

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені Ігоря Сікорського»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

**Лабораторна робота №1**

з дисципліни

**«Бази даних і засоби управління»**

Тема: «**Проектування бази даних та ознайомлення з базовими операціями СУБД PostgreSQL та PgAdmin 4**»

Виконав: студентка 3 курсу

Група: КВ-03

Костюченко А. В.

Перевірив: Петрашенко А. В.

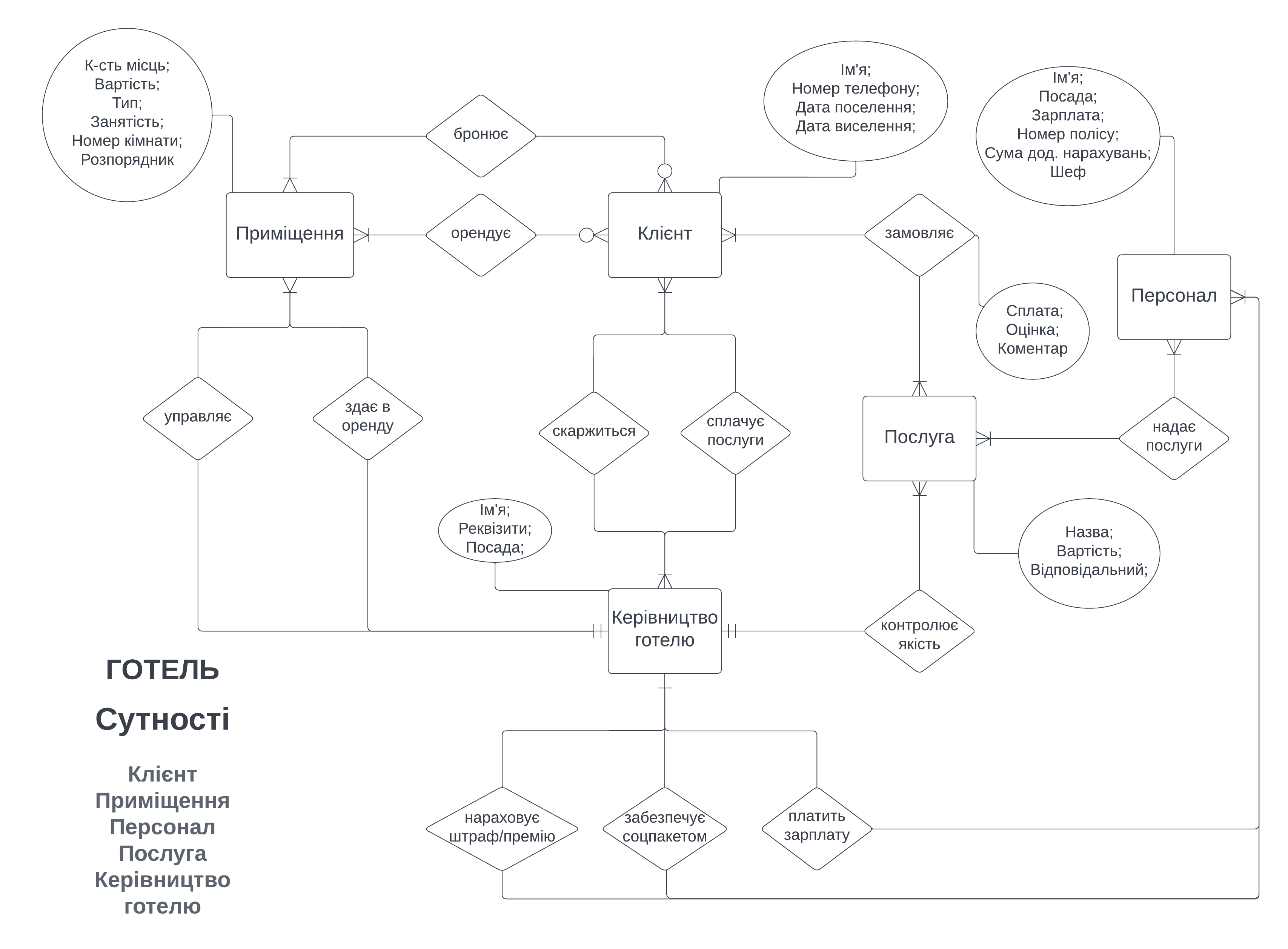
Київ – 2022

*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.

**Етап 1** - Розробка моделі “сутність-звʼязок” для предметної галузі “Готель”(нотація Crow’s foot)



*ER-діаграма моделі*

***Примітки***

* Атрибути однієї сутності зібрані в одному еліпсі. Кожен атрибут починається з нового рядка, атрибути розділені крапкою з комою. Так довелось зробити, бо у безкоштовній версії кількість фігур обмежена, а картку я потім забуду відв'язати. Якщо необхідно, можу спробувати якось переробити.
* Вважаємо, що у нас один директор на кожну галузь, кожен директор має у підпорядкуванні щонайменше одного працівника. Це дозволяє нам використати посаду як первинний ключ. Випадок, коли робітник працює на двох посадах у двох різних відділах не розглядаємо, тому отримуємо зв'язок 1:N.
* Номер телефона клієнта записуємо без +38. У такому разі розмірності integer якраз вистачає, хоч нуль на початку і не записується. Ідентифікувати людей за номером телефона чи страхового полісу дивно і незручно з практичної точки зору, але це дозволяє нам не додавати окремі ідентифікатори.
* Випадок, коли працівнику нараховують і штраф, і премію, не розглядаємо. Тому записуємо все в один атрибут.
* Клієнт має орендувати або забронювати хоча б одне приміщення, інакше він не клієнт. Але необов'язково робити і те, і інше, тому з його боку належність сутності необовязкова.
* Дати бронювання та поселення наведені у довгостроковій перспективі, тому зв'язок M:N. У нашому випадку номер записано на одну людину, незалежно від кількості мешканців - це полегшує збереження атомарності атрибутів. Атрибут is\_available таблиці Room показує, чи готовий номер для поселення саме сьогодні. Тобто він заброньований кимось через два тижні, але зараз він вільний. Тим не менш, під датами проживання в таблиці Client ми маємо на увазі ще й дати бронювання.
* Відгук щодо послуги можна залишити навіть не оплативши її, це на випадок поганого сервісу (як з піццою). Послуга оцінюється від 1 до 5 балів за бажанням.
* Скарги, фірми-партнери, бухгалтерські розрахунки і решта сутностей проігноровані, бо інакше схема розростається до космічного масштабу. Тож зараз у результаті отримуємо сферичний готель у вакуумі.
* Це кінцевий варіант бази даних, але можливо доведеться її переробити у разі проблем з іншими етапами.

***Опис моделі***

У нас п'ять основних сутностей: **Клієнт**, **Персонал**, **Послуга**, **Приміщення** і **Керівництво готелю**. Сутності можуть існувати окремо одна від одної, тому всі вони сильні.

Сутність **Клієнт** має чотири атрибути: *номер телефону* (PK), *ім'я*, *дата поселення* та *дата виселення*. Усі атрибути *обов'язкові* для присвоєння значень.

Сутність **Послуга** має три атрибути: *назва послуги* (РК), *вартість* та *керівник*, відповідальний за якість. Усі атрибути *обов'язкові* для присвоєння значень.

Сутність **Керівництво** готелю має три атрибути: *посада* (РК), *ім'я* та *реквізити*. Усі атрибути *обов'язкові* для присвоєння значень.

Сутність **Приміщення** має шість атрибутів: *номер кімнати* (РК), *тип*, *кількість місць*, *розпорядник* (власник), *вартість* та *занятість*. Усі атрибути *обов'язкові* для присвоєння значень.

Сутність **Персонал** має шість атрибутів: *номер страхового полісу* (РК), *ім'я*, *посада*, *зарплата*, *премія або штраф*, *керівник*. Атрибут *премія або штраф* є *необов'язковим* для присвоення значеннь.

Сутності **Клієнт** та **Приміщення** мають два зв'язки: бронювання та оренда. Зв'язки однакові, M:N, де М>=0, N>0.

Сутності **Клієнт** та **Керівництво** готелю мають два зв'язки: скарга та оплата послуг. Зв'язки однакові, M:N, де М>0, N>0 (ці зв'язки не утворюють нові таблиці, але зумовлюють появу атрибутів *rating*, *comment* та *is\_paid* в таблиці **Ordering**).

Сутності **Клієнт** та **Послуга** мають один зв'язок: замовлення. Зв'язок M:N, де М>0, N>0. Він має три власні атрибути: *сплата*, *оцінка* та *коментар*. Атрибути сплата та коментар необов'язкові для присвоєння значень.

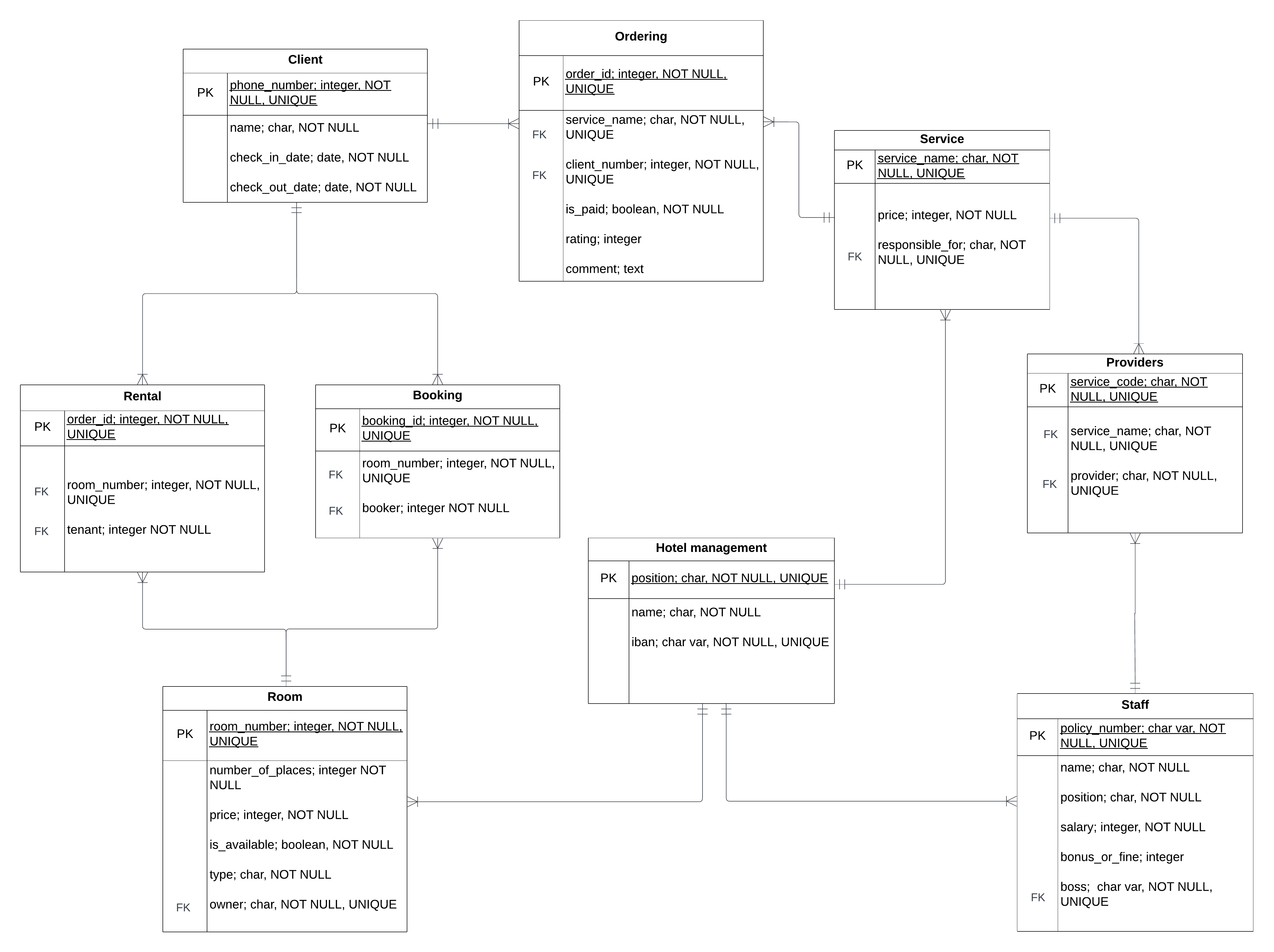
Сутності **Приміщення** та **Керівництво готелю** мають два зв'язки: управління та здача в оренду. Зв'язки однакові, 1:М, де М>0 (зв'язки не мають чіткого вираження в базі даних, але є важливими у функціонуванні готелю).

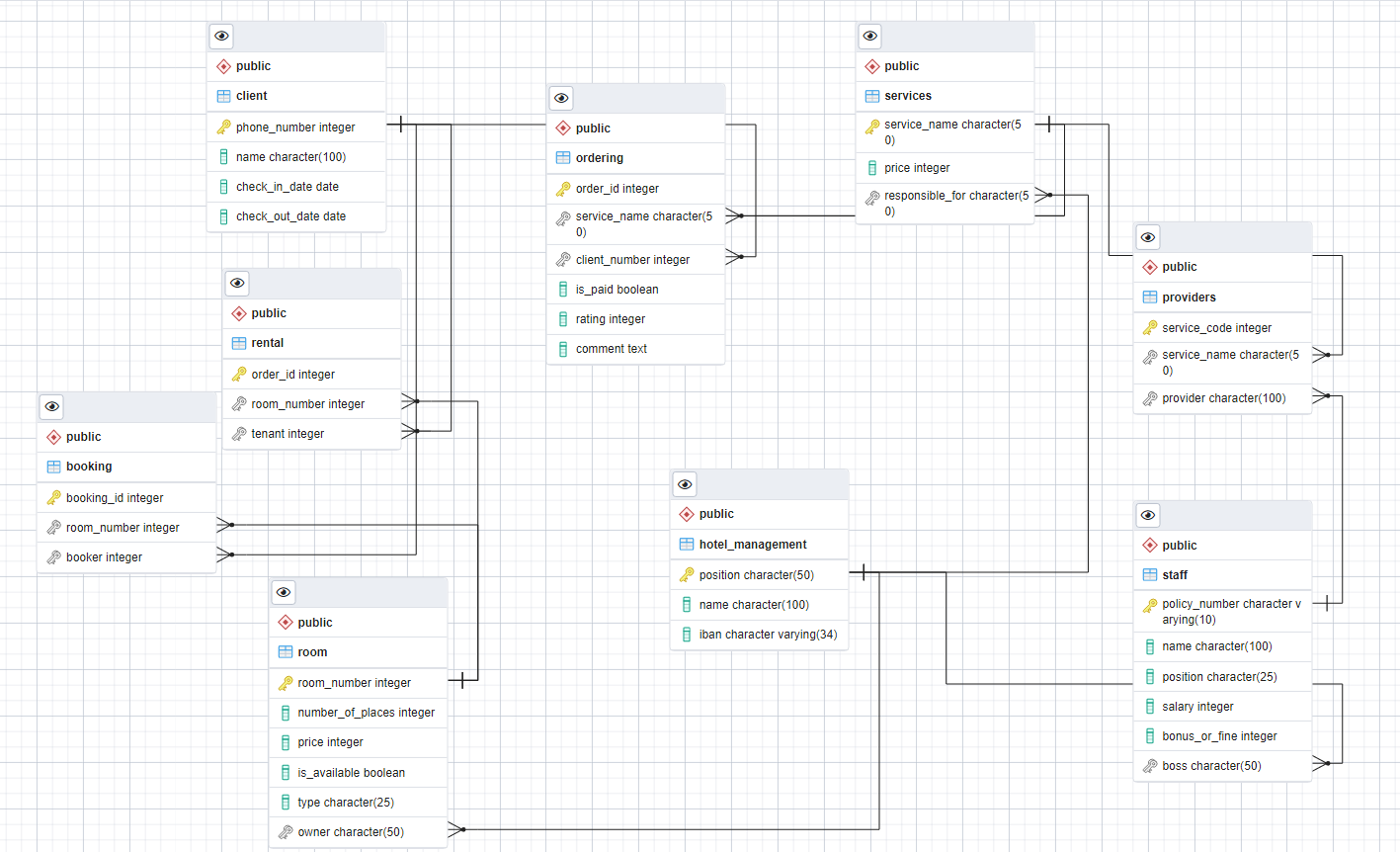
Сутності **Керівництво готелю** та **Персонал** мають три зв'язки: нарахування штрафу/премії, забезпечення соцпакетом та плата зарплати. Зв'язки однакові, 1:М, де М>0 (ці зв'язки не утворюють нові таблиці, але зумовлюють появу атрибутів *policy\_number*, *bonus\_or\_fine* та *salary* в таблиці **Staff**).

Сутності **Керівництво готелю** та **Послуга** мають один зв'язок: контроль якості. Зв'язок 1:М, де М>0 (цей зв'язок не утворює нову таблицю, але зумовлює появу атрибутів *rating*, *comment* та *bonus\_or\_fine* в таблицях *Ordering* та *Staff*).

Сутності **Послуга** та **Персонал** мають один зв'язок: надання послуг. Зв'язок M:N, де М>0, N>0. N>0.

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

*Графічне перетворення*



*Таблиця в pgAdmin 4*

***Перетворення***

Сутність **Клієнт** перетворюємо в таблицю **Client**, сутність **Керівництво готелю** - в **Hotel\_management**, сутність **Приміщення** - в таблицю **Room**, сутність **Персонал** - в таблицю **Staff**, сутність **Послуга** - в таблицю **Services**.

Зв'язок **Клієнт** та **Приміщення** зумовило появу таблиць **Booking** та **Rental** з відповідними зовнішніми ключами. Зв'язок сутностей **Клієнт** та **Послуга** зумовило появу таблиці **Ordering** з відповідними зовнішніми ключами. Зв'язок сутностей **Послуга** та **Персонал** зумовило появу таблиці **Providers**. Зв'язки з таблицею **Hotel\_management** зумовили появу FK *owner* у таблиці **Room**, FK *responsible\_for* у таблиці **Services** та FK *boss* у таблиці **Staff**.

***Опис значень атрибутів***

**Client** — таблиця мешканців готелю або орендаторів спеціальних приміщень та їхніми контактами

*phone\_number* — номер телефону мешканця, PK, записується без +38;

*name* — ПІБ клієнта;

*check\_in\_date* — дата поселення/початку періоду бронювання;

*check\_out\_date* — дата виселення/кінця періоду бронювання.

**Services** — таблиця зі списком послуг та їх вартістю

*service\_name* — назва послуги, РК;

*price* — ціна послуги в гривнях;

*responsible\_for* — посада керівника, відповідального за сервіс. Людина, яка опрацьовує скарги, FK.

**Room** — таблиця зі списком доступних приміщень

*room\_number* — номер кімнати, РК;

*number\_of\_places* — кількість житлових місць або місць для відвідувачів;

*type* — тип кімнати, відповідний до її використання;

*owner* — керівник, у розпорядженні якого знаходиться приміщення, FK;

*is\_available* — показує, чи зайнята кімната станом на сьогодні.

**Hotel\_management** — таблиця керівників готелю, кожен з яких відповідає за свою галузь, послуги та приміщення

*position* — назва посади керівника, РК;

*name* — ПІБ керівника;

*iban* — номер рахунку для сплати послуг, які надає керівник.

**Staff** — таблиця персоналу, підпорядкованого керівництву готелю

*policy\_number* — номер страхового полісу, РК;

*bonus\_or\_fine* — поле для нарахування штрафу або премії за якість роботи;

*salary* — стандартна зарплата працівника без штрафів та премій;

*name* — ПІБ працівника;

*position* — посада;

*boss* — керівник, якому підпорядковується працівник, FK.

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

***Функціональні залежності***

Client (**phone\_number**, name, check\_in\_date, check\_out\_date)

**phone\_number** -> name, check\_in\_date, check\_out\_date

**phone\_number** -> name

**phone\_number** -> check\_in\_date

**phone\_number** -> check\_out\_date

Services (**service\_name**, price, responsible\_for)

**service\_name** -> price, responsible\_for

**service\_name** -> responsible\_for

**service\_name** -> price

Room (**room\_number**, number\_of\_places, type, owner, is\_available)

**room\_number** -> number\_of\_places, type, owner, is\_available

**room\_number** -> number\_of\_places

**room\_number** -> type

**room\_number** -> owner

**room\_number** -> is\_available

Hotel\_management (**position**, name, iban)

**position** -> name, iban

**position** -> name

**position** -> iban

Staff (**policy\_number**, bonus\_or\_fine, salary, name, position, boss)

**policy\_number** -> bonus\_or\_fine, salary, name, position, boss

**policy\_number** -> bonus\_or\_fine

**policy\_number** -> name

**policy\_number** -> salary

**policy\_number** -> position

**policy\_number** -> boss

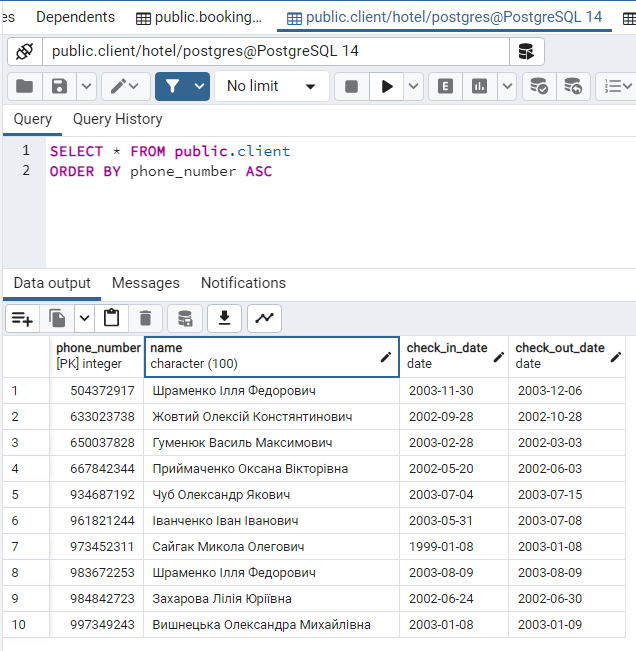
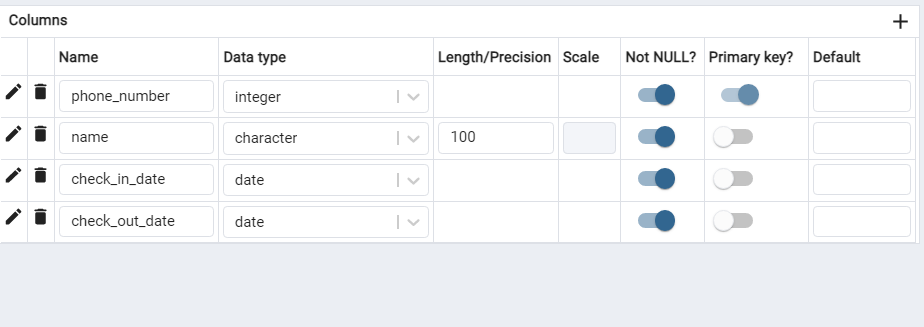
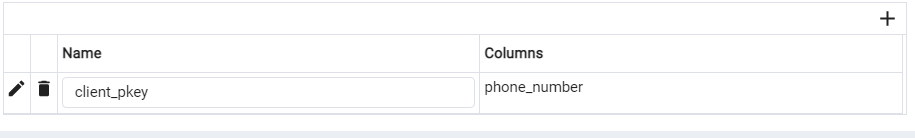
***Нормалізація***

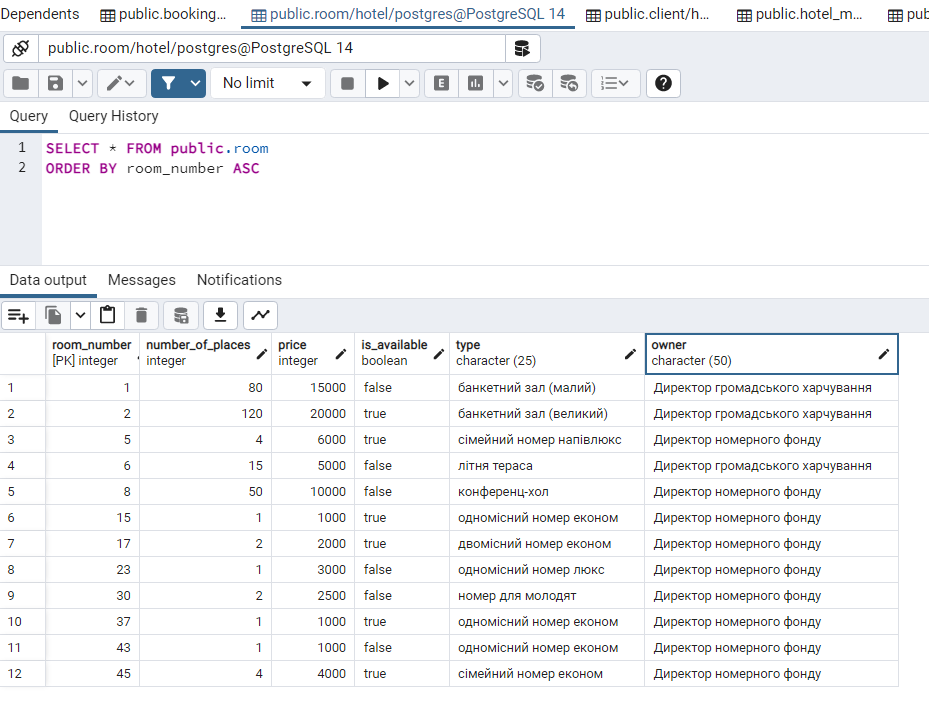
Схема цієї бази даних відповідає нормам 1НФ, бо кожна таблиця має основний ключ, всі рядки унікальні, а всі атрибути - атомарні.

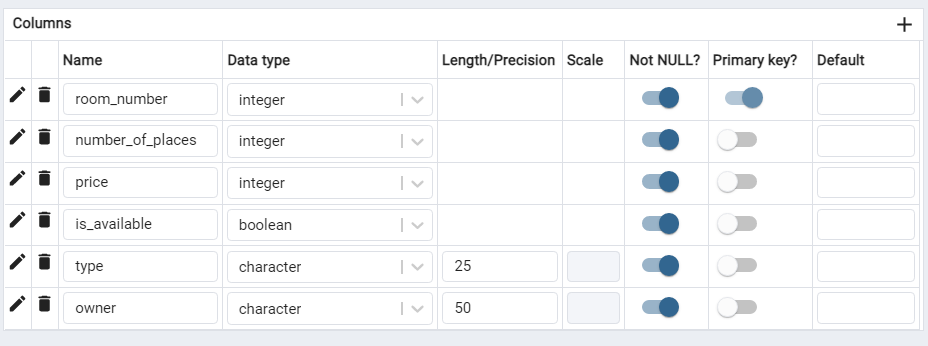
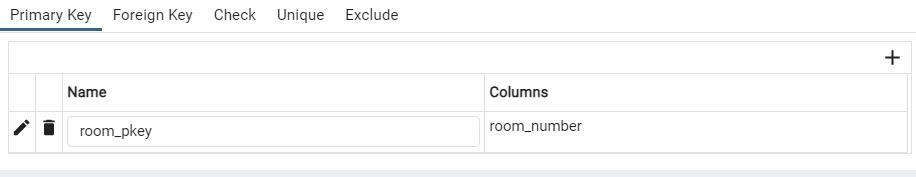
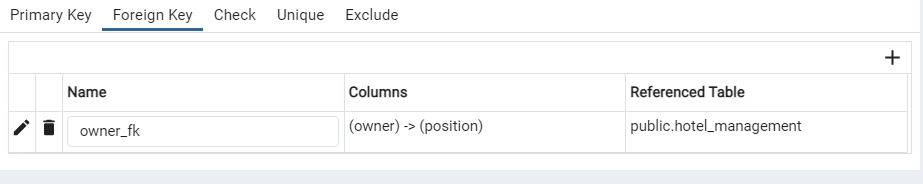
Схема цієї бази даних відповідає нормам 2НФ, бо схема всі її неключові атрибути загалом функціонально залежні від одного ключа і виконується 1НФ.

Схема цієї бази даних відповідає нормам 3НФ, бо немає функціональних залежностей не ключових атрибутів від ключових і виконується 2НФ.

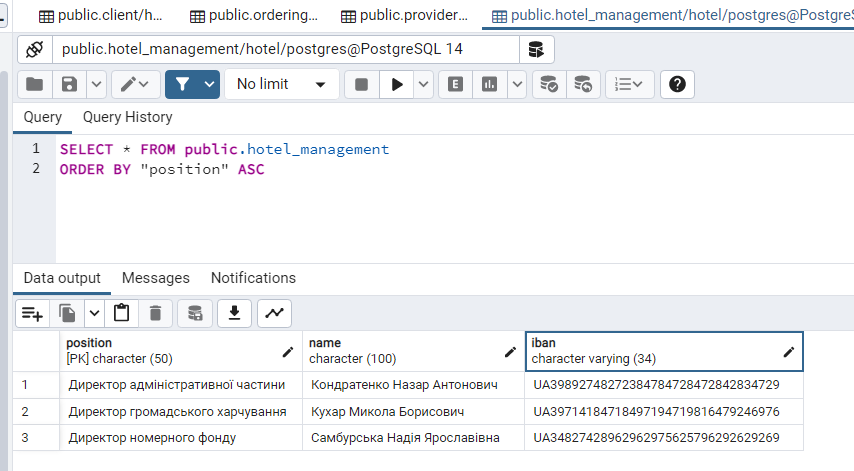
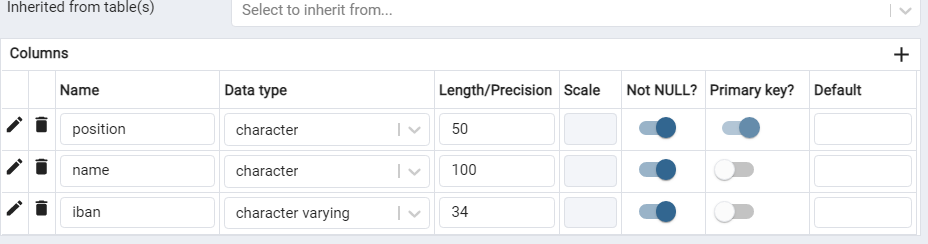
**Етап 4** - наведення скріншотів роботи з PostgreSQL та pgAdmin 4

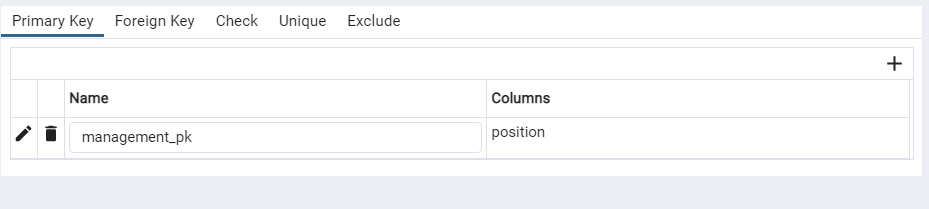
*Список усіх таблиць у боковому меню*  
  
*Таблиця Client*  
  
  


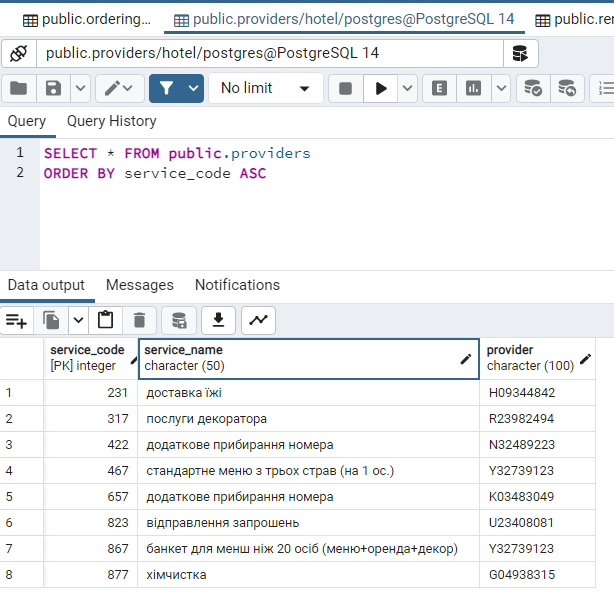
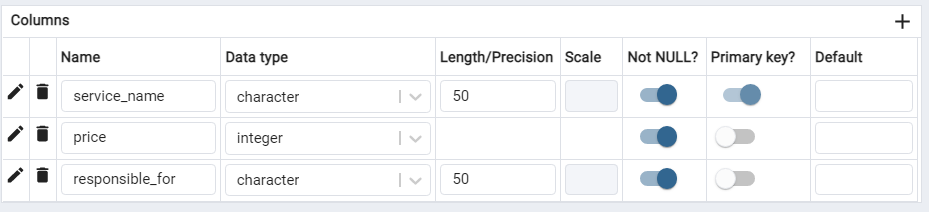
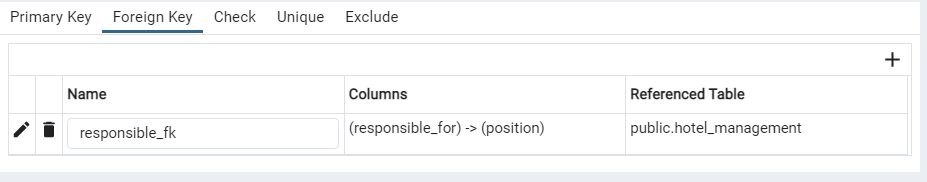
*Таблиця Room*  


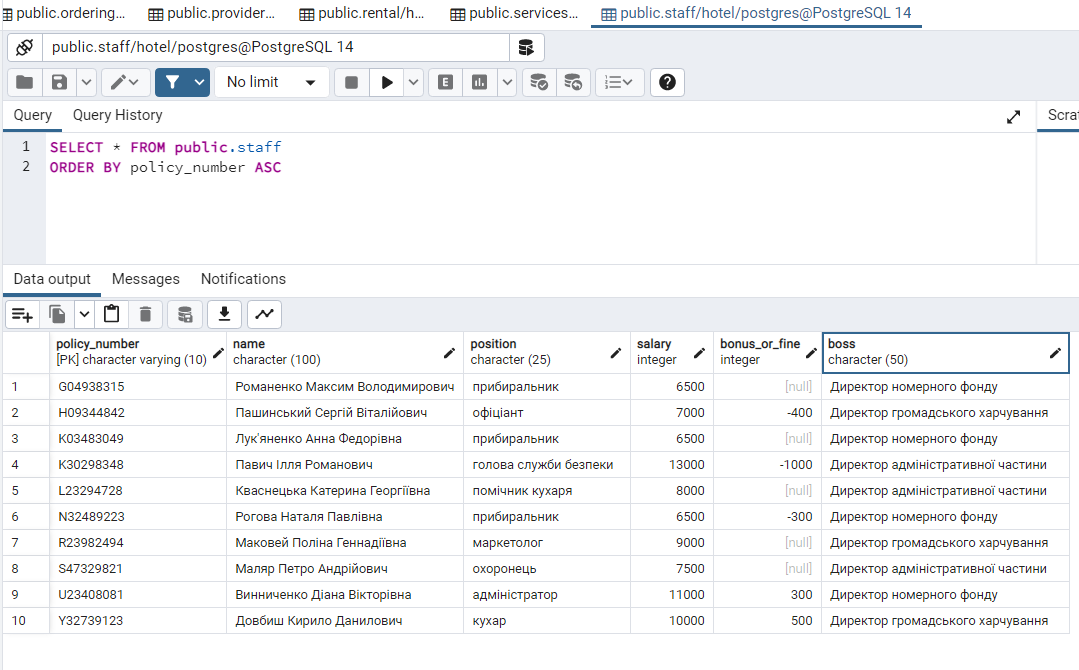
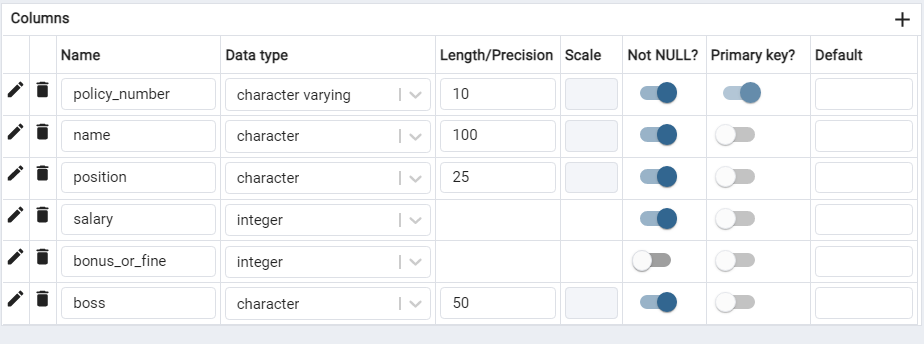
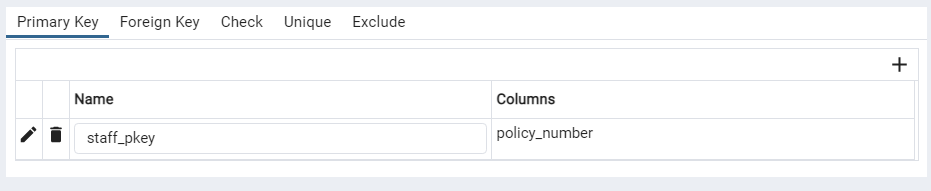
  
  


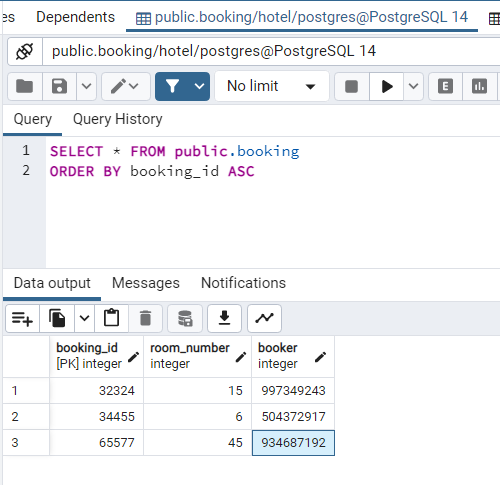
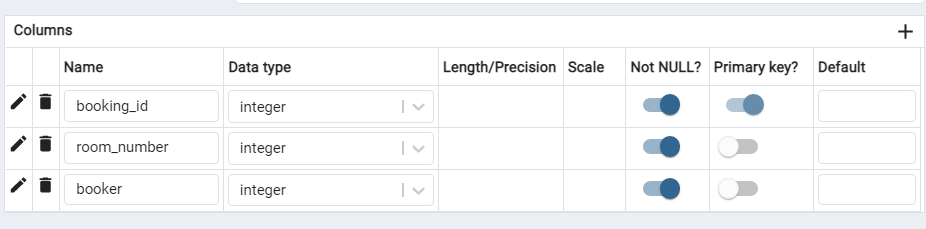
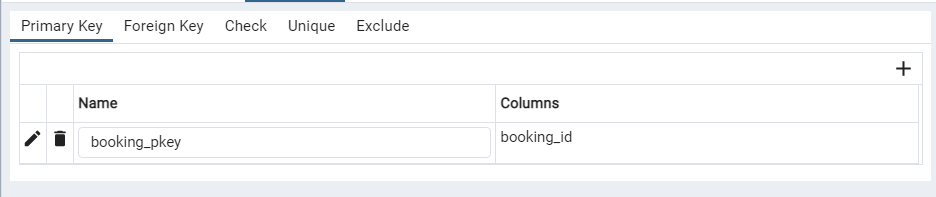
*Таблиця Hotel\_management*

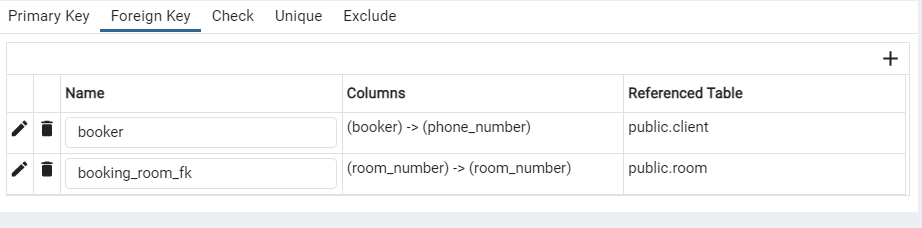
  
  


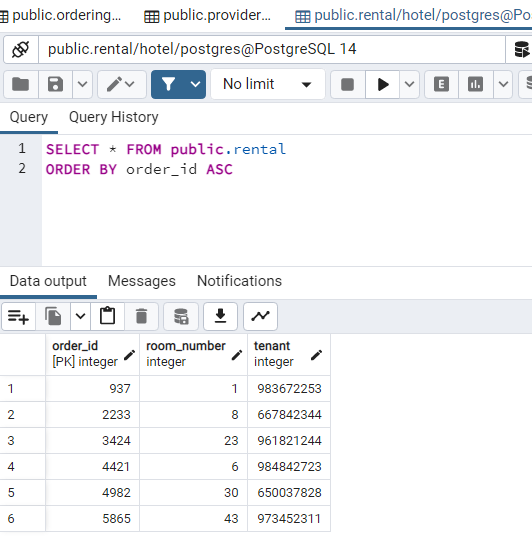
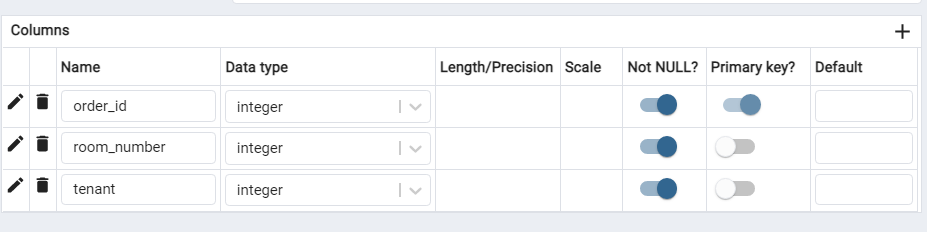


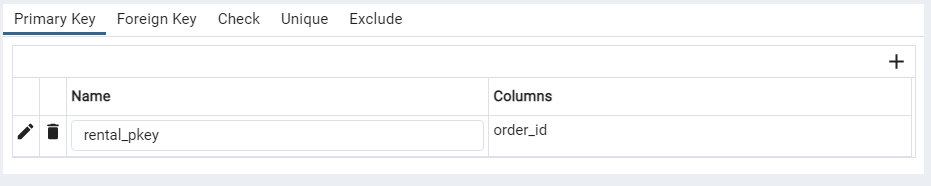
*Таблиця Services*  
  
  
  


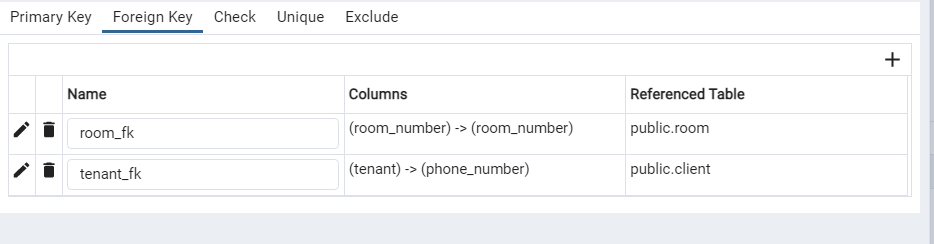
*Таблиця Staff*  
  
  
  

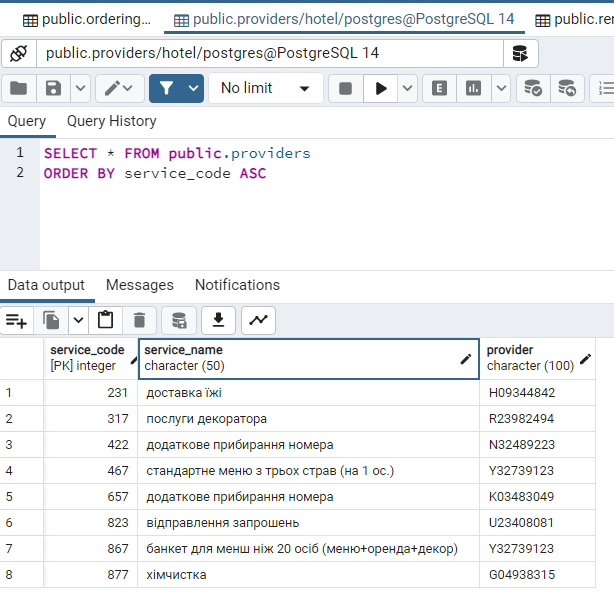
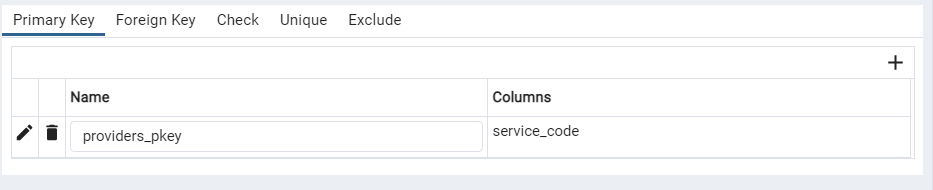
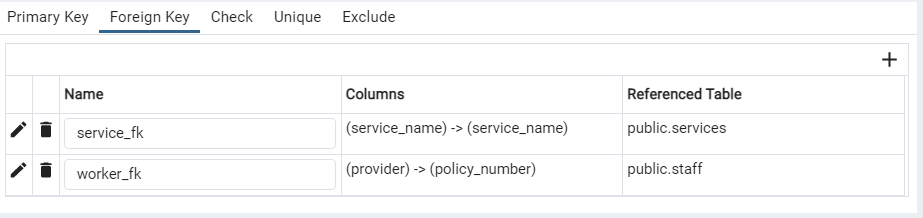

*Таблиця Booking*  
  
  


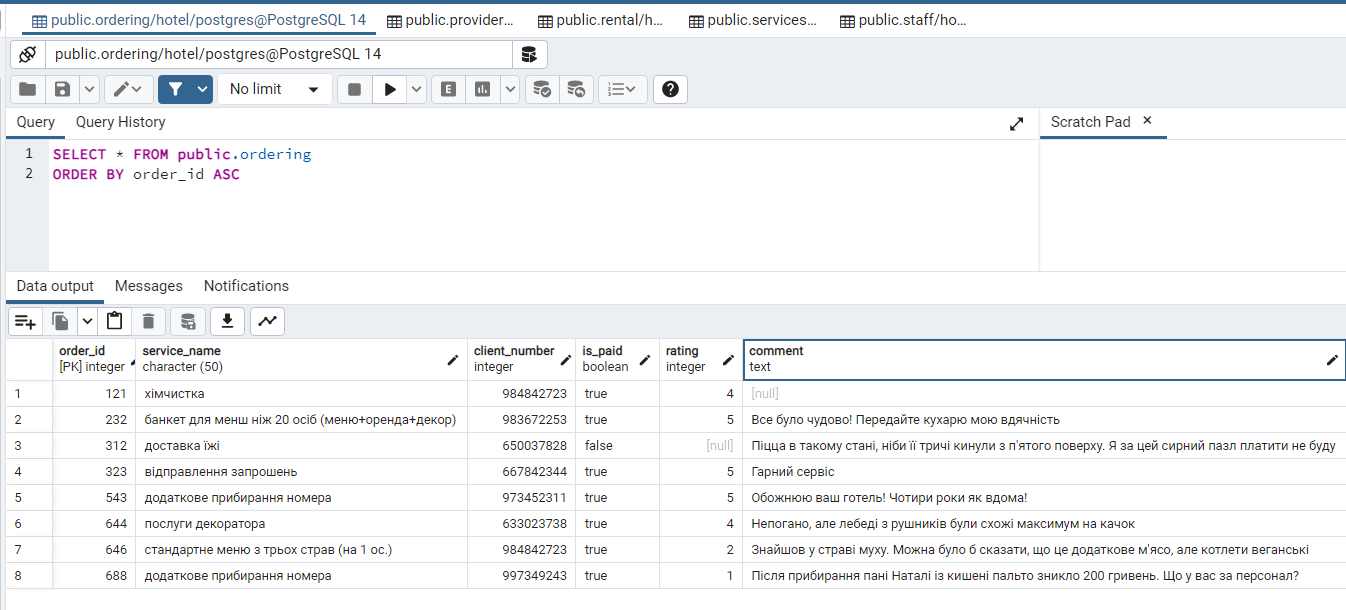


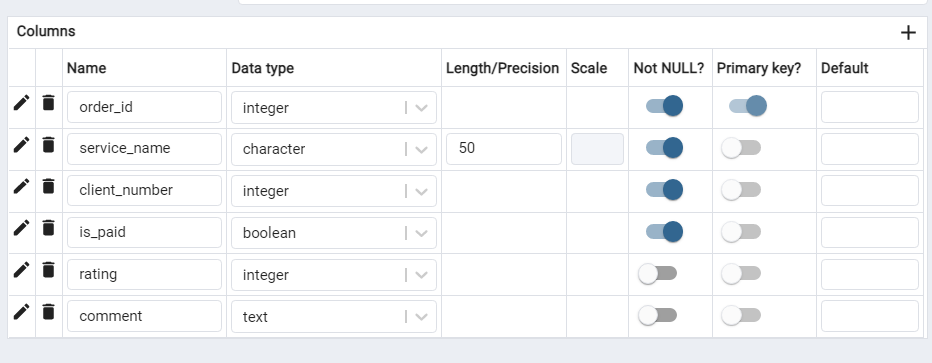
*Таблиця Rental*  
  


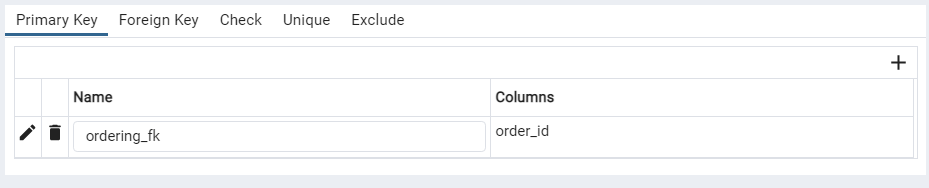


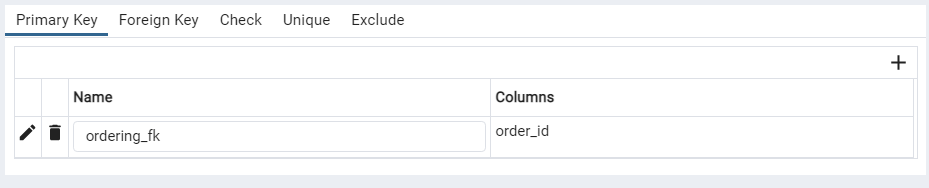
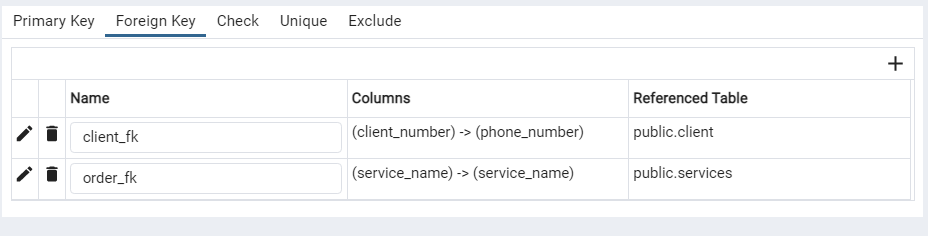


*Таблиця Providers*  
  
  
  


*Таблиця Ordering* 





Github: <https://github.com/reginakasteen/data-bases>Telegram: @rkasteen