**Федеральное государственное образовательное бюджетное учреждение**

**высшего образования**

**«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ РФ»**

**Департамент анализа данных и машинного обучения**

**Отчет по практике №5-6**

по дисциплине «системы управления базами данных»

Студента группы ПМ23-1

Факультета информационных технологий и анализа больших данных

Тищенко И.С.

Преподаватель

Карамышев Е.П.

Москва 2024

# Упражнение 1

Для создания базы данных в pgAdmin, которая будет использоваться для систематизации и прослушивания музыки, можно спроектировать следующие таблицы:

1. Таблица Users (Пользователи)

Описание: Содержит информацию о пользователях сайта.

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Тип данных | Описание | Ограничения |
| user\_phone | VARCHAR(11) | Номер телефона | PRIMARY KEY, NOT NULL |
| username | VARCHAR(50) | Имя пользователя | UNIQUE, NOT NULL |
| email | VARCHAR(100) | Электронная почта | UNIQUE, NOT NULL |
| password | VARCHAR(255) | Хэшированный пароль | NOT NULL |
| registration\_date | DATE | Дата регистрации | DEFAULT CURRENT\_DATE |

2. Таблица Artists (Исполнители)

Описание: Содержит информацию о музыкальных исполнителях.

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Тип данных | Описание | Ограничения |
| artist\_email | VARCHAR(50) | Электронная почта | PRIMARY KEY, NOT NULL |
| artist\_name | VARCHAR(100) | Имя исполнителя | UNIQUE, NOT NULL |
| country | VARCHAR(50) | Страна происхождения | NOT NULL |
| birth\_date | DATE | Дата рождения исполнителя |  |
| genre | VARCHAR(50) | Жанр | CHECK (genre IN ('Pop', 'Rock', 'Jazz', 'Classical', 'Electronic', 'Hip-Hop')) |

3. Таблица Albums (Альбомы)

Описание: Содержит информацию об альбомах, выпущенных исполнителями.

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Тип данных | Описание | Ограничения |
| album\_id | SERIAL | Уникальный идентификатор | PRIMARY KEY, NOT NULL |
| album\_name | VARCHAR(100) | Название альбома | NOT NULL |
| release\_date | DATE | Дата выхода альбома | NOT NULL |
| artist\_id | INT | Ссылка на исполнителя | FOREIGN KEY (artist\_id) REFERENCES Artists(artist\_id) ON DELETE CASCADE ON UPDATE CASCADE |
| total\_tracks | INT | Количество треков | CHECK (total\_tracks > 0), NOT NULL |

4. Таблица Songs(Песни)

Описание: Содержит информацию о песнях.

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Тип данных | Описание | Ограничения |
| song\_id | SERIAL | Уникальный идентификатор | PRIMARY KEY, NOT NULL |
| song\_name | VARCHAR(100) | Название песни | NOT NULL |
| album\_id | INT | Ссылка на альбом | FOREIGN KEY (album\_id) REFERENCES Albums(album\_id) ON DELETE CASCADE ON UPDATE CASCADE |
| duration | INTERVAL | Длительность песни | NOT NULL |
| plays\_count | INT | Количество прослушиваний | CHECK (plays\_count >= 0) |

5. Таблица Playlists (Плейлисты)

Описание: Содержит информацию о пользовательских плейлистах.

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Тип данных | Описание | Ограничения |
| playlist\_id | SERIAL | Уникальный идентификатор | PRIMARY KEY, NOT NULL |
| playlist\_name | VARCHAR(100) | Название плейлиста | NOT NULL |
| user\_id | INT | Ссылка на пользователя | FOREIGN KEY (user\_id) REFERENCES Users(user\_id) ON DELETE CASCADE ON UPDATE CASCADE |
| created\_date | DATE | Дата создания плейлиста | DEFAULT CURRENT\_DATE |
| is\_public | BOOLEAN | Статус публичности | DEFAULT FALSE, NOT NULL |

Таблица для связи плейлистов и песен: Playlist\_Songs

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Тип данных | Описание | Ограничения |
| playlist\_id | INT | Ссылка на плейлист | FOREIGN KEY (playlist\_id) REFERENCES Playlists(playlist\_id) ON DELETE CASCADE ON UPDATE CASCADE |
| song\_id | INT | Ссылка на песню | FOREIGN KEY (song\_id) REFERENCES Songs(song\_id) ON DELETE CASCADE ON UPDATE CASCADE |

Итоговые ограничения:

1. PRIMARY KEY для каждой таблицы.

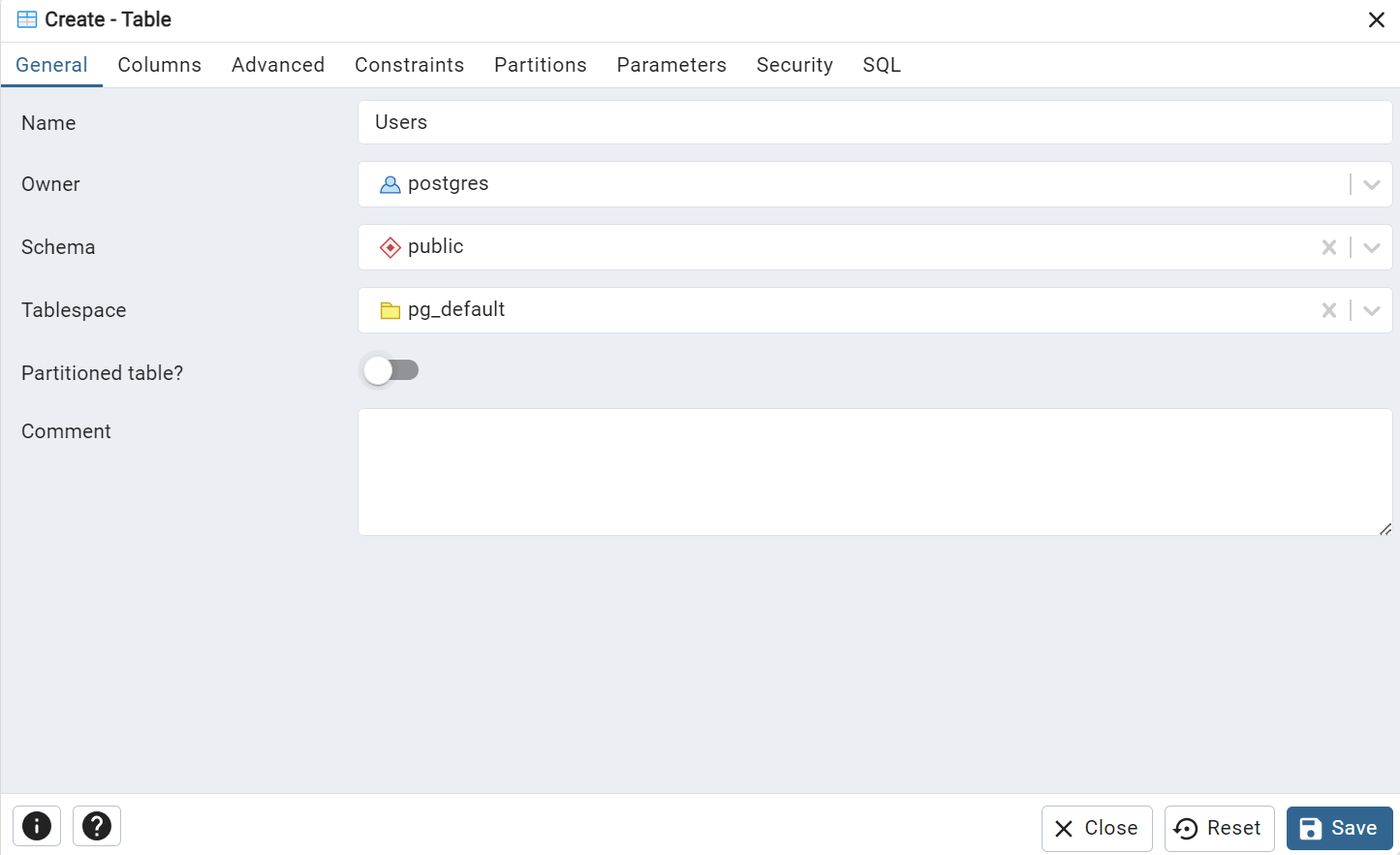
2. Внешние ключи для таблиц Albums, Songs, Playlists, Playlist\_Songs.

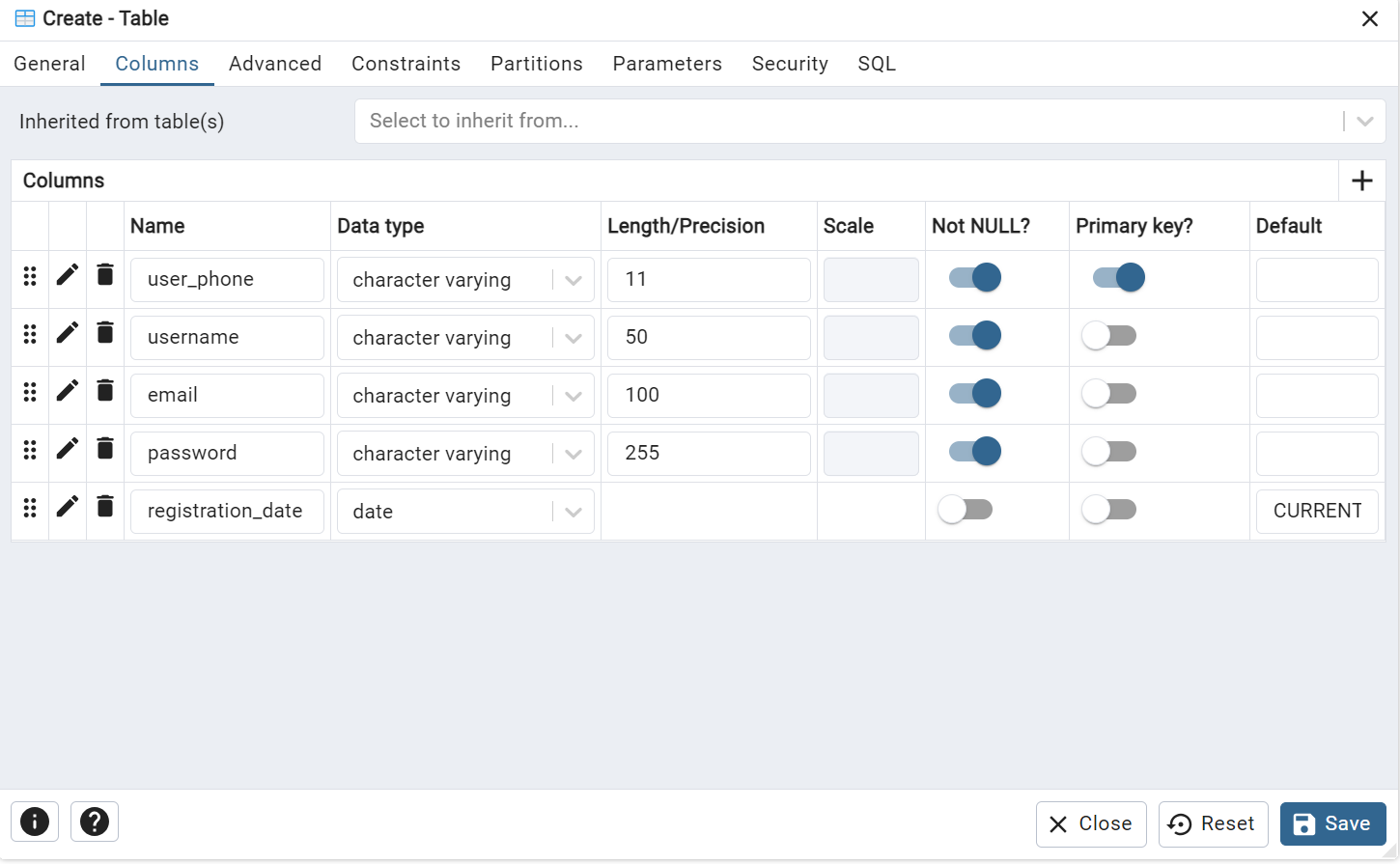
3. CHECK для жанра, длительности и количества треков.

4. Уникальные значения для username, email`, artist\_name`.

5. DEFAULT для полей `registration\_date`, `created\_date`, `is\_public`.

6. RESTRICT/CASCADE для внешних ключей.







-- Создание таблицы пользователей

CREATE TABLE Users (

user\_phone VARCHAR(11) PRIMARY KEY NOT NULL,

username VARCHAR(50) UNIQUE NOT NULL,

email VARCHAR(100) UNIQUE NOT NULL,

password VARCHAR(255) NOT NULL,

registration\_date DATE DEFAULT CURRENT\_DATE

);

-- Создание таблицы исполнителей

CREATE TABLE Artists (

artist\_email VARCHAR(50) PRIMARY KEY UNIQUE NOT NULL ,

artist\_name VARCHAR(100) UNIQUE NOT NULL,

country VARCHAR(50) NOT NULL,

birth\_date DATE,

genre VARCHAR(50) CHECK (genre IN ('Pop', 'Rock', 'Jazz', 'Classical', 'Electronic', 'Hip-Hop'))

);

-- Создание таблицы альбомов

CREATE TABLE Albums (

album\_id SERIAL PRIMARY KEY,

album\_name VARCHAR(100) NOT NULL,

release\_date DATE NOT NULL,

artist\_email VARCHAR(50) UNIQUE NOT NULL ,

total\_tracks INT CHECK (total\_tracks > 0) NOT NULL,

FOREIGN KEY (artist\_email) REFERENCES Artists(artist\_email)

ON DELETE CASCADE

ON UPDATE CASCADE

);

-- Создание таблицы песен

CREATE TABLE Songs (

song\_id SERIAL PRIMARY KEY,

song\_name VARCHAR(100) NOT NULL,

album\_id INT NOT NULL,

duration INTERVAL NOT NULL,

plays\_count INT CHECK (plays\_count >= 0),

FOREIGN KEY (album\_id) REFERENCES Albums(album\_id)

ON DELETE CASCADE

ON UPDATE CASCADE

);

-- Создание таблицы плейлистов

CREATE TABLE Playlists (

playlist\_id SERIAL PRIMARY KEY,

playlist\_name VARCHAR(100) NOT NULL,

user\_phone VARCHAR(11) NOT NULL,

created\_date DATE DEFAULT CURRENT\_DATE,

is\_public BOOLEAN DEFAULT FALSE NOT NULL,

FOREIGN KEY (user\_phone) REFERENCES Users(user\_phone)

ON DELETE CASCADE

ON UPDATE CASCADE

);

-- Создание таблицы для связи песен и плейлистов

CREATE TABLE Playlist\_Songs (

playlist\_id INT NOT NULL,

song\_id INT NOT NULL,

FOREIGN KEY (playlist\_id) REFERENCES Playlists(playlist\_id)

ON DELETE CASCADE

ON UPDATE CASCADE,

FOREIGN KEY (song\_id) REFERENCES Songs(song\_id)

ON DELETE CASCADE

ON UPDATE CASCADE,

PRIMARY KEY (playlist\_id, song\_id)

);