**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МО ЭВМ**

отчет

**по лабораторной работе №2**

**по дисциплине «Базы данных»**

Тема: Реализация базы данных в СУБД PostgreSQL

|  |  |  |
| --- | --- | --- |
| Студент гр. 1303 |  | Иевлев Е.А. |
| Преподаватель |  | Заславский М.М. |

Санкт-Петербург

2023

**Цель работы.**

Развернуть локально PostgreSQL, написать запросы для создания и заполнения таблиц, написать запросы к БД, отвечающие на вопросы в задания.

**Задание.**

Вариант 10

Пусть требуется создать программную систему, предназначенную для работников справочной службы кинотеатров города. Такая система должна обеспечивать хранение сведений о кинотеатрах города, о фильмах, которые в них демонстрируются, о сеансах и билетах на эти сеансы. Сведения о кинотеатре — это его название, район города, где расположен кинотеатр, категория, вместимость. Сведения о фильме — это название фильма, режиссер, оператор, актеры, сыгравшие главные роли, жанр; производство, наличие призов кинофестивалей, продолжительность сеанса, кадр из фильма для рекламы. Кроме того, должна храниться информация о репертуаре кинотеатров на месяц, то есть о том какие фильмы, когда и где демонстрируются, о ценах на билеты и о количестве свободных мест на тот или иной сеанс. На разных сеансах в одном кинотеатре могут идти разные фильмы, а если в кинотеатре несколько залов, то и на одном. Кинотеатр может ввести новый фильм в репертуар или убрать фильм из репертуара. Работник справочной службы может корректировать перечень фильмов, находящихся в прокате – добавлять новые фильмы и снимать с проката, а также перечень кинотеатров, поскольку кинотеатры могут открываться или закрываться, причем иногда временно, например, на ремонт. Цена билета определяется прокатной стоимостью копии фильма, сеансом и категорией кинотеатра. Справочной службе могут потребоваться следующие сведения о текущем состоянии проката фильмов в городе:

* Репертуар кинотеатра?
* Адрес и район кинотеатра?
* Число свободных мест на данный сеанс в указанном кинотеатре?
* Цена билетов на данный сеанс в указанном кинотеатре?
* Жанр, производство и режиссер данного фильма?
* Какие фильмы имеют награды, когда и в каких кинотеатрах они демонстрируются?
* В каких кинотеатрах в указанный день на указанных сеансах демонстрируется комедия?

**Выполнение работы.**

Создана база данных под названием «Cinema».

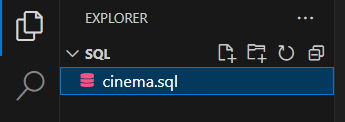


Рисунок 1. Создание БД "Cinema"

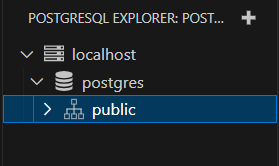


Рисунок 2. Вид созданной БД.

Созданы таблицы «Актёр», «Актёр главной роли», «Кинотеатр», «Приз кинофестиваля», «Приз фильма», «Сеанс», «Фильм» в соответствии со структурой БД и написаны запросы на их заполнение.

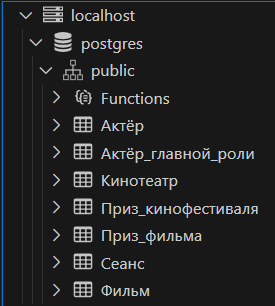


Рисунок 3. Созданные таблицы.

Реализованы запросы на вывод нужных сведений для работников справочной службы кинотеатров города:

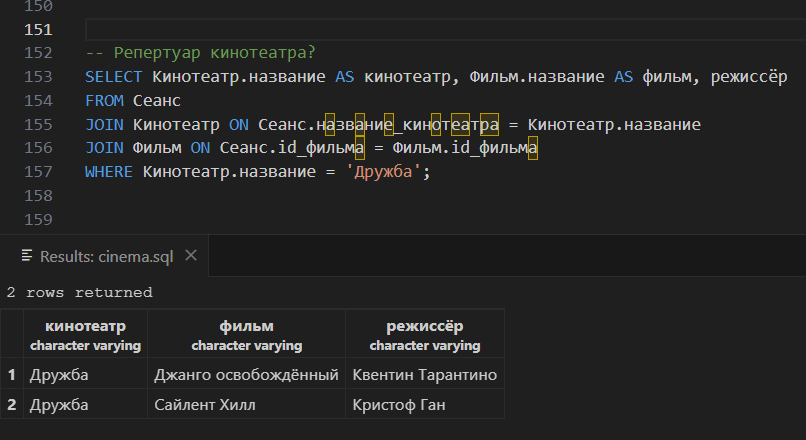


Рисунок 4. Запрос на вывод: репертуар кинотеатра.

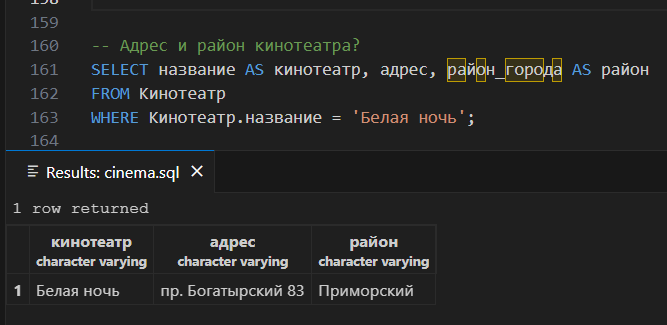


Рисунок 5. Запрос на вывод: адрес и район кинотеатра.

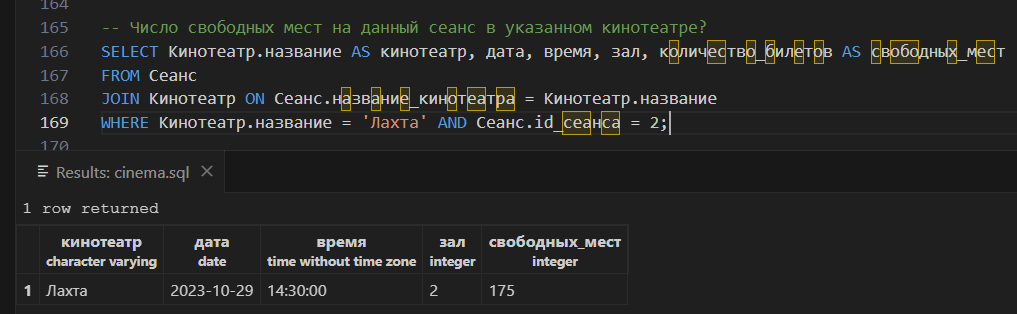


Рисунок 6. Запрос на вывод: число свободных мест на данный сеанс в указанном кинотеатре.

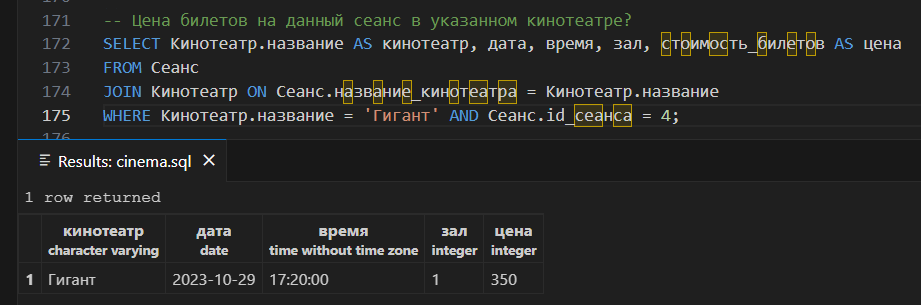


Рисунок 7. Запрос на вывод: цена билетов на данный сеанс в указанном кинотеатре.

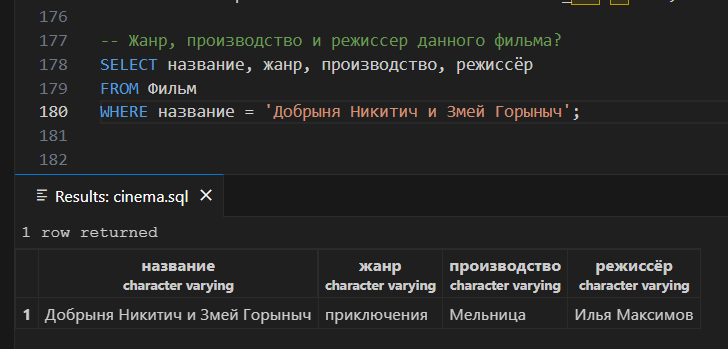


Рисунок 8. Запрос на вывод: жанр, производство и режиссёр данного фильма.

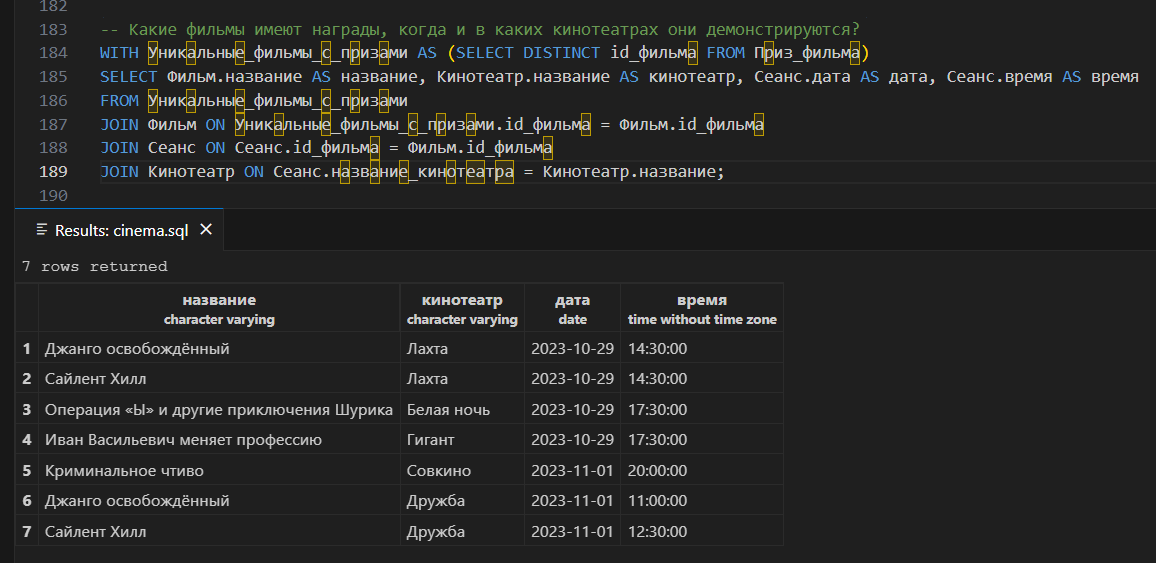


Рисунок 9. Запрос на вывод: какие фильмы имеют награды, когда и в каких кинотеатрах они демонстрируются.

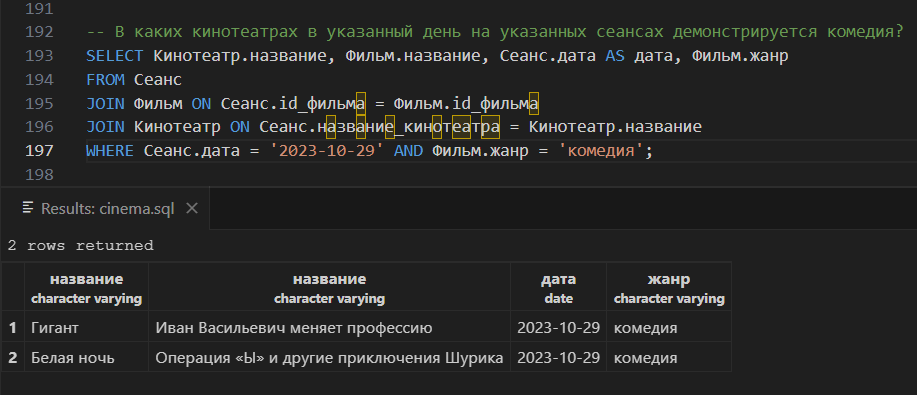


Рисунок 10. Запрос на вывод: в каких кинотеатрах в указанный день на указанных сеансах демонстрируется комедия.

**Вывод.**

В ходе выполнения работы был развернут локально PostgreSQL, написаны запросы для создания и заполнения таблиц в соответствии со структурой БД, написаны запросы к БД, отвечающие на вопросы в задания.

**Приложение А**

**Ссылки**

Pull Request: https://github.com/moevm/sql-2023-1303/pull/36

DB Fiddle: https://www.db-fiddle.com/f/cvUKV3D8kjDEMr9GJE7n1U/3

**Приложение Б**

**Исходный код**

Файл lab2.sql:

CREATE TABLE Фильм

(

id\_фильма SMALLSERIAL PRIMARY KEY,

название TEXT NOT NULL,

режиссёр TEXT NOT NULL,

оператор TEXT,

жанр TEXT,

производство TEXT NOT NULL,

продолжительность TIME NOT NULL,

постер TEXT

);

CREATE TABLE Актёр

(

id\_актёра SERIAL PRIMARY KEY,

фамилия TEXT NOT NULL,

имя TEXT NOT NULL

);

CREATE TABLE Приз\_кинофестиваля

(

название TEXT PRIMARY KEY

);

CREATE TABLE Кинотеатр

(

название TEXT PRIMARY KEY,

район\_города TEXT NOT NULL,

адрес TEXT NOT NULL,

категория TEXT NOT NULL,

вместимость INTEGER NOT NULL

);

CREATE TABLE Сеанс

(

id\_сеанса SERIAL PRIMARY KEY,

дата DATE NOT NULL,

время TIME NOT NULL,

зал INTEGER NOT NULL,

стоимость\_билетов INTEGER NOT NULL,

количество\_билетов INTEGER NOT NULL,

id\_фильма INTEGER,

название\_кинотеатра TEXT,

FOREIGN KEY (id\_фильма) REFERENCES Фильм(id\_фильма) ON DELETE CASCADE,

FOREIGN KEY (название\_кинотеатра) REFERENCES Кинотеатр(название) ON DELETE CASCADE

);

CREATE TABLE Актёр\_главной\_роли

(

id\_фильма INTEGER,

id\_актёра INTEGER,

PRIMARY KEY (id\_фильма, id\_актёра),

FOREIGN KEY (id\_фильма) REFERENCES Фильм(id\_фильма) ON DELETE CASCADE,

FOREIGN KEY (id\_актёра) REFERENCES Актёр(id\_актёра) ON DELETE SET NULL

);

CREATE TABLE Приз\_фильма

(

id\_фильма INTEGER,

название\_приза TEXT,

PRIMARY KEY (id\_фильма, название\_приза),

FOREIGN KEY (id\_фильма) REFERENCES Фильм(id\_фильма) ON DELETE SET NULL,

FOREIGN KEY (название\_приза) REFERENCES Приз\_кинофестиваля(название) ON DELETE CASCADE

);

INSERT INTO Фильм (название, режиссёр, оператор, жанр, производство, продолжительность, постер) VALUES

('Джанго освобожденный', 'Квентин Тарантино', 'Роберт Ричардсон', 'спагетти-вестерн', 'Columbia Pictures', '2:45:00', NULL),

('Добрыня Никитич и Змей Горыныч', 'Илья Максимов', 'Татьяна Присяч', 'приключения', 'Мельница', '1:08:09', NULL),

('Мегамозг', 'Том Макграт', NULL, 'фантастика', 'DreamWorks', '1:36:00', NULL),

('Сайлент Хилл', 'Кристоф Ган', 'Дан Лаустсен', 'психологический фильм ужасов', 'Konami', '2:07:00', NULL),

('Иван Васильевич меняет профессию', 'Леонид Гайдай', 'Сергей Полуянов', 'комедия', 'Мосфильм', '1:28:00', NULL),

('Операция «Ы» и другие приключения Шурика', 'Леонид Гайдай', 'Константин Бровин', 'комедия', 'Мосфильм', '1:30:00', NULL),

('Криминальное чтиво', 'Квентин Тарантино', 'Анджей Секула', 'криминальный фильм', 'A Band Apart', '1:30:00', NULL);

INSERT INTO Кинотеатр (название, район\_города, адрес, категория, вместимость) VALUES

('Гигант', 'Центральный', 'пр. Московский 8', 'эконом', 1400),

('Совкино', 'Центральный', 'наб. Дворцовая 10', 'стандарт', 1000),

('Лахта', 'Приморский', 'ул. Савушкина 30', 'премиум', 600),

('Белая ночь', 'Приморский', 'пр. Богатырский 83', 'эконом', 1500),

('Дружба', 'Петроградский', 'пр. Каменоостровский', 'премиум', 600);

INSERT INTO Актёр (фамилия, имя) VALUES

('Яковлев', 'Юрий'),

('Демьяненко', 'Александр'),

('Вальц', 'Кристоф'),

('Фокс', 'Джейми'),

('Джексон', 'Сэмюэл Лерой'),

('Траволта', 'Джон'),

('Никулин', 'Юрий'),

('Бин', 'Шон'),

('Митчелл', 'Рада');

INSERT INTO Сеанс (дата, время, зал, стоимость\_билетов, количество\_билетов, id\_фильма, название\_кинотеатра) VALUES

('2023-10-29', '14:30:00', 1, 400, 300, 3, 'Белая ночь'),

('2023-10-29', '14:30:00', 2, 650, 175, 1, 'Лахта'),

('2023-10-29', '14:30:00', 1, 500, 400, 4, 'Лахта'),

('2023-10-29', '17:20:00', 1, 350, 350, 2, 'Гигант'),

('2023-10-29', '17:30:00', 1, 400, 600, 6, 'Белая ночь'),

('2023-10-29', '17:30:00', 2, 400, 300, 5, 'Гигант'),

('2023-11-01', '20:00:00', 2, 450, 300, 7, 'Совкино'),

('2023-11-01', '11:00:00', 2, 700, 150, 1, 'Дружба'),

('2023-11-01', '11:00:00', 1, 300, 400, 2, 'Совкино'),

('2023-11-01', '12:30:00', 1, 450, 270, 4, 'Дружба');

INSERT INTO Приз\_кинофестиваля (название) VALUES

('красная слива'),

('золотая малина'),

('оскар'),

('шоколадная медаль'),

('ведро');

INSERT INTO Приз\_фильма VALUES

(1, 'оскар'),

(1, 'ведро'),

(4, 'красная слива'),

(5, 'шоколадная медаль'),

(6, 'шоколадная медаль'),

(7, 'оскар'),

(7, 'красная слива');

INSERT INTO Актёр\_главной\_роли VALUES

(5, 2),

(5, 1),

(6, 2),

(6, 7),

(4, 8),

(4, 9),

(1, 3),

(1, 4),

(1, 5),

(7, 5),

(7, 6);

-- Репертуар кинотеатра?

SELECT Кинотеатр.название AS кинотеатр, Фильм.название AS фильм, режиссёр

FROM Сеанс

JOIN Кинотеатр ON Сеанс.название\_кинотеатра = Кинотеатр.название

JOIN Фильм ON Сеанс.id\_фильма = Фильм.id\_фильма

WHERE Кинотеатр.название = 'Дружба';

-- Адрес и район кинотеатра?

SELECT название AS кинотеатр, адрес, район\_города AS район

FROM Кинотеатр

WHERE Кинотеатр.название = 'Белая ночь';

-- Число свободных мест на данный сеанс в указанном кинотеатре?

SELECT Кинотеатр.название AS кинотеатр, дата, время, зал, количество\_билетов AS свободных\_мест

FROM Сеанс

JOIN Кинотеатр ON Сеанс.название\_кинотеатра = Кинотеатр.название

WHERE Кинотеатр.название = 'Лахта' AND Сеанс.id\_сеанса = 2;

-- Цена билетов на данный сеанс в указанном кинотеатре?

SELECT Кинотеатр.название AS кинотеатр, дата, время, зал, стоимость\_билетов AS цена

FROM Сеанс

JOIN Кинотеатр ON Сеанс.название\_кинотеатра = Кинотеатр.название

WHERE Кинотеатр.название = 'Гигант' AND Сеанс.id\_сеанса = 4;

-- Жанр, производство и режиссер данного фильма?

SELECT название, жанр, производство, режиссёр

FROM Фильм

WHERE название = 'Добрыня Никитич и Змей Горыныч';

-- Какие фильмы имеют награды, когда и в каких кинотеатрах они демонстрируются?

WITH Уникальные\_фильмы\_с\_призами AS (SELECT DISTINCT id\_фильма FROM Приз\_фильма)

SELECT Фильм.название AS название, Кинотеатр.название AS кинотеатр, Сеанс.дата AS дата, Сеанс.время AS время

FROM Уникальные\_фильмы\_с\_призами

JOIN Фильм ON Уникальные\_фильмы\_с\_призами.id\_фильма = Фильм.id\_фильма

JOIN Сеанс ON Сеанс.id\_фильма = Фильм.id\_фильма

JOIN Кинотеатр ON Сеанс.название\_кинотеатра = Кинотеатр.название;

-- В каких кинотеатрах в указанный день на указанных сеансах демонстрируется комедия?

SELECT Кинотеатр.название, Фильм.название, Сеанс.дата AS дата, Фильм.жанр

FROM Сеанс

JOIN Фильм ON Сеанс.id\_фильма = Фильм.id\_фильма

JOIN Кинотеатр ON Сеанс.название\_кинотеатра = Кинотеатр.название

WHERE Сеанс.дата = '2023-10-29' AND Фильм.жанр = 'комедия';