

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**

*на тему: “Проектування бази даних та ознайомлення з базовими операціями СУБД PostgreSQL”*

Виконав: студент ІII курсу

групи КВ-22 Ковкін В. В. Перевірив:

Павловский В. І.

Київ – 2024

*Метою роботи* є здобуття вмінь проектування бази даних 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 (3 НФ).

4. Ознайомитись із інструментарієм PostgreSQL та pgAdmin4 та внести декілька рядків даних у кожну з таблиць засобами pgAdmin4.

**Модель «сутність-зв’язок»**

Обрана предметна галузь передбачає систему управління процесом вакцинації населення, зберігання даних про пацієнтів, медичні установи, вакцини та лікарів.

**Сутності**

Згідно цієї області для побудови бази даних було виділено наступні сутності:

Громадянин з атрибутами:, код громадянина, ім’я, адреса, номер телефону Призначення: збереження інформації про громадян які мають отримати вакцинацію.

Вакцина з атрибутами: код вакцини, необхідна кількість доз. Призначення: збереження інформації про вакцини, що використовуються в процесі вакцинації.

Лікар з атрибутами: код лікаря, ім’я, номер телефону. Призначення: збереження інформації про медичних працівників, що здійснюють вакцинацію.

Медичний заклад з атрибутами: код закладу, адреса. Призначення: збереження інформації про медичні заклади, де проводиться вакцинація.

**Опис зв’язків**

Один медичний заклад може мати багато лікарів і один лікар може працювати в різних медичних закладах. Тому між сутностями Медичний заклад і Лікар існує зв’язок M:N.

В одному медичному закладі може бути доступно багато вакцин і також вакцини можуть зберігатися у різних медичних закладах. Тому між сутностями Медичний заклад і Вакцина існує зв'язок M:N.

Для вакцинації кожен раз використовується один громадянин, одна вакцина, один лікар і один медичний заклад. Для цього зв’язку додана сутність **Вакцинація**. **Вакцинація** це сутність-зв’язок 4-х сутностей **Громадянин – Вакцина – Лікар – Лікарня**.

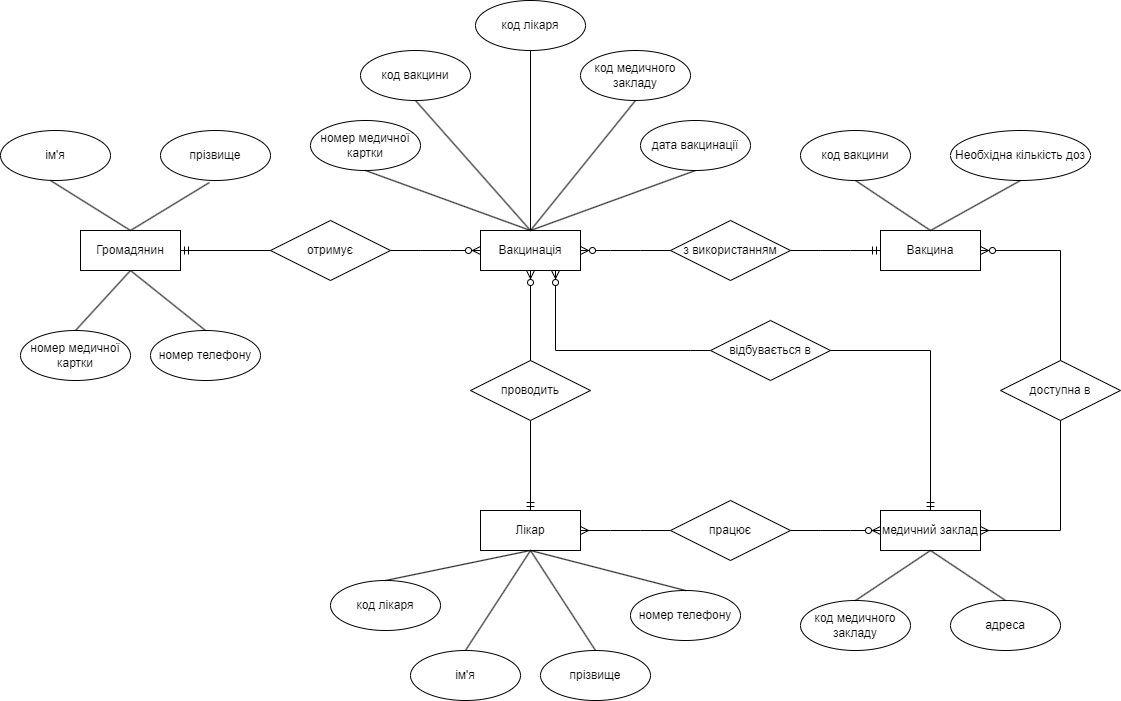


Рисунок 1 - ER-діаграма, побудована за нотацією Чена

**Перетворення концептуальної моделі у логічну модель та схему бази даних**

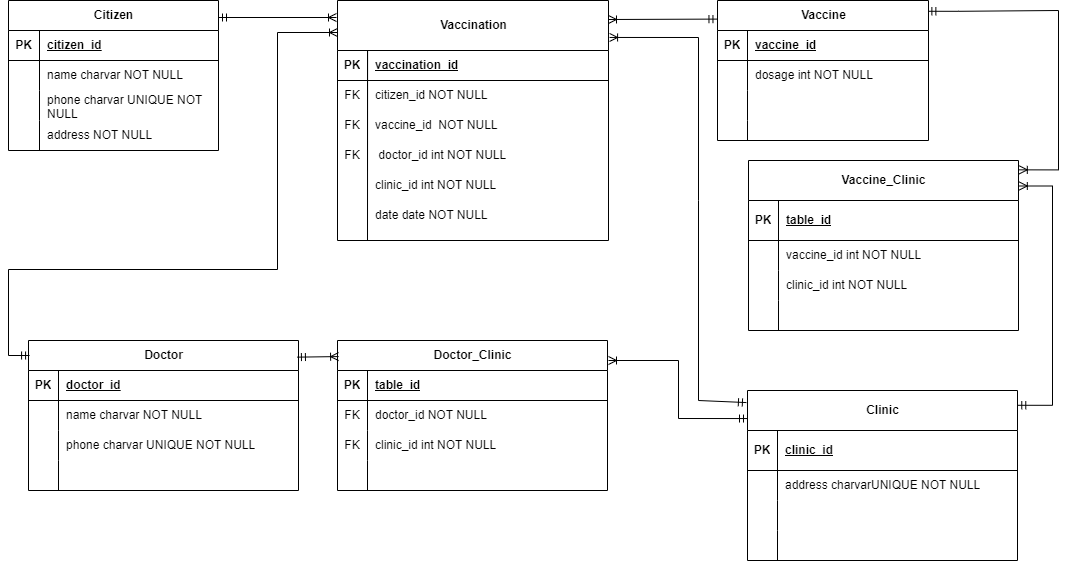
В логічній моделі неможливий безпосередній зв'язок М:N, а в концептуальній моделі він існує між сутностями Лікар і Медичний заклад та Медичний заклад і Вакцина. Для їх представлення було введено допоміжні таблиці Лікар\_МедичнийЗаклад, Вакцина\_Медичний заклад.

Тернарний зв’язок буде перетворено на сутність Вакцинація, з якою кожна сутність у зв’язку, а саме Громадянин, Лікар і Вакцина будуть мати зв’язок 1:N.

Назви таблиць і полів було перекладено англійською, для зручності наведено таблицю нижче.

Таблиця 1 – Опис структури бази даних «Система управління вакцинацією населення»

|  |  |  |
| --- | --- | --- |
| Сутність | Атрибут | Тип атрибуту |
| **Citizen –** містить дані про громадян | **citizen\_id** – унікальний ідентифікатор того, хто має отримати вакцинацію  **name** – ім’я громадянина  **phone –** номер телефону громадянина  **address –** адреса громадянина | **integer** *(числовий)*  **character varying** *(рядок)*  **character varying** *(рядок)*  **character varying** *(рядок)* |
| **Vaccination –** зберігає дані про кожну здійснену вакцинацію. | **vaccination\_id –** унікальний ідентифікатор вакцинації  **citizen\_id –** унікальний ідентифікатор того, хто отримав вакцинацію  **doctor\_id –** унікальний ідентифікатор лікаря, який проводив вакцинацію  **vaccine\_id –** унікальний ідентифікатор вакцини, яка була використана  **clinic\_id –** унікальний ідентифікатор медичного закладу  **date –** дата вакцинації | **integer** *(числовий)*  **integer** *(числовий)*  **integer** *(числовий)*  **integer** *(числовий)*  **integer** *(числовий)*  **date** |
| **Vaccine –** містить інформацію про вакцини | **vaccine\_id –** унікальний ідентифікатор вакцини  **dosage –** необхідна кількість вакцинацій цією вакциною | **integer** *(числовий)*  **integer** *(числовий)* |
| **Doctor –** містить інформацію про лікарів (тих, хто здійснює/надає вакцинацію) | **doctor\_id --** унікальний ідентифікатор лікаря  **name –** ім’я лікаря  **phone –** номер телефону лікаря | **character varying** *(рядок)*  **character varying** *(рядок)*  **character varying** *(рядок)*  **integer** *(числовий)* |
| **Clinic –** містить інформацію про медичні заклади в яких працюють лікарі | **clinic\_id –** унікальний ідентифікатор медичного закладу  **address –** фізична адреса медичного закладу | **integer** *(числовий)*  **character varying** *(рядок)* |
| **Doctor\_Clinic –** містить інформацію про відповідність лікаря і лікарні, де він працює | **id\_tab –** унікальний ідентифікатор таблиці вдіповідності  **doctor\_id –** унікальний ідентифікатор лікаря  **hospital\_id –** унікальний ідентифікатор лікарні | **integer** *(числовий)*  **integer** *(числовий)*  **integer** *(числовий)* |
| **Vaccine\_Clinic –** містить інформацію про наявність вакцин у медичних закладах | **table\_id –** унікальний ідентифікатор таблиці  **vaccine\_id –** унікальний ідентифікатор вакцини  **clinic\_id –** унікальний ідентифікатор медичного закладу | **integer** *(числовий)*  **integer** *(числовий)*  **integer** *(числовий)* |

Рисунок 2 – Схема бази даних

**Функціональні залежності**

**Citizen:**

**citizen\_id –** унікальний ідентифікатор громадянина

**name –** ім’я громадянина

**phone –** номер телефону

**address –** адреса

**citizen\_id →** name, phone, address

citizen\_id **→**phone**→** name, address (транзитивне відношення, але включає в себе ключовий атрибут)

phone → citizen\_id, name, address

**Doctor:**

**doctor\_id –** унікальний ідентифікатор лікаря

**name –** ім’я лікаря

**phone** – номер телефону лікаря

**doctor\_id** **→ name, phone**

phone → doctor\_id, name

**Vaccine:**

**vaccine\_id –** унікальний ідентифікатор вакцини

**dosage –**потрібна кількість доз

vaccine\_id → dosage

**Clinic:**

**clinic\_id –** унікальний ідентифікатор лікарні

address – адреса медичного закладу

clinic\_id → address

**Vaccine\_Clinic:**

**table\_id –** унікальний ідентифікатор таблиці

vaccine\_id – унікальний ідентифікатор вакцини FK

clinic\_id – унікальний ідентифікатор медичного закладу FK

table\_id → clinic\_id, vaccine\_id

clinic\_id, vaccine\_id → table\_id

**Doctor\_Clinic**

**table\_id –** унікальний ідентифікатор таблиці

doctor\_id – унікальний ідентифікатор лікаря FK

clinic\_id – унікальний ідентифікатор медичного закладу FK

table\_id → doctor\_id, vaccine\_id

clinic\_id, vaccine\_id → table\_id

**Vaccination:**

**vaccination\_id → citizen\_id, vaccine\_id, doctor\_id, clinic\_id, date**

citizen\_id, vaccine\_id, doctor\_id, clinic\_id, date → vaccination\_id

Схема відповідає 1НФ, тому що:

Кожен атрибут в кожній таблиці є атомарним. Тобто кажна клітинка містить єдине значення і кожен запис є унікальний.

Схема відповідає 2НФ, тому що:

Кожний неключовий атрибут функціонально залежить від цілого ключа. Кожна таблиця схеми має унікальний ідентифікатор від якого залежать всі її неключові атрибути.

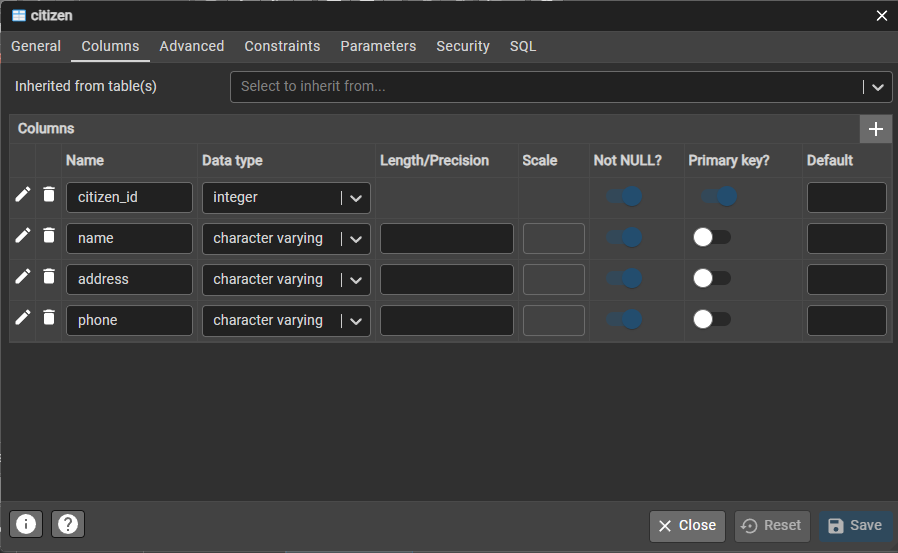
Схема перебуває в 3НФ, тому що:

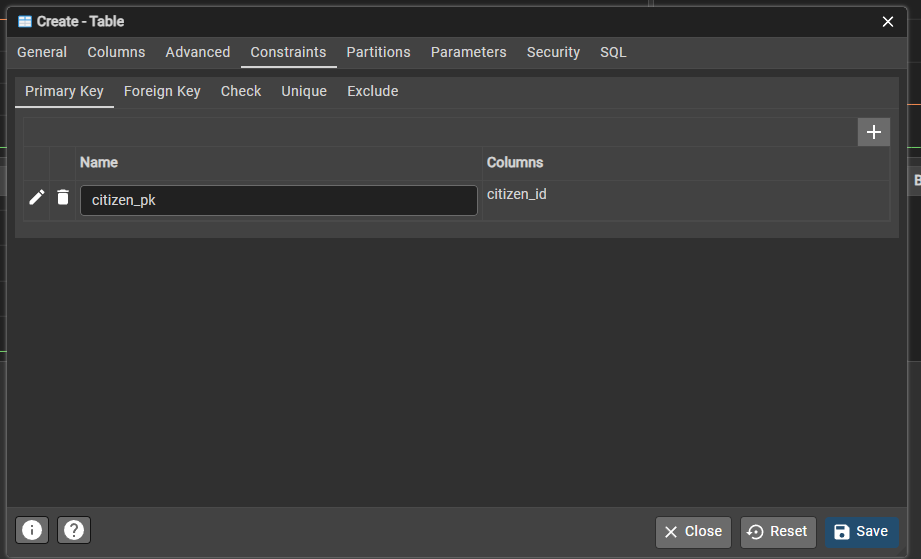
Схема перебуває в 2НФ

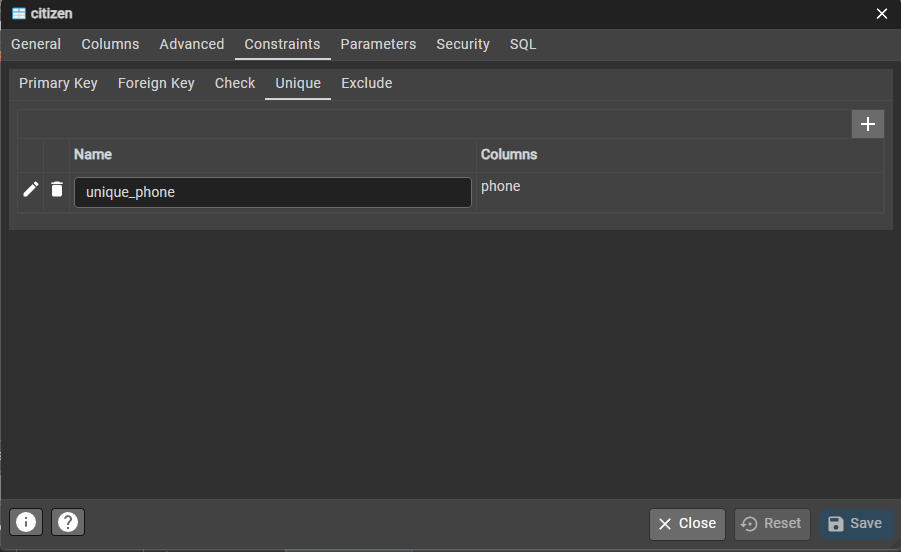
Жодна таблиця не містить неключових атрибутів, які б залежали від інших неключових атрибутів.

**Таблиці бази даних у pgAdmin4**

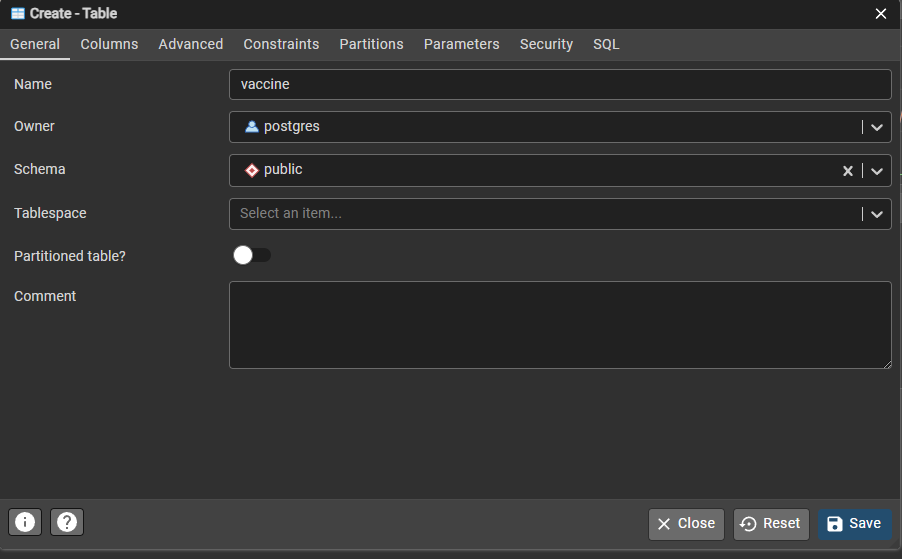
**Citizen**

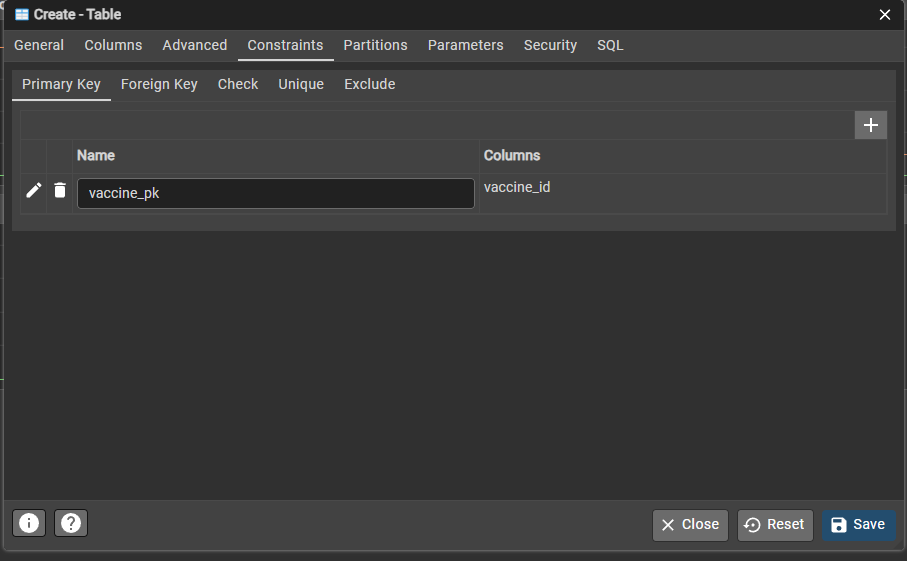
****

****

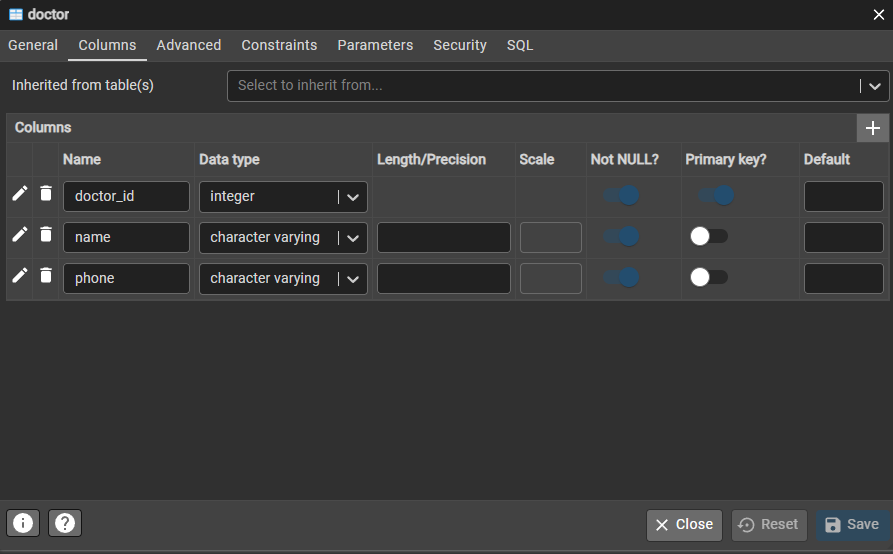
****

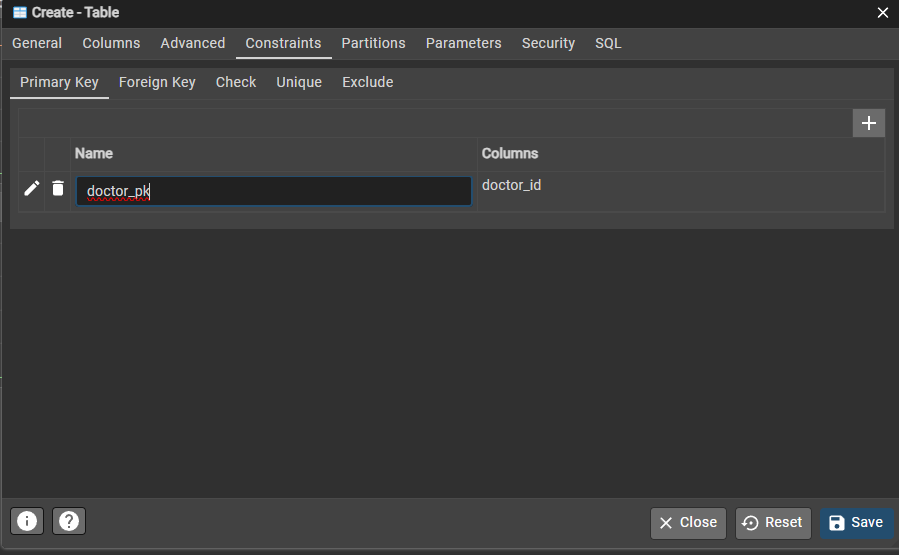
**Vaccine**

****

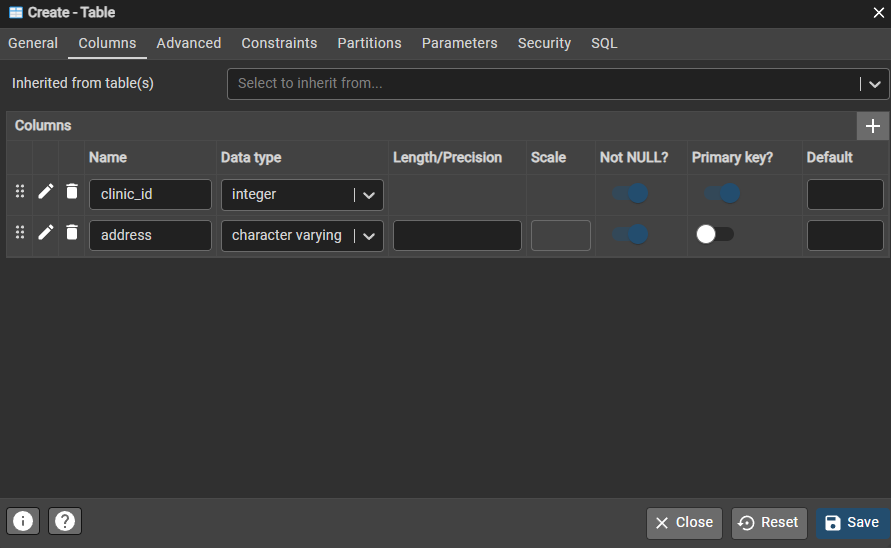
****

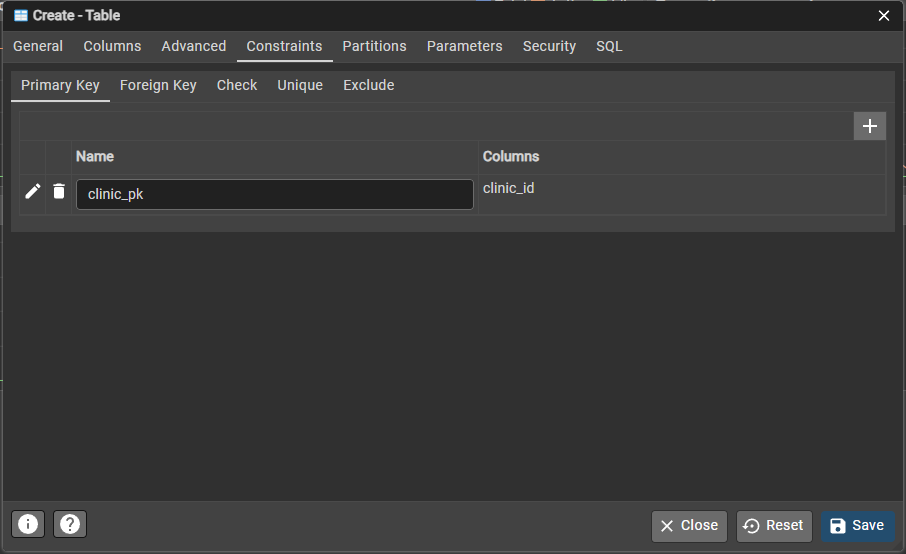
**Doctor**

****

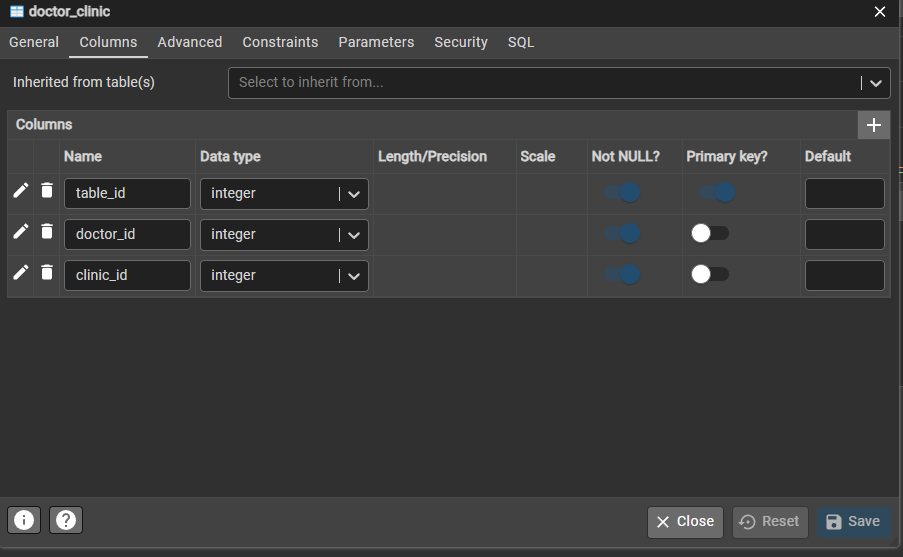
****

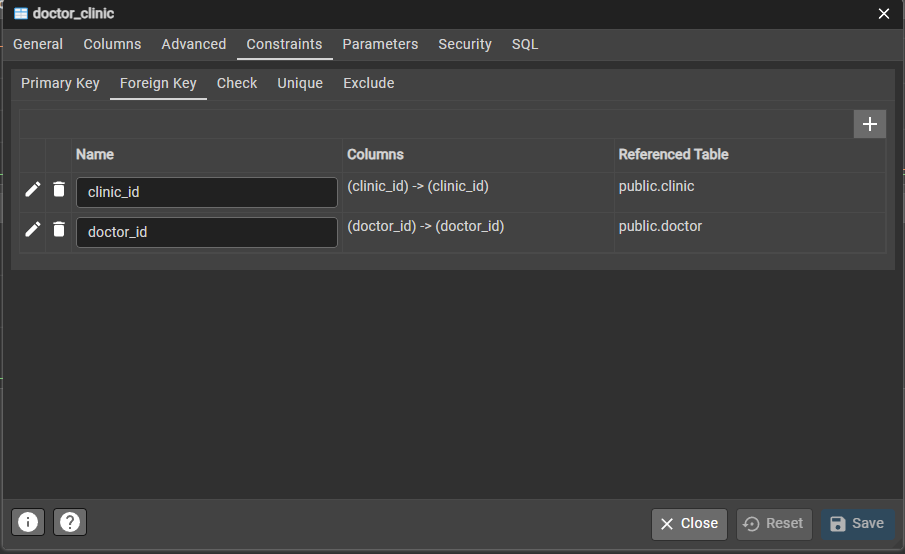
**Clinic**

****

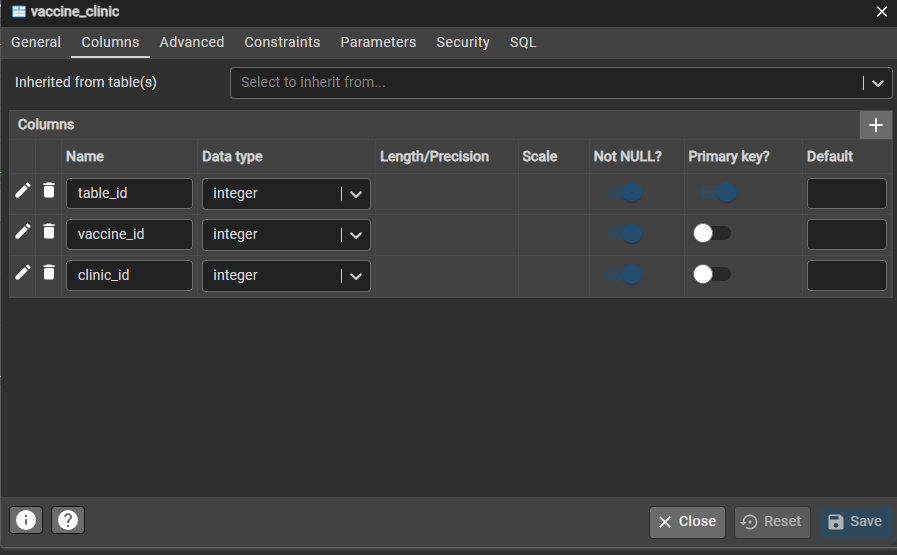
****

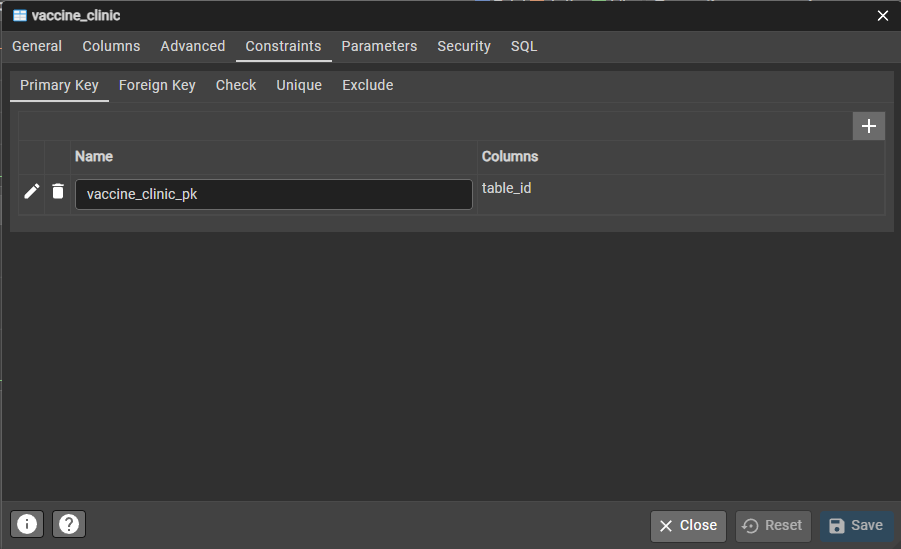
**Doctor\_Clinic**

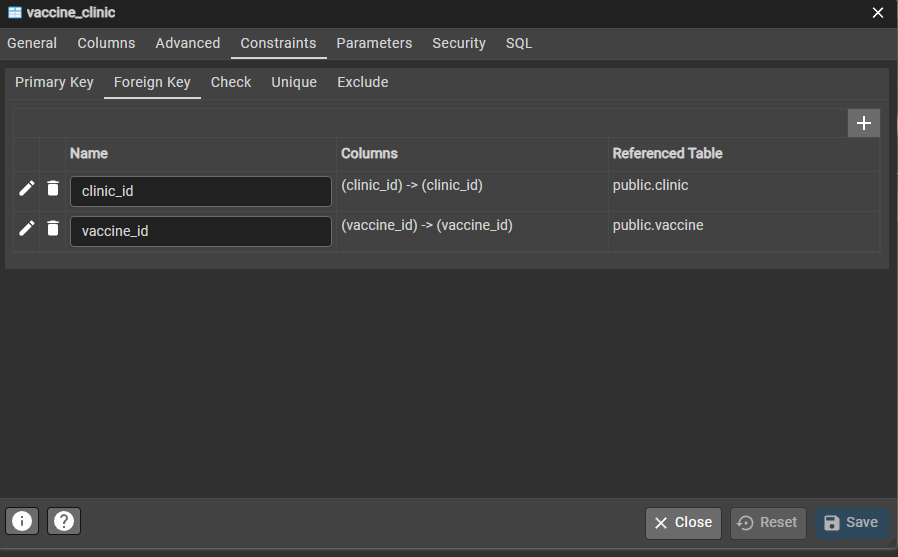
****

****

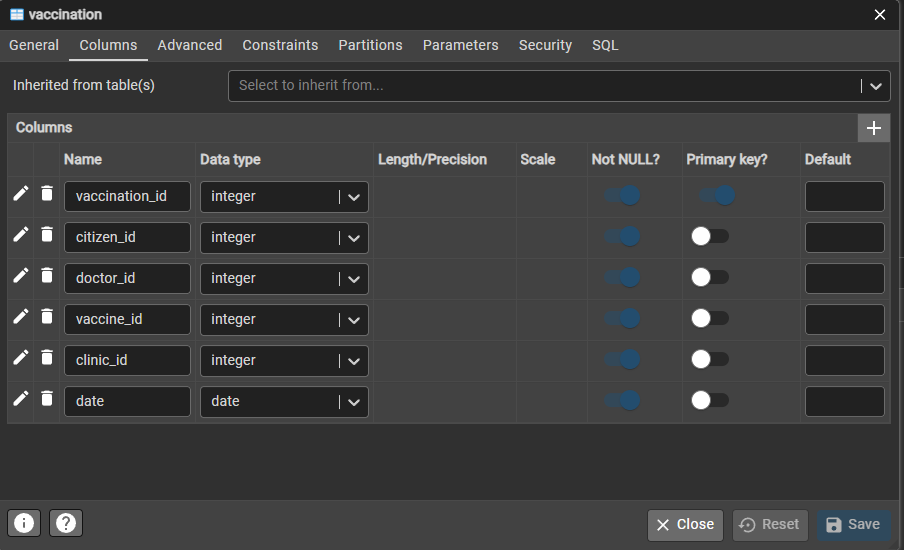
**Vaccine\_Clinic**

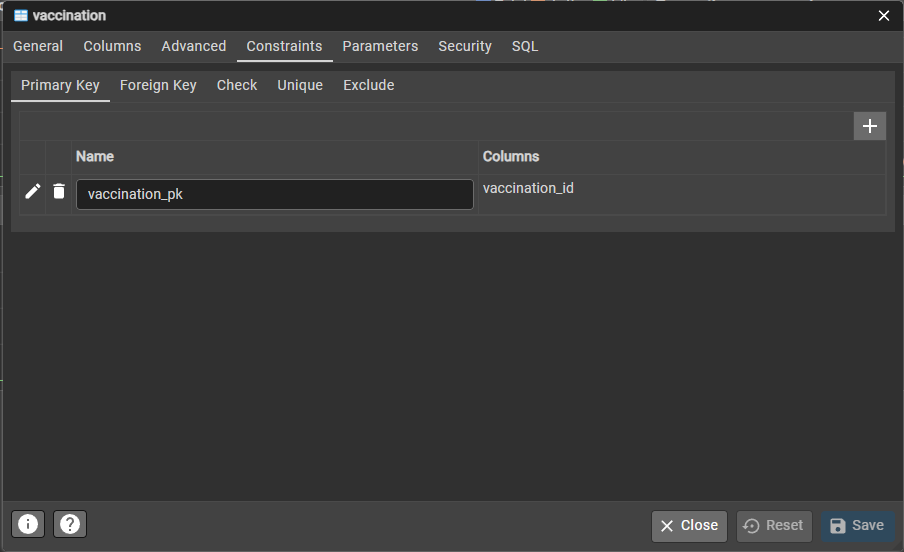
****

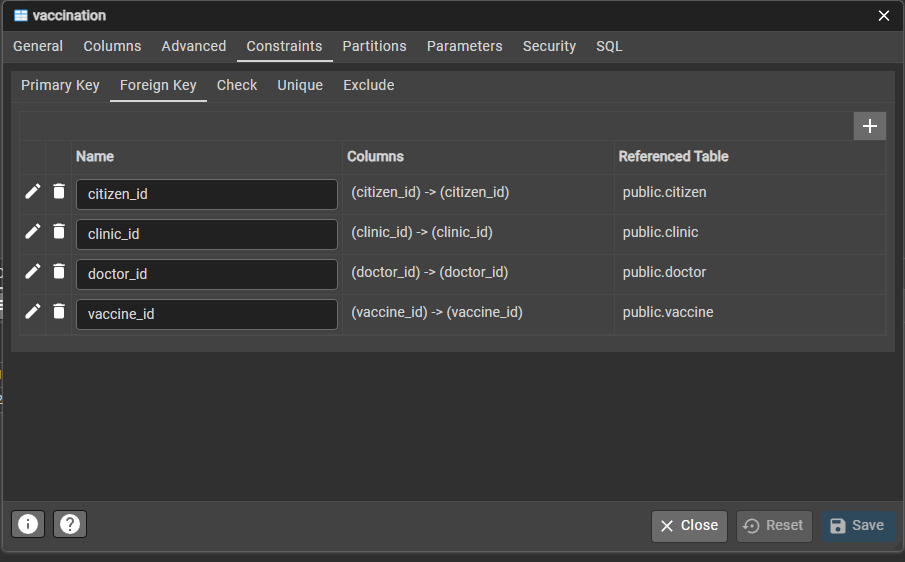
****

****

**Vaccination**

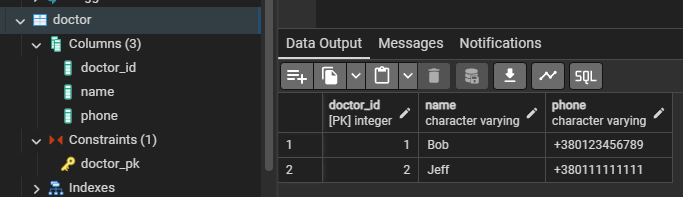
****

****

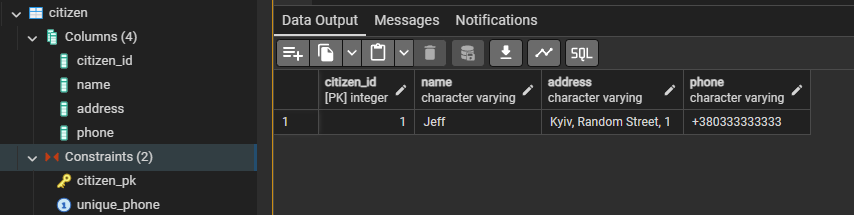
****

**Таблиці заповнені даними**

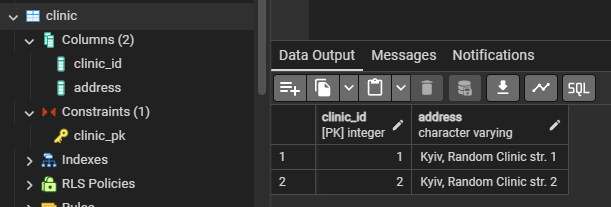
**Doctor**

****

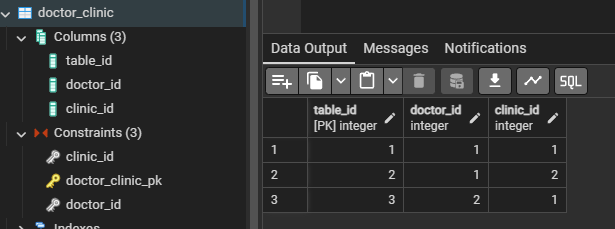
**Citizen**

****

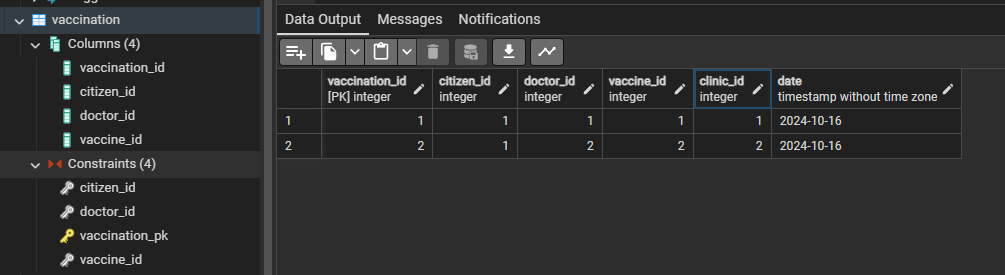
**Clinic**

****

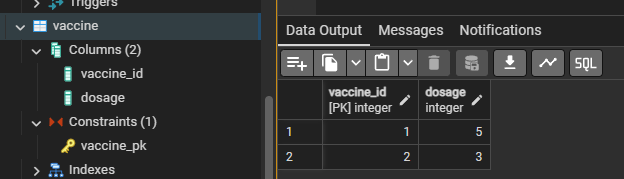
**Doctor\_Clinic**

****

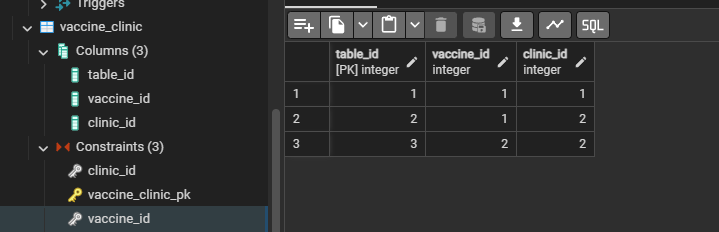
**Vaccination**

****

**Vaccine**

****

**Vaccine\_clinic**

****

**Висновки**

У ході виконання лабораторної роботи було розроблено ER-модель предметної області, яка відповідає вимогам. Модель включає сутності з атрибутами і зв’язками між ними. Для побудови моделі використовувалась нотація Чена. Це дозволило правильно відобразити всі зв’язки між сутностями, зокрема зв'язки типу 1:N та M:N.

Після перетворення ER-моделі на реляційну схему бази даних, було проведено нормалізацію до третьої нормальної форми (3НФ). Схема відповідає 1НФ, оскільки всі атрибути атомарні. Вона також відповідає 2НФ, оскільки кожен неключовий атрибут залежить від повного первинного ключа. У 3НФ виключено транзитивні залежності, забезпечено правильну функціональну залежність атрибутів.

За допомогою PostgreSQL та інструмента pgAdmin4 були створені таблиці бази даних, визначені типи даних атрибутів, встановлені первинні та зовнішні ключі, а також обмеження на стовпці (NOT NULL, UNIQUE). Це гарантувало цілісність даних та відповідність заданим обмеженням.

У результаті роботи було заповнено таблиці бази даних реальними даними, що дозволило перевірити коректність створеної структури та виконання базових операцій додавання і збереження даних у системі.

**Посилання на репозиторій Github:** <https://github.com/kovkinvladyslav/bd_labs.git>