

**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 заданиях.

### **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».

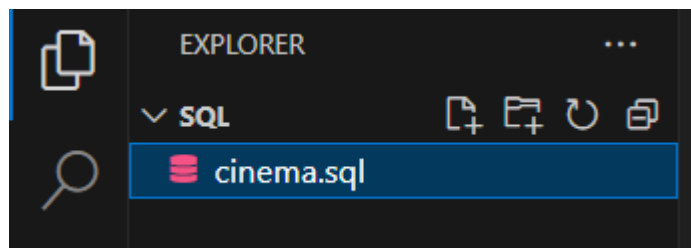


Рисунок 1. Создание БД "Cinema"

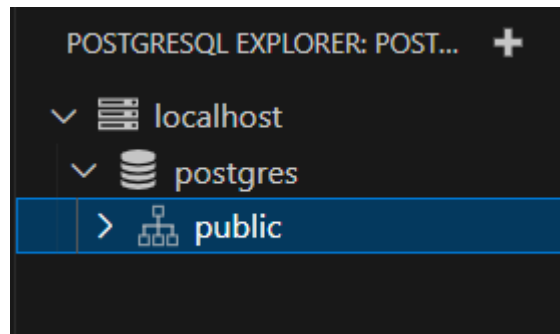


Рисунок 2. Вид созданной БД.

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

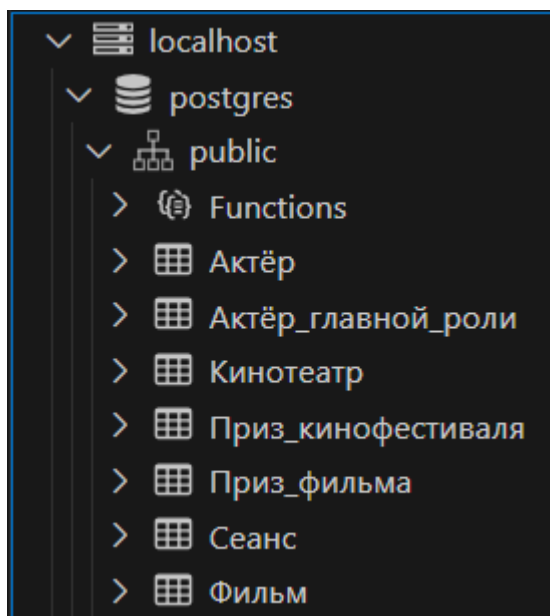


Рисунок 3. Созданные таблицы.

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

```

150
151
152 -- Репертуар кинотеатра?
153 SELECT Кинотеатр.название AS кинотеатр, Фильм.название AS фильм, режиссёр
154 FROM Сеанс
155 JOIN Кинотеатр ON Сеанс.название_кинотеатра = Кинотеатр.название
156 JOIN Фильм ON Сеанс.id_фильма = Фильм.id_фильма
157 WHERE Кинотеатр.название = 'Дружба';
158
159

```

Results: cinema.sql X

2 rows returned

	кинотеатр character varying	фильм character varying	режиссёр character varying
1	Дружба	Джанго освобождённый	Квентин Тарантино
2	Дружба	Сайлент Хилл	Кристоф Ган

Рисунок 4. Запрос на вывод: репертуар кинотеатра.

```
158
159
160 -- Адрес и район кинотеатра?
161 SELECT название AS кинотеатр, адрес, район_города AS район
162 FROM Кинотеатр
163 WHERE Кинотеатр.название = 'Белая ночь';
164
```

Results: cinema.sql X

1 row returned

	кинотеатр character varying	адрес character varying	район character varying
1	Белая ночь	пр. Богатырский 83	Приморский

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

```
164
165 -- Число свободных мест на данный сеанс в указанном кинотеатре?
166 SELECT Кинотеатр.название AS кинотеатр, дата, время, зал, количество_билетов AS свободных_мест
167 FROM Сеанс
168 JOIN Кинотеатр ON Сеанс.название_кинотеатра = Кинотеатр.название
169 WHERE Кинотеатр.название = 'Лахта' AND Сеанс.id_сеанса = 2;
170
```

Results: cinema.sql X

1 row returned

	кинотеатр character varying	дата date	время time without time zone	зал integer	свободных_мест integer
1	Лахта	2023-10-29	14:30:00	2	175

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

```

171 -- Цена билетов на данный сеанс в указанном кинотеатре?
172 SELECT Кинотеатр.название AS кинотеатр, дата, время, зал, стоимость_билетов AS цена
173 FROM Сеанс
174 JOIN Кинотеатр ON Сеанс.название_кинотеатра = Кинотеатр.название
175 WHERE Кинотеатр.название = 'Гигант' AND Сеанс.id_сеанса = 4;
176

```

Results: cinema.sql X

1 row returned

	кинотеатр character varying	дата date	время time without time zone	зал integer	цена integer
1	Гигант	2023-10-29	17:20:00	1	350

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

```

176
177 -- Жанр, производство и режиссер данного фильма?
178 SELECT название, жанр, производство, режиссёр
179 FROM Фильм
180 WHERE название = 'Добрыня Никитич и Змей Горыныч';
181
182

```

Results: cinema.sql X

1 row returned

	название character varying	жанр character varying	производство character varying	режиссёр character varying
1	Добрыня Никитич и Змей Горыныч	приключения	Мельница	Илья Максимов

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

```

182
183 -- Какие фильмы имеют награды, когда и в каких кинотеатрах они демонстрируются?
184 WITH Уникальные_фильмы_с_призами AS (SELECT DISTINCT id_фильма FROM Приз_фильма)
185 SELECT Фильм.название AS название, Кинотеатр.название AS кинотеатр, Сеанс.дата AS дата, Сеанс.время AS время
186 FROM Уникальные_фильмы_с_призами
187 JOIN Фильм ON Уникальные_фильмы_с_призами.id_фильма = Фильм.id_фильма
188 JOIN Сеанс ON Сеанс.id_фильма = Фильм.id_фильма
189 JOIN Кинотеатр ON Сеанс.название_кинотеатра = Кинотеатр.название;
190

```

Results: cinema.sql X

7 rows returned

	название character varying	кинотеатр character varying	дата date	время time without time zone
1	Джанго освобождённый	Лакта	2023-10-29	14:30:00
2	Сайлент Хилл	Лакта	2023-10-29	14:30:00
3	Операция «Ы» и другие приключения Шурика	Белая ночь	2023-10-29	17:30:00
4	Иван Васильевич меняет профессию	Гигант	2023-10-29	17:30:00
5	Криминальное чтиво	Совкино	2023-11-01	20:00:00
6	Джанго освобождённый	Дружба	2023-11-01	11:00:00
7	Сайлент Хилл	Дружба	2023-11-01	12:30:00

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

```

191
192 -- В каких кинотеатрах в указанный день на указанных сеансах демонстрируется комедия?
193 SELECT Кинотеатр.название, Фильм.название, Сеанс.дата AS дата, Фильм.жанр
194 FROM Сеанс
195 JOIN Фильм ON Сеанс.id_фильма = Фильм.id_фильма
196 JOIN Кинотеатр ON Сеанс.название_кинотеатра = Кинотеатр.название
197 WHERE Сеанс.дата = '2023-10-29' AND Фильм.жанр = 'комедия';
198

```

Results: cinema.sql X

2 rows returned

	название character varying	название character varying	дата date	жанр character varying
1	Гигант	Иван Васильевич меняет профессию	2023-10-29	комедия
2	Белая ночь	Операция «Ы» и другие приключения Шурика	2023-10-29	комедия

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 заданиях.

## **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';
```