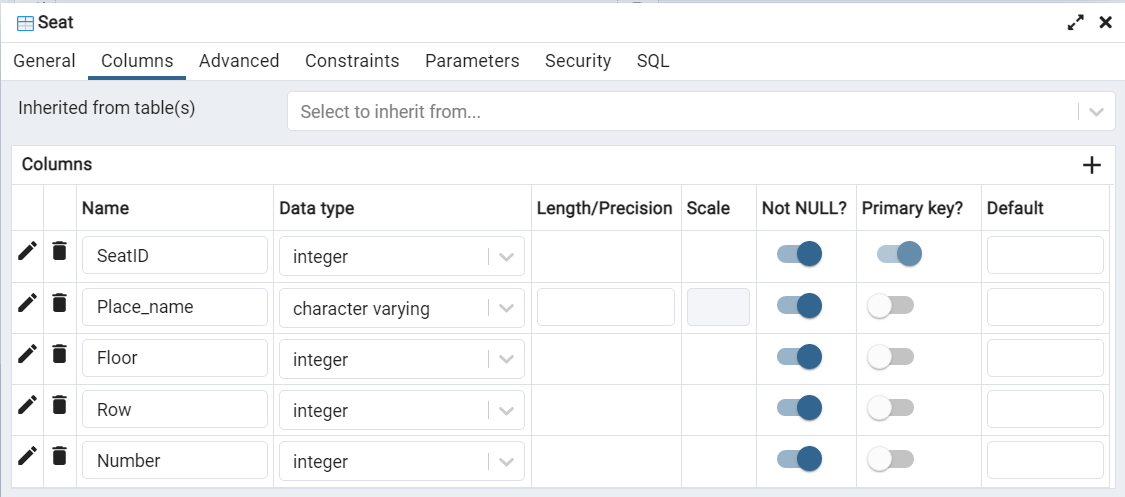


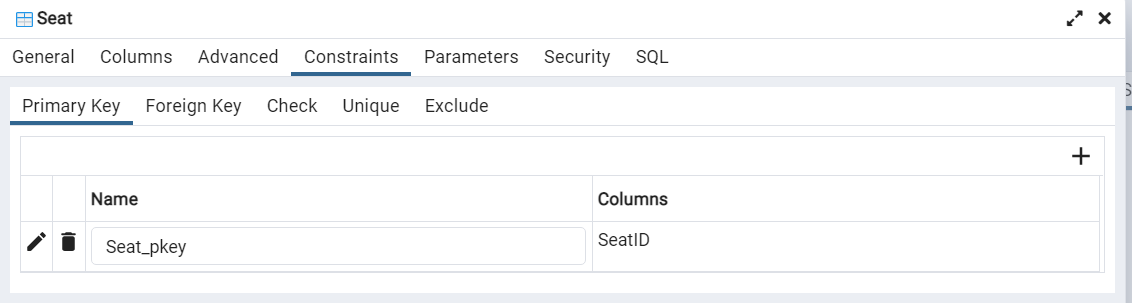




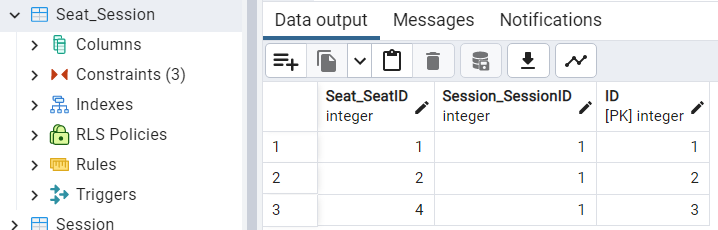
Seat:

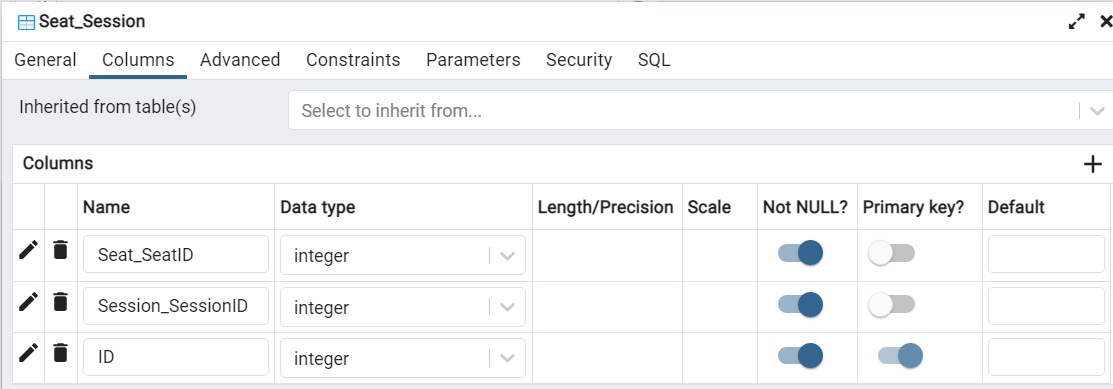


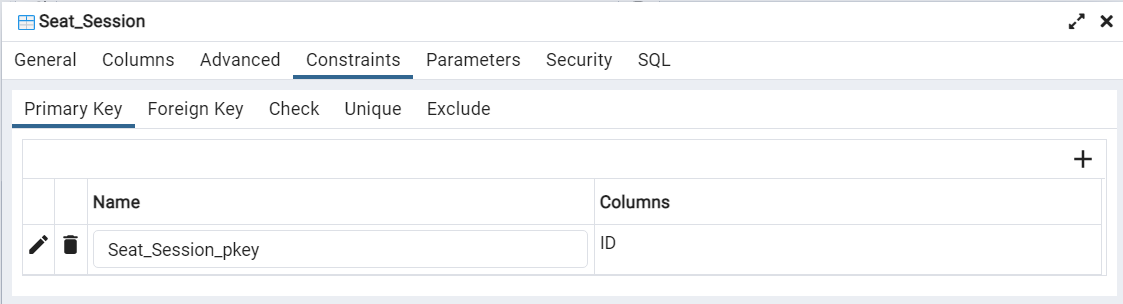




Seat\_Session:







**SQL текст**