



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 III курсу
групи KB-93

Вітковський В.Б.

Перевірив: Павловський В.І.

Київ – 2021

Мета роботи

Здобуття вмінь проектування бази даних та практичних навичок створення реляційних баз даних за допомогою PostgreSQL.

Завдання лабораторної роботи

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.

Модель «сутність-зв'язок» Сервіс для прослуховування музики

Обрана предметна галузь передбачає можливість користувачеві вибрати зі списку пісні, та додавати їх у власну бібліотеку. Згідно цієї області для побудови бази даних було виділено наступні сутності:

1. Сутність “customer” містить такі атрибути як ID, ім'я, пошта, та інформація про те, має користувач підписку на сервіс, чи ні. Слугує для зберігання інформації про користувача.

2. Сутність “song” містить такі атрибути як ID, автор, альбом, ім'я, тривалість пісні, рік виходу. Слугує для зберігання інформації про пісню.

3. Сутність “song_photo” містить такі атрибути як ID, адрес зображення, ідентифікатор пісні до якої воно відноситься. Слугує для зберігання інформації про фото до пісні.

4. Сутність “customer_library” містить такі атрибути як ID, дата коли була додана перша пісня до бібліотеки, ідентифікатор користувача до якої вона відноситься. Слугує для зберігання інформації про бібліотеку користувача.

5. Сутність “customer_library_songs” містить такі атрибути як ID, ідентифікатор бібліотеки до якої вона відноситься, ідентифікатор пісень які належить бібліотеці. Слугує для зберігання інформації про пісні, які знаходяться в бібліотеці.

Опис зв’язків

Клієнт може мати одну бібліотеку, тому між сутностями “customer” і “customer_library” існує зв’язок 1:1. Бібліотека може мати тільки один каталог пісень, тому між сутностями “customer_library” і “customer_library_songs” існує зв’язок 1:1. Бібліотека пісень може мати багато пісень, а пісня може входити в багато бібліотек, тому між сутностями “customer_library_songs” і “song” існує зв’язок М:М. Одне фото може відноситись до багатьох пісень, а одна пісня має тільки одне фото, тому між сутностями “song” і “song_photo” існує зв’язок 1:М.

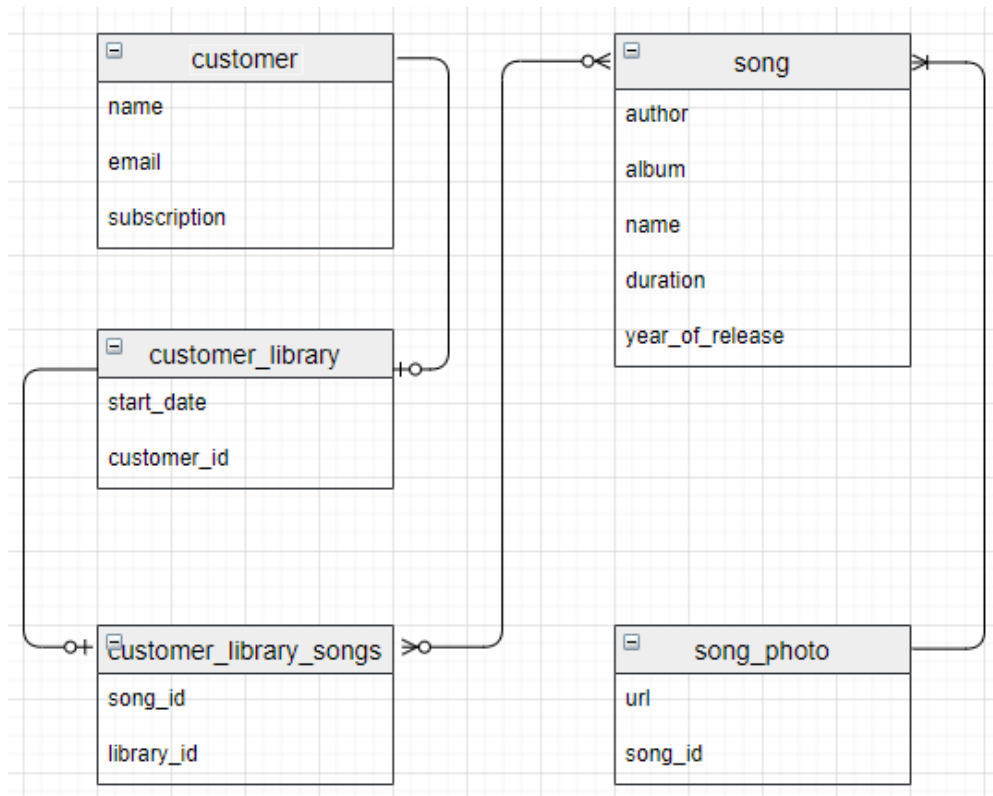


Рисунок 1. Концептуальна модель предметної області «Сервіс для прослуховування музики» (засобами draw.io)

Опис структури БД «Сервіс для прослуховування музики»

Відношення	Атрибут	Тип (розмір)
Відношення “customer” Вміщує інформацію про користувача сервісу	id - унікальний ID користувача name - ім'я користувача email - електронна пошта користувача subscription - інформація про те, має користувач підписку на сервіс, чи немає	Serial primary key Текстовий (255) Текстовий (255) Булевий тип (0,1)
Відношення “song” Вміщує інформацію про пісню, доступну на сервісі	id - унікальний ID пісні author - ім'я автора пісні album - назва альбому, з якого пісня name - назва пісні duration - тривалість пісні (в секундах) year_of_release - рік виходу пісні	Serial primary key Текстовий (255) Текстовий (255) Текстовий (255) Числовий Текстовий (255)

Відношення “song_photo” Вміщує інформацію про фото до пісні	id - унікальний ID фото пісні url - адреса зображення song_id - атрибут, який посилається на ID пісні	Serial primary key Текстовий (255) Числовий
Відношення “customer_library” Вміщує інформацію про бібліотеку пісень користувача	id - унікальний ID бібліотеки start_date - дата коли була додана перша пісня до бібліотеки customer_id - атрибут, який посилається на ID користувача	Serial primary key Числовий Числовий
Відношення “customer_library_songs” Вміщує інформацію про пісні, які є у бібліотеці користувача	id - унікальний ID користувача library_id - атрибут, який посилається на ID бібліотеки song_id - атрибут, який посилається на ID пісні	Serial primary key Числовий Числовий

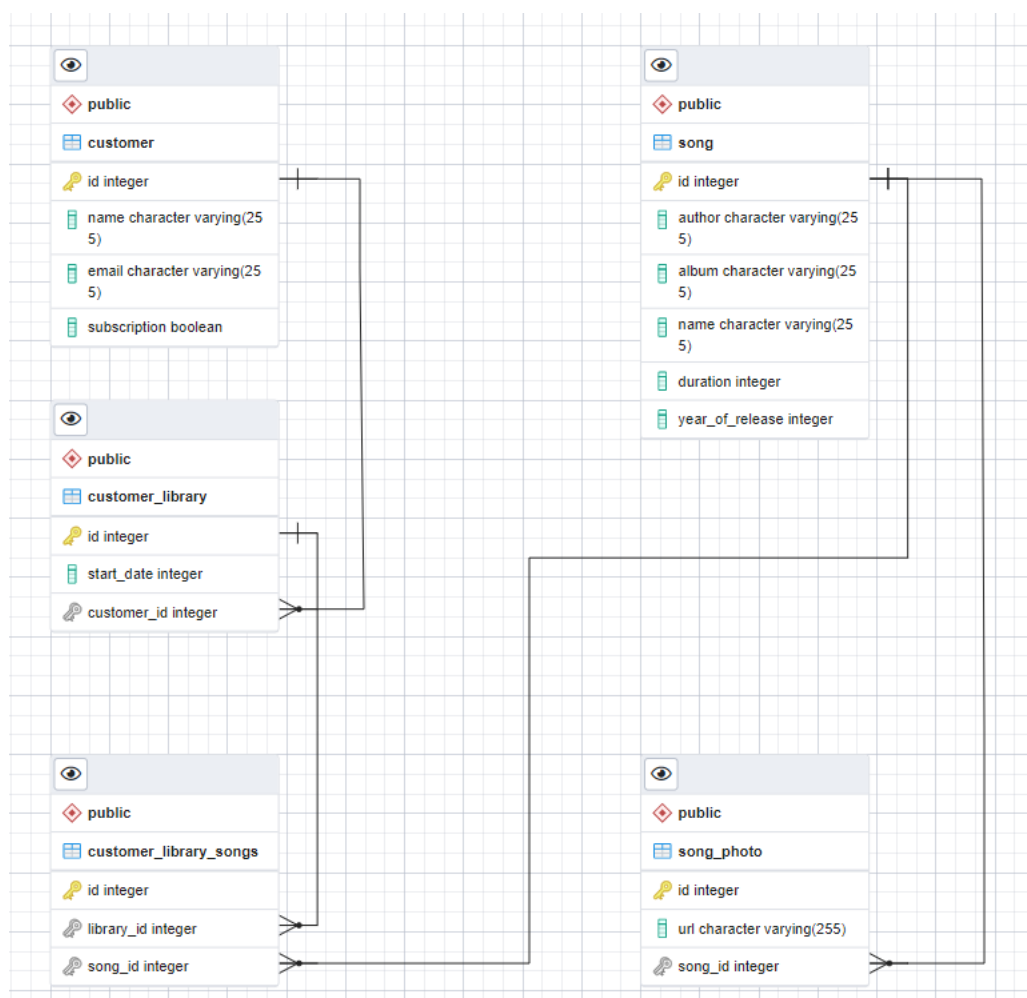


Рисунок 2. Схема БД «Сервіс для прослуховування музики» у pgAdmin 4

Нормалізація схеми бази даних до третьої нормальної форми

Таблиці бази даних «Сервіс для прослуховування музики» відповідають першій нормальній формі оскільки в них немає повторюючих рядків та масивів даних.

Таблиці бази даних відповідають другій нормальній формі, тому що вони відповідають першій нормальній формі та кожна таблиця має первинний ключ (Primary key), від якого залежать всі неключові атрибути.

Таблиці бази даних відповідають третій нормальній формі, тому що вони відповідають другій нормальній та не мають транзитивних залежностей для не ключових атрибутів.

Таблиці бази даних «Сервіс для прослуховування музики» у pgAdmin4

customer

customer

General

Columns

Advanced

Constraints

Parameters

Security

SQL

Inherited from table(s)

Select to inherit from...

Columns

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
	id	integer			<div>Yes</div>	<div>Yes</div>
	name	character varying	255		<div>Yes</div>	<div>No</div>
	email	character varying	255		<div>Yes</div>	<div>No</div>
	subscription	boolean			<div>Yes</div>	<div>No</div>

	id [PK] integer	name character varying (255)	email character varying (255)	subscription boolean
1	1	Andriy	andriy@gmail.com	true
2	2	Vitaliy	vitaliy@gmail.com	false
3	3	Bogdan	bogdan@gmail.com	true
4	4	Oleg	oleg@gmail.com	true

customer_library

customer_library

General Columns Advanced Constraints Parameters Security SQL

Inherited from table(s) Select to inherit from...

Columns

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
	id	integer			Yes	Yes
	start_date	integer			Yes	No
	customer_id	integer			Yes	No

	id [PK] integer	start_date integer	customer_id integer
1	1	191832	1
2	2	191900	3
3	3	192312	4

customer_library_songs

customer_library_songs

Общие Столбцы Дополнительно Ограничения Параметры Безопасность SQL

Наследуется из таблиц(ы) Выберите источник нас

Столбцы

	Имя	Тип данных	Length/Precision	Масштаб	Не NULL?	Первичный ключ?
	id	integer			Да	Да
	library_id	integer			Да	Нет
	song_id	integer			Да	Нет

	id [PK] integer	library_id integer	song_id integer
1	1	1	1
2	2	1	2
3	3	1	3
4	4	1	4
5	5	2	1
6	6	2	2
7	7	3	2
8	8	3	3
9	9	3	4

song













song

Общие Столбцы Дополнительно Ограничения Параметры Безопасность SQL

Наследуется из таблиц(ы)

Выберите источник нас

Столбцы

	Имя	Тип данных	Length/Precision	Масштаб	Не NULL?	Первичный ключ?
 	id	integer			<div>Да</div>	<div>Да</div>
 	author	character varying	255		<div>Да</div>	<div>Нет</div>
 	album	character varying	255		<div>Да</div>	<div>Нет</div>
 	name	character varying	255		<div>Да</div>	<div>Нет</div>
 	duration	integer			<div>Да</div>	<div>Нет</div>
 	year_of_release	integer			<div>Да</div>	<div>Нет</div>

	id [PK] integer	author character varying (255)	album character varying (255)	name character varying (255)	duration integer	year_of_release integer
1		1 Author_1	Album_1	song_1	240	1999
2		2 Author_2	Album_2	song_2	250	2000
3		3 Author_3	Album_3	song_3	260	2001
4		4 Author_4	Album_4	song_4	270	2002

song_photo







song_photo

Общие Столбцы Дополнительно Ограничения Параметры Безопасность SQL

Наследуется из таблиц(ы)

Выберите источник нас

Столбцы

	Имя	Тип данных	Length/Precision	Масштаб	Не NULL?	Первичный ключ?
 	id	integer			<div>Да</div>	<div>Да</div>
 	url	character varying	255		<div>Да</div>	<div>Нет</div>
 	song_id	integer			<div>Да</div>	<div>Нет</div>

	id [PK] integer	url character varying (255)	song_id integer
1		1 song_1 photo	1
2		2 song_2 photo	2
3		3 song_3 photo	3
4		4 song_4 photo	4

Згенерований SQL-текст

```
BEGIN;

CREATE TABLE IF NOT EXISTS customer
(
    id integer NOT NULL,
    name character varying(255) NOT NULL,
    email character varying(255) NOT NULL,
    subscription boolean
);

CREATE TABLE IF NOT EXISTS customer_library
(
    id integer NOT NULL,
    start_date integer NOT NULL
    customer_id integer NOT NULL
);

CREATE TABLE IF NOT EXISTS customer_library_songs
(
    id integer NOT NULL,
    customer_id integer NOT NULL,
    customer_id integer NOT NULL
);

CREATE TABLE IF NOT EXISTS song
(
    id integer NOT NULL,
    address character varying(255) NOT NULL,
    manager_name character varying(50) NOT NULL,
    manager_surname character varying(150) NOT NULL,
    PRIMARY KEY (id)
);

CREATE TABLE IF NOT EXISTS song_photo
(
    id integer NOT NULL,
    author character varying(255) NOT NULL,
    album character varying(255) NOT NULL,
    name character varying(255) NOT NULL,
    duration integer NOT NULL,
    year_of_release integer NOT NULL
    PRIMARY KEY (id)
);

ALTER TABLE customer_library
    ADD FOREIGN KEY (customer_id)
    REFERENCES customer (id)
    NOT VALID;

ALTER TABLE customer_library_songs
    ADD FOREIGN KEY (library_id)
    REFERENCES customer_library (id)
    NOT VALID;

ALTER TABLE customer_library_songs
    ADD FOREIGN KEY (song_id)
    REFERENCES song (id)
    NOT VALID;
```

```
ALTER TABLE song_photo
  ADD FOREIGN KEY (song_id)
    REFERENCES song (id)
  NOT VALID;

END;
```