

**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 дисциплін

«Бази даних та засоби управління»

Група: КВ-13

Виконав: Соболев Т.Ю.

Оцінка:

Київ-2023

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 кожен з таблиць засобами pgAdmin 4.

Концептуальна модель предметної області “Платформа для оренди та бронювання спортивних об'єктів.”

В концептуальній моделі предметної області “Платформа для оренди та бронювання спортивних об'єктів.” (Рисунок 1) виділяються наступні сутності та зв'язки між ними:

1. Сутність “Client” з атрибутами: id, name, email;
2. Сутність “Sport_object” з атрибутами: id, name, price;
3. Зв'язок “Reservation” з атрибутами: id, client_id, object_id, booking_start_date, booking_end_date;
4. Сутність “Company” з атрибутами: id, name;

Один клієнт може оформити декілька підтверджень на оренду спортивних об'єктів, а кожен факт підтвердження оренди може бути наданий одному клієнту (зв'язок 1:N).

Один спортивний об'єкт може бути орендований декількома людьми на різній час, тому до одного об'єкта може належати одразу декілька фактів його оренди (зв'язок 1:N).

Виходячи з вищенаведеного одна людина може орендувати декілька спортивних об'єктів, а один об'єкт можуть бути орендовані різними людьми. Тобто маємо зв'язок багато до багатьох N:M.

Також один клієнт може бути у одній компанії, а компаній може бути декілька (зв'язок 1:N).

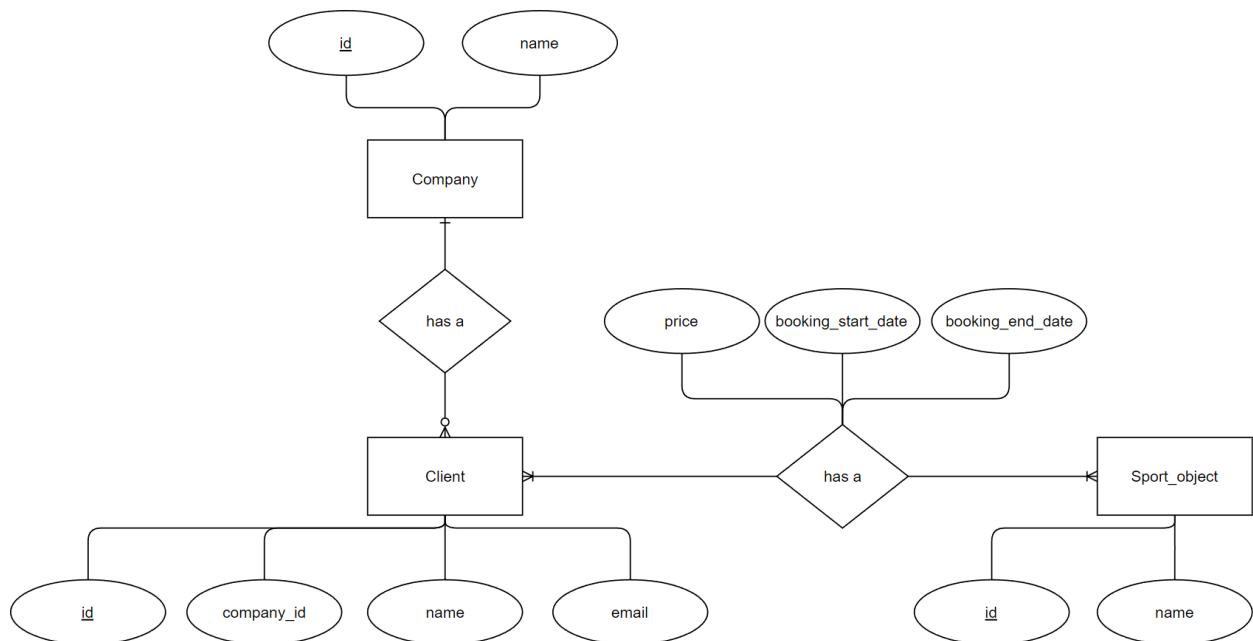


Рисунок 1 – Концептуальна модель предметної області.

Логічна модель (схема) БД “Платформа для оренди та бронювання спортивних об’єктів”.

В логічній моделі (Рисунок 2):

1. Сутність “Client” перетворена в таблицю “Client”;
2. Сутність “Sport_object” перетворена в таблицю “Sport_object”;
3. Зв’язок “Client” та “Sport_object” перетворений в таблицю “Reservation”;
4. Сутність “Company” перетворена в таблицю “Company”;

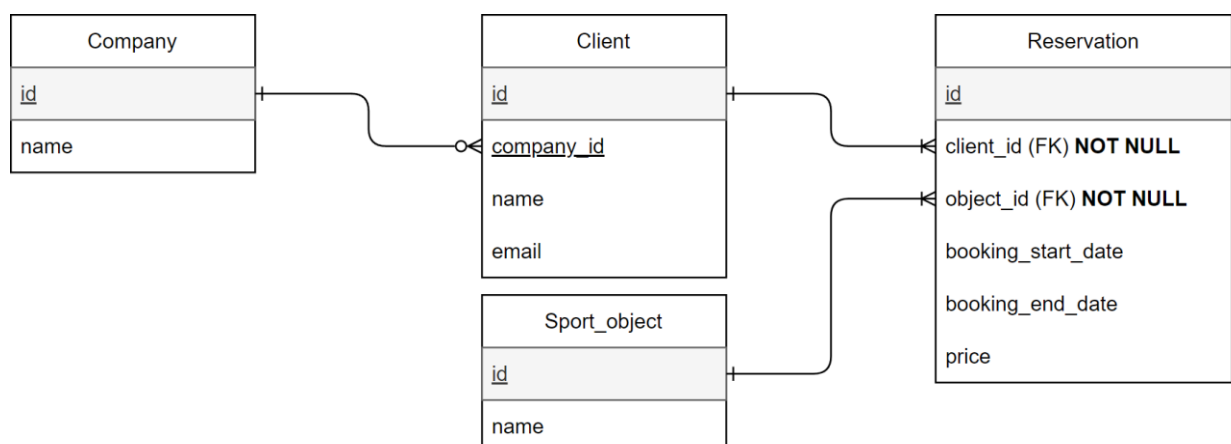


Рисунок 2 - Логічна модель предметної області

Опис структури БД

Сутність	Атрибут	Тип
Client (інформація про клієнта)	id - унікальний ID клієнта	Число
	name – повне ім'я клієнта	Текст
	email – пошта клієнта	Текст
	company_id — зовнішній ключ	Число
Sport_object (інформація про об'єкт)	id - унікальний ID об'єкта	Число
	name — ім'я об'єкту	Текст
Reservation (інформація про оренду)	id - унікальний ID оренди	Число
	client_id — зовнішній ключ	Число
	object_id — зовнішній ключ	Число
	booking_start_date – час початку оренди	Дата
	booking_end_date – час кінця оренди	Дата
	price — ціна	Число
Company (інформація про компанію)	id - унікальний ID компанії	Число
	name – повне ім'я компанії	Текст

Відповідність схеми БД до третьої нормальної форми

Функціональні залежності схеми.

Сутність “Client”:

- id → name (атрибут “name” функціонально залежний від “id”)
- id → email (атрибут “email” функціонально залежний від “id”)

Сутність “Sport_object”:

- id → name (атрибут “name” функціонально залежний від “id”)

Сутність “Reservation”:

- $id \rightarrow booking_start_date$ (атрибут “booking_start_date” функціонально залежний від “id”)
- $id \rightarrow booking_end_date$ (атрибут “booking_start_date” функціонально залежний від “id”)
- $id \rightarrow price$ (атрибут “price” функціонально залежний від “id”)

Сутність “Company”:

- $id \rightarrow name$ (атрибут “name” функціонально залежний від “id”)

З вищевказаних функціональних залежностей

- **НФ1 (перша нормальна форма):** Схема відповідає НФ1, оскільки всі атрибути є атомарними, і у всіх таблицях є первинні ключі (вони всі названі однаково “id”).
- **НФ2 (друга нормальна форма):** Схема відповідає НФ2, оскільки всі атрибути які не є ключами, функціонально залежні від всіх часткових ключів. У нашому випадку, всі атрибути таблиць функціонально залежні від ключових атрибутів.
- **НФ3 (третя нормальна форма):** Схема відповідає НФ3, оскільки всі атрибути які не є ключами, функціонально залежні від первинного ключа таблиці, а не від інших неключових атрибутів.

Отже дана БД відповідає умовам трьом нормальних форм.

Структура БД “Платформа для оренди та бронювання спортивних об'єктів” в pgAdmin 4

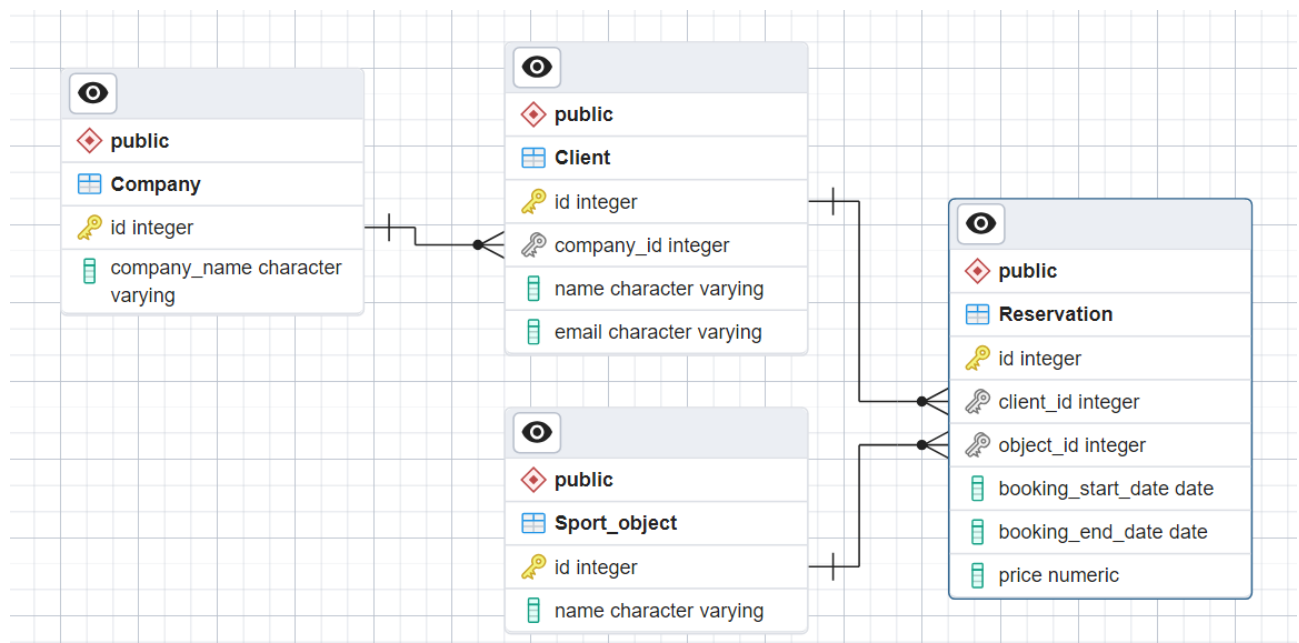


Рисунок 3 - Структура предметної області “Платформа для оренди та бронювання спортивних об'єктів” в pgAdmin 4

Client

- Columns (4)
 - id
 - company_id
 - name
 - email
- Constraints (2)
 - Client_company_id_fkey
 - Client_pkey
- Indexes
- RLS Policies
- Rules
- Triggers

Sport_object

- Columns (2)
 - id
 - name
- Constraints (1)
 - Sport_object_pkey
- Indexes
- RLS Policies
- Rules
- Triggers

Company

- Columns (2)
 - id
 - company_name
- Constraints (1)
 - Company_pkey
- Indexes
- RLS Policies
- Rules
- Triggers

Reservation

- Columns (6)
 - id
 - client_id
 - object_id
 - booking_start_date
 - booking_end_date
 - price
- Constraints (3)
 - Reservation_client_id_fkey
 - Reservation_object_id_fkey
 - Reservation_pkey
- Indexes
- RLS Policies
- Rules
- Triggers

> Operators				
> Procedures				
> Sequences				
> Tables (4)				
> Client				
> Company				
> Reservation				
> Sport_object				

	id [PK] integer	company_id integer	name character varying	email character varying
1	1	1	Данііл	[null]
2	2	1	Олег	[null]
3	3	[null]	Богдан	[null]

Github - <https://github.com/balalay4ik/DB1>