НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ “КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ

ІМЕНІ ІГОРЯ СІКОРСЬКОГО”

Факультет прикладної математики Кафедра прикладної математики

Звіт

з лабораторної роботи №1

з дисципліни “Вступ до баз даних та інформаційних систем” на тему:

*Створення таблиць. Виконання простих запитів на мові SQL та за допомогою алгебри Кодда. Використання вбудованих функцій мови SQL*

Виконав:

студент групи КМ-03

*Куцалаба Н. В.*

Керівник:

*ст. викладач Бай Ю. П.*

Київ 2022

# ЗАВДАННЯ

**ВАРІАНТ №8**

**Завдання 1.** Спроектувати базу даних, що дозволить відобразити наступні події (*5 балів*):

**Людина має лікарняну картку, що містить записи про історію хвороби.**

**1a)** Визначити сутності та їх атрибути, встановити зв’язки між сутностями. Побудувати ER-діаграму.

**1b)** Побудувати логічну схему таблиць, використовуючи «crow's foot notation».

**1с)** За допомогою команд мови SQLстворити таблиці в СУБД PostgreSQL. Визначити поля та типи. Первинні та зовнішні ключі створювати окремо від таблиць, використовуючи команду ALTER TABLE.

**Завдання 2.** Згенерувати базу даних з книги Б. Форта та виконати запити (потрібні для виконання завдань файли *create.txt, populate.txt* можна завантажити, наприклад, з <https://github.com/alinbxSorcerer/SQL-in-10-minutes-with-notes.git>) (*6 балів*):

**2a) Скільки продано найдешевшого товару?**

**2b) Як звуть покупця з найдовшим іменем – поле назвати long\_name?**

**2c) Вивести ID та імена постачальників, що не мають жодного товару.   
Результат вивести у нижньому регістрі, як єдине поле vendor\_name.**

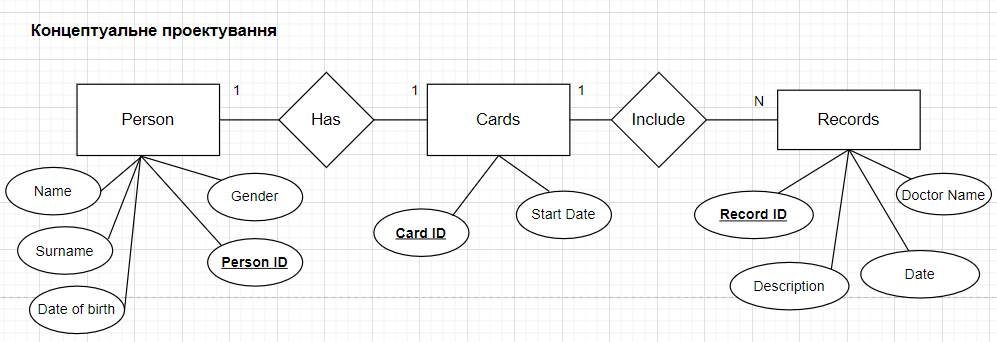
**Завдання 3.** Виконати запити 2a), 2b) з попереднього завдання, використовуючи операції реляційної алгебри Кодда та агрегатні функції мови SQL (*4 бали*).

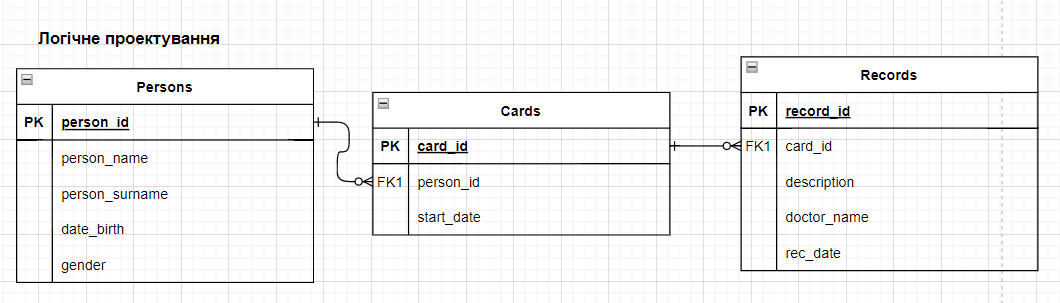
# Завдання 1

Спроектувати базу даних, що дозволить відобразити наступні події:

# Людина має лікарняну картку, що містить записи про історію хвороби.

**1a)** Визначити сутності та їх атрибути, встановити зв’язки між сутностями. Побудувати ER-діаграму.



**1b)** Побудувати логічну схему таблиць, використовуючи «crow's foot notation». 

**1с)** За допомогою команд мови SQLстворити таблиці в СУБД PostgreSQL. Визначити поля та типи. Первинні та зовнішні ключі створювати окремо від таблиць, використовуючи команду ALTER TABLE.

Для зв’язку “1 : 1” у відношенні створюється зовнішній ключ. Зовнішній ключ приймає значення тільки з множини значень первинного ключа відношення, що може існувати самостійно.

Для зв’язку “1 : N” будемо використовувати так звану з’єднуючу таблицю, яка містить значення первинних ключів.

Команди створення таблиць:

CREATE TABLE persons (

person\_id int UNIQUE NOT NULL,

person\_name char(50) NOT NULL,

person\_surname char(50) NOT NULL,

date\_birth date NOT NULL,

gender char(50) NOT NULL

);

CREATE TABLE cards (

card\_id int UNIQUE NOT NULL,

person\_id int NOT NULL,

start\_date date NOT NULL

);

CREATE TABLE records (

record\_id int UNIQUE NOT NULL,

card\_id int UNIQUE NOT NULL,

description char(200) NULL,

doctor\_name char(50) NOT NULL,

rec\_date date NOT NULL

);

Команди налаштування первинних та зовнішніх ключів:

ALTER TABLE persons ADD PRIMARY KEY (person\_id);

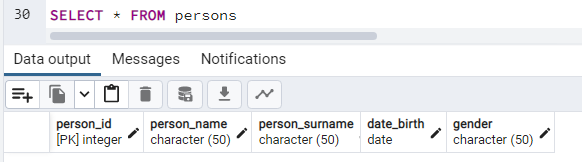
ALTER TABLE cards ADD PRIMARY KEY (card\_id);

ALTER TABLE records ADD PRIMARY KEY (record\_id);

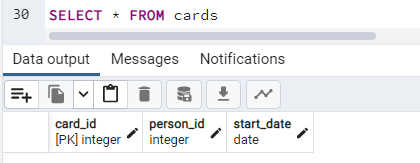
ALTER TABLE cards ADD CONSTRAINT FK\_cards\_persons FOREIGN KEY (person\_id) REFERENCES persons (person\_id);

ALTER TABLE records ADD CONSTRAINT FK\_records\_cards FOREIGN KEY (card\_id) REFERENCES cards (card\_id);

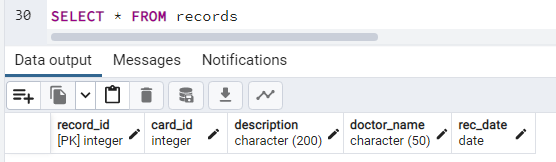
Таблиця persons:



Таблиця cards:



Таблиця records:



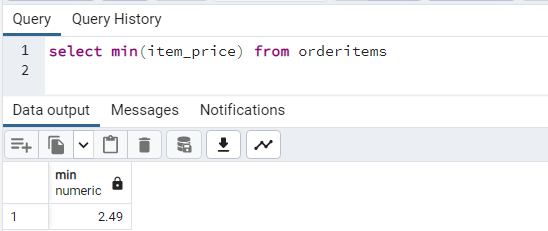
# Завдання 2

Згенерувати базу даних з книги Б. Форта (*create.txt, populate.txt)*, та виконати запити:

# 2a) Скільки продано найдешевшого товару?

Визначимо мінімальну вартість товарів:

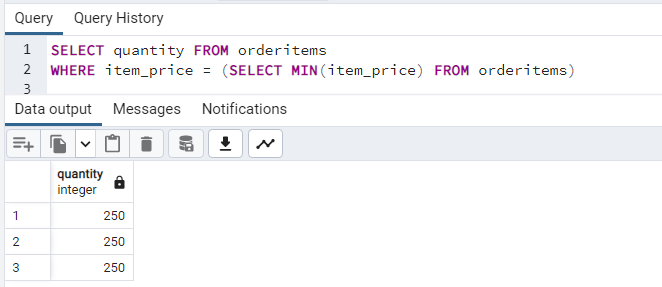
SELECT MIN(item\_price) FROM orderitems



Використаємо це значення у вигляді вкладеного запиту, аби отримати quantity всіх проданих товарів із найменшою ціною:

SELECT quantity FROM orderitems

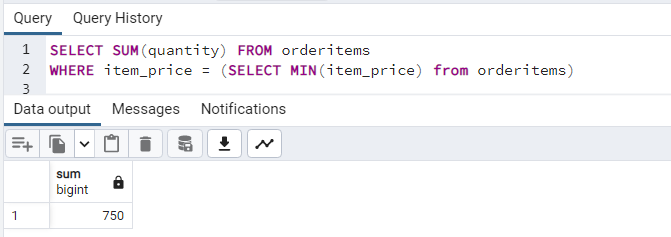
WHERE item\_price = (SELECT MIN(item\_price) FROM orderitems)



Нарешті визначимо кількість продано найдешевшого товару:

SELECT SUM(quantity) FROM orderitems

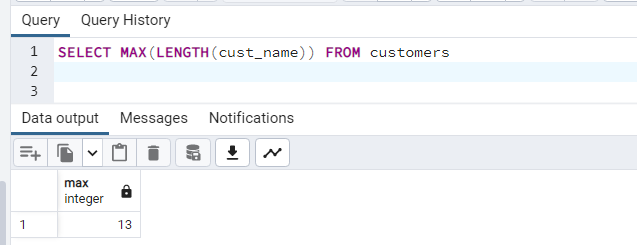
WHERE item\_price = (SELECT MIN(item\_price) from orderitems)



# 2b) Як звуть покупця з найдовшим іменем – поле назвати long\_name?

Визначимо довжину найдовшого імені покупця:

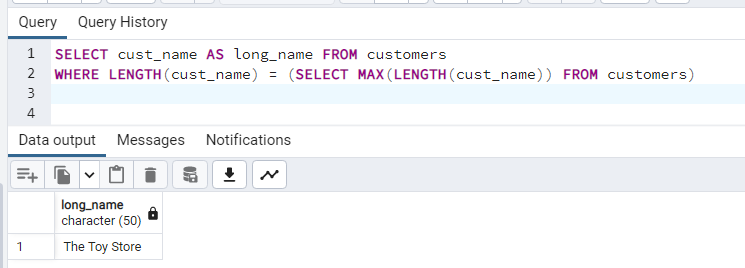
SELECT MAX(LENGTH(cust\_name)) FROM customers



Отримане значення використовуємо для отримання cust\_name покупця, що має найдовше ім’я:

SELECT cust\_name AS long\_name FROM customers

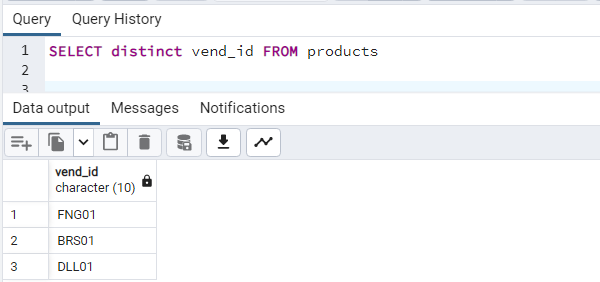
WHERE LENGTH(cust\_name) = (SELECT MAX(LENGTH(cust\_name)) FROM customers)



# 2c)  Вивести ID та імена постачальників, що не мають жодного товару. Результат вивести у нижньому регістрі, як єдине поле vendor\_name.

Виберемо vend\_id всіх товарів(без повторів):

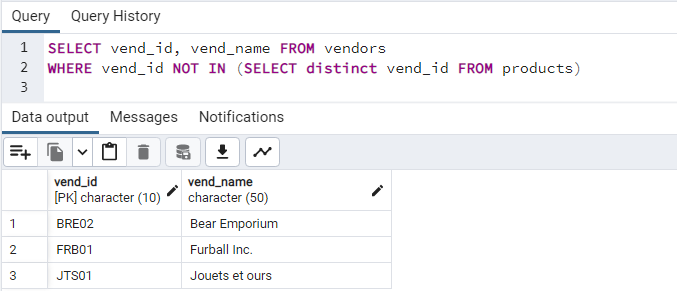
SELECT distinct vend\_id FROM products



Наступним кроком виберемо vend\_id та vend\_name тих постачальників, які не мають товарів:

SELECT vend\_id, vend\_name FROM vendors

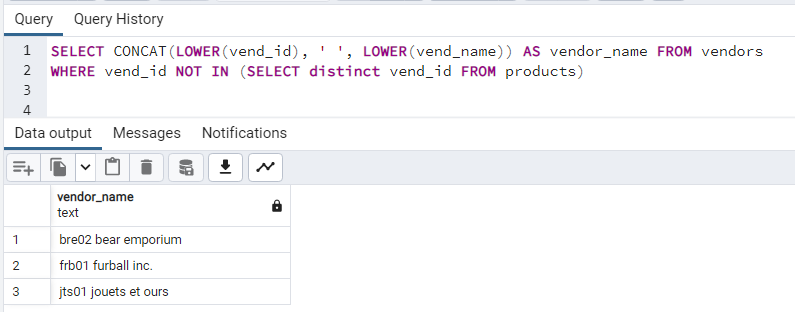
WHERE vend\_id NOT IN (SELECT distinct vend\_id FROM products)



Знайдемо та виведемо у нижньому регістрі id та імена постачальників які не мають товарів, як єдине поле vendor\_name:

SELECT CONCAT(LOWER(vend\_id), ' ', LOWER(vend\_name)) AS vendor\_name FROM vendors

WHERE vend\_id NOT IN (SELECT distinct vend\_id FROM products)



# Завдання 3

Виконати запити 2a), 2b) з попереднього завдання, використовуючи операції реляційної алгебри Кодда та агрегатні функції мови SQL.

# 2a)

Введемо допоміжне позначення мінімальної ціни товару:

𝑀𝑝 = 𝑀𝐼𝑁(π (*orderitems*));

𝑖𝑡𝑒𝑚\_𝑝𝑟𝑖𝑐𝑒

Далі визначимо quantity проданих товарів з найменшою ціною:

*Q* = π (σ (*orderitems*))

*quantity* 𝑖𝑡𝑒𝑚\_𝑝𝑟𝑖𝑐𝑒 = 𝑀𝑝

Нарешті визначимо кількість продано найдешевшого товару:

SUM(Q)

# 2b)

Позначимо кількість літер у найдовшій назві товару наступним чином:

𝑀𝑙 = 𝑀𝐴𝑋(π𝐿𝐸𝑁𝐺𝑇𝐻(*cust\_name)*(*customers*))

Визначимо cust\_name покупця, що має найдовше ім’я:

*Ln* = π*cust\_name*(σ𝐿𝐸𝑁𝐺𝑇𝐻(*cust\_name*) = 𝑀𝑙(*customers*))

Замінимо назву поля на *long\_name*:

*Plong\_name*(Ln)

# Список літератури

1. Дейт К. Введение в системы баз данных. − Пер. с англ. − 8-е изд. − К.: Изд. дом “Вильямс”, 2006. − 1326 с.
2. Берко А. Ю., Верес О. М., Пасічник В. В. Системи баз даних та знань. Книга 1. Організація баз даних та знань: Навчальний посібник. − Львів: “Магнолія 2006”, 2008. − 456 с.
3. Конноли Т. Базы данных. Проектирование, реализация и сопровождение. Теория и практика / Т. Конноли, К. Бегг. − 3-е изд. − М.: Изд. дом “Вильямс”, 2003. − 1440 с.
4. Форта Б. Освой самостоятельно SQL. 3-е изд.: Пер. с англ. − М.: Изд. дом “Вильямс”, 2006. − 288 с.