

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 ІII курсу групи КВ-92

Михайлов О. Г.

Перевірив:

Петрашенко А.В.

Київ – 2021

*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 кожну з таблиць засобами pgAdmin 4.

**Модель «сутність-зв’язок»:**

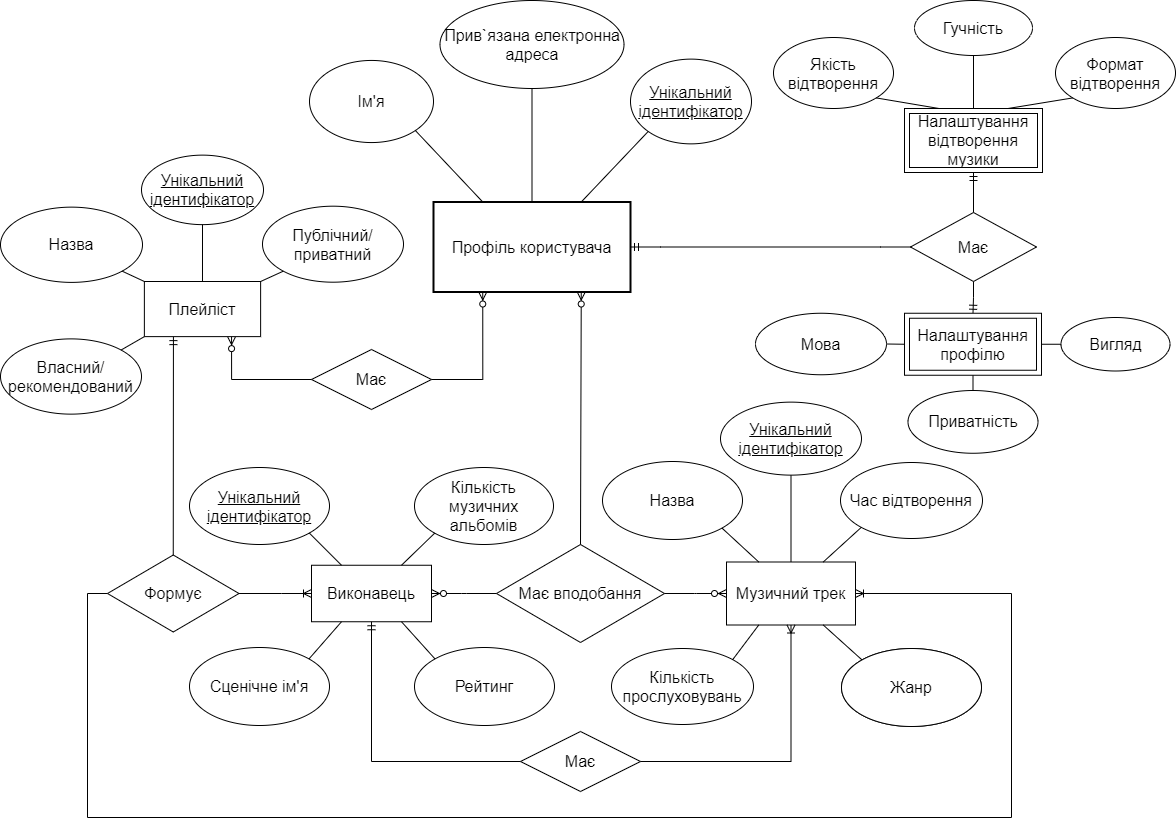


Рис1. ER-діаграма за нотацією Чена

Сервіс потокового аудіо, згідно якого побудована база даних з сутностями:

1. Профіль користувача, з атрибутами:

* Унікальний ідентифікатор
* Ім`я
* Прив`язка електронної пошти до акаунту

1. Налаштування відтворення музики, з атрибутами:

* Якість відтворення
* Гучність
* Формат відтворення

1. Налаштування профілю, з атрибутами:

* Мова
* Вигляд
* Приватність профілю

1. Плейліст, з атрибутами:

* Унікальний ідентифікатор
* Назва
* Приватність плейлісту
* Створювач плейлісту

1. Виконавець, з атрибутами:

* Унікальний ідентифікатор
* Кількість музичних альбомів
* Сценічне ім`я
* Рейтинг

6) Музичний трек, з атрибутами:

- Унікальний ідентифікатор

- Назва

- Жанр

- Час відтворення

- Кількість прослуховувань

**Опис зв’язків:**

Профіль користувача має налаштування відтворення музики та налаштування профілю, з якими відповідно утворюються зв`язки 1:1 та 1:1, оскільки одному профілю повинен відповідати унікальний шаблон раніше визначених налаштувань.

Також профіль користувача має вподобання серед виконавців та музичних треків. Оскільки багатьом профілям можуть сподобатися нуль або багато виконавців та музичних треків, то відповідно утворюються з`язки М:N та М:N.

В свою чергу, оскільки виконавець зобов`язаний мати хоча б один музичний трек, а музичний трек — тільки одного виконавця, то між сутностями виконавець та музичний трек утворюється зв`язок 1:N.

Вподобані користувачем виконавці та музичні треки є унікальними складовими сутності плейліст, тобто: кожен виконавець та його музичний трек може з`явитися тільки у одному плейлісті. Отже між сутністю плейліст та сутностями виконавець та музичний трек відповідно попарно утворюються зв`язки 1:N та 1:N.

Оскільки плейлісти створюються користувачами або рекомендуються їм на основі раніше вподобаних виконавців та музичних треків, то сутність профіль утворює з сутністю плейліст зв`язок М:N.

**Опис процесу перетворення:**

- Сутність Профіль користувача була перетворена в таблицю User profile.

- Сутність Налаштування відтворення музики була перетворена в таблицю Music settings, а її зв`язок з сутністю Профіль користувача забезпечується первинним ключем таблиці Music settings.

- Сутність Налаштування профілю було перетворено в таблицю Profile settings, її зв`язок з сутністю Профіль користувача забезпечується первинним ключем таблиці Profile settings.

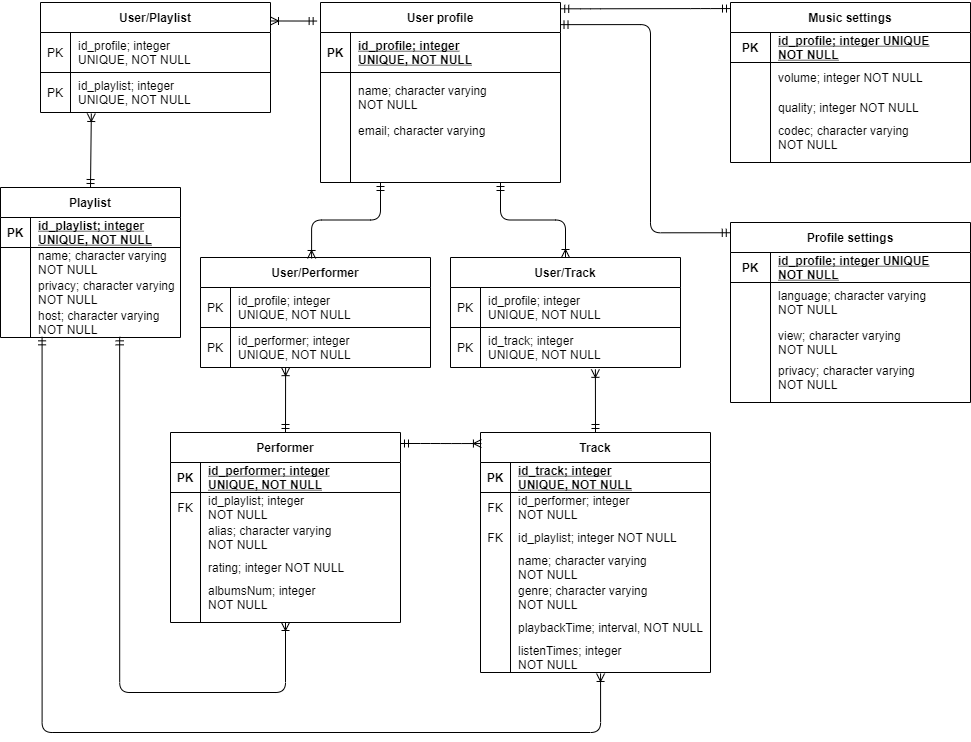
- Сутність Виконавець була перетворена в таблицю Performer. Її зв`язок з сутністю Профіль користувача забезпечується таблицею декомпозиції User/Performer, а зв`язок з сутністю Плейліст - зовнішнім ключем id\_playlist.

- Сутність Музичний трек була перетворена в таблицю Track. Її зв`язок з сутністю Профіль користувача забезпечується таблицею декомпозиції User/Track, з сутністю Виконавець - зовнішнім ключем id\_performer, а з сутністю Плейліст - зовнішнім ключем id\_playlist.

- Сутність Плейліст була перетворена в таблицю Playlist. Її зв`язок з сутністю Профіль користувача забезпечується таблицею декомпозиції User/Performer.

**Схеми бази даних:**

Рис2. Схема бази даних



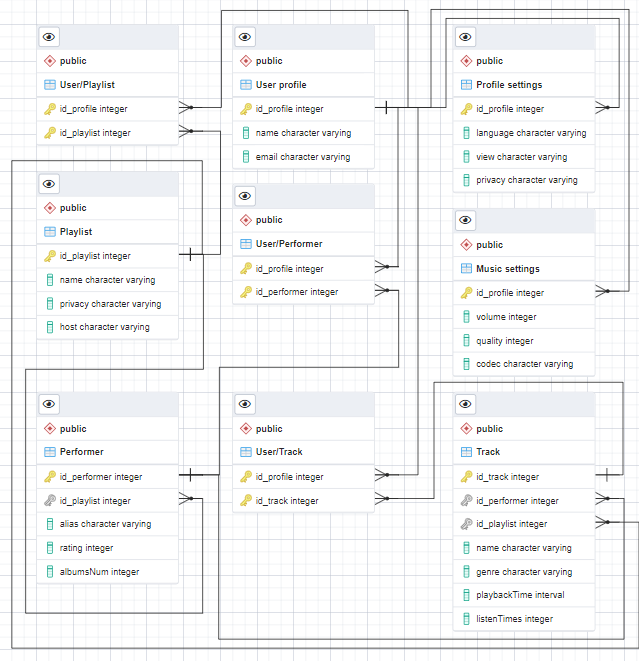


Рис3. Схема бази даних в pgAdmin4

|  |  |  |
| --- | --- | --- |
| Сутність | Атрибут | Тип атрибуту |
| **User profile** – містить інформацію про користувача | **id\_profile** – первинний ключ, унікальний ідентифікатор профілю  **name** – ім`я користувача  **email** – прив`язана електронна пошта | **integer**  **character varying**  **character varying** |
| **Music settings** – містить інформацію про налаштування музики | **id\_profile** – первинний ключ, що містить значення унікального ідентифікатора профілю  **volume** – гучність  **quality** – якість відтворення  **codec** – формат відтворення | **integer**  **integer**  **integer**  **character varying** |
| **Profile settings** –  містить інформацію про налаштування профілю | **id\_profile –** первинний ключ, що містить значення унікального ідентифікатора  **language** – мова профілю  **view** – зовнішній вигляд акаунту  **privacy –** приватність профілю | **integer**  **character varying**  **character varying**  **character varying** |
| **Performer** – містить інформацію про виконавця | **id\_performer** – первинний ключ, унікальний ідентифікатор виконавця  **id\_playlist** – ідентифікатор плейлісту, до якого належить виконавець  **alias** – сценічне ім`я виконавця  **rating** – рейтинг виконавця  **albumsNum** – кількість музичних альбомів виконавця | **integer**  **integer**  **character varying**  **integer**  **integer** |
| **User/Performer –** містить інформацію про відповідність між користувачами та вподобаними виконавцями | **id\_profile** – первинний ключ, унікальний ідентифікатор профілю  **id\_performer** – первинний ключ, унікальний ідентифікатор виконавця | **integer**  **integer** |
| **Track** – містить інформацію про музичні треки | **id\_track** – первинний ключ, унікальний ідентифікатор музичного треку  **id\_performer –** ідентифікатор виконавця, якому належить музичний трек  **id\_playlist** – ідентифікатор плейлісту, до якого належить трек  **name** – назва музичного треку  **genre** – назва жанру треку  **playbackTime** – час відтворення треку  **listenTimes** – кількість прослуховувань | **integer**  **integer**  **integer**  **character varying**  **character varying**  **interval**  **integer** |
| **User/Track –** містить інформацію про відповідність між користувачами та вподобаними треками | **id\_profile** – первинний ключ, унікальний ідентифікатор профілю  **id\_track** – первинний ключ, унікальний ідентифікатор музичного треку | **integer**  **integer** |
| **Playlist** – містить інформацію про плейлісти | **id\_playlist** – первинний ключ, унікальний ідентифікатор плейлісту  **name** – назва плейлісту  **privacy** — приватність плейлісту  **host** – створювач плейлісту (власний/рекомендований) | **integer**  **character varying**  **character varying**  **character varying** |
| **User/Playlist –** містить інформацію про відповідність між користувачами та плейлістами | **id\_profile** – первинний ключ, унікальний ідентифікатор профілю  **id\_playlist** – первинний ключ, унікальний ідентифікатор плейлісту | **integer**  **integer** |

///////////////////////////////////////////////////////

**Нормалізація схеми бази даних до третьої нормальної форми**

**Функціональні залежності:**

**User profile**:

*id\_profile → name, email*

id\_ profile → name(ім`я профілю зв’язане з його унікальним ідентифікатором)

id\_ profile → email(електронна пошта зв’язана з унікальним ідентифікатором профілю)

name → email

id\_ profile → name → email(транзитивна залежність з ключовим атрибутом)

email → name

id\_ profile → email → name(транзитивна залежність з ключовим атрибутом)

**Music settings**:

*id\_ profile → volume, quality, codec*

id\_ profile → volume(гучність відтворення музики прив’язана до унікального ідентифікатора профілю)

id\_ profile → quality(якість відтворення музики зв’язана з унікальним ідентифікатором профілю)

id\_ profile → codec(формат відтворення музики зв’язаний з унікальним ідентифікатором профілю)

codec → quality

id\_ profile → codec → quality(транзитивна залежність з ключовим атрибутом)

**Profile settings**:

*id\_ profile → language, view, privacy*

id\_ profile → language(мова профілю зв’язана з його унікальним ідентифікатором)

id\_ profile → view(вид профілю зв’язаний з його унікальним ідентифікатором)

id\_ profile → privacy(налаштування приватності профілю зв’язані з його унікальним ідентифікатором)

**Performer**:

*id\_ performer → alias, rating, albumNum*

id\_ performer → alias(сценічне ім`я залежить від унікального ідентифікатору виконавця)

id\_ performer → rating(рейтинг залежить від унікального ідентифікатору виконавця)

id\_ performer → albumNum(кількість музичних альбомів залежить від унікального ідентифікатору виконавця)

alias → rating

id\_ performer → alias → rating(транзитивна залежність з ключовим атрибутом)

alias → albumNum

id\_ performer → alias → albumNum(транзитивна залежність з ключовим атрибутом)

albumNum → rating

id\_ performer → albumNum → rating(транзитивна залежність з ключовим атрибутом)

rating → albumNum

id\_ performer → rating → albumNum(транзитивна залежність з ключовим атрибутом)

**Track**:

*id\_track → name, genre, playbackTime, listenTimes*

id\_ track → name(назва треку зв’язана з його унікальним ідентифікатором)

id\_ track → genre(жанр треку залежить від його унікального ідентифікатора)

id\_ track → playbackTime(час відтворення залежить від унікального ідентифікатора треку)

id\_ track → listenTimes(кількість прослуховувань залежить від унікального ідентифікатора треку)

name → playbackTime

id\_ track → name → playbackTime(транзитивна залежність з ключовим атрибутом)

name → listenTimes

id\_ track → name → listenTimes(транзитивна залежність з ключовим атрибутом)

**Playlist**:

*id\_ playlist → name, privacy, host*

id\_ playlist → name(ім`я плейлісту зв’язане з його унікальним ідентифікатором)

id\_ playlist → privacy(налаштування приватності плейлісту зв’язані з його унікальним ідентифікатором)

id\_ playlist → host(створювач плейлісту зв’язаний з його унікальним ідентифікатором)

host → name

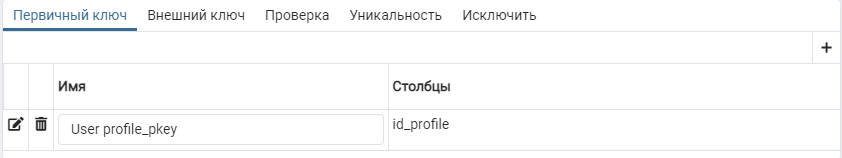
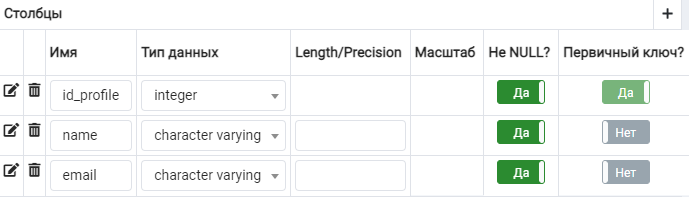
id\_ playlist → host → name(транзитивна залежність з ключовим атрибутом)

host → privacy

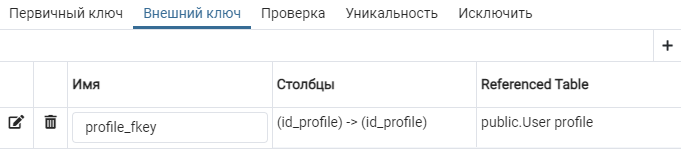
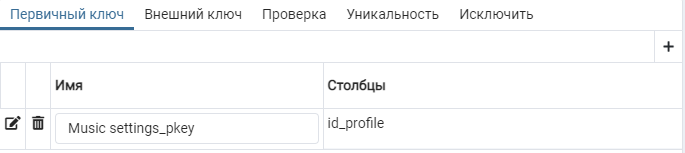
id\_ playlist → host → privacy(транзитивна залежність з ключовим атрибутом)

**Копії екрану з pgAdmin4:**

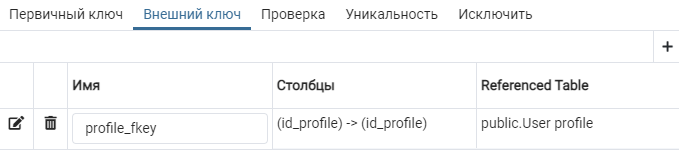
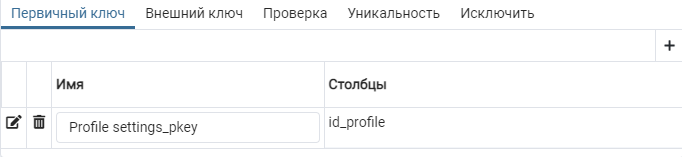
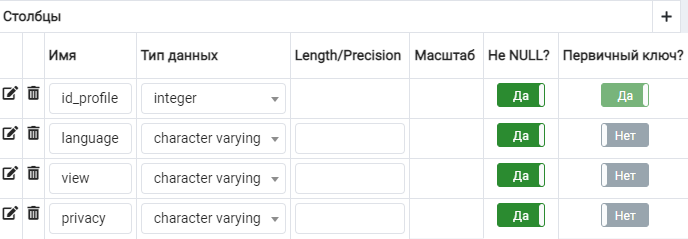
**User profile:**

****

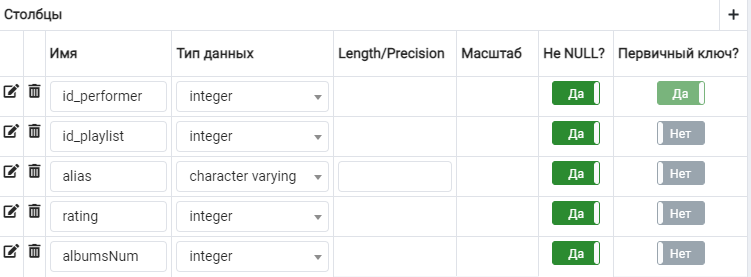
**Music settings:**

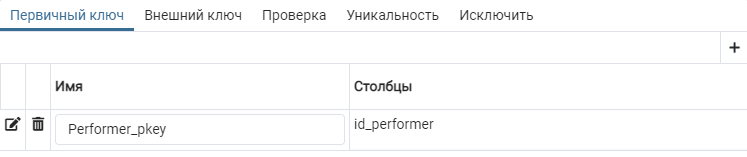
****

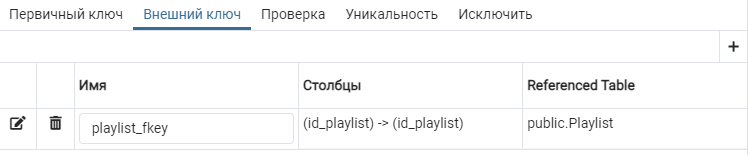
**Profile settings:**

****

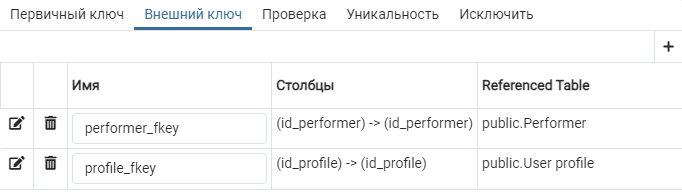
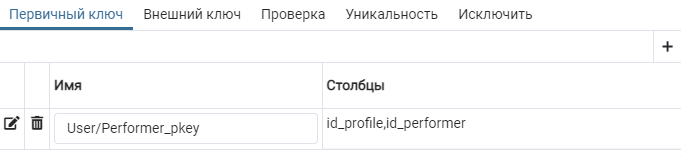
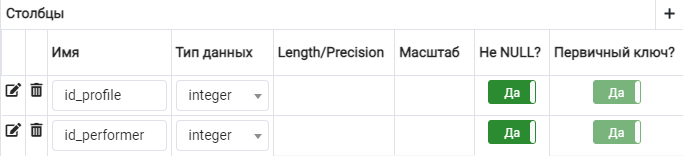
**Performer:**

****

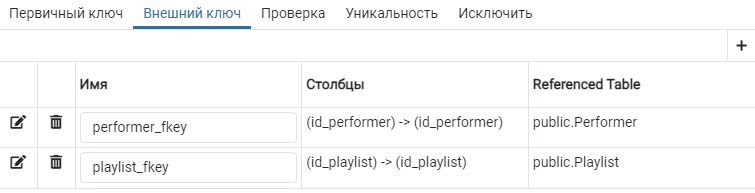
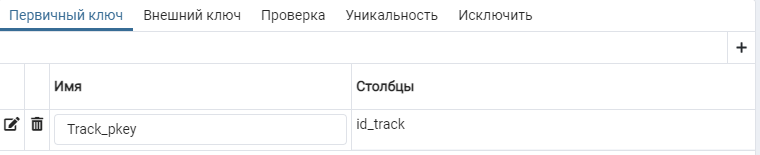
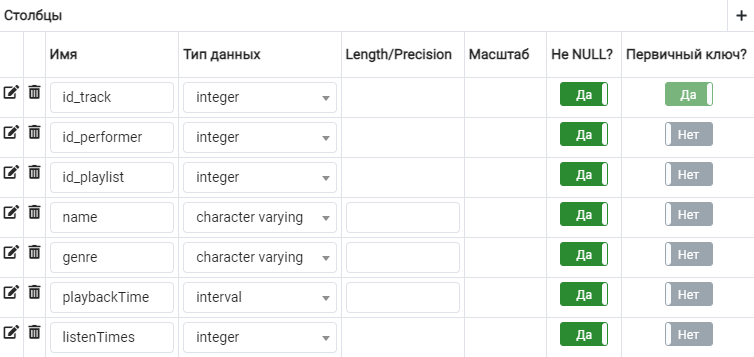
****

****

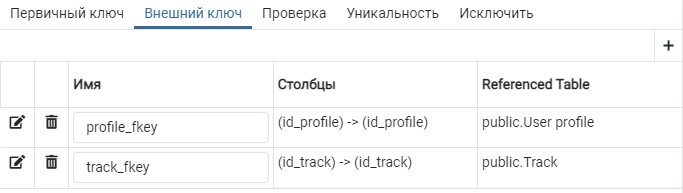
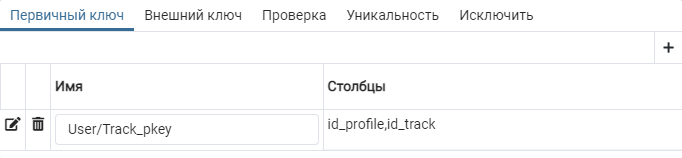
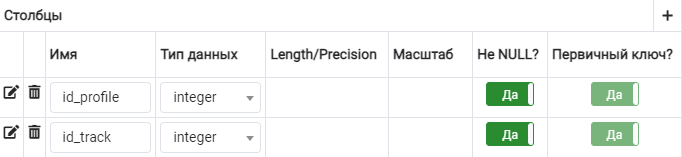
**User/Performer:**

****

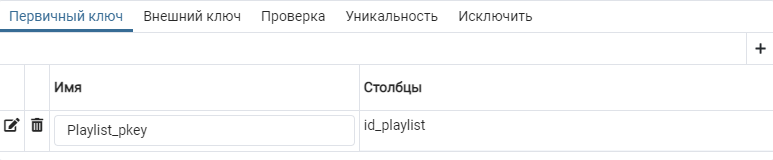
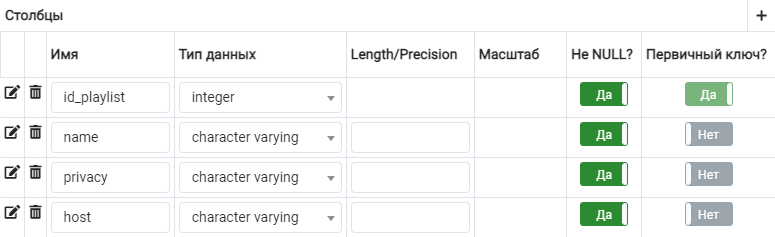
**Track:**

****

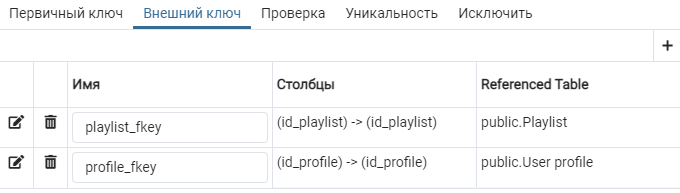
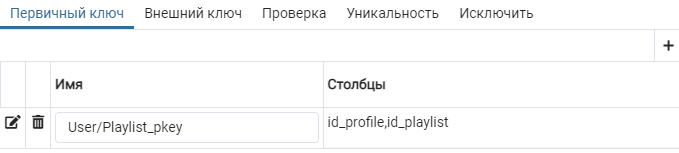
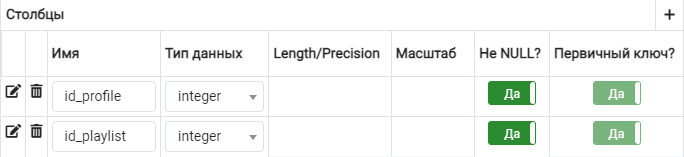
**User/Track:**

****

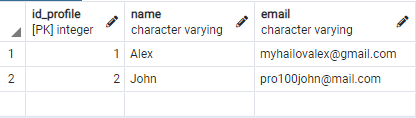
**Playlist:**

****

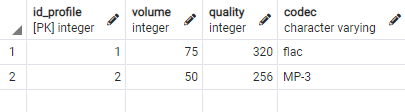
**User/Playlist:**

****

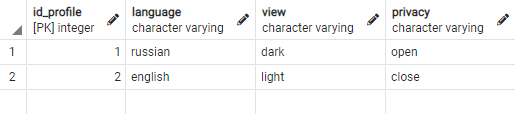
**User profile:**

****

**Music settings:**

****

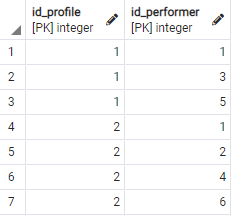
**Profile settings:**

****

**Performer:**

****

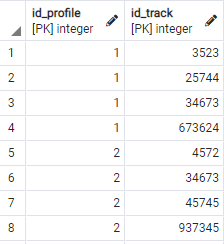
**User/Performer:**

****

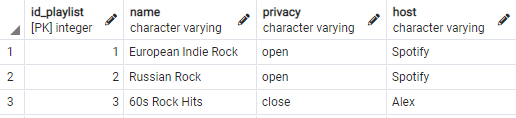
**Track:**

****

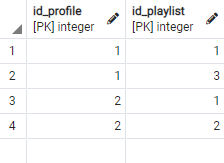
**User/Track:**

****

**Playlist:**

****

**User/Playlist:**

****