

**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**

Студентка гр. 1303

\_\_\_\_\_

Королева П.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 12

Пусть требуется создать программную систему, предназначенную для организаторов соревнований по футболу в рамках первенства страны. Такая система должна обеспечивать хранение сведений о командах, участвующих в первенстве, об игроках команд, о расписании встреч и их результатах, о цене билетов на игры. Сведения о команде — название команды, город, где она базируется, имя главного тренера, место в таблице прошлого сезона, расписание встреч. В один день команда может участвовать только в одной встрече. Сведения об игроке включают в себя фамилию и имя игрока, его возраст, номер и амплуа в команде. Сведения о стадионе, на котором происходит встреча содержат город, в котором он находится, название стадиона, и его вместимость. Цена билета на матч зависит от вместимости стадиона и положения встречающихся команд в турнирной таблице прошлого сезона (наибольшая - при игре тройки призеров, наименьшая — при игре тройки аутсайдеров). Организаторы соревнований должны иметь возможность внести изменения в данные о составе команд, перенести встречу. Им могут потребоваться следующие сведения:

- Даты встреч указанной команды, ее противники и счет?
- Номера и фамилии игроков команд, участвовавших во встрече, которая проходила в указанный день в указанном городе?
- Цена, билета на матч между указанными командами?
- Игрок, забивший в турнире наибольшее количество мячей?
- Команды, имеющие наилучшую и наихудшую разницу забитых и пропущенных мячей?
- Самый молодой участник турнира?

- Команды, занявшие призовые места?

### Выполнение работы.

Создана база данных «football\_manager», ее наполнение представлено на рисунке 1. Попутно созданы 6 схем «Игрок», «Команда», «Команды\_в\_игре», «Организатор», «Расписание\_игр», «Стадион».

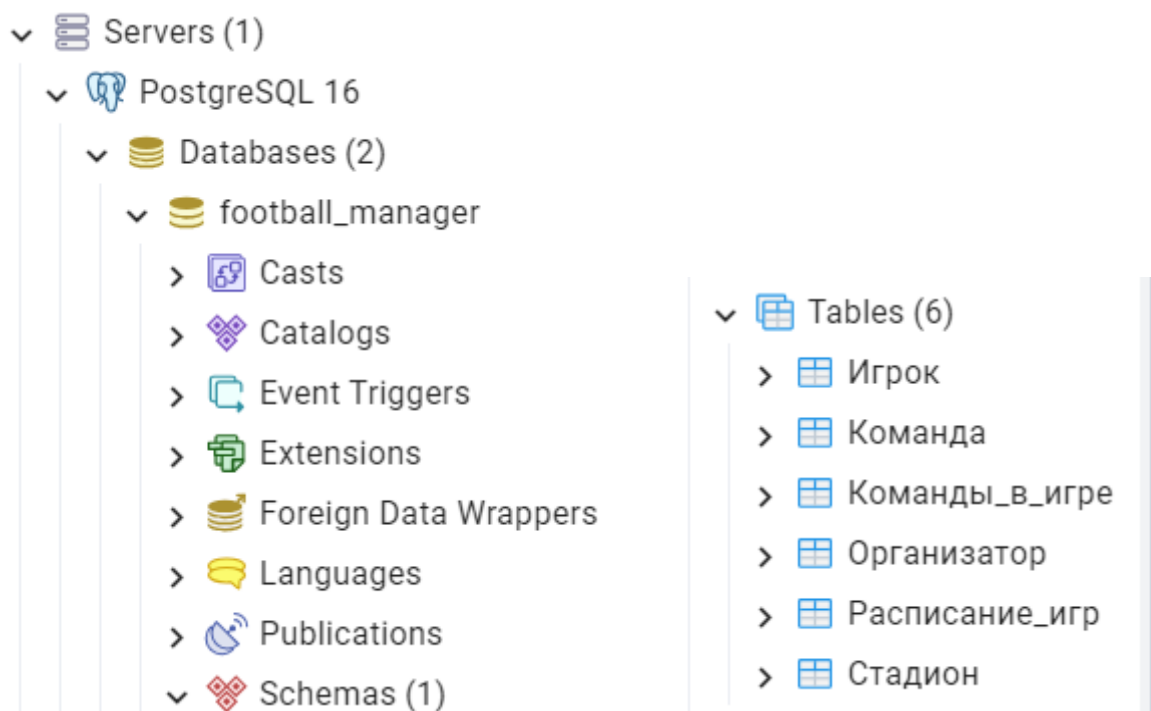


Рисунок 1. БД «football\_manager»

Схемы заполнены значениями, в каждой 5-10 кортежей. Результат представлен на рисунках 2-7.

	id_игрока [PK] integer	Название_команды text	Фамилия text	Имя text	Возраст smallint	Амплуа text	Номер smallint	Забитые_голы smallint
1	1	Балтика	Лазарев	Владислав	21	Полузащитник	89	1
2	2	Балтика	Шишкин	Федор	20	Нападающий	52	7
3	3	Зенит	Мостовой	Андрей	25	Полузащитник	17	0
4	4	Зенит	Самохин	Кирилл	22	Нападающий	8	12
5	5	Динамо	Смолов	Фёдор	33	Нападающий	10	5
6	6	Динамо	Дудко	Максим	33	Вратарь	2	28
7	7	Локомотив	Дзюба	Артём	35	Нападающий	7	3
8	8	Локомотив	Баринов	Дмитрий	27	Полузащитник	6	0
9	9	Рубин	Рыбус	Мацей	30	Защитник	31	0
10	10	Рубин	Кашуба	Данил	19	Нападающий	37	5

Рисунок 2 – Наполнение схемы «Игрок»

	Название_команды [PK] text	Город_команды text	Тренер text	Место_в_прошлом_сезоне smallint
1	Балтика	Калининград	Королев А.Л.	1
2	Динамо	Москва	Смирнов Д.Ю.	3
3	Зенит	Санкт-Петербург	Гирман А.В.	2
4	Локомотив	Москва	Королева П.А.	4
5	Рубин	Казань	Иевлев Е.А.	5

Рисунок 3 – Наполнение схемы «Команда»

	Название_команды1 text	Название_команды2 text	id_игры [PK] integer
1	Балтика	Зенит	1
2	Зенит	Локомотив	2
3	Локомотив	Рубин	3
4	Рубин	Динамо	4
5	Динамо	Балтика	5

Рисунок 4 – Наполнение схемы «Команды\_в\_игре»

	Название_организации [PK] text
1	Fonbet
2	OLIMPBET
3	Лига Чемпионов
4	Мир Российская Премьер Лига
5	ЧЕ-2024

Рисунок 5 – Наполнение схемы «Организатор»

	id_игры [PK] integer	Дата date	Счет1 smallint	Счет2 smallint	Цена_на_билет integer	Название_стадиона text	Название_организации text
1	1	2023-12-13	3	1	7000	Лужники	Fonbet
2	2	2023-12-17	1	0	7000	Лужники	Fonbet
3	3	2024-02-05	2	7	3000	Газпром Арена	ЧЕ-2024
4	4	2024-03-01	0	1	3000	Газпром Арена	ЧЕ-2024
5	5	2024-04-10	6	9	1000	Динамо	OLIMPBET

Рисунок 6 – Наполнение схемы «Расписание\_игр»

	Название_стадиона [PK] text	Город_стадиона text	Вместимость integer
1	Барс	Казань	45379
2	Газпром Арена	Санкт-Петербург	64448
3	Динамо	Воронеж	10880
4	Лужники	Москва	76880
5	Сибсельмаш	Новосибирск	13000

Рисунок 7 – Наполнение схемы «Стадион»

Написаны запросы для ответа на вопросы в задании. Результаты приведены на рисунках 8-14.

```

1
2  --Даты встреч команды, ее противники и счет
3  --Команда Локомотив
4  SELECT Дата, Команды_в_игре.Название_команды1 as Команда1, Команды_в_игре.Название_команды2 as Команда2, Счет1, Счет2
5  FROM Расписание_игр
6  JOIN Команды_в_игре ON Команды_в_игре.id_игры = Расписание_игр.id_игры
7  WHERE Команды_в_игре.Название_команды1 = 'Локомотив' OR Команды_в_игре.Название_команды2 = 'Локомотив';
8
9

```

	Дата date	Команда1 text	Команда2 text	Счет1 smallint	Счет2 smallint
1	2023-12-17	Зенит	Локомотив	1	0
2	2024-02-05	Локомотив	Рубин	2	7

Рисунок 8. Даты встреч команды, ее противники и счет.

```

1  --Номера и фамилии игроков команд, участвовавших во встрече, которая проходила в указанный день в указанном городе
2  --День 2024-02-05, город Санкт-Петербург
3  SELECT Номер, Фамилия
4  FROM Игрок
5  JOIN Команды_в_игре ON Команды_в_игре.Название_команды1 = Игрок.Название_команды
6  OR
7  Команды_в_игре.Название_команды2 = Игрок.Название_команды
8  JOIN Расписание_игр ON Команды_в_игре.id_игры = Расписание_игр.id_игры
9  JOIN Стадион ON Расписание_игр.Название_стадиона = Стадион.Название_стадиона
10 WHERE Стадион.Город_стадиона = 'Санкт-Петербург' AND Расписание_игр.Дата = '2024-02-05';

```

	Номер smallint	Фамилия text
1	7	Дзюба
2	6	Баринов
3	31	Рыбус
4	37	Кашуба

Рисунок 9. Номера и фамилии игроков команд, участвовавших во встрече, которая проходила в указанный день в указанном городе.

```

1  --Цена билета на матч между указанными командами
2  --Команды 'Балтика' и 'Динамо'
3  SELECT Цена_на_билет
4  FROM Расписание_игр
5  JOIN Команды_в_игре ON Команды_в_игре.id_игры = Расписание_игр.id_игры
6  WHERE Команды_в_игре.Название_команды1 = 'Балтика' AND Команды_в_игре.Название_команды2 = 'Динамо'
7  OR Команды_в_игре.Название_команды1 = 'Динамо' AND Команды_в_игре.Название_команды2 = 'Балтика';
8

```

	Цена_на_билет integer
1	1000

Рисунок 10. Цена, билета на матч между указанными командами.

1

--Игрок, забивший в турнире наибольшее количество мячей

2

SELECT Имя, Фамилия, Забитые\_голы

3

FROM Игрок

4

WHERE Забитые\_голы = (SELECT MAX(Забитые\_голы) FROM Игрок);

5

6

/\*

Data Output

Messages

Notifications

≡+

📄

▼

📋

▼

🗑️

🗑️

📦

⬇️

📈

	Имя text	Фамилия text	Забитые_голы smallint
1	Максим	Дудко	28

Рисунок 11. Игрок, забивший в турнире наибольшее количество мячей.

1	--Команды с наилучшей и наихудшей разницей забитых и пропущенных мячей
2	
3	CREATE TEMPORARY TABLE Разница AS
4	SELECT Название_команды1 AS Команда, (Расписание_игр.Счет1 - Расписание_игр.Счет2) AS Разница_забитых_и_пропущенных
5	FROM Команды_в_игре
6	JOIN Расписание_игр ON Команды_в_игре.id_игры = Расписание_игр.id_игры
7	UNION
8	SELECT Название_команды2 AS Команда, (Расписание_игр.Счет2 - Расписание_игр.Счет1) AS Разница_забитых_и_пропущенных
9	FROM Команды_в_игре
10	JOIN Расписание_игр ON Команды_в_игре.id_игры = Расписание_игр.id_игры;
11	
12	
13	SELECT Команда, Разница_забитых_и_пропущенных
14	FROM Разница
15	WHERE Разница_забитых_и_пропущенных =
16	(SELECT MIN(Разница_забитых_и_пропущенных) FROM Разница)
17	OR
18	Разница_забитых_и_пропущенных =
19	(SELECT MAX(Разница_забитых_и_пропущенных) FROM Разница);
20	

Data Output	Messages	Notifications
<div> <div>≡+</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑️</div> <div>🗑️</div> <div>📦</div> <div>⬇️</div> <div>📈</div> </div>		
	Команда text	Разница_забитых_и_пропущенных smallint
1	Рубин	5
2	Локомотив	-5

Рисунок 12. Команды, имеющие наилучшую и наихудшую разницу забитых и пропущенных мячей.

```

1  --Самый молодой участник турнира
2  SELECT Имя, Фамилия, Возраст
3  FROM Игрок
4  WHERE Возраст = (SELECT MIN(Возраст) FROM Игрок);
5
6
7  /*

```

	Имя text	Фамилия text	Возраст smallint
1	Данил	Кашуба	19

Рисунок 13. Самый молодой участник турнира.

```

1
2  --Команды, занявшие призовые места
3  SELECT Название_команды, Место_в_прошлом_сезоне AS Место
4  FROM Команда
5  WHERE Место_в_прошлом_сезоне < 4;
6
7  /*

```

	Название_команды [PK] text	Место smallint
1	Балтика	1
2	Зенит	2
3	Динамо	3

Рисунок 14. Команды, занявшие призовые места.

## Вывод.

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**

DB Fiddle: <https://www.db-fiddle.com/f/8P9RYRw1Db3HcXQywc6TsU/0>

## Приложение Б

### Исходный код

Файл lab2.sql:

```
CREATE TABLE Команда (
    Название_команды TEXT PRIMARY KEY,
    Город_команды TEXT,
    Тренер TEXT,
    Место_в_прошлом_сезоне SMALLSERIAL
);

CREATE TABLE Игрок (
    id_игрока SERIAL PRIMARY KEY,
    Название_команды TEXT,
    Фамилия TEXT,
    Имя TEXT,
    Возраст SMALLSERIAL,
    Амплуа TEXT,
    Номер SMALLSERIAL,
    Забитые_голы SMALLSERIAL,
    FOREIGN KEY (Название_команды) REFERENCES
Команда(Название_команды)
);

CREATE TABLE Стадион (
    Название_стадиона TEXT PRIMARY KEY,
    Город_стадиона TEXT,
    Вместимость SERIAL
);

CREATE TABLE Организатор (
    Название_организации TEXT PRIMARY KEY
);

CREATE TABLE Расписание_игр (
    id_игры SERIAL PRIMARY KEY,
    Дата date,
    Счет1 SMALLSERIAL,
    Счет2 SMALLSERIAL,
    Цена_на_билет SERIAL,
    Название_стадиона TEXT,
    Название_организации TEXT,
    FOREIGN KEY (Название_стадиона) REFERENCES
Стадион(Название_стадиона),
    FOREIGN KEY (Название_организации) REFERENCES
Организатор(Название_организации)
);

CREATE TABLE Команды_в_игре (
```

```

        Название_команды1 TEXT,
        Название_команды2 TEXT,
        id_игры SERIAL PRIMARY KEY,
        FOREIGN KEY (Название_команды1) REFERENCES
Команда(Название_команды),
        FOREIGN KEY (Название_команды2) REFERENCES
Команда(Название_команды),
        FOREIGN KEY (id_игры) REFERENCES Расписание_игр(id_игры)
);

```

```

INSERT INTO Команда(Название_команды, Город_команды, Тренер,
Место_в_прошлом_сезоне ) VALUES
('Балтика', 'Калининград', 'Королев А.Л.', 1),
('Зенит', 'Санкт-Петербург', 'Гирман А.В.', 2),
('Динамо', 'Москва', 'Смирнов Д.Ю.', 3),
('Локомотив', 'Москва', 'Королева П.А.', 4),
('Рубин', 'Казань', 'Иевлев Е.А.', 5);

```

```

INSERT INTO Игрок(Название_команды, Фамилия, Имя, Возраст, Амплуа,
Номер, Забитые_голы) VALUES
('Балтика', 'Лазарев', 'Владислав', 21, 'Полузащитник', 89,
1),
('Балтика', 'Шишкин', 'Федор', 20, 'Нападающий', 52, 7),
('Зенит', 'Мостовой', 'Андрей', 25, 'Полузащитник', 17, 0),
('Зенит', 'Самохин', 'Кирилл', 22, 'Нападающий', 8, 12),
('Динамо', 'Смолов', 'Фёдор', 33, 'Нападающий', 10, 5),
('Динамо', 'Дудко', 'Максим', 33, 'Вратарь', 2, 28),
('Локомотив', 'Дзюба', 'Артём', 35, 'Нападающий', 7, 3),
('Локомотив', 'Баринов', 'Дмитрий', 27, 'Полузащитник', 6,
0),
('Рубин', 'Рыбус', 'Мацей', 30, 'Защитник', 31, 0),
('Рубин', 'Кашуба', 'Данил', 19, 'Нападающий', 37, 5);

```

```

INSERT INTO Организатор(Название_организации) VALUES
('Fonbet'),
('Мир Российская Премьер Лига'),
('ЧЕ-2024'),
('Лига Чемпионов'),
('OLIMPBET');

```

```

INSERT INTO Стадион(Название_стадиона, Город_стадиона,
Вместимость) VALUES
('Лужники', 'Москва', 76880),
('Газпром Арена', 'Санкт-Петербург', 64448),
('Динамо', 'Воронеж', 10880),
('Сибсельмаш', 'Новосибирск', 13000),
('Барс', 'Казань', 45379);

```

```

INSERT INTO Расписание_игр(id_игры, Дата, Счет1, Счет2,
Цена_на_билет, Название_стадиона, Название_организации) VALUES

```

```

(1, '2023-12-13', 3, 1, 7000, 'Лужники', 'Fonbet'),
(2, '2023-12-17', 1, 0, 7000, 'Лужники', 'Fonbet'),
(3, '2024-02-05', 2, 7, 3000, 'Газпром Арена', 'ЧЕ-2024'),
(4, '2024-03-01', 0, 1, 3000, 'Газпром Арена', 'ЧЕ-2024'),
(5, '2024-04-10', 6, 9, 1000, 'Динамо', 'ОЛИМПБЕТ');

INSERT INTO Команды_в_игре(id_игры, Название_команды1,
Название_команды2) VALUES
(1, 'Балтика', 'Зенит'),
(2, 'Зенит', 'Локомотив'),
(3, 'Локомотив', 'Рубин'),
(4, 'Рубин', 'Динамо'),
(5, 'Динамо', 'Балтика');

*/
--Даты встреч команды, ее противники и счет
--Команда Локомотив
SELECT Дата, Команды_в_игре.Название_команды1 as Команда1,
Команды_в_игре.Название_команды2 as Команда2, Счет1, Счет2
FROM Расписание_игр
JOIN Команды_в_игре ON Команды_в_игре.id_игры =
Расписание_игр.id_игры
WHERE Команды_в_игре.Название_команды1 = 'Локомотив' OR
Команды_в_игре.Название_команды2 = 'Локомотив';

--Номера и Фамилии игроков команд, участвовавших во встрече,
которая проходила в указанный день в указанном городе
--День 2024-02-05, город Санкт-Петербург
SELECT Номер, Фамилия
FROM Игрок
JOIN Команды_в_игре ON Команды_в_игре.Название_команды1 =
Игрок.Название_команды OR Команды_в_игре.Название_команды2 =
Игрок.Название_команды
JOIN Расписание_игр ON Команды_в_игре.id_игры =
Расписание_игр.id_игры
JOIN Стадион ON Расписание_игр.Название_стадиона =
Стадион.Название_стадиона
WHERE Стадион.Город_стадиона = 'Санкт-Петербург' AND
Расписание_игр.Дата = '2024-02-05';

--Цена билета на матч между указанными командами
--Команды 'Балтика' и 'Динамо'
SELECT Цена_на_билет
FROM Расписание_игр
JOIN Команды_в_игре ON Команды_в_игре.id_игры =
Расписание_игр.id_игры
WHERE Команды_в_игре.Название_команды1 = 'Балтика' AND
Команды_в_игре.Название_команды2 = 'Динамо'
OR Команды_в_игре.Название_команды1 = 'Динамо' AND
Команды_в_игре.Название_команды2 = 'Балтика';

--Игрок, забивший в турнире наибольшее количество мячей
SELECT Имя, Фамилия, Забитые_голы

```

```

FROM Игрок
WHERE Забитые_голы = (SELECT MAX(Забитые_голы) FROM Игрок);

--Самый молодой участник турнира
SELECT Имя, Фамилия, Возраст
FROM Игрок
WHERE Возраст = (SELECT MIN(Возраст) FROM Игрок);

--Команды, занявшие призовые места
SELECT Название_команды, Место_в_прошлом_сезоне AS Место
FROM Команда
WHERE Место_в_прошлом_сезоне < 4;

--Команды с наилучшей и наихудшей разницей забитых и пропущенных мячей

CREATE TEMPORARY TABLE Разница AS
SELECT Название_команды1 AS Команда, (Расписание_игр.Счет1 -
Расписание_игр.Счет2) AS Разница_забитых_и_пропущенных
FROM Команды_в_игре
JOIN Расписание_игр ON Команды_в_игре.id_игры =
Расписание_игр.id_игры
UNION
SELECT Название_команды2 AS Команда, (Расписание_игр.Счет2 -
Расписание_игр.Счет1) AS Разница_забитых_и_пропущенных
FROM Команды_в_игре
JOIN Расписание_игр ON Команды_в_игре.id_игры =
Расписание_игр.id_игры;

SELECT Команда, Разница_забитых_и_пропущенных
FROM Разница
WHERE Разница_забитых_и_пропущенных =
(SELECT MIN(Разница_забитых_и_пропущенных) FROM Разница)
OR
Разница_забитых_и_пропущенных =
(SELECT MAX(Разница_забитых_и_пропущенных) FROM Разница);

```