



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: *«Створити БД "Бібліотека" в СУБД PostgreSQL  
з допомогою конструктора PgAdmin 4»*

Виконав: студент 3 курсу

ФПМ групи КВ-83

Глеб В. Ю.

Перевірів: Павловський В.І.

Київ – 2020

Лабораторна робота №1.  
**Створити БД "Бібліотека" в СУБД PostgreSQL  
з допомогою конструктора PgAdmin 4**

**Мета роботи:** створити БД Бібліотека та сформувані обмеження цілісності на значення даних.

**Порядок виконання роботи**

1. Розробити концептуальну модель вибраного предметного середовища. Концептуальну модель предметного середовища Бібліотека наводиться в Додатку А до лабораторної роботи;
2. Розробити логічну модель (схему) даних БД. Логічна модель (схема) даних БД Бібліотека наводиться в Додатку ? до лабораторної роботи;
3. Вивчити склад та правила роботи з СУБД PostgreSQL;
4. Створити в СУБД PostgreSQL БД Бібліотека, використовуючи конструктори таблиць та стовпчиків (краще колонок). Схема даних БД Бібліотека наводиться в Додатку Б до лабораторної роботи. Перелік атрибутів наводиться в Додатку В до лабораторної роботи;
5. Сформувані обмеження цілісності, що забезпечують:
  - унікальність та обов'язковість вводу первинних ключів для всіх таблиць;
  - перевірка на відповідність зовнішніх ключів таблиць;
  - обмеження на значення даних для атрибутів "expiredln", "name", "sex", "age", "surname" і вивід відповідних повідомлень при їх порушенні (опис всіх атрибутів наводиться нижче в Додатку В до лабораторної роботи);
  - обов'язковість вводу атрибутів "expiredln", "name", "sex", "age", "surname" та інших.
6. Фізична модель (схема) даних БД Бібліотека наводиться в Додатку ? до лабораторної роботи;
7. Заповнити створену БД даними (порядку 5-10 записів в кожній таблиці).

**Зміст звіту**

1. Склад СУБД PostgreSQL.
2. Концептуальна модель предметної області.
3. Логічна модель (схема) БД.
4. Склад обмежень цілісності в термінах СУБД PostgreSQL.
5. Фізична модель БД в термінах СУБД PostgreSQL.
6. Представлення БД в pgAdmin 4

## Додаток А. Концептуальна модель предметної області "Бібліотека"

В концептуальній моделі предметної області "Бібліотека" (Рисунок 1) виділяються наступні сутності та зв'язки між ними.

Сутність "Читачі" з атрибутами: id, ім'я, фамілія, вік, стать;

Сутність "Абонементи" з атрибутами : "термін дії" та "id";

Сутність "Книги з атрибутами "автори" і "назва";

Сутність "Автори" з атрибутами : id, Ім'я, Фамілія, Живий, Національність;

Між сутностями "Читачі" та "Абонементи" зв'язок R(1:1), тому що 1 читач може мати тільки 1 абонемент.

1 абонемент може мати багато книг, але 1 книга може бути тільки в 1 абонементі. Тому між сутностями "Абонементи" і "Книги" зв'язок R(1:M).

Автори пишуть книги, книга може мати одного автора, багато авторів або бути без автора. Але один автор може мати багато книг. writes

Отже тут відношення R(M:N)

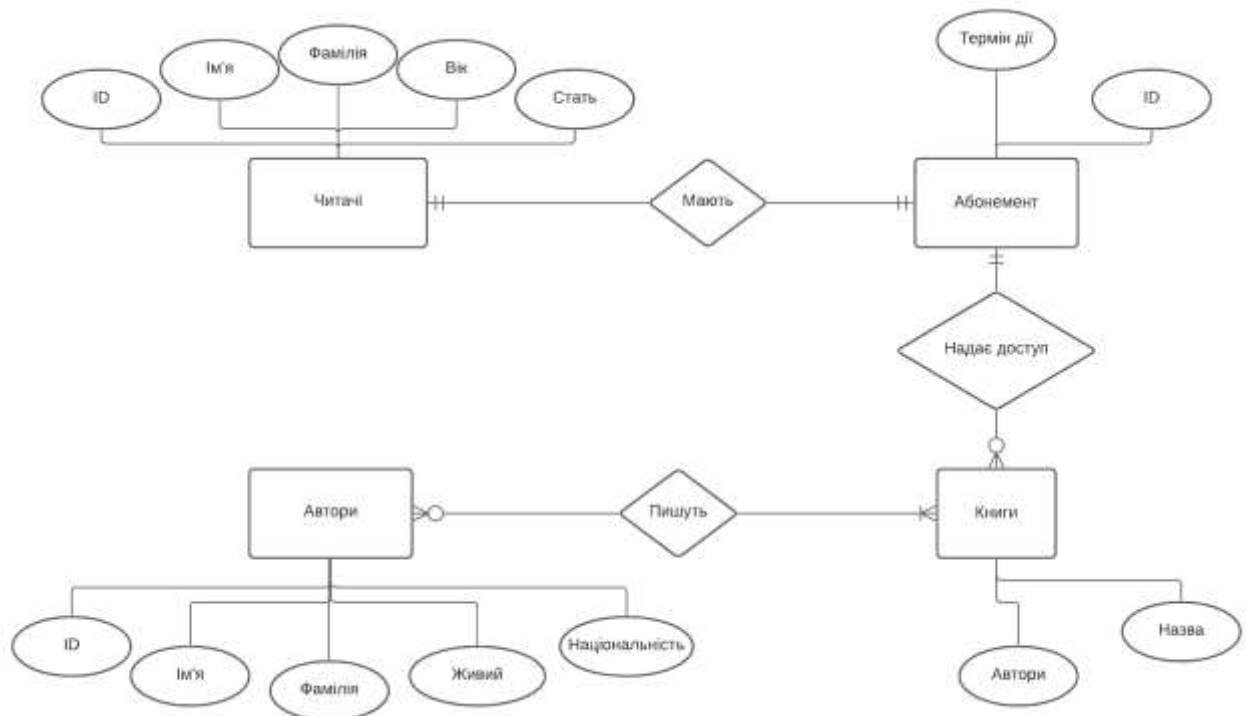


Рисунок 1 - Концептуальна модель предметної області "Бібліотека"

## Додаток Б. Логічна модель (схема) БД "Бібліотека"

В логічній моделі (Рисунок 2):

Сутність "Читачі" перетворена в таблицю "Readers".

Сутність "Абонементи" була перетворена в таблицю "Abonements".

Сутність "Книги" була перетворена в таблицю "Books".

Сутність "Автори" була перетворена в таблицю "Writers".

Оскільки відношення між авторами і книгами R(M:N), потрібно створити ще одну перехідну таблицю з авторами і їх книгами під назвою "Writes".

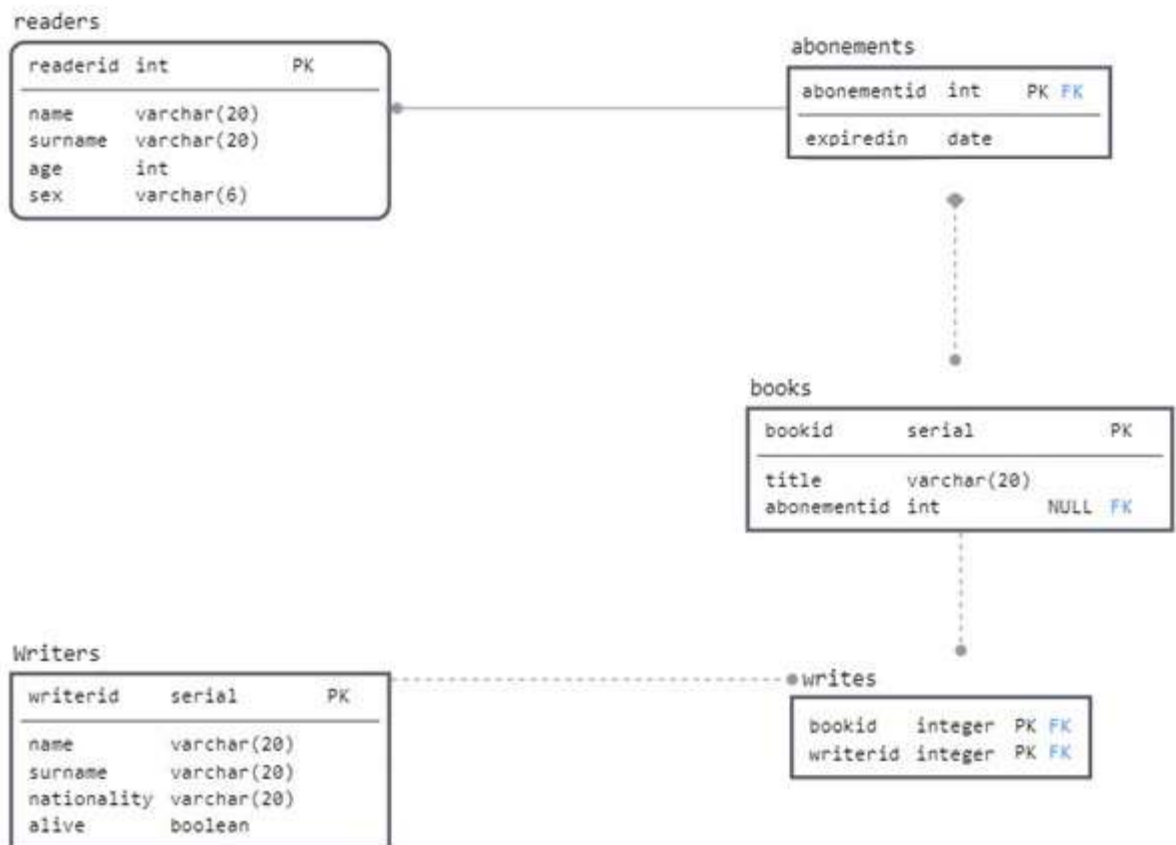


Рисунок 2 - Логічна модель предметної області "Бібліотека"

Схема бази даних відповідає 1НФ тому, що всі рядки унікальні, всі атрибути прості і не мають нереляційних структур(масивів, об'єктів і т.д..

Схема бази даних відповідає 2НФ тому, що всі таблиці мають первинні ключі, і всі атрибути описують первинний ключ повністю.

Схема бази даних відповідає 3НФ тому, що всі атрибути залежні тільки від первинного ключа.

## Додаток В. Структура БД “Бібліотека”

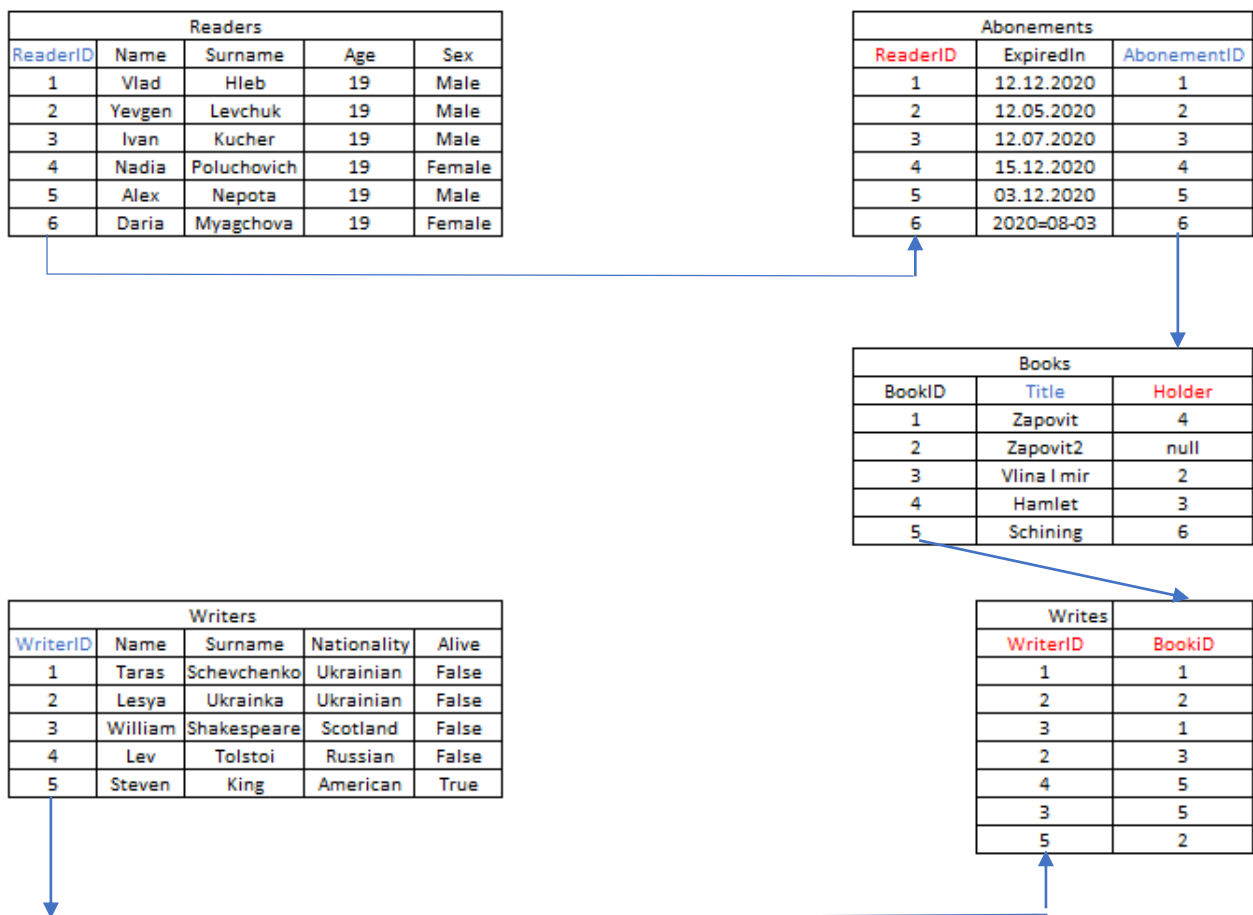
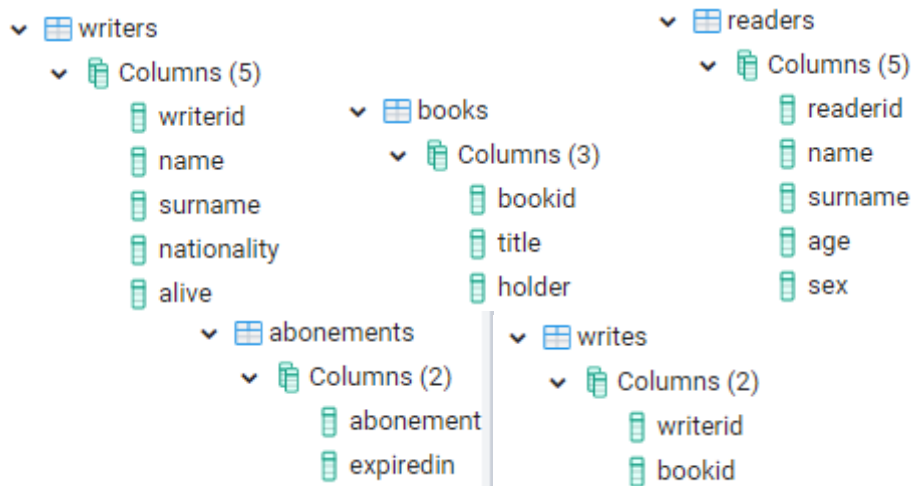


Рисунок 3 - Структурна модель предметної області "Бібліотека"

**Додаток Г. Опис структури БД "Бібліотека"**  
Текстове представлення логічної моделі (схеми) БД

Відношення	Атрибут	Тип
Відношення "Readers" містить інформацію про читачів в бібліотеці	<b>ReaderID</b> – унікальний номер читача в БД Name – ім'я читача. Не допускає NULL. Surname – прізвище читача. Не допускає NULL. Age – вік читача. Не допускає NULL Sex – стать читача.	Числовий, SERIAL PK Текстовий(20) Текстовий(20) Числовий Текстовий(6)
Відношення "Writers" містить інформацію про авторів	<b>WriterID</b> – унікальний номер автора в БД. Name – ім'я автора. Не допускає NULL. Surname – прізвище автора. Не допускає NULL. Nationality – національність автора. Не допускає NULL. Alive – являється автор живим (чи може написати ще роботи). Не допускає NULL.	Числовий, SERIAL PK Текстовий(20) Текстовий(20) Текстовий(20) Двійковий
Відношення "Abonements" містить інформацію про абонементи і їх валідність	<b>AbonementID</b> – унікальний номер абонементу в БД. ExpiredIn – кінцева дата валідності. Не допускає NULL <b>ReaderID</b> – унікальний номер студента, який володіє абонементом. Не допускає NULL	Числовий, SERIAL PK Дата DATE Числовий FK
Відношення "Books" містить інформацію про книжки їх наявність в бібліотеці, назву, і кому вони видані.	<b>BookID</b> - унікальний номер книжки в БД <b>Title</b> – Назва книжки. Не допускає NULL <b>Holder</b> - ідентифікатор абонементу, який взяв книгу в бібліотеці. Може містити NULL, якщо книжка знаходиться в бібліотеці.	Числовий, SERIAL PK Текстовий(20) Текстовий(20) FK Числовий
Відношення "Writes" містить інформацію про авторів кожної книги.	<b>WriterID</b> – унікальний номер автора. Не допускає NULL <b>Title</b> – Назва книжки. Не допускає NULL	Числовий SERIAL PK FK Текстовий(20) PK

## Додаток Г. Структура БД "Бібліотека" в pgAdmin 4



## Опис таблиць БД "Бібліотека" в pgAdmin 4

### Таблиця "Readers"

```

create table readers(
    readerid SERIAL primary key,
    name varchar(20) NOT NULL,
    surname varchar(27) NOT NULL,
    age INT not null,
    sex varchar(6) not null
);

```

	readerid [PK] integer	name character varying (20)	surname character varying (27)	age integer	sex character varying (6)
1	1	Vlad	Hleb	19	Male
2	2	Yevgen	Levchuk	19	Male
3	3	Ivan	Kucher	19	Male
4	4	Nadia	Poluchovich	19	Female
5	5	Alex	Nepota	19	Male
6	6	Daria	Myagchova	19	Female

### Таблиця "Abonements"

```

create table abonements(
    abonementid int primary key,
    expiredin DATE not null,
    foreign key(abonementid) references readers(readerid)
);

```

	abonementid [PK] integer	expiredin date
1	1	2020-12-12
2	2	2020-05-12
3	3	2020-07-12
4	4	2020-12-15
5	5	2020-12-03

Таблиця “Writers”

```
create table writers(
  writerid serial PRIMARY KEY,
  name varchar(20) not null,
  surname varchar(20) not null,
  nationality varchar(20) not null,
  alive boolean not null
);
```

	writerid [PK] integer	name character varying (20)	surname character varying (20)	nationality character varying (20)	alive boolean
1	1	Taras	Shevchaenko	Ukrainian	false
2	2	Lesya	Ukrainka	Ukrainian	false
3	3	William	Shakespeare	Scotland	false
4	4	Lev	Tolstoi	Russian	false
5	5	Steven	Sking	American	true

Таблиця “Writes”

	writerid [PK] integer	bookid [PK] integer
1	1	1
2	2	2
3	1	3
4	2	3
5	3	4
6	4	3

Таблиця “Books”



```

create table books(
  bookid serial primary key,
  title varchar(20) not null,
  holder int,
  foreign key(holder) references abonements(abonementid)
);

```

	bookid [PK] integer	title character varying (20)	holder integer
1	1	Zapovit	4
2	2	Romeo and Juliet	[null]
3	3	Shining	[null]
4	4	Hamlet	[null]
5	5	War and peace	[null]