

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

Виконав: студент групи КВ-13  
Шпилька І.В.  
Перевірив: Петрашенко А.В.

Київ-2023

## Лабораторна робота № 1.

### 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.

#### *Вимоги до ER-моделі*

1. Сутності моделі предметної галузі мають містити зв'язки типу 1:N або N:M.
2. Кількість сутностей у моделі – 3-4. Кількість атрибутів у кожній сутності: від двох до п'яти.
3. Передбачити наявність зв'язку з атрибутом.
4. Для побудови ER-діаграм використовувати одну із нотацій: Чена, “Пташиної лапки (Crow's foot)”, UML.

#### *Вимоги до інструментарію*

1. Створення ER-діаграм: Google Docs (Drawing) або <https://www.draw.io/> або <https://www.lucidchart.com>
2. Середовище для створення таблиць відлагодження SQL-запитів до бази даних – pgAdmin 4.
3. СУБД - PostgreSQL 13-15 (<https://www.postgresql.org/download/>).

## *Вимоги до оформлення лабораторної роботи у електронному вигляді*

Опис лабораторної роботи, створеної за допомогою сервісу Google Docs і розміщеної у Google Classroom включає: назву лабораторної роботи, варіант студента (опис обраної предметної галузі), посилання на репозиторій Github, контакт студента в Телеграм та вимоги до звітування щодо пунктів 1-4 завдання, які наведено нижче:

*У звіті щодо пункту №1 завдання має бути:*

- перелік сутностей з описом їх призначення;
- графічний файл розробленої моделі «сутність-зв'язок»;
- назва нотації.

*У звіті щодо пункту №2 завдання має бути:*

- опис процесу перетворення (наприклад, “сутність А було перетворено у таблицю А, а зв'язок R (M:N) зумовив появу додаткової таблиці R1 тощо);
- схему бази даних у графічному вигляді з **назвами таблиць (!) та зв'язками між ними, а також необхідно намалювати перетворену ER-діаграму у ТАБЛИЦІ БД! Це означає, що тут не може бути зв'язку N:M, мають бути позначені первинні та зовнішні ключі, обмеження NOT NULL та UNIQUE і внести типи даних атрибутів.**

*У звіті щодо пункту №3 завдання має бути:*

- пояснення (**обґрунтування!**) щодо відповідності схеми бази даних нормальним формам НФ1, НФ2 та НФ3. Пояснення **полягає у наведенні функціональних залежностей**, що демонструють висновки. У випадку невідповідності надати опис необхідних змін у схемі;
- У випадку проведення змін у схемі бази даних надати оновлену версію схеми, інакше - не наводити схему.

*У звіті щодо пункту №4 завдання має бути:*

- навести копії екрану з pgAdmin4, що відображають назви, типи та обмеження на стовпці (доступне у закладці “Columns” та “Constraints” властивостей “Properties” таблиць дерева об'єктів у pgAdmin4);
- навести копії екрану з pgAdmin4, що відображають вміст таблиць бази даних у PostgreSQL. Таблиці на зображенні обов'язково **повинні мати назву!**

## *Опис предметної галузі*

Обрана предметна галузь – «Платформа для зберігання історій хвороб пацієнтів». Були створені наступні сутності:

1. hospital:

Призначення: має інформацію про лікарню, у якій працюють лікарі.

Атрибути: hospital\_id (ідентифікатор лікарні), hospital\_name (назва лікарні).

2. doctor:

Призначення: має інформацію про лікаря, що працює у лікарні та займається лікуванням пацієнтів.

Атрибути: doctor\_id (ідентифікаційний номер лікаря), doctor\_full\_name (повне ім'я лікаря), doctor\_specialization (інформація про спеціальність лікаря)

3. patient:

Призначення: представляє інформацію про пацієнтів, що страждають від хвороб та проходять лікування під наглядом лікарів.

Атрибути: patient\_id (ідентифікаційний номер пацієнта), patient\_full\_name (ім'я пацієнта), date\_of\_hospitalization (дата госпіталізації), date\_of\_discharge (дата виписки із лікарні)

4. illness:

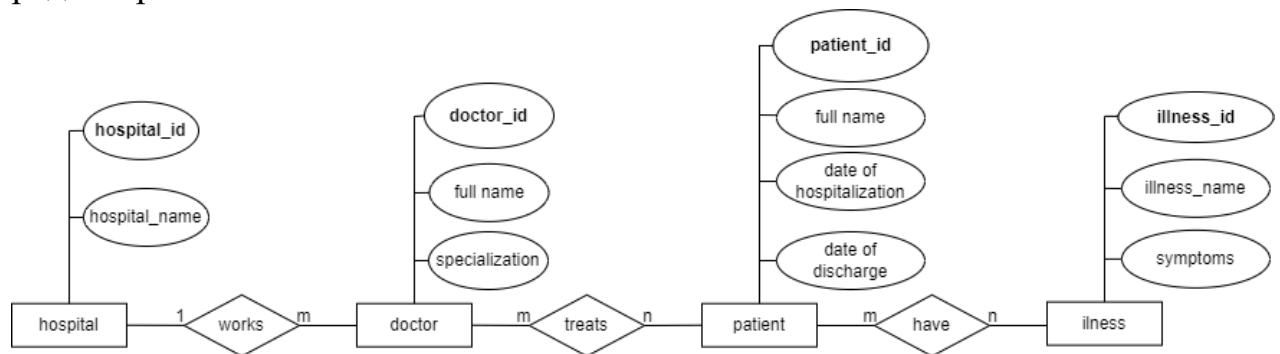
Призначення: Містить інформацію про хворобу, від якої страждає пацієнт.

Атрибути: illness\_id (ідентифікатор хвороби), illness\_name (назва хвороби), symptoms (симптоми)

### Зв'язки:

1. У лікарні може працювати багато лікарів, але кожен лікар працює лише в одній лікарні (1:M – doctor:hospital)
2. У кожного пацієнта може бути багато лікарів, що за ним доглядають, та у кожного лікаря – багато пацієнтів (N:M – patient:doctor)
3. Кожен пацієнт може мати багато хвороб, і кожна хвороба може бути у багатьох пацієнтів (M:N – illness:patient)

ER-діаграма предметної області «Платформа для зберігання історій хвороб пацієнтів», побудована відповідно до норм «Нотації Чена» за допомогою редактора draw.io:

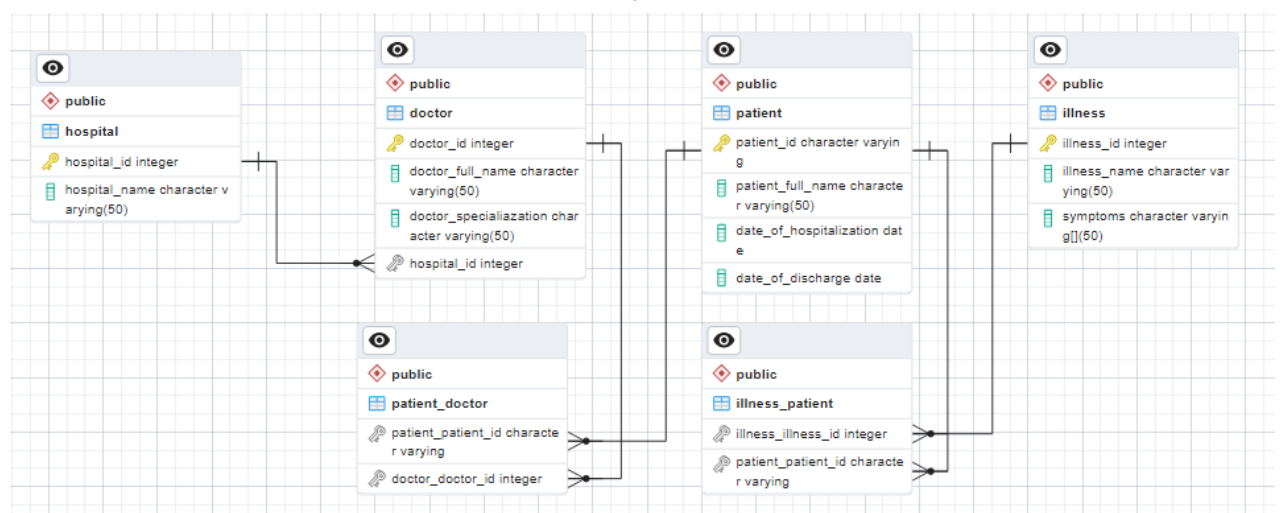


### *Процес перетворення*

Зв'язок між patient та doctor (зв'язок N:M) потребував утворення таблиці «patient\_doctor» для збереження інформації про те, який лікар доглядає за пацієнтом.

Зв'язок між illness та patient (зв'язок N:M) потребував утворення таблиці «illness\_patient» для збереження інформації про те, від якої хвороби страждає пацієнт.

Логічна модель БД «Платформа для зберігання історій хвороб пацієнтів»:



Модель побудована за допомогою утиліти pgAdmin4

## ***Відповідність нормальним формам***

1. Схема відповідає 1НФ, оскільки:

- Кожна таблиця має Primary key та мінімальний набір атрибутів, які ідентифікують запис;
- Кожен із атрибутів має лише одне значення, а не множину значень.

2. Схема відповідає 2НФ, оскільки:

- Схема відповідає 1НФ;
- Кожен не ключовий атрибут функціонально залежить від основного атрибуту.

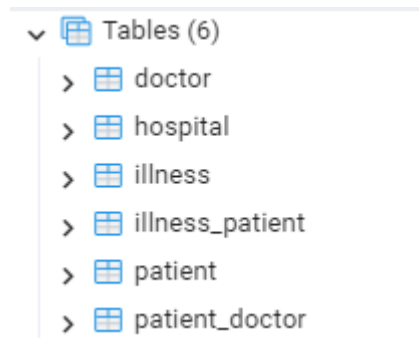
Приклад: У таблиці "doctor" атрибути doctor\_id, doctor\_full\_name, doctor\_specialization, і hospital\_id повністю залежать від doctor\_id, який є первинним ключем.

3. Схема відповідає 3НФ, оскільки:

- Схема відповідає вимогам 2НФ;
- Дані в таблиці залежать лише від основного ключа

Приклад: У таблиці «patient» атрибути patient\_full\_name, date\_of\_hospitalization, і date\_of\_discharge не залежать один від одного.

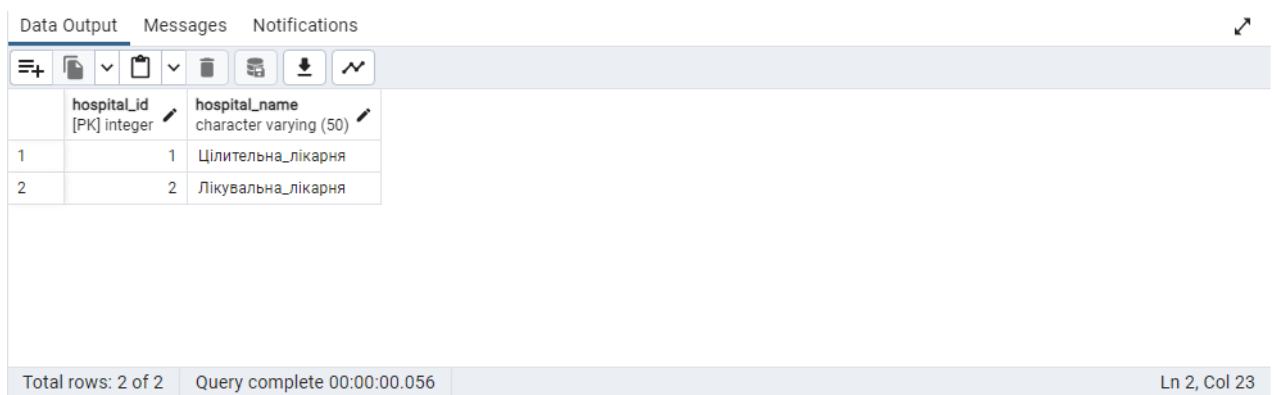
## Скріни з pgAdmin4



A screenshot of the pgAdmin4 interface showing a tree view of database tables. The 'Tables (6)' folder is expanded, revealing six tables: doctor, hospital, illness, illness\_patient, patient, and patient\_doctor. Each table is represented by a small table icon and its name.

Tables (6)
doctor
hospital
illness
illness_patient
patient
patient_doctor

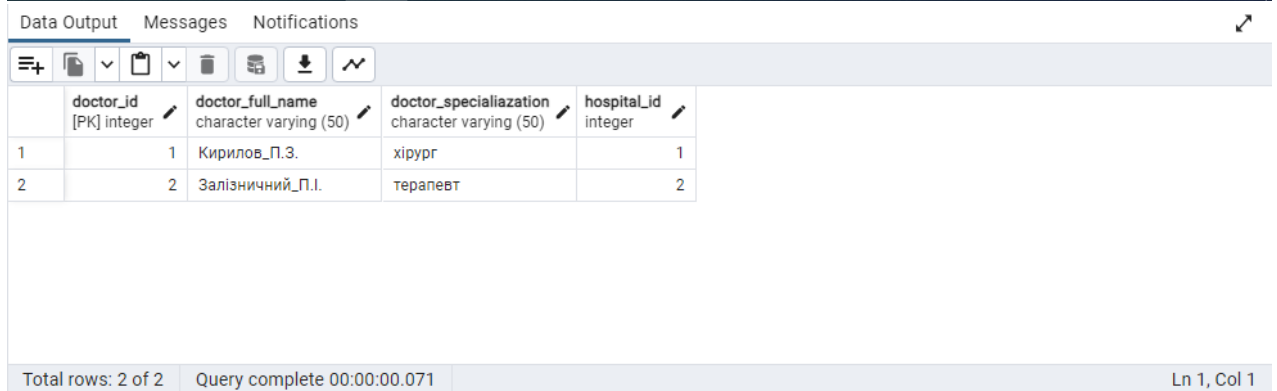
## Скріни вмісту таблиць



A screenshot of the pgAdmin4 Data Output window for the 'hospital' table. The window has tabs for 'Data Output', 'Messages', and 'Notifications'. The 'Data Output' tab is active, showing a table with two columns: 'hospital\_id' (integer, PK) and 'hospital\_name' (character varying (50)). There are two rows of data. The status bar at the bottom indicates 'Total rows: 2 of 2' and 'Query complete 00:00:00.056'.

	hospital_id [PK] integer	hospital_name character varying (50)
1	1	Цілительна_лікарня
2	2	Лікувальна_лікарня

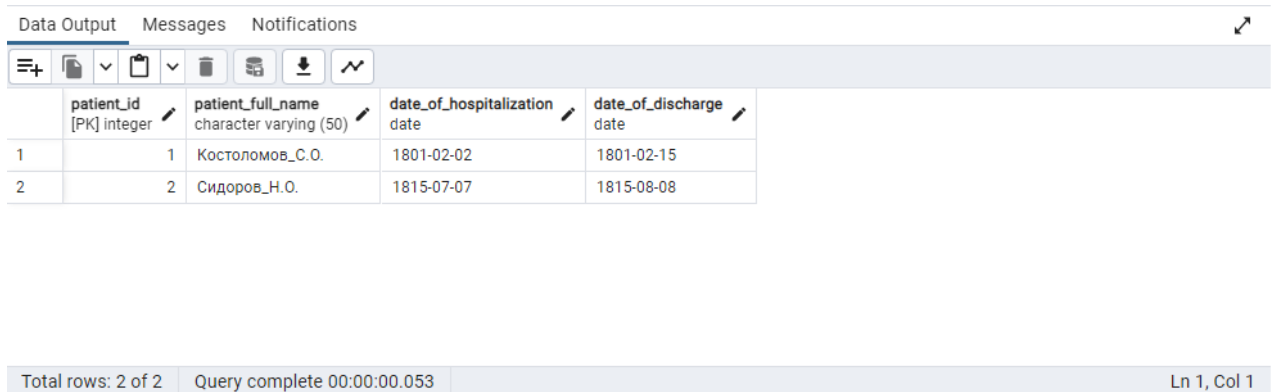
Total rows: 2 of 2    Query complete 00:00:00.056    Ln 2, Col 23



A screenshot of the pgAdmin4 Data Output window for the 'doctor' table. The window has tabs for 'Data Output', 'Messages', and 'Notifications'. The 'Data Output' tab is active, showing a table with four columns: 'doctor\_id' (integer, PK), 'doctor\_full\_name' (character varying (50)), 'doctor\_specialization' (character varying (50)), and 'hospital\_id' (integer). There are two rows of data. The status bar at the bottom indicates 'Total rows: 2 of 2' and 'Query complete 00:00:00.071'.

	doctor_id [PK] integer	doctor_full_name character varying (50)	doctor_specialization character varying (50)	hospital_id integer
1	1	Кирилов_П.З.	хірург	1
2	2	Залізничний_П.І.	терапевт	2

Total rows: 2 of 2    Query complete 00:00:00.071    Ln 1, Col 1



A screenshot of the pgAdmin4 Data Output window for the 'patient' table. The window has tabs for 'Data Output', 'Messages', and 'Notifications'. The 'Data Output' tab is active, showing a table with four columns: 'patient\_id' (integer, PK), 'patient\_full\_name' (character varying (50)), 'date\_of\_hospitalization' (date), and 'date\_of\_discharge' (date). There are two rows of data. The status bar at the bottom indicates 'Total rows: 2 of 2' and 'Query complete 00:00:00.053'.

	patient_id [PK] integer	patient_full_name character varying (50)	date_of_hospitalization date	date_of_discharge date
1	1	Костоломов_С.О.	1801-02-02	1801-02-15
2	2	Сидоров_Н.О.	1815-07-07	1815-08-08

Total rows: 2 of 2    Query complete 00:00:00.053    Ln 1, Col 1

	patient_patient_id integer	doctor_doctor_id integer
1	1	1
2	2	2
Total rows: 2 of 2    Query complete 00:00:00.056    Ln 1, Col 1		

Data Output    Messages    Notifications			
	illness_id [PK] integer	illness_name character varying (50)	symptoms character varying (50)
1	1	перелом_ноги	біль_у_нозі
2	2	коронавірус	лихоманка
Total rows: 2 of 2    Query complete 00:00:00.060    Ln 1, Col 1			

	illness_illness_id integer	patient_patient_id integer
1	1	1
2	2	2
Total rows: 2 of 2    Query complete 00:00:00.050    Ln 1, Col 1		

## Контакти:

1. [github](#)
2. Telegram: <https://t.me/ihorshpilka>