

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»

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

Виконав: студент III курсу ФПМ групи КВ-21 Юдін Дмитро

Оцінка:

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 (3HФ).
- 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

Завлання №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».

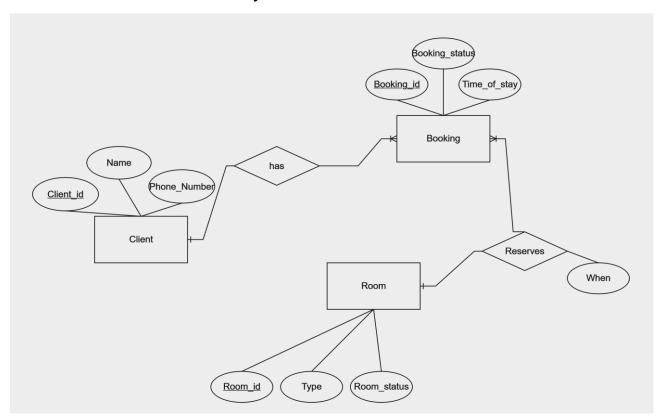


Рисунок 1. ER-діаграма побудована за нотацією Чена

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.

Завлання №2

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

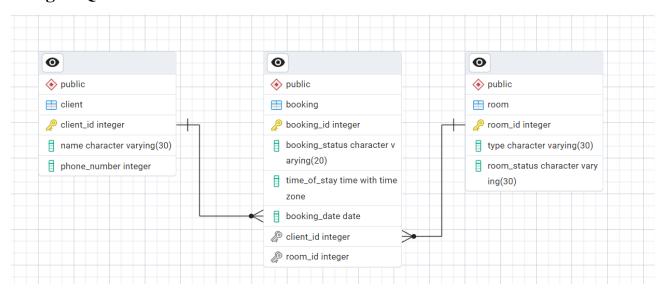


Рисунок 2. Схема бази даних у pgAdmin4

Опис процесу перетворення:

Сутність Client перетворилась на таблицю client. В таблиці роль первинного ключа викону ϵ ідентифікатор client_id, а атрибутами ϵ : name та phone_number.

Сутність Booking стала таблицею booking. Первинний ключ у цій таблиці — booking_id, а зовнішнім ключем ϵ room_id та client_id для реалізації зв'язку між booking-room та booking-client відповідно. Ця таблиця ма ϵ 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 ϵ type та room_status.

Завдання №3

Функціональні залежності:

1. Client (client_id, name, phone_number).

```
client_id \rightarrow name
client_id \rightarrow phone_number
```

2. Booking (booking_id, booking_status, time_of_stay).

booking_id → booking_status booking_id → time_of_stay

3. Room (room_id, type, room_status).

```
room\_id \rightarrow type

room\_id \rightarrow room\_status
```

Аби схема бази даних була 1НФ потрібно щоб:

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

Схема відповідає $1H\Phi$. 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:

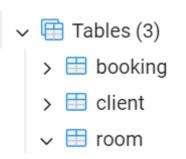
- База даних відповідала 2НФ
- Всі атрибути залежать тільки від первинного ключа, а не від інших атрибутів.

Схема відповідає ЗНФ. Кожен неключевий атрибут не ϵ транзитивно залежним від іншого неключового атрибута.

Завлання №4

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

Таблиці бази даних:



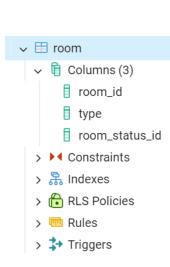
> Constraints
> Indexes
> Indexes
> RLS Policies
> Rules
> Triggers

> Iclient

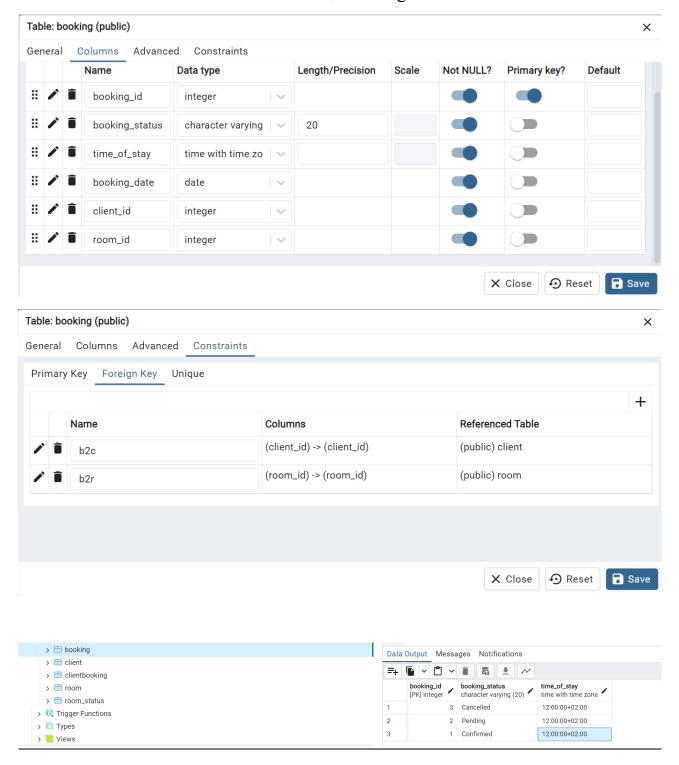
Columns (3)
Iclient_id
Iname
Iphone_number
> Constraints
> Indexes
> RLS Policies

> m Rules

> **Triggers**



Таблиця booking



Таблиця client

