“САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ” (СПбГУ)

Математико-механический факультет

“Искусственный интеллект и наука о данных”

Итоговый проект по предмету “Основы баз данных”

Хрисанкова София, 23Б16-мм

Преподаватель:

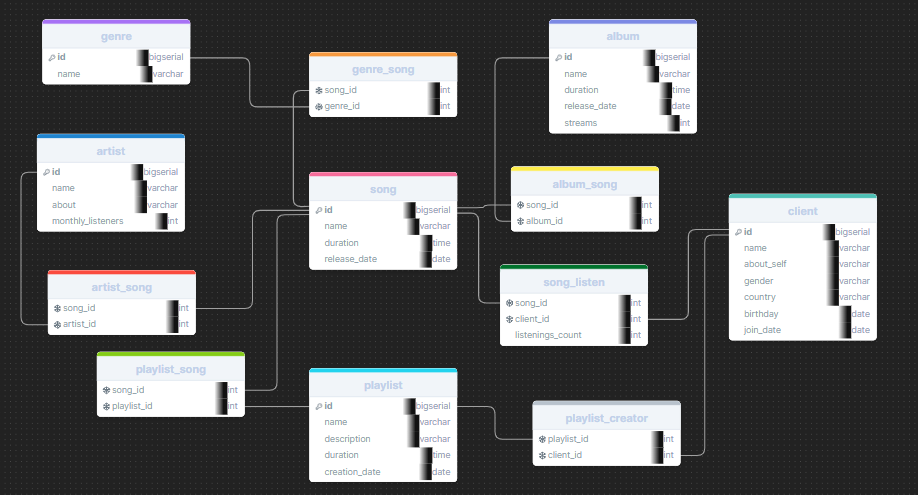
Шевнин Лев Ярославович

Санкт-Петербург

2024

**1. Проектирование базы данных**

Я решил создать базу данных, которая будет хранить в себе наименования песен, альбомов, жанров, музыкантов, а также информацию о пользователях, плейлистах, которые они могли бы создавать, и о песнях, которые они прослушали.



Пояснение к таблицам:  
1. song

Эта таблица содержит некоторую информацию о песнях (ее название, длительность, дата выпуска). Уникально идентифицирует песни по id.

2. genre

Эта таблица содержит названия жанров. Уникально идентифицирует жанры по id.

3. album

Таблица хранит некоторые сведения об альбомах, такие как длительность, название, и т.д. Уникально идентифицирует альбомы по id.

4. artist

В этой таблице содержится некоторая информация об артистах (описание, сценическое имя, количество слушателей в месяц). Уникально идентифицирует музыкантов по id.

5. client

Это таблица о пользователях приложения, которое могло бы пользоваться моей базой данных. Она будет включать в себя информацию о пользователе (пол, возраст, страна, и т.д.). Уникально идентифицирует пользователей по id.

6. playlist

Таблица будет хранить информацию о плейлисте. Например: название, описание, длительность. Уникально идентифицирует плейлисты по id

7. playlist\_songs

Эта таблица создана для того, чтобы можно было получить информацию о том, какая песня в какой плейлист добавлена.

8. song\_listen

Таблица создана для того, чтобы хранить информацию о том, сколько раз и какую песню пользователь прослушал. Она будет полезна для создания рекомендаций пользователю.

9. album\_song

Таблица хранит информацию о песнях, принадлежащих к альбому.

10. genre\_song

Таблица хранит информацию о жанрах песен.

11. artist\_song

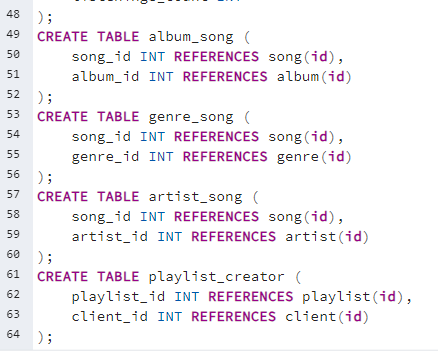
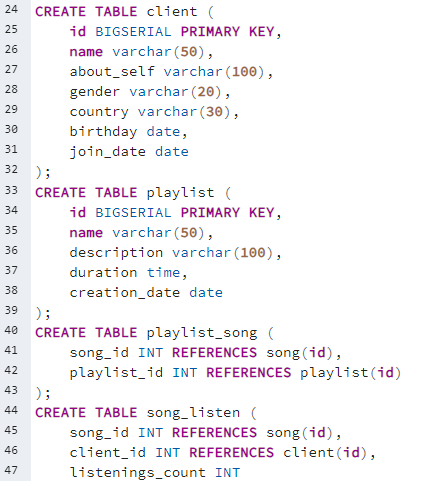
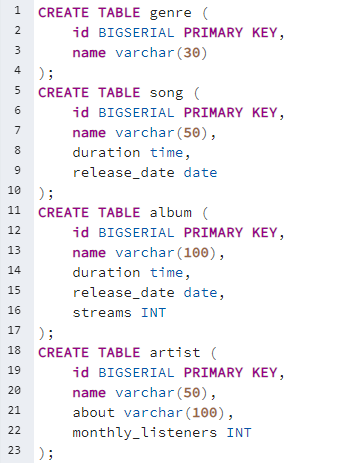
Таблица хранит информацию о песнях исполнителей.

12. playlist\_creator

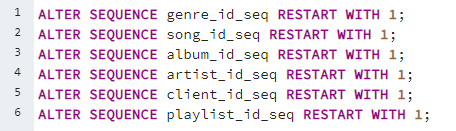
Таблица хранит информацию об авторах плейлистов.

**2. Создание и заполнение таблиц**

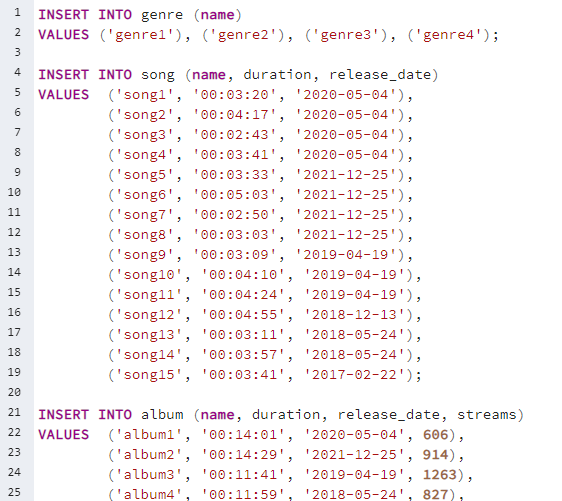
Итак, сначала нужно написать запросы для создания таблиц:

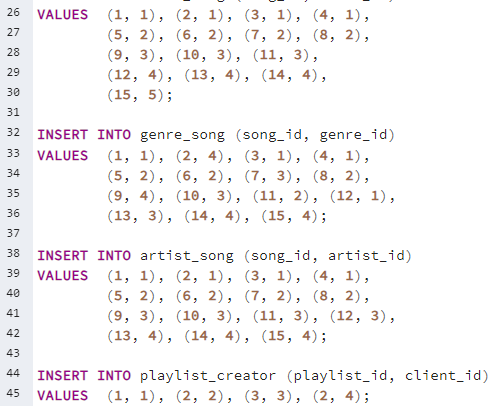
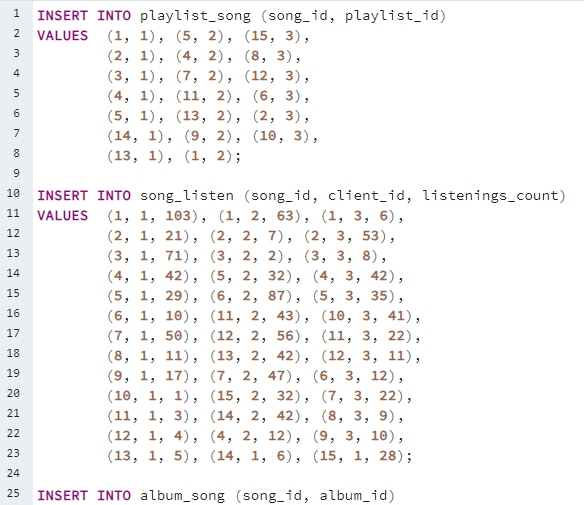


Будет удобно, если последовательности будут начинаться с единицы. Запрос для этого:

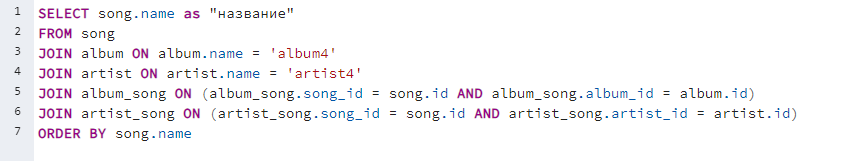


Теперь заполнение таблиц. Запросы на заполнение таблиц:

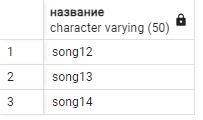




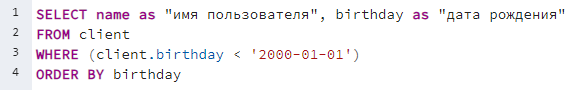
**3. Запросы**

Запрос для получения списка песен из конкретного альбома конкретного исполнителя: 

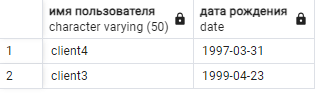
Результат:



Запрос на получение списка пользователей, родившихся в 20 веке:

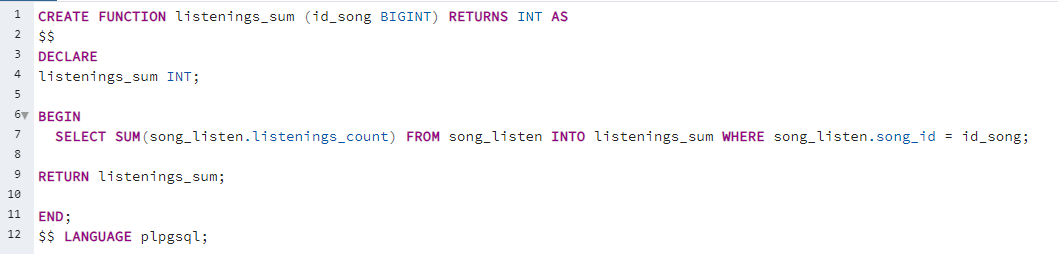


Результат:

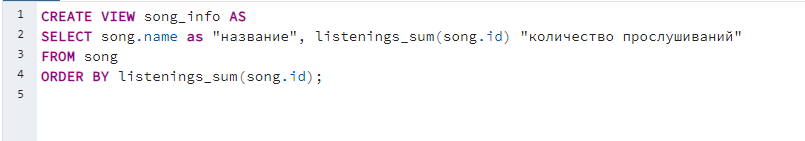


Было бы удобно получать информацию об общем количестве прослушиваний песен (то есть, количество прослушиваний от каждого пользователя).

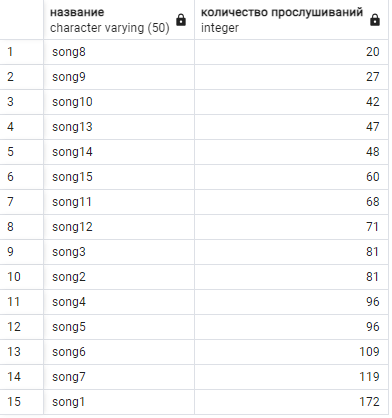
Поэтому я написал для этого функцию для этого функцию:



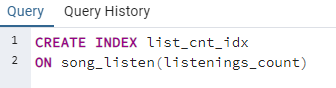
И создал представление для быстрого доступа к информации:



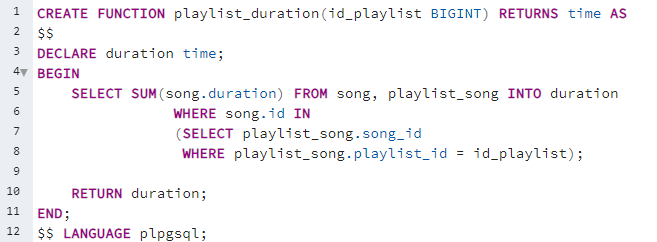
Данные из этого представления:



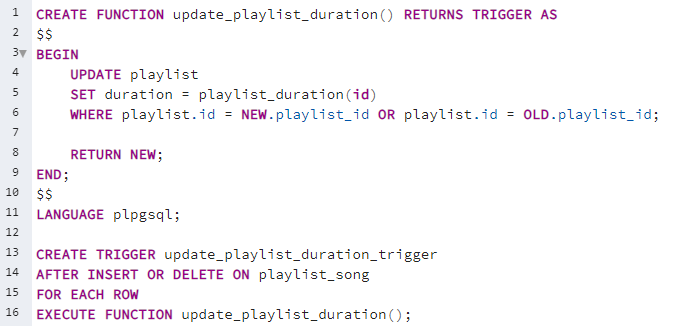
Теперь можно добавить индекс к столбцу о количестве прослушиваний композиций:



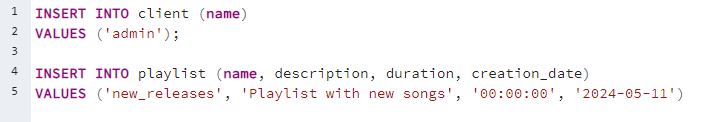
Функция, которая вычисляет длительность плейлиста:



Далее я написал функцию, которая обновляет длительность плейлиста, и триггер, который будет срабатывать при изменении плейлиста:



Создал клиента (админа) и плейлист с новыми песнями, который будет принадлежать этому клиенту.



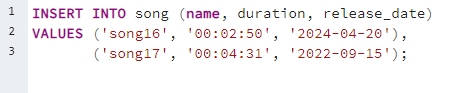
Добавление этого клиента автором плейлиста с новыми релизами:

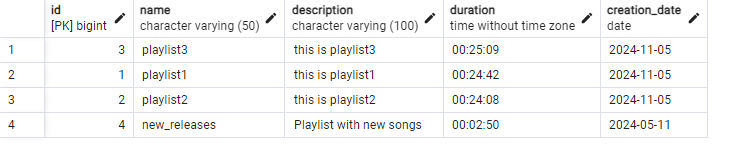


Далее, я добавил функцию которая добавляет песни, отличающиеся от сегодняшей даты не более, чем на сто дней, и убирать остальные, если такие будут. Также я создал тригнер, который будет вызывать эту функцию при добавлении новой песни (таблица song).



Тестирование:





Как можно заметить, песня “song16” была добавлена в плейлист, а “song17” нет. Все работает