

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені Ігоря Сікорського»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

# Кафедра системного програмування та спеціалізованих комп’ютерних систем

**Лабораторна робота №1**

з дисципліни

**«Бази даних і засоби управління»**

Група: КВ-22

Виконала: Гончар В.

Оцінка:

Київ – 2024

### 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 кожну з таблиць засобами pgAdmin 4.Модель «сутність-зв’язок» предметної галузі “Система управління клієнтами для компаній”

#### Основні сутності та їх атрибути:

1. **Компанія**:
   * Атрибути: Id, Назва, Власник, Країна.
2. **Клієнт**:
   * Атрибути: Id, Ім’я, Вік, Стать.
3. **Покупка**:
   * Атрибути: Id, Опис, Дата, Сума.
4. **Платіжна система**:
   * Атрибути: Id, Назва, Вебсайт.

#### Зв’язки між сутностями:

* **Компанія ↔ Клієнт** (зв’язок N:M):
  + Одна компанія може обслуговувати багатьох клієнтів.
  + Один клієнт може обслуговуватись у багатьох компаніях.
* **Клієнт ↔ Покупка** (зв’язок 1:N):
  + Один клієнт може здійснити багато покупок.
  + Одна покупка належить лише одному клієнту.
* **Компанія ↔ Покупка** (зв’язок 1:N):
  + Одна компанія може мати багато покупок.
  + Одна покупка завжди належить лише одній компанії.
* **Покупка ↔ Платіжна система** (зв’язок 1:N):
  + Одна покупка оплачується лише однією платіжною системою.
  + Один платіжний сервіс може приймати оплати багатьох покупок.

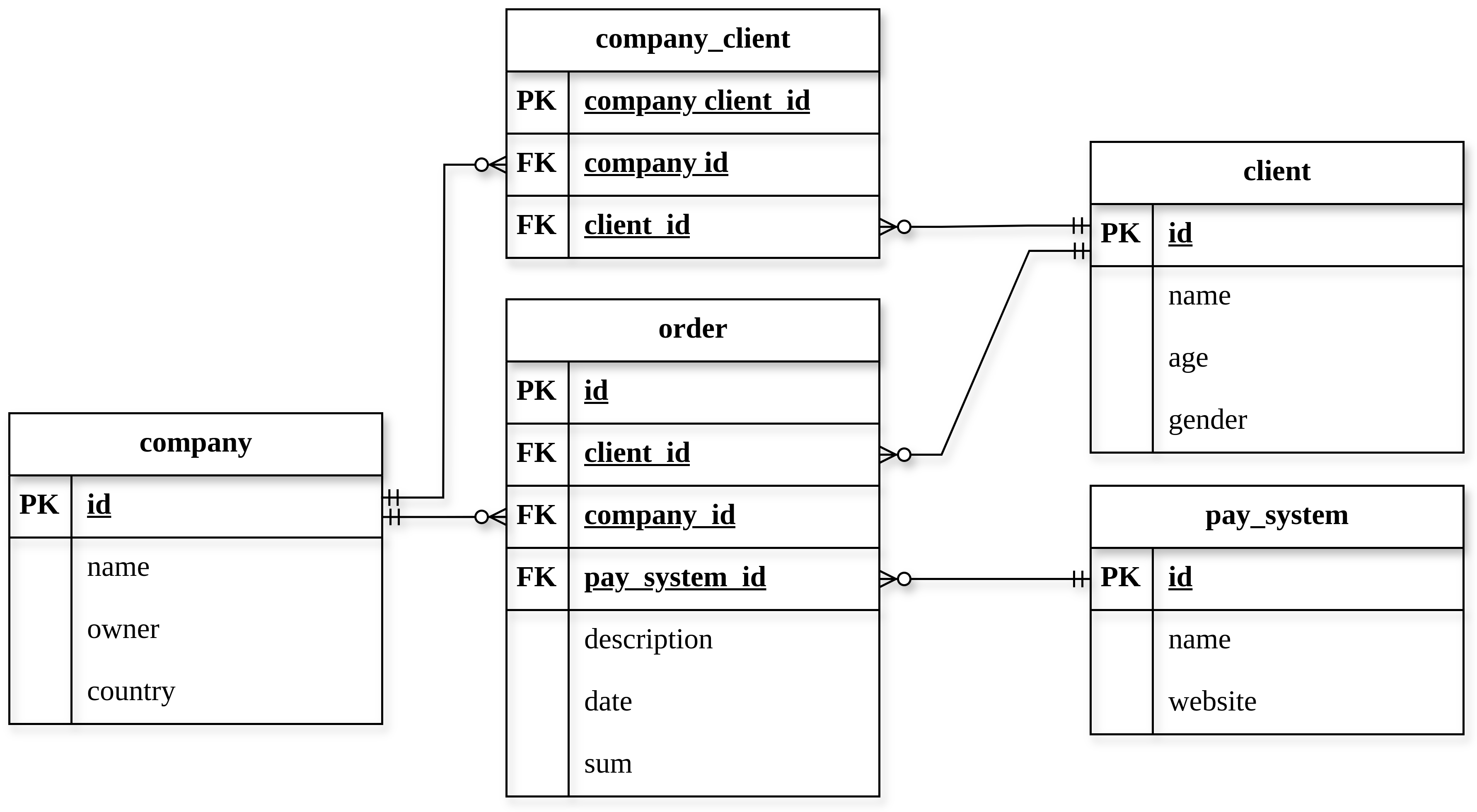


Рис. 1 - ER-діаграма. Нотація «Пташиної лапки» (Crow’s foot)

### Логічна модель (схема) БД

#### 1. Таблиця ****company****

* **Attributes**:
  + id (PK) – Унікальний ідентифікатор компанії.
  + name – Назва компанії.
  + owner – Власник компанії.
  + country – Країна розташування компанії.

#### 2. Таблиця ****client****

* **Attributes**:
  + id (PK) – Унікальний ідентифікатор клієнта.
  + name – Ім’я клієнта.
  + age – Вік клієнта.
  + gender – Стать клієнта.

#### 3. Таблиця ****order****

* **Attributes**:
  + id (PK) – Унікальний ідентифікатор замовлення.
  + client\_id (FK) – Зовнішній ключ, який посилається на таблицю **client**.
  + company\_id (FK) – Зовнішній ключ, який посилається на таблицю **company**.
  + pay\_system\_id (FK) – Зовнішній ключ, який посилається на таблицю **pay\_system**.
  + description – Опис замовлення.
  + date – Дата замовлення.
  + sum – Сума замовлення.

#### 4. Таблиця ****pay\_system****

* **Attributes**:
  + id (PK) – Унікальний ідентифікатор платіжної системи.
  + name – Назва платіжної системи.
  + website – Вебсайт платіжної системи.

#### 5. Таблиця ****company\_client****

* **Attributes**:
  + company\_client\_id (PK) – Унікальний ідентифікатор зв’язку.
  + company\_id (FK) – Зовнішній ключ, який посилається на таблицю **company**.
  + client\_id (FK) – Зовнішній ключ, який посилається на таблицю **client**.

### Опис процесу перетворення моделі у схему бази даних

#### 1. Сутність company перетворена у таблицю company:

* **Створення таблиці**:

CREATE TABLE company (

id SERIAL PRIMARY KEY,

name VARCHAR(255) NOT NULL,

owner VARCHAR(255) NOT NULL,

country VARCHAR(100) NOT NULL

);

* Таблиця company містить атрибути id (PK), name, owner та country.

#### 2. Сутність client перетворена у таблицю client:

* **Створення таблиці**:

CREATE TABLE client (

id SERIAL PRIMARY KEY,

name VARCHAR(255) NOT NULL,

age INT NOT NULL,

gender VARCHAR(10) NOT NULL

);

* Таблиця client містить атрибути id (PK), name, age та gender.

#### 3. Сутність order перетворена у таблицю order:

* **Створення таблиці**:

CREATE TABLE "order" (

id SERIAL PRIMARY KEY,

client\_id INT NOT NULL,

company\_id INT NOT NULL,

pay\_system\_id INT NOT NULL,

description TEXT,

date DATE NOT NULL,

sum NUMERIC(10, 2) NOT NULL,

FOREIGN KEY (client\_id) REFERENCES client (id),

FOREIGN KEY (company\_id) REFERENCES company (id),

FOREIGN KEY (pay\_system\_id) REFERENCES pay\_system (id)

);

* Таблиця order містить атрибути id (PK), client\_id (FK), company\_id (FK), pay\_system\_id (FK), description, date та sum.
* Зовнішні ключі (client\_id, company\_id, pay\_system\_id) реалізують відношення 1:N із відповідними таблицями.

#### 4. Сутність pay\_system перетворена у таблицю pay\_system:

* **Створення таблиці**:

CREATE TABLE pay\_system (

id SERIAL PRIMARY KEY,

name VARCHAR(255) NOT NULL,

website VARCHAR(255) NOT NULL UNIQUE

);

* Таблиця pay\_system містить атрибути id (PK), name та website.
* Атрибут website визначений як UNIQUE, щоб гарантувати унікальність.

#### 5. Зв’язок company - client (N:M) призвів до створення таблиці company\_client:

* **Створення таблиці**:

CREATE TABLE company\_client (

company\_client\_id SERIAL PRIMARY KEY,

company\_id INT NOT NULL,

client\_id INT NOT NULL,

FOREIGN KEY (company\_id) REFERENCES company (id),

FOREIGN KEY (client\_id) REFERENCES client (id)

);

* Таблиця company\_client містить атрибути company\_client\_id (PK), company\_id (FK) та client\_id (FK).
* Ця таблиця реалізує зв’язок N:M між таблицями company та client.

### Нормалізація схеми бази даних до третьої нормальної форми (3НФ)

#### 1. Таблиця company

* **Атрибути**: id (PK), name, owner, country.

1. **Перша нормальна форма (1НФ)**:
   * Всі значення атомарні.
   * Таблиця має унікальний ідентифікатор id як первинний ключ.
   * **Відповідає 1НФ.**
2. **Друга нормальна форма (2НФ)**:
   * Таблиця має простий первинний ключ.
   * Всі неключові атрибути (наприклад, name, owner, country) залежать від первинного ключа id.
   * **Відповідає 2НФ.**
3. **Третя нормальна форма (3НФ)**:
   * У таблиці немає транзитивних залежностей. Атрибути не залежать один від одного, лише від первинного ключа id.
   * **Відповідає 3НФ.**

#### 2. Таблиця client

* **Атрибути**: id (PK), name, age, gender.

1. **Перша нормальна форма (1НФ)**:
   * Всі атрибути мають атомарні значення.
   * Таблиця має унікальний первинний ключ id.
   * **Відповідає 1НФ.**
2. **Друга нормальна форма (2НФ)**:
   * Всі неключові атрибути залежать лише від первинного ключа id.
   * **Відповідає 2НФ.**
3. **Третя нормальна форма (3НФ)**:
   * Атрибути не залежать один від одного, лише від id.
   * **Відповідає 3НФ.**

#### 3. Таблиця order

* **Атрибути**: id (PK), client\_id (FK), company\_id (FK), pay\_system\_id (FK), description, date, sum.

1. **Перша нормальна форма (1НФ)**:
   * Всі значення атомарні, таблиця має унікальний первинний ключ id.
   * **Відповідає 1НФ.**
2. **Друга нормальна форма (2НФ)**:
   * Всі неключові атрибути залежать від id (наприклад, description, date, sum).
   * Зовнішні ключі забезпечують зв’язок із відповідними таблицями.
   * **Відповідає 2НФ.**
3. **Третя нормальна форма (3НФ)**:
   * Усі атрибути залежать лише від id. Транзитивних залежностей немає.
   * **Відповідає 3НФ.**

#### 4. Таблиця pay\_system

* **Атрибути**: id (PK), name, website.

1. **Перша нормальна форма (1НФ)**:
   * Всі значення атомарні.
   * Таблиця має унікальний первинний ключ id.
   * **Відповідає 1НФ.**
2. **Друга нормальна форма (2НФ)**:
   * Всі атрибути залежать лише від первинного ключа id.
   * **Відповідає 2НФ.**
3. **Третя нормальна форма (3НФ)**:
   * У таблиці немає транзитивних залежностей. Наприклад, website залежить тільки від id.
   * **Відповідає 3НФ.**

#### 5. Таблиця company\_client

* **Атрибути**: company\_client\_id (PK), company\_id (FK), client\_id (FK).

1. **Перша нормальна форма (1НФ)**:
   * Всі значення атомарні.
   * Таблиця має унікальний первинний ключ company\_client\_id.
   * **Відповідає 1НФ.**
2. **Друга нормальна форма (2НФ)**:
   * Всі атрибути залежать лише від первинного ключа company\_client\_id.
   * Зовнішні ключі забезпечують зв’язок із таблицями company та client.
   * **Відповідає 2НФ.**
3. **Третя нормальна форма (3НФ)**:
   * Усі атрибути залежать тільки від company\_client\_id. Транзитивних залежностей немає.
   * **Відповідає 3НФ.**

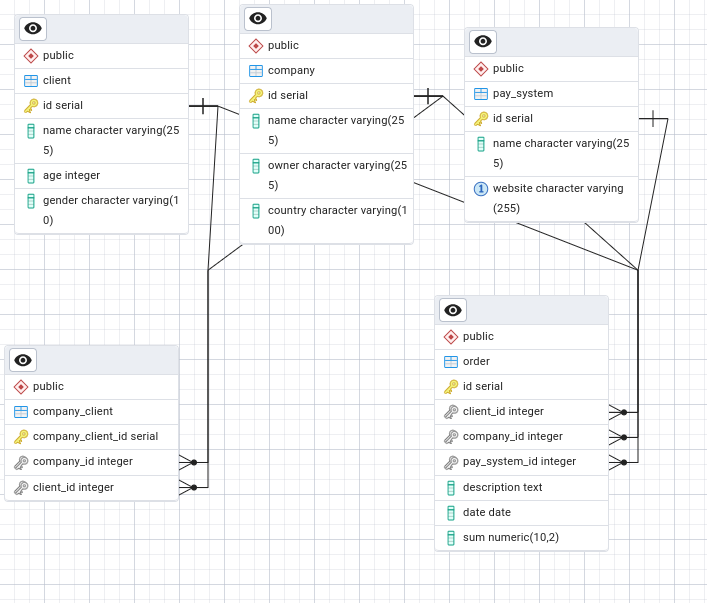


Рис. 2 - Логічна модель предметної області

### Внесення даних у кожну таблицю засобами pgAdmin 4

#### Таблиця company

INSERT INTO company (name, owner, country) VALUES

('Apple Inc.', 'Tim Cook', 'USA'),

('Microsoft', 'Satya Nadella', 'USA'),

('Tesla', 'Elon Musk', 'USA'),

('Toyota', 'Akio Toyoda', 'Japan'),

('BMW', 'Oliver Zipse', 'Germany');

#### Таблиця client

INSERT INTO client (name, age, gender) VALUES

('John Doe', 30, 'Male'),

('Jane Smith', 25, 'Female'),

('Michael Johnson', 40, 'Male'),

('Emily Davis', 35, 'Female'),

('Chris Brown', 29, 'Male');

#### Таблиця pay\_system

INSERT INTO pay\_system (name, website) VALUES

('PayPal', 'https://www.paypal.com'),

('Stripe', 'https://www.stripe.com'),

('Square', 'https://www.squareup.com'),

('Visa', 'https://www.visa.com'),

('MasterCard', 'https://www.mastercard.com');

#### Таблиця order

INSERT INTO "order" (client\_id, company\_id, pay\_system\_id, description, date, sum) VALUES

(1, 1, 1, 'Purchase of MacBook Pro', '2024-11-20', 2500.00),

(2, 2, 2, 'Purchase of Microsoft Office', '2024-11-21', 150.00),

(3, 3, 3, 'Purchase of Tesla Model 3', '2024-11-22', 45000.00),

(4, 4, 4, 'Purchase of Toyota Corolla', '2024-11-23', 20000.00),

(5, 5, 5, 'Purchase of BMW X5', '2024-11-24', 60000.00);

#### Таблиця company\_client

INSERT INTO company\_client (company\_id, client\_id) VALUES

(1, 1),

(2, 2),

(3, 3),

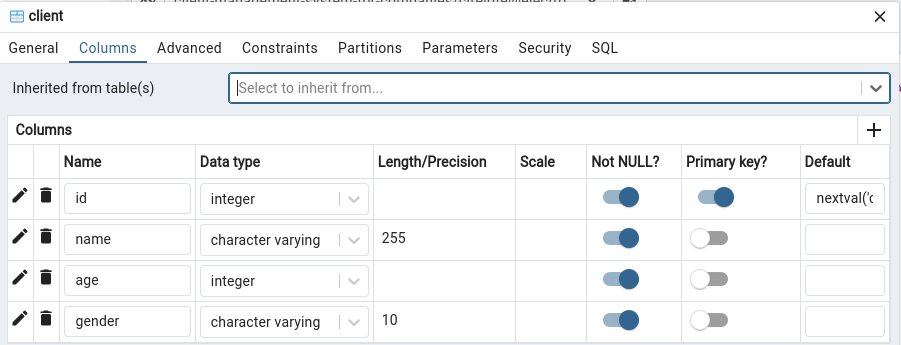
(4, 4),

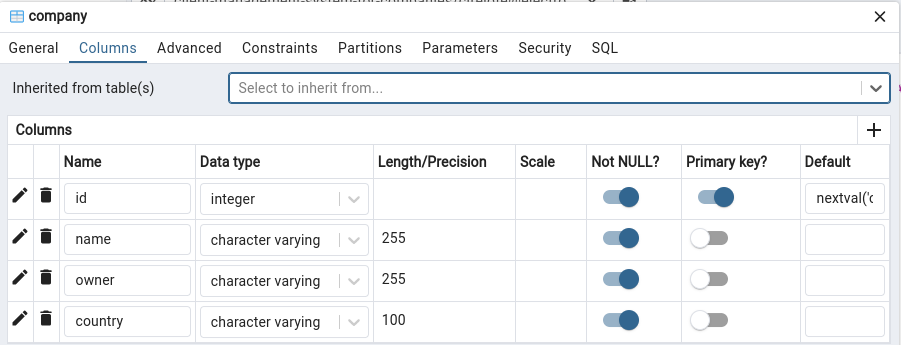
(5, 5),

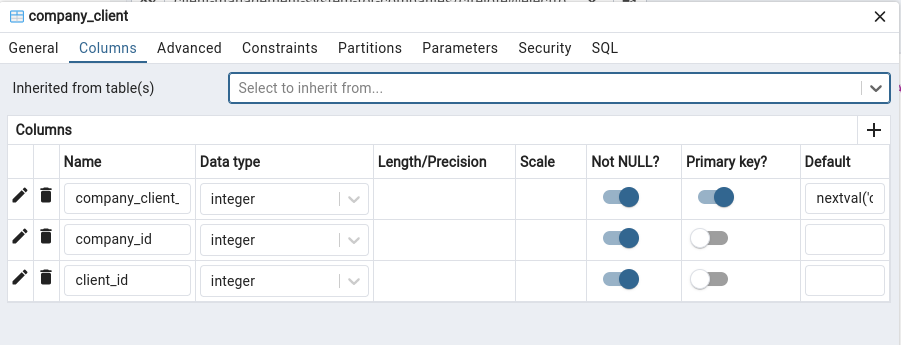
(1, 2),

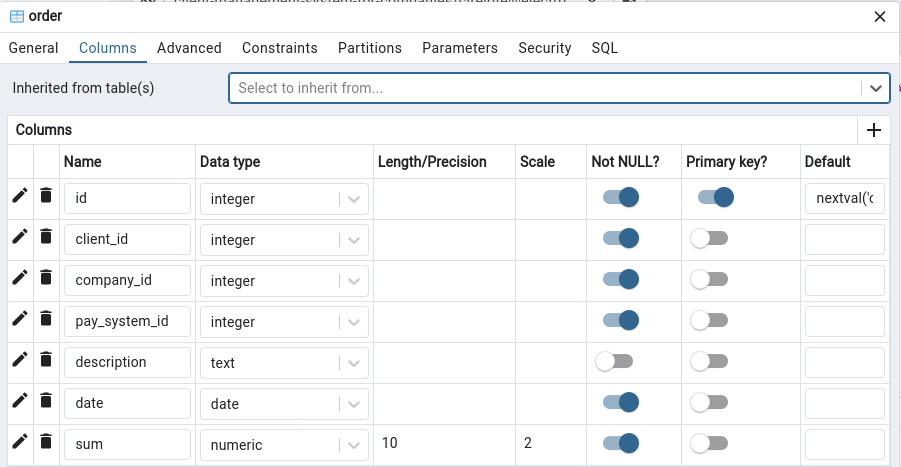
(3, 4);

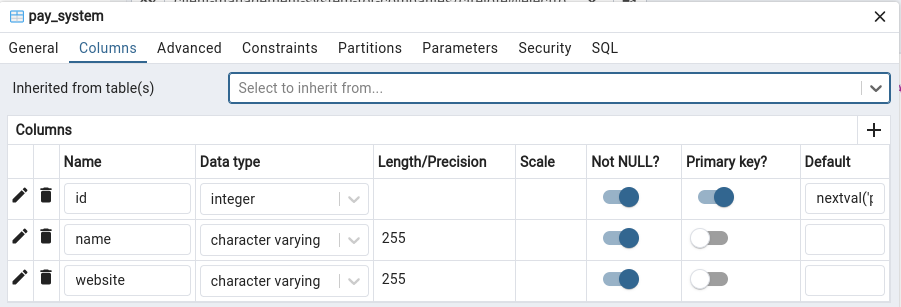
### Назви, типи та обмеження на стовпці









Вміст таблиць бази даних у PostgreSQL

