



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО”

Факультет прикладної математики
Кафедра програмного забезпечення комп’ютерних систем

Лабораторна робота № 1
з дисципліни “Бази даних ”

Виконав

студент II курсу

групи КП-03

Заїка Максим Олександрович

Перевірив

“ ____ ” “ _____ ” 20__ р.

викладач

Радченко Костянтин

Олександрович

варіант № 4

Київ 2021

Тема

Проектування бази даних та ознайомлення з базовими операціями СУБД PostgreSQL

Мета

Здобути вміння проектування бази даних та практичні навички створення реляційних баз даних за допомогою PostgreSQL.

Завдання

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.

Вимоги:

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 10 або 11.

Хід роботи

На рисунку 1.1 наведено UML діаграму ER-моделі. У якості моделі обрано бібліотеку: таблиця читачів, авторів, абонементів і книг.

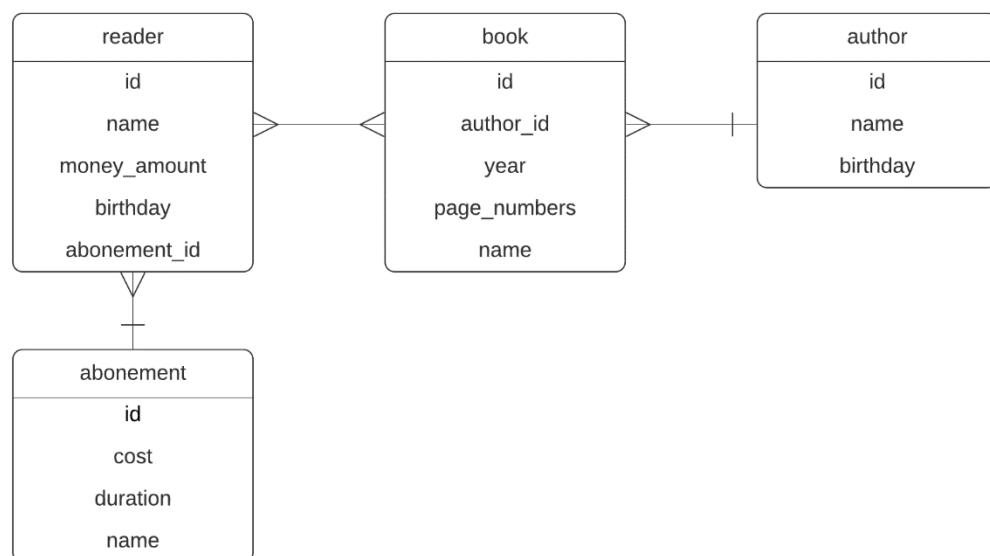


Рис. 1.1

Як помітно, сутності знаходяться у таких відношеннях:

- 1) abonements – reader : one-to-many
- 2) reader – book : many-to-many
- 3) author – book : one-to-many

Нижче зображено графічне представлення нормалізованих до третьої нормальної форми таблиць (для відношення many-to-many створена додаткова таблиця readers_books):

	 id [PK] bigint 	name character varying (50) 	money_amount integer 	birthday date 	abonement_id integer 
1	1	Taras Shevchenko	1000	1814-03-09	3
2	2	Some Person	100	2000-02-01	1
3	3	Olexander Kovalenko	500	1975-11-21	2

Рис. 1.2 – readers







	 id [PK] bigint 	year integer 	pages integer 	author_id integer 	name text 
1	1	2012	900	2	Mathematical analysis. Part 1
2	2	2013	700	2	Mathematical analysis. Part 2
3	3	2015	1200	2	Mathematical analysis. Part 3
4	4	2016	200	3	Dota 2: Tutorial
5	5	2017	400	3	Dota 2: Lining
6	6	2017	800	3	Dota 2: Fights
7	7	2017	800	3	Dota 2: 12K MMR

Рис. 1.3 – books





	 id [PK] bigint 	birthday date 	name text 
1	1	1980-02-01	Anton Vasilenko
2	2	2002-10-16	Danylo Shalack
3	3	2002-10-02	Roman Tymoshchuk

Рис. 1.4 – authors






	 id [PK] bigint 	cost integer 	duration integer 	name text 
1	1	100	1	Normal
2	2	500	6	Premium
3	3	800	12	VIP

Рис. 1.5 – abonements




	id bigint 	reader_id bigint 	book_id bigint 
1	1	2	3
2	2	1	3
3	3	3	1

Рис. 1.6 – readers_books

Висновок

У даній роботі було досліджено взаємодію з СУБД PostgreSQL за допомогою pgAdmin 4. Вивчено 3 форми нормалізації даних у таблицях баз даних. Здобуто практичні навички створені UML-діаграм ER-моделей і подальшого їх перетворення в SQL-запит для створення відповідної таблиці в базі даних.