

**Національний технічний університет України**

**“Київський політехнічний інститут ім. Ігоря Сікорського”**

**Факультет прикладної математики**

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

**ЛАБОРАТОРНА РОБОТА № 1**

***з дисципліни***

***«Бази даних та засоби управління*»**

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

**Виконав:** студент ІII курсу

ФПМ групи КВ-21

Юдін Дмитро

**Оцінка:**

**Київ – 2024**

**Мета:** здобуття вмінь проектування бази даних та практичних навичок створення реляційних баз даних за допомогою PostgreSQL.

**Завдання лабораторної роботи:**

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.

**Предметна область:**

Електронна база даних для клієнтів хостелу.

GitHub репозиторій: https://github.com/montrealkiss/database

Контакт в Telegram: <https://t.me/living_unicornn>

**Завдання №1**

Розробка моделі «сутність-зв’язок» предметної галузі для проектування бази даних «Electronic database for hostel clients». Предметна галузь – 101 «Електронна база даних для клієнтів хостелу».

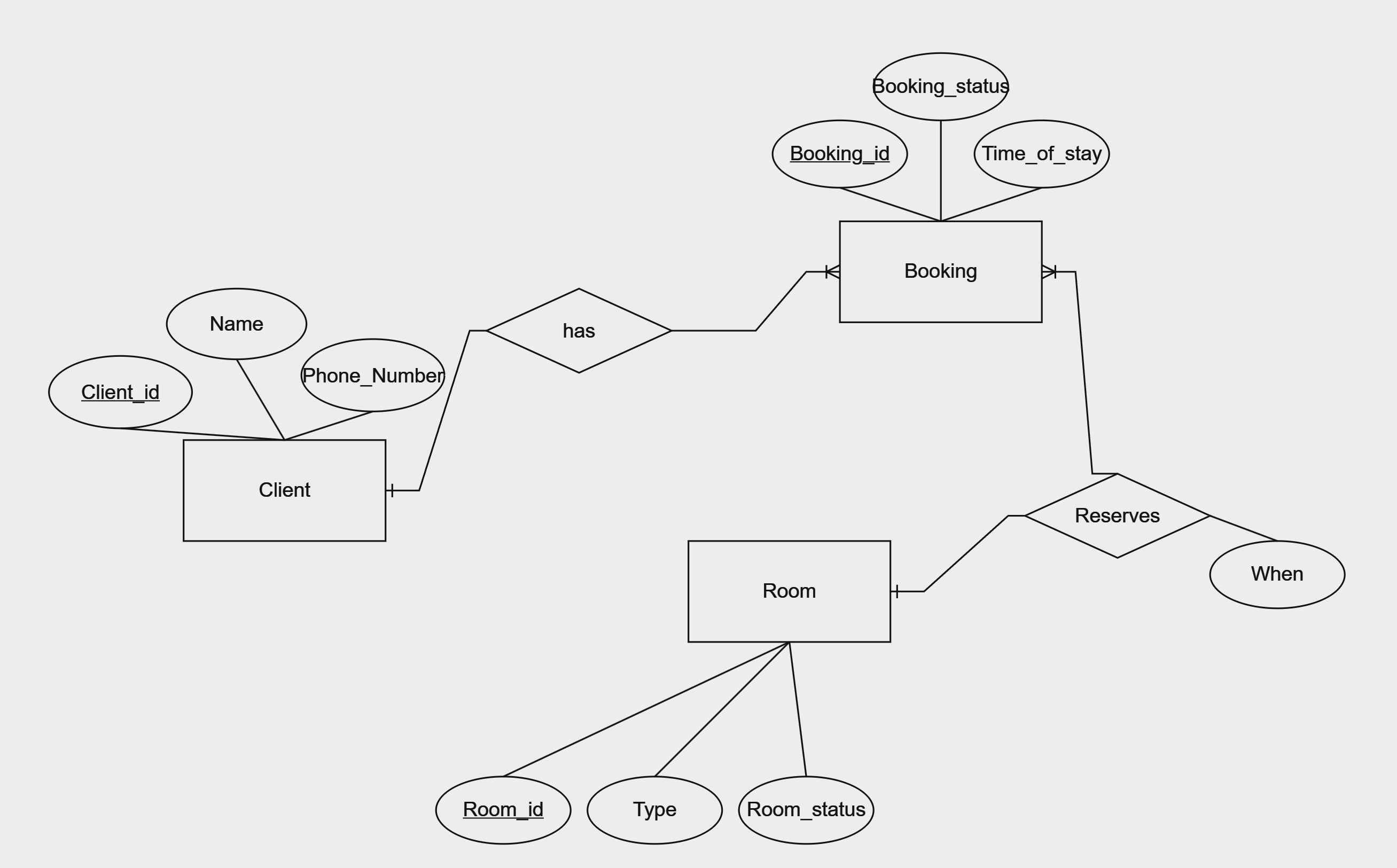


Рисунок 1. ER-діаграма побудована за нотацією Чена

**Сутності з описом призначення:**

Предметна галузь «Electronic database for hostel clients» включає в себе 3 сутності, кожна сутність містить по 3 атрибути:

1. Client (Client\_id, Name, Phone\_Number).
2. Booking (Booking\_id, Booking\_status, Time\_of\_stay).
3. Room (Room\_id, Type, Room\_status).

Сутність Client описує клієнта хостелу. Кожен клієнт має свій ідентифікатор, ім’я та номер мобільного телефону для зв’язку.

Сутність Booking описує бронювання номеру. Кожне бронювання має ідентифікатор бронювання, статус бронювання та час до якого заброньоване перебування в хостелі.

Сутність Room описує кімнату в хостелі. Кожна кімната має свій ідентифікатор, тип кімнати (кількість кімнат), та статус кімнати (вільна, зайнята, потребує прибирання).

**Зв’язки між сутностями:**

Зв’язок між Client та Booking:

Один клієнт може мати кілька бронювань. Це означає, що клієнт може робити різні бронювання в різний час. Одне бронювання належить лише одному клієнту. Зв’язок 1:N.

Зв’язок між Booking та Room:

Одне бронювання стосується конкретного номера. Один номер може бути пов'язаний з багатьма бронюваннями, але кожне бронювання відбувається на різні дати. Зв’язок 1:N.

**Завдання №2**

**Перетворення розробленої моделі «сутність-зв’язок» у схему бази даних PostgreSQL.**

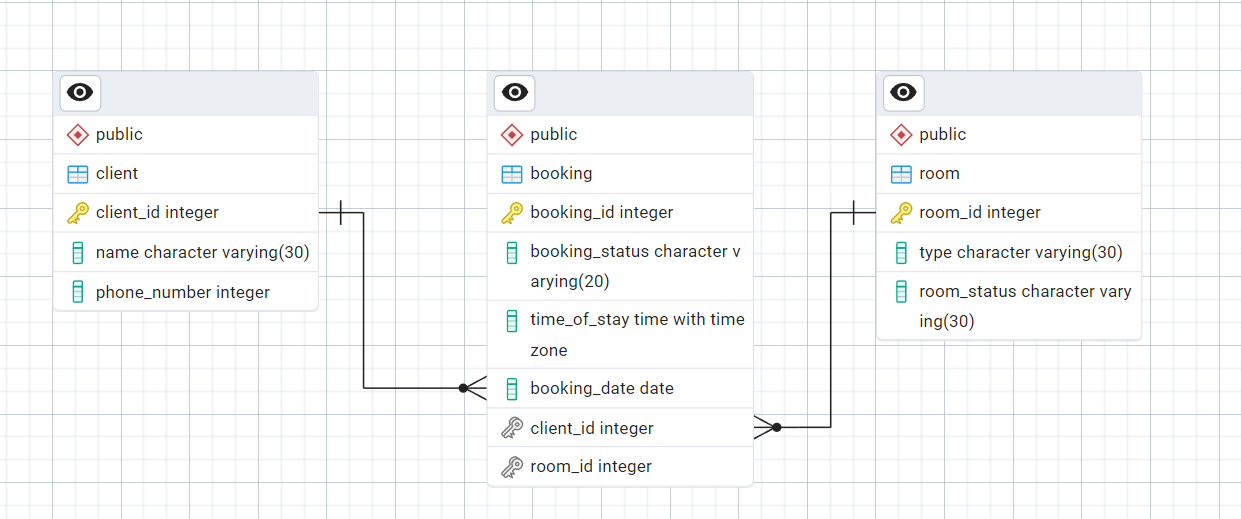
****

Рисунок 2. Схема бази даних у pgAdmin4

**Опис процесу перетворення:**

Сутність Client перетворилась на таблицю client. В таблиці роль первинного ключа виконує ідентифікатор client\_id, а атрибутами є: name та phone\_number.

Сутність Booking стала таблицею booking. Первинний ключ у цій таблиці – booking\_id, а зовнішнім ключем є room\_id та client\_id для реалізації зв’язку між booking-room та booking-client відповідно. Ця таблиця має найбільшу кількість атрибутів: time\_of\_stay, booking\_date, booking\_status. В ER-діаграмі був атрибут «When» у зв’язку Reserves, він перетворився на атрибут booking\_date типу date, що підпорядковується таблиці booking.

Сутність Room була перетворена на таблицю room. Первинний ключ: room\_id. Атрибутами є type та room\_status.

**Завдання №3**

Функціональні залежності:

1. Client (client\_id, name, phone\_number).

client\_id **→** name

client\_id **→** phone\_number

1. Booking (booking\_id, booking\_status, time\_of\_stay).

booking\_id **→** booking\_status

booking\_id **→** time\_of\_stay

1. Room (room\_id, type, room\_status).

room\_id **→** type

room\_id **→** room\_status

Аби схема бази даних була 1НФ потрібно щоб:

* В кожній клітинці таблиці було лише 1 значення
* Не було повторних рядків

Схема відповідає 1НФ. В кожній клітинці таблиці є лише по 1 значенню, та ми не маємо повторних рядків.

Аби схема бази даних була 2НФ потрібно щоб:

* Схема відповідала 1НФ
* Був в наявності первинний ключ
* Всі атрибути залежать від первинного ключа повністю, а не від якоїсь його частини

Схема відповідає 2НФ. Оскільки вона відповідає нормам 1НФ, в наявності всюди є первинні ключі, а всі атрибути залежать лише від одного первинного ключа в своїй таблиці.

Аби схема бази даних була 3НФ потрібно щоб:

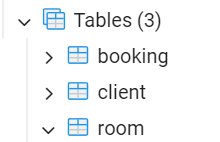
* База даних відповідала 2НФ
* Всі атрибути залежать тільки від первинного ключа, а не від інших атрибутів.

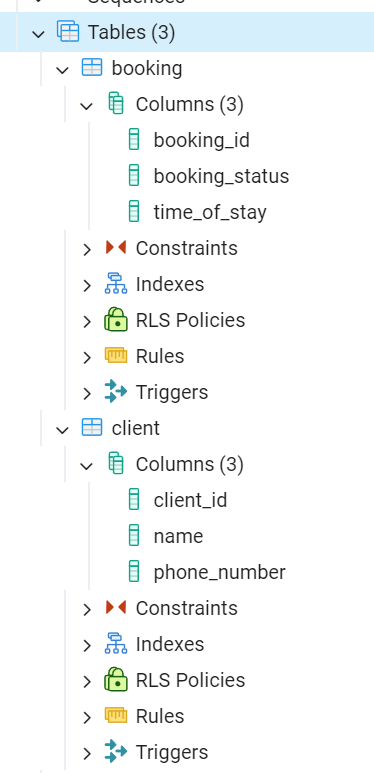
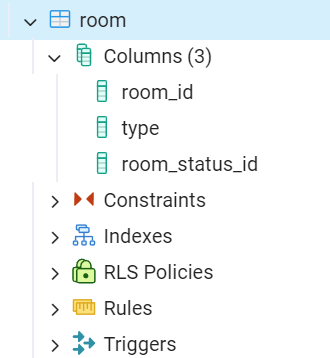
Схема відповідає 3НФ. Кожен неключевий атрибут не є транзитивно залежним від іншого неключового атрибута.

**Завдання №4**

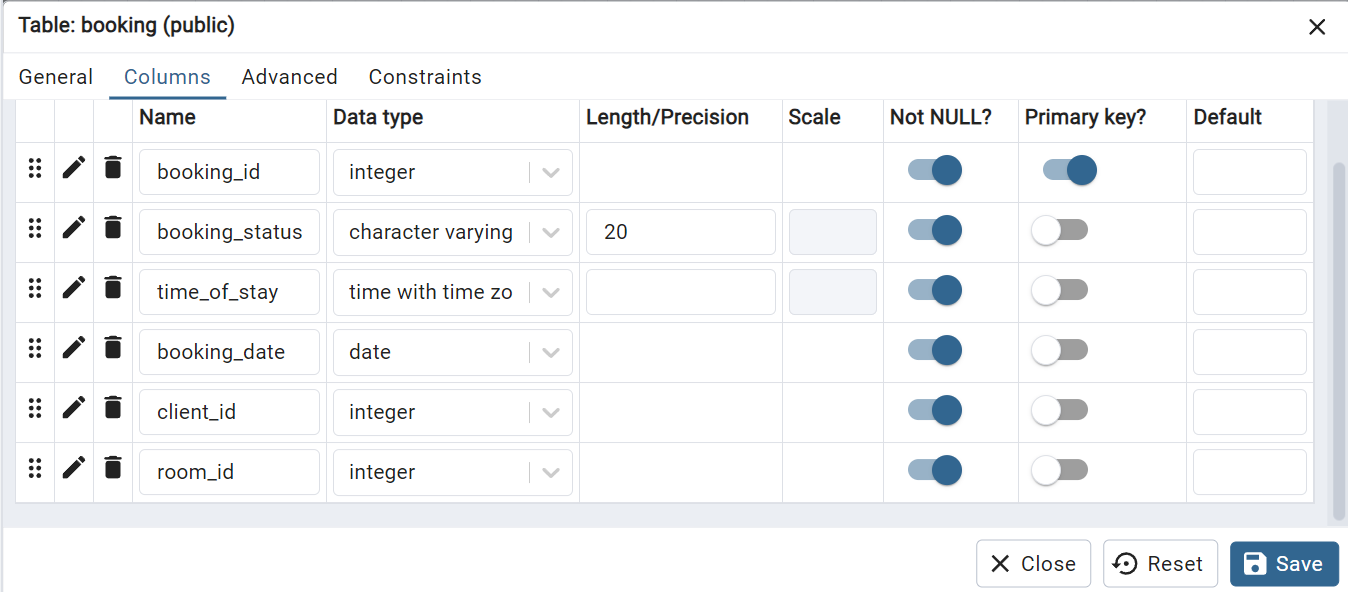
**Ознайомлення із інструментарієм PostgreSQL та pgAdmin 4 та внесення даних у кожну з таблиць засобами pgAdmin 4.**

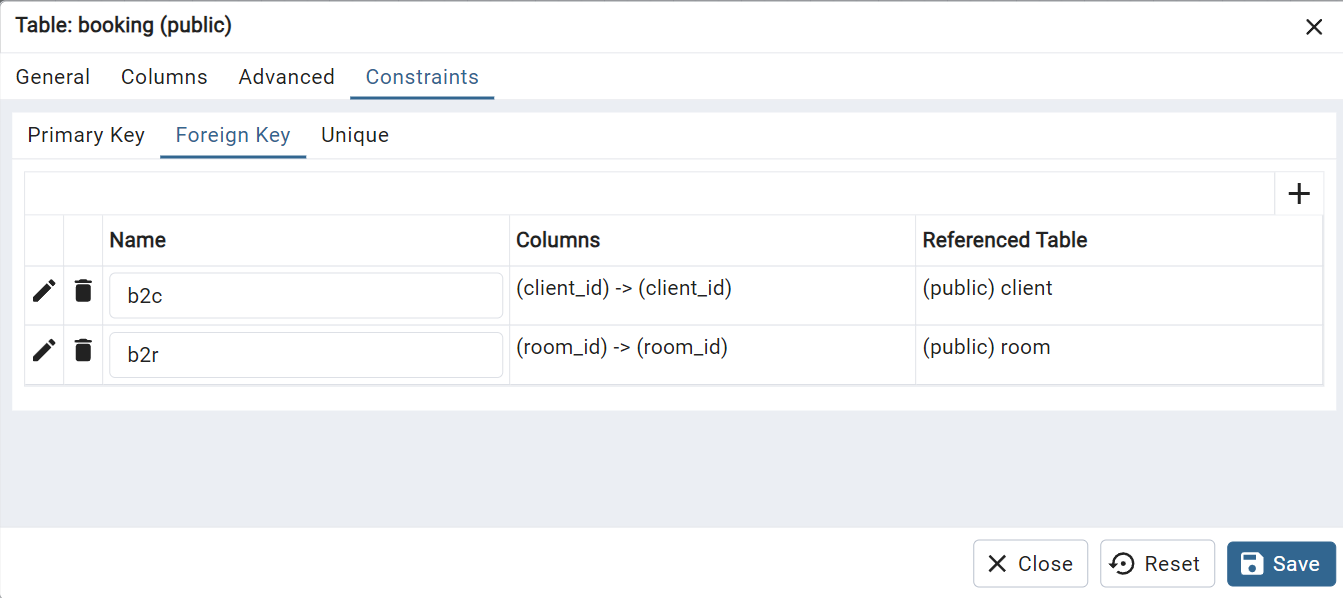
Таблиці бази даних:

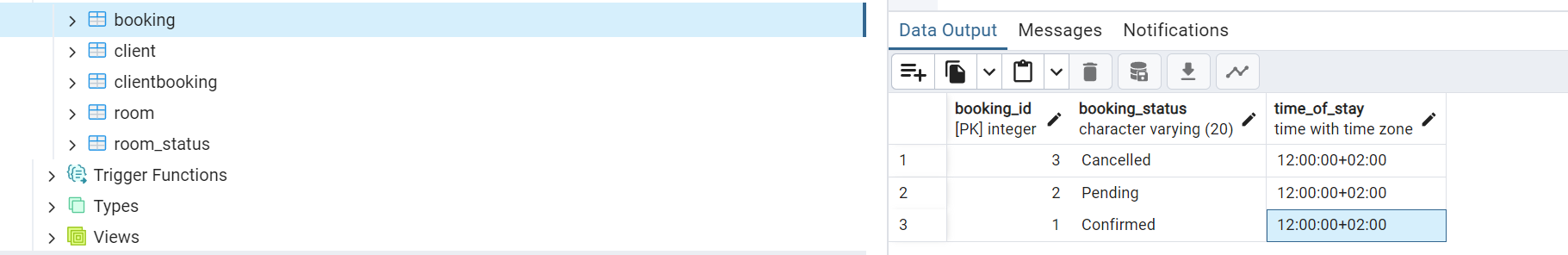


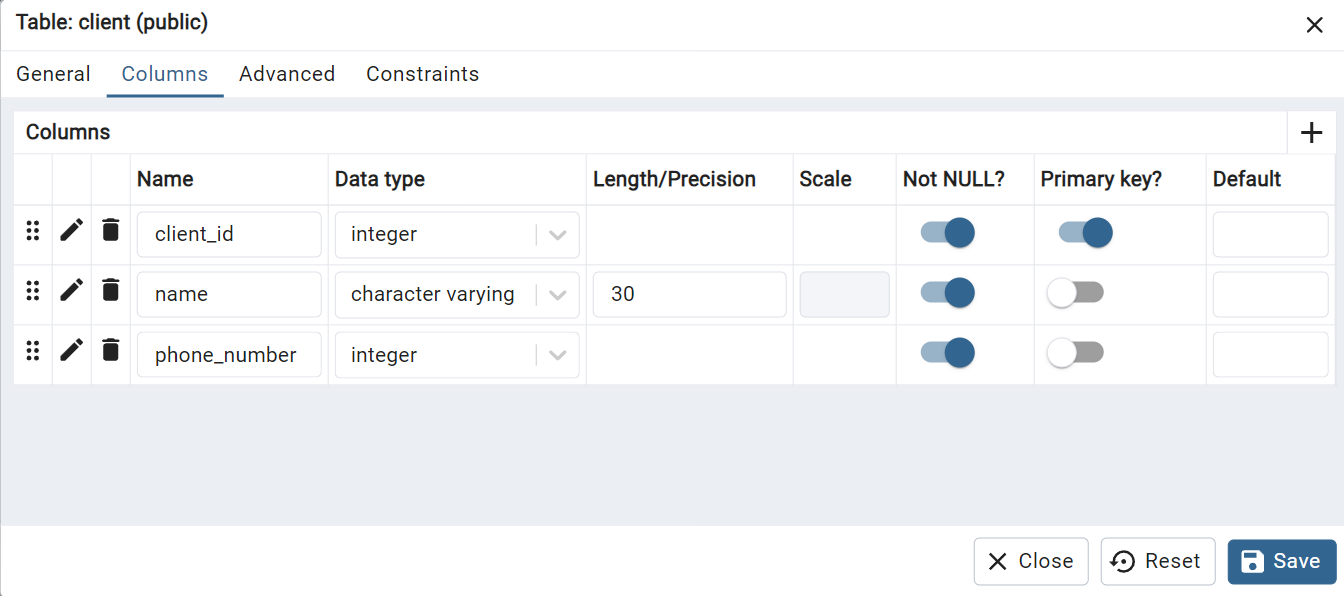
Таблиця booking

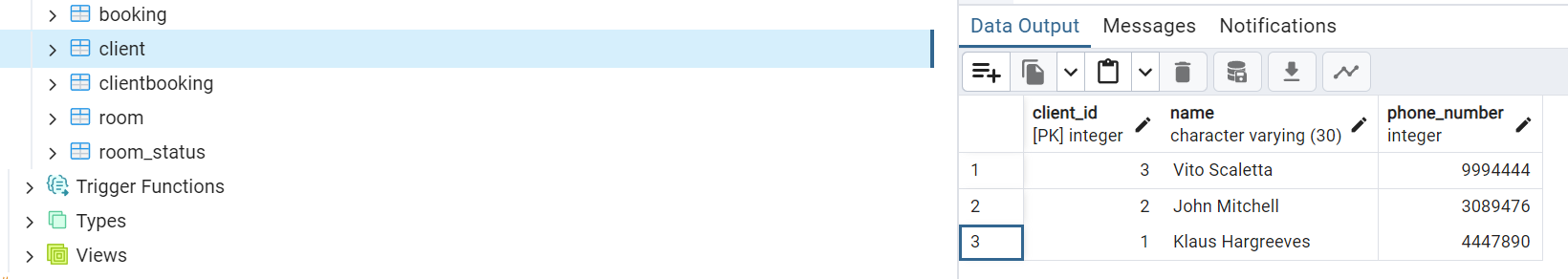






Таблиця client





Таблиця room

