НАЦIОНАЛЬНИЙ ТЕХНIЧНИЙ УНIВЕРСИТЕТ УКРАЇНИ «КИЇВСЬКИЙ ПОЛIТЕХНIЧНИЙ IНСТИТУТ IМЕНI IГОРЯ СIКОРСЬКОГО» ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

**Кафедра системного програмування і спеціалізованих комп'ютерних систем**

**Лабораторна робота №1**

з дисципліни «Бази даних і засоби управління»

Тема: «**Проектування бази даних та ознайомлення з базовими операціями СУБД PostgreSQL**»

Виконали: студенти ІІІ курсу

ФПМ групи КВ-81

Ядуха Б.В.

Викладач: ПетрашенкоА.В.

***Завдання***

*У звіті щодо пункту №1 завдання має бути:*

* перелік сутностей з описом їх призначення;
* графічний файл розробленої моделі «сутність-зв’язок»;
* назва нотації.

*У звіті щодо пункту №2 завдання має бути:*

* опис процесу перетворення (наприклад, “сутність А було перетворено у таблицю А, а зв’язок R (M:N) зумовив появу додаткової таблиці R1 тощо);
* схему бази даних у графічному вигляді **з назвами таблиць (!) та зв’язками між ними, а також необхідно намалювати перетворену ER-діаграму у ТАБЛИЦІ БД! Це означає, що тут не може бути зв'язку N:M, мають бути позначені первинні та зовнішні ключі, обмеження NOT NULL та UNIQUE і бажано внести типи даних атрибутів.**

*У звіті щодо пункту №3 завдання має бути:*

* пояснення (**обґрунтування!**) щодо відповідності схеми бази даних нормальним формам НФ1, НФ2 та НФ3. Пояснення ***полягає у наведенні функціональних залежностей***, що демонструють висновки. У випадку невідповідності надати опис необхідних змін у схемі;
* У випадку проведення змін у схемі бази даних надати оновлену версію схеми, інакше - не наводити схему.

*У звіті щодо пункту №4 завдання має бути:*

* навести копії екрану з pgAdmin4, що відображають назви, типи та обмеження на стовпці (доступне у закладці “Columns” та “Constraints” властивостей “Properties” таблиць дерева об’єктів у pgAdmin4);
* навести копії екрану з pgAdmin4, що відображають вміст таблиць бази даних у PostgreSQL. Таблиці на зображенні обов'язково **повинні мати назву**!

Пункт №1

**Перелік сутностей**

**Film -** зберігає дані про фільм: Movie-title - назва фільму, Director - режисер, MPAA - віковий рейтинг фільму.

**Hall** - зберігає дані про зал: Number - номер залу, Size - розмір залу

**Performance** - зберігає дані про сеанс: Time - час сеансу

**Ticket** - зберігає дані про продані квитки: seat - місце, row - ряд

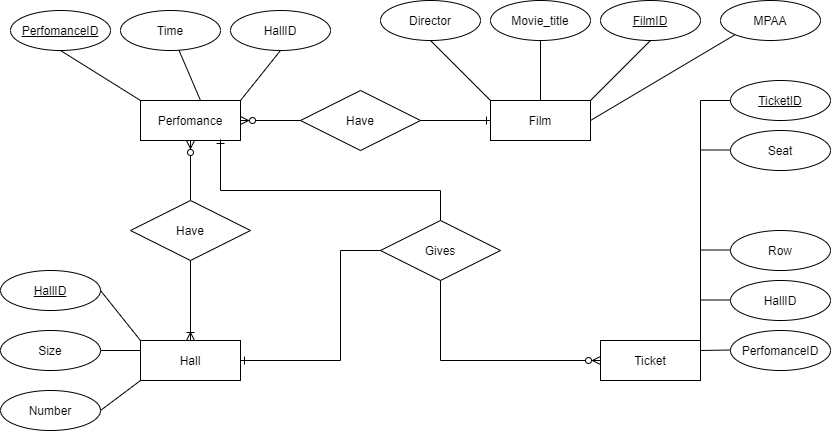
**Зв’язки**

**Film-Performance** - 1 : M (на один фільм існує 0 або багато сеансів)

**Film-Ticket -** 1 : M (на один фільм куплено від 0 до багатьох білетів)

**Hall-Ticket -** 1 : M (на один зал куплено від 0 до багатьох білетів)

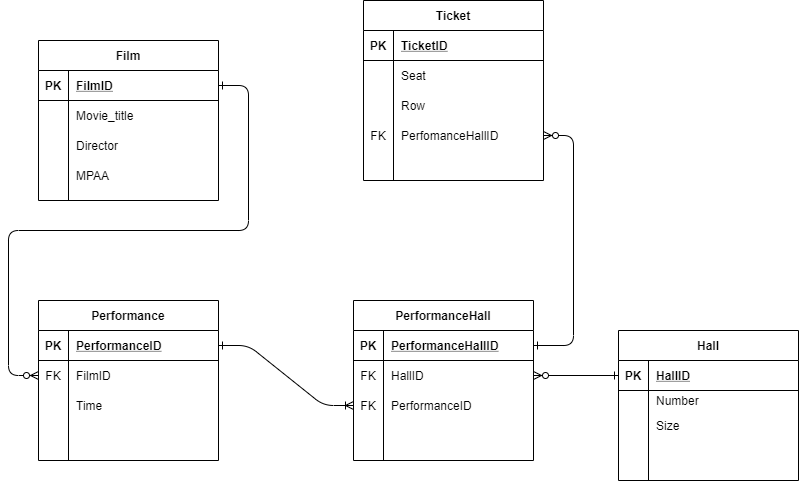
**Performance-Ticket -** 1 : M (на один сеанс куплено від 0 до багатьох білетів)

**Hall-Performance -** N : M (Один сеанс може проводитися в багатьох залах, а в одному залі можуть проводитися багато сеансів. Існування залу не означає, що в ньому проводиться сеанс. Якщо сеанс існує його проводять хоча б в одному залі)Нотація Чена

Пункт №2

Для забезпечення зв’язку між таблицями:

* Всі сутності перетворенні в одноймені таблиці
* Були додані первинні ключі для кожної таблиці (ID)
* Через зв’язок “Performance”-”Hall” (M:N) було створено додаткову таблицю “Performance/Hall”
* Для зв’язку “Performance”-”Ticket” було додано зовнішній ключ “PerformanceID” в таблиці “Ticket”
* Для зв’язку “Hall”-”Ticket” було додано зовнішній ключ “HallID” в таблиці “Ticket”
* Для зв’язку “Film”-”Performance” було додано зовнішній ключ “FilmID” в таблиці “Performance”



Пункт №3

Таблиці бази даних знаходяться у 1НФ тому, що кожне значення є атомарним. В стовпці “Time” таблиці Performance зберігається одне значення аналогічно для інших таблиць.

Таблиці бази даних знаходяться у 2НФ тому, що вони знаходяться в 1НФ і кожне значення залежить лише від одного ключа і не існує композитних первинних ключів.

Таблиця **Film**(FilmID - PK):

FilmID -> Movie\_title

FilmID -> Director

FilmID -> MPAA

Таблиця **Hall**(HallID - PK):

HallID -> Number

HallID -> Size

Таблиця **Performance**(PerformanceID - PK):

PerformanceID -> Time

PerformanceID -> FilmID

Таблиця **Ticket**(TicketID - PK):

TicketID -> Seat

TicketID -> Row

TicketID -> PerformanceHallID

Таблиця **PerformanceHall**(PerformanceHallID - PK):

PerformanceHallID -> PerformanceID

PerformanceHallID -> HallID

Таблиці бази даних знаходяться у 3НФ тому, що вони знаходяться в 2НФ і транзитивно не залежать один від одного.

Таблиця **Film**(FilmID - PK):

FilmID -> Movie\_title

FilmID -> Director

FilmID -> MPAA

Всі характеристики окрім первинного ключа не є унікальними і тому через них не можна дізнатися інші характеристики.

Таблиця **Hall**(HallID - PK):

HallID -> Number

HallID -> Size

Всі характеристики окрім первинного ключа не є унікальними і тому через них не можна дізнатися інші характеристики.

Таблиця **Performance**(PerformanceID - PK):

PerformanceID -> Time

PerformanceID -> FilmID

Всі характеристики окрім первинного ключа не є унікальними і тому через них не можна дізнатися інші характеристики.

Таблиця **Ticket**(TicketID - PK):

TicketID -> Seat

TicketID -> Row

TicketID -> PerformanceHallID

Всі характеристики окрім первинного ключа не є унікальними і тому через них не можна дізнатися інші характеристики.

Таблиця **PerformanceHall**(PerformanceHallID - PK):

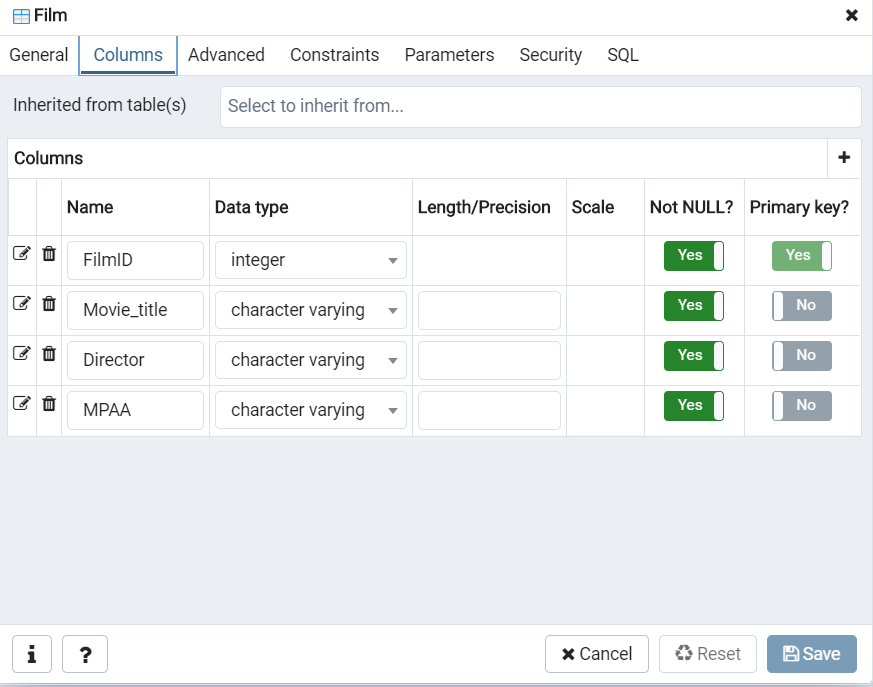
PerformanceHallID -> PerformanceID

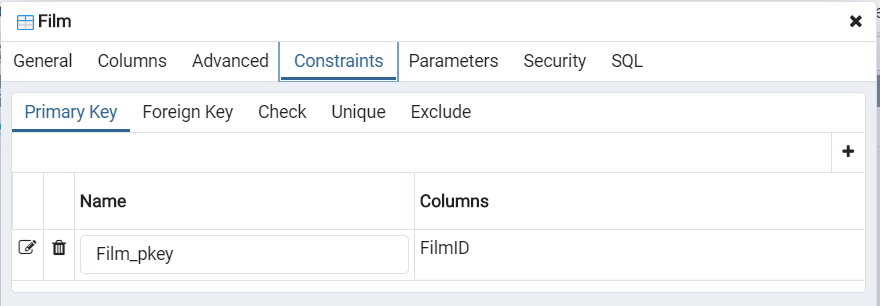
PerformanceHallID -> HallID

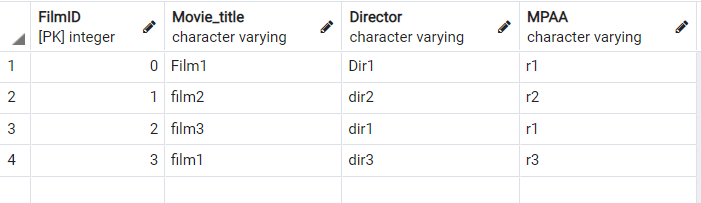
Всі характеристики окрім первинного ключа не є унікальними і тому через них не можна дізнатися інші характеристики.

Пункт №4

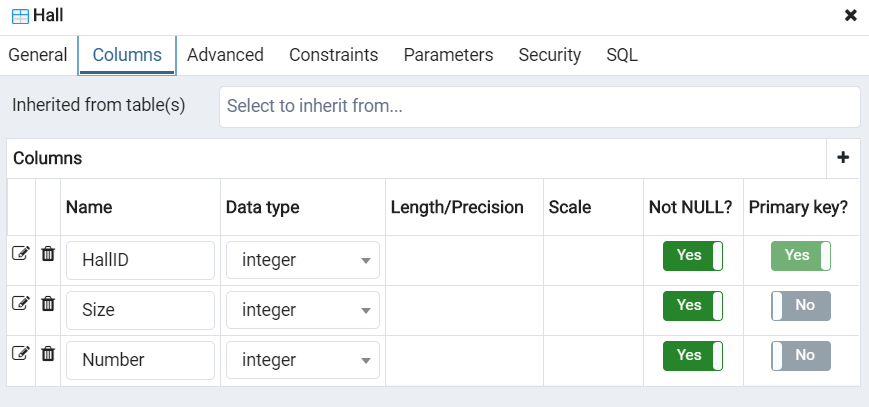
Таблиця **Film**:

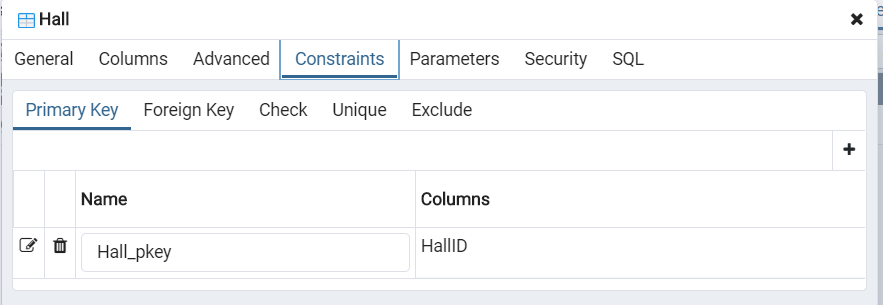


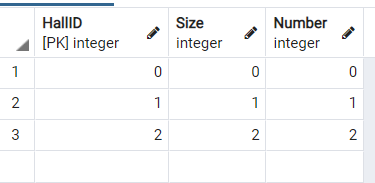




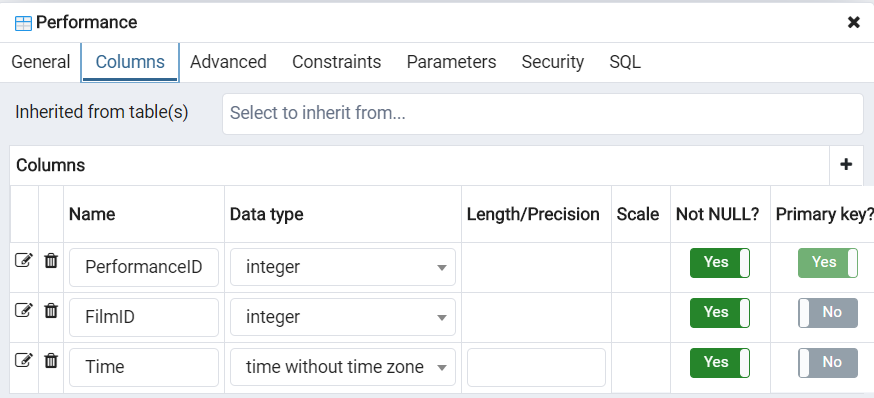
Таблиця **Hall**:

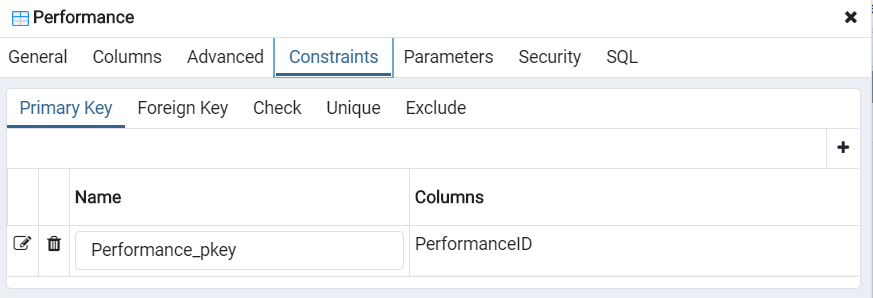


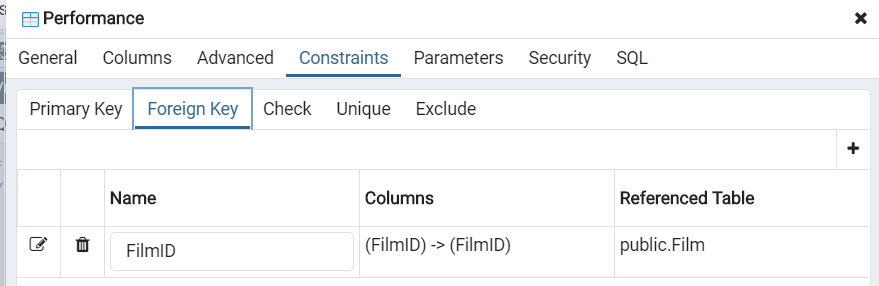


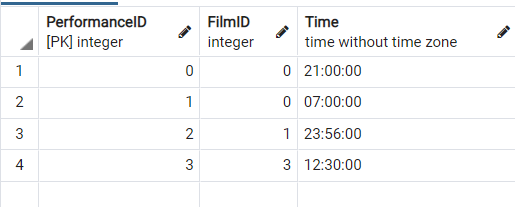


Таблиця **Performance**:

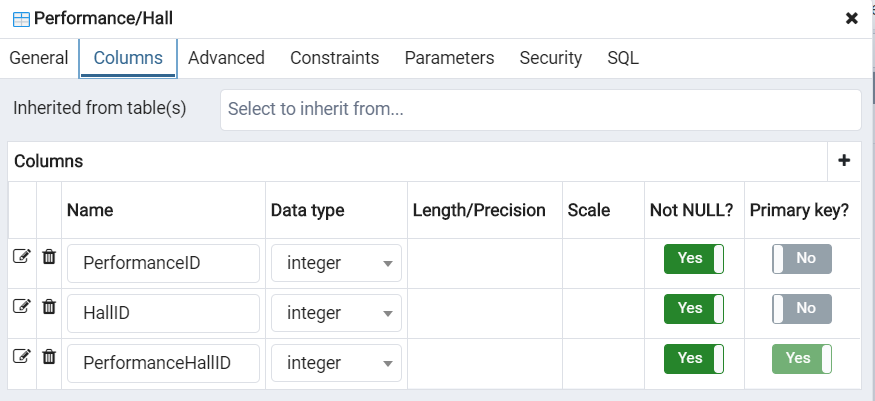


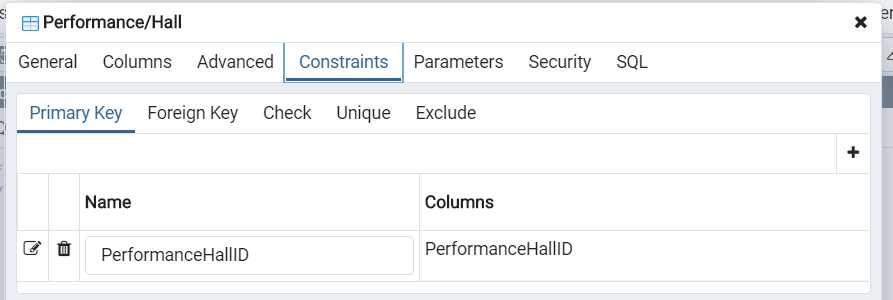


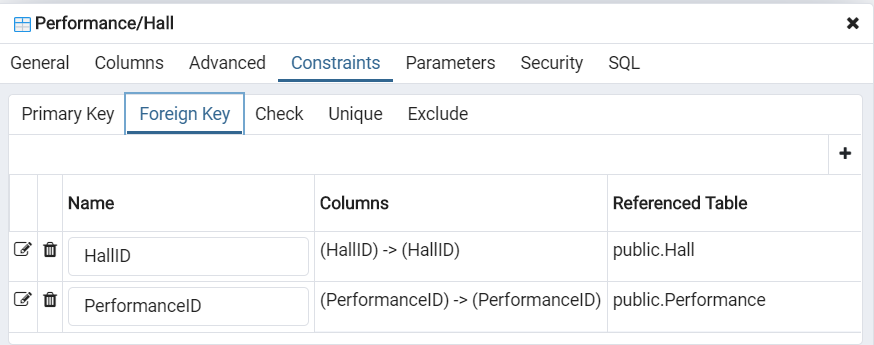


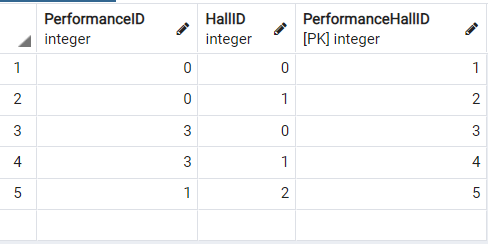


Таблиця **PerformanceHall**:









Таблиця **Ticket**:

