

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

Гирман А.В.

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 5

Пусть требуется создать программную систему, предназначенную для работников приемной комиссии высшего учебного заведения. Она должна обеспечивать хранение, просмотр и изменение сведений об абитуриентах, а также о расписании экзаменов и консультаций. Результатом работы приемной комиссии должен быть список абитуриентов, зачисленных в институт. Секретарь приемной комиссии регистрирует абитуриентов. Для каждого абитуриента в базу данных заносятся следующие сведения: фамилия, имя, отчество, паспортные данные, какое учебное заведение, где и когда окончил, наличие золотой или серебряной медали, название кафедры и факультета, на которые поступает абитуриент. При регистрации абитуриенту выдают экзаменационный лист, имеющий уникальный номер, и сообщают номер группы и потока. Группы формируются на период вступительных экзаменов и объединяются в потоки по 3-4 группы. Для каждой группы по каждому предмету в базу данных заносится экзаменационная ведомость. Оценка, полученная абитуриентом, может быть изменена на апелляции. Абитуриент может не только подать, но и забрать документы, а также перевести их на другую кафедру. Для каждого потока формируется расписание консультаций и экзаменов по предметам. Медалисты сдают только один экзамен. Известно количество мест на каждый факультет. Приемная комиссия по результатам экзаменов должна сформировать списки абитуриентов, зачисленных в институт. Секретарю приемной комиссии могут потребоваться следующие сведения:

- Список абитуриентов на заданный факультет?
- Оценки, полученные указанным абитуриентом?

- Когда и в какой аудитории будет консультация и экзамен у заданного абитуриента по указанному предмету?
- Где, когда и по каким предметам будут проходить экзамены у заданной группы?
- Конкурс на каждый факультет?
- Средний балл по каждому предмету на каждом факультете?

Выполнение работы.

Создана база данных под названием «Приемная_комиссия».

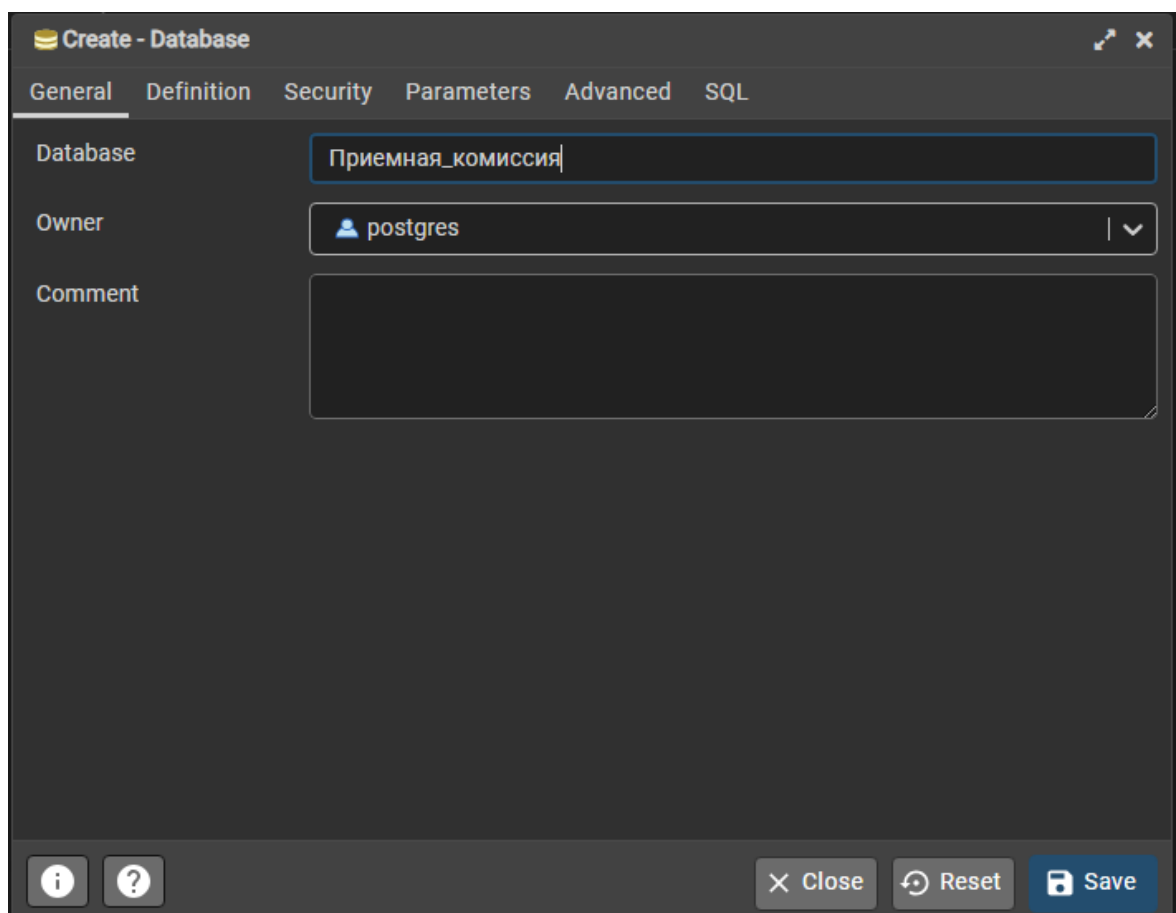


Рисунок 1. Создание БД "Приемная_комиссия"

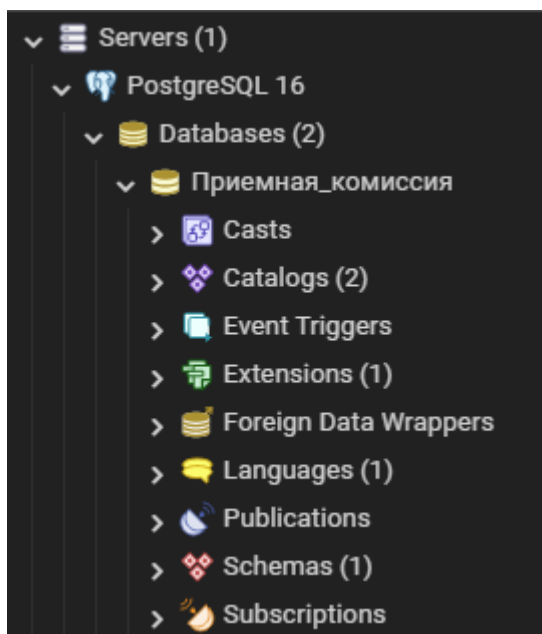


Рисунок 2. Вид созданной БД.

Созданы таблицы «Абитуриент», «Ведомость», «Группа», «Кафедра», «Консультация», «Поток», «Предмет», «Факультет», «Экзамен» в соответствии со структурой БД и написаны запросы на их заполнение.

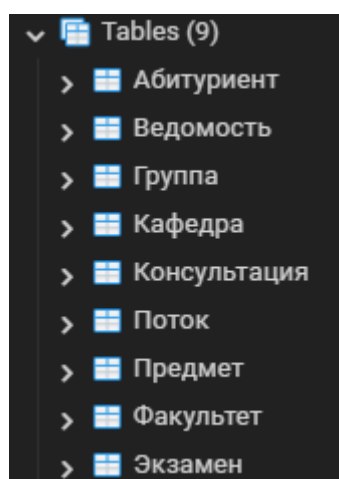


Рисунок 3. Созданные таблицы.

Написаны запросы на вывод нужных сведений для секретаря приемной комиссии:

Query

Query History

1

SELECT Уникальный_номер, Фамилия, Имя, Отчество, Факультет.Название AS Факультет

2

FROM Абитуриент

3

JOIN Кафедра ON Абитуриент.Кафедра_id = Кафедра.Кафедра_id

4

JOIN Факультет ON Кафедра.Факультет_id = Факультет.Факультет_id

5

WHERE Факультет.Название = 'ФКТИ';

Data Output

Messages

Notifications

	Уникальный_номер integer	Фамилия text	Имя text	Отчество text	Факультет text	
1	15089468	Шестаков	Вальтер	Всеволодович	ФКТИ	
2	19828121	Меркушев	Альфред	Протасьевич	ФКТИ	
3	21952948	Носова	Рамина	Михаиловна	ФКТИ	
4	66862382	Лукина	Анэля	Аристарховна	ФКТИ	
5	98664711	Кузнецова	Юнона	Ильяовна	ФКТИ	

Рисунок 4. Запрос на вывод: список абитуриентов на заданный факультет.

Query

Query History

1

SELECT

Фамилия, Имя, Отчество, Название

AS

Предмет, Оценка

2

FROM

Ведомость

3

JOIN

Абитуриент

ON

Ведомость.Уникальный_номер = Абитуриент.Уникальный_номер

4

JOIN

Экзамен

ON

Ведомость.Экзамен_id = Экзамен.Экзамен_id

5

JOIN

Предмет

ON

Экзамен.Предмет_id = Предмет.Предмет_id

6

WHERE

Абитуриент.Уникальный_номер = 19828121;

Data Output

Messages

Notifications

≡+

📄

▼

📋

▼

🗑

🗄

⬇

📈

	Фамилия text 🔒	Имя text 🔒	Отчество text 🔒	Предмет text 🔒	Оценка smallint 🔒
1	Меркушев	Альфред	Протасьевич	Математика	4
2	Меркушев	Альфред	Протасьевич	Физика	5
3	Меркушев	Альфред	Протасьевич	Информатика	3

Рисунок 5. Запрос на вывод: оценки, полученные указанным абитуриентом.

Query	Query History
1	SELECT Факультет.Название, COUNT (Уникальный_номер) AS Конкурс
2	FROM Абитуриент
3	JOIN Кафедра ON Абитуриент.Кафедра_id = Кафедра.Кафедра_id
4	JOIN Факультет ON Кафедра.Факультет_id = Факультет.Факультет_id
5	GROUP BY Факультет.Название;

Data Output	Messages	Notifications									
<div> <div>≡+</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑️</div> <div>📦</div> <div>⬇️</div> <div>📈</div> </div> <table> <tr> <th></th><th>Название text</th><th>Конкурс bigint</th></tr> <tr> <td>1</td><td>ГФ</td><td>3</td></tr> <tr> <td>2</td><td>ФКТИ</td><td>5</td></tr> </table>		Название text	Конкурс bigint	1	ГФ	3	2	ФКТИ	5		
	Название text	Конкурс bigint									
1	ГФ	3									
2	ФКТИ	5									

Рисунок 8. Запрос на вывод: конкурс на каждый факультет.

Query

Query History

1

SELECT

Факультет.Название

AS

Факультет, Предмет.Название

AS

Предмет, ROUND(AVG(Оценка), 2)

AS

Средний_бал

2

FROM

Абитуриент

3

JOIN

Кафедра

ON

Абитуриент.Кафедра_id = Кафедра.Кафедра_id

4

JOIN

Факультет

ON

Кафедра.Факультет_id = Факультет.Факультет_id

5

JOIN

Ведомость

ON

Абитуриент.Уникальный_номер = Ведомость.Уникальный_номер

6

JOIN

Экзамен

ON

Ведомость.Экзамен_id = Экзамен.Экзамен_id

7

JOIN

Предмет

ON

Экзамен.Предмет_id = Предмет.Предмет_id

8

GROUP BY

Факультет, Предмет

9

ORDER BY

1;

Data Output

Messages

Notifications

≡+

📄

▼

📋

▼

🗑️

📦

⬇️

📈

	Факультет text	Предмет text	Средний_бал numeric
1	ГФ	Английский Язык	5.00
2	ГФ	Литература	4.33
3	ГФ	Русский Язык	3.50
4	ФКТИ	Информатика	3.50
5	ФКТИ	Математика	3.40
6	ФКТИ	Физика	4.50

Рисунок 9. Запрос на вывод: средний балл по каждому предмету на каждом факультете.

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/21>

DB Fiddle: <https://www.db-fiddle.com/f/izZR3d787LEDK7hxc4631A/4>

Приложение Б

Исходный код

Файл lab2.sql:

```
CREATE TABLE Факультет (  
    Факультет_id SMALLSERIAL PRIMARY KEY,  
    Название TEXT NOT NULL,  
    Мест SMALLINT NOT NULL  
);  
  
CREATE TABLE Кафедра (  
    Кафедра_id SMALLSERIAL PRIMARY KEY,  
    Название TEXT NOT NULL,  
    Факультет_id SMALLINT,  
    FOREIGN KEY (Факультет_id) REFERENCES Факультет(Факультет_id)  
ON DELETE RESTRICT  
);  
  
CREATE TABLE Поток (  
    Поток_id SMALLSERIAL PRIMARY KEY,  
    Номер SMALLINT NOT NULL  
);  
  
CREATE TABLE Группа (  
    Группа_id SERIAL PRIMARY KEY,  
    Номер SMALLINT NOT NULL,  
    Поток_id SMALLINT,  
    FOREIGN KEY (Поток_id) REFERENCES Поток(Поток_id) ON DELETE  
SET NULL  
);  
  
CREATE TABLE Абитуриент (  
    Уникальный_номер SERIAL PRIMARY KEY,  
    Фамилия TEXT NOT NULL,  
    Имя TEXT NOT NULL,
```

```

        Отчество TEXT NOT NULL,
        Серия_паспорта SMALLINT NOT NULL,
        Номер_паспорта INTEGER NOT NULL,
        Учебное_заведение TEXT,
        Город TEXT,
        Дата_окончания_УЗ DATE,
        Медаль TEXT,
        Статус_документов TEXT,
        Количество_экзаменов SMALLINT NOT NULL,
        Кафедра_id SMALLINT,
        FOREIGN KEY (Кафедра_id) REFERENCES Кафедра(Кафедра_id) ON
DELETE SET NULL,
        Группа_id INTEGER,
        FOREIGN KEY (Группа_id) REFERENCES Группа(Группа_id) ON
DELETE SET NULL
    );

```

```

CREATE TABLE Предмет (
    Предмет_id SMALLSERIAL PRIMARY KEY,
    Название TEXT NOT NULL
);

```

```

CREATE TABLE Экзамен (
    Экзамен_id SMALLSERIAL PRIMARY KEY,
    Поток_id SMALLINT,
    FOREIGN KEY (Поток_id) REFERENCES Поток(Поток_id) ON DELETE
SET NULL,
    Предмет_id SMALLINT,
    FOREIGN KEY (Предмет_id) REFERENCES Предмет(Предмет_id) ON
DELETE CASCADE,
    Дата DATE NOT NULL,
    Время TIME NOT NULL,
    Аудитория INTEGER
);

```

```

CREATE TABLE Консультация (
    Консультация_id SERIAL PRIMARY KEY,
    Экзамен_id INTEGER,
    FOREIGN KEY (Экзамен_id) REFERENCES Экзамен(Экзамен_id) ON
DELETE SET NULL,
    Дата DATE NOT NULL,
    Время TIME NOT NULL,
    Аудитория INTEGER
);

```

```

CREATE TABLE Ведомость (
    Экзамен_id SMALLINT,
    Уникальный_номер INTEGER,
    PRIMARY KEY (Экзамен_id, Уникальный_номер),
    FOREIGN KEY (Экзамен_id) REFERENCES Экзамен(Экзамен_id) ON
DELETE SET NULL,
    FOREIGN KEY (Уникальный_номер) REFERENCES
Абитуриент(Уникальный_номер) ON DELETE SET NULL,
    Оценка SMALLINT NOT NULL,
    Апелляция BOOLEAN NOT NULL
);

```

```

INSERT INTO Факультет(Название, Мест) VALUES
    ('ФРТ', 155),
    ('ФЭЛ', 198),
    ('ФКТИ', 308),
    ('ФЭА', 243),
    ('ФИБС', 215),
    ('ИНПРОТЕХ', 52),
    ('ГФ', 20);

```

```

INSERT INTO Кафедра(Название, Факультет_id) VALUES
    ('РЭС', 1),
    ('ФЭТ', 2),
    ('МО ЭВМ', 3),

```

```
('КСУ', 4),  
( 'БТС', 5),  
( 'МСК', 6),  
( 'ИНЯЗ', 7);
```

```
INSERT INTO Поток(Номер) VALUES
```

```
(1),  
(2);
```

```
INSERT INTO Группа(Номер, Поток_id) VALUES
```

```
(1301, 1),  
(1302, 1),  
(1303, 1),  
(1304, 1),  
(1305, 2),  
(1306, 2),  
(1307, 2);
```

```
INSERT INTO Абитуриент VALUES
```

```
(15834161, 'Коновалов', 'Гавриил', 'Анатольевич', 5719,  
529588, 'МАОУ СОШ №3', 'Пермь', '2023-05-20', 'Золотая', 'Поданы',  
1, 7, 5),  
(15089468, 'Шестаков', 'Вальтер', 'Всеволодович', 2419,  
666949, 'Лицей №67', 'Иваново', '2023-05-25', 'Золотая', 'Поданы',  
1, 3, 1),  
(13669205, 'Авдеев', 'Леонтий', 'Эдуардович', 6119, 975207,  
'Многопрофильная школа № 17', 'Рязань', '2023-05-31', NULL, NULL,  
4, 7, 5),  
(19828121, 'Меркушев', 'Альфред', 'Протасьевич', 8919,  
447416, 'Школа №27', 'Мордовия', '2023-05-31', NULL, 'Поданы', 4,  
3, 1),  
(21952948, 'Носова', 'Рамина', 'Михаиловна', 9620, 602158,  
'Школа №48', 'Грозный', '2023-05-15', 'Золотая', NULL, 1, 3, 3),
```

```

(66862382, 'Лукина', 'Анэля', 'Аристарховна', 7919, 269571,
'Школа №15', 'Майкоп', '2023-05-31', 'Серебряная', 'Поданы', 1, 3,
2),
(98664711, 'Кузнецова', 'Юнона', 'Ильяовна', 3620, 328391,
'МБОУ Школа №32', 'Самара', '2023-05-31', NULL, 'Поданы', 4, 3,
3),
(12636043, 'Мартынова', 'Карина', 'Мартыновна', 4519, 473248,
'Школа №91', 'Москва', '2023-05-31', NULL, 'Перевод', 4, 7, 6);

```

```

INSERT INTO Предмет(Название) VALUES

```

```

('Математика'),
('Физика'),
('Информатика'),
('Физкультура'),
('Русский Язык'),
('Английский Язык'),
('Литература');

```

```

INSERT INTO Экзамен(Поток_id, Предмет_id, Дата, Время, Аудитория)
VALUES

```

```

(1, 1, '2023-08-10', '10:00', 3322),
(1, 2, '2023-08-15', '9:00', 3102),
(1, 3, '2023-08-20', '15:00', 5413),
(2, 7, '2023-08-16', '11:00', 3413),
(2, 6, '2023-08-19', '9:00', 5413),
(2, 5, '2023-08-23', '14:00', 3102);

```

```

INSERT INTO Консультация(Экзамен_id, Дата, Время, Аудитория)
VALUES

```

```

(1, '2024-08-05', '18:00', NULL),
(2, '2024-08-14', '13:00', 3102),
(3, '2024-08-19', '15:00', NULL),
(4, '2024-08-08', '15:30', NULL),
(5, '2024-08-14', '17:00', NULL),
(6, '2024-08-17', '18:30', NULL);

```

```
INSERT INTO Ведомость VALUES
```

```
(1, 15089468, 5, 'no'),  
(1, 19828121, 4, 'no'),  
(1, 21952948, 3, 'no'),  
(1, 66862382, 2, 'yes'),  
(1, 98664711, 3, 'yes'),  
(2, 19828121, 5, 'no'),  
(2, 98664711, 4, 'no'),  
(3, 19828121, 3, 'yes'),  
(3, 98664711, 4, 'no'),  
(4, 15834161, 5, 'no'),  
(4, 13669205, 4, 'no'),  
(4, 12636043, 4, 'yes'),  
(5, 13669205, 5, 'no'),  
(5, 12636043, 5, 'no'),  
(6, 13669205, 4, 'no'),  
(6, 12636043, 3, 'no');
```

```
-- Список абитуриентов на заданный факультет?
```

```
SELECT      Уникальный_номер,      Фамилия,      Имя,      Отчество,  
Факультет.Название AS Факультет  
FROM Абитуриент  
JOIN Кафедра ON Абитуриент.Кафедра_id = Кафедра.Кафедра_id  
JOIN Факультет ON Кафедра.Факультет_id = Факультет.Факультет_id  
WHERE Факультет.Название = 'ФКТИ';
```

```
-- Оценки, полученные указанным абитуриентом?
```

```
SELECT Фамилия, Имя, Отчество, Название AS Предмет, Оценка  
FROM Ведомость  
JOIN      Абитуриент      ON      Ведомость.Уникальный_номер      =  
Абитуриент.Уникальный_номер  
JOIN Экзамен ON Ведомость.Экзамен_id = Экзамен.Экзамен_id  
JOIN Предмет ON Экзамен.Предмет_id = Предмет.Предмет_id  
WHERE Абитуриент.Уникальный_номер = 19828121;
```

```
-- Когда и в какой аудитории будет консультация и экзамен у
заданного абитуриента по указанному предмету?
SELECT    Фамилия,      Имя,      Отчество,      Название      AS      Предмет,
Консультация.Дата AS Дата_Конс,
          Консультация.Время AS Время_Конс, Консультация.Аудитория AS
Аудитория_Конс,
          Экзамен.Дата AS Дата_Экз, Экзамен.Время AS Время_Экз,
          Экзамен.Аудитория AS Аудитория_Экз
FROM Абитуриент
JOIN Группа ON Абитуриент.Группа_id = Группа.Группа_id
JOIN Поток ON Группа.Поток_id = Поток.Поток_id
JOIN Экзамен ON Поток.Поток_id = Экзамен.Поток_id
JOIN Предмет ON Экзамен.Предмет_id = Предмет.Предмет_id
JOIN Консультация ON Экзамен.Экзамен_id = Консультация.Экзамен_id
WHERE Абитуриент.Уникальный_номер = 19828121 AND Название =
'Математика';
```

```
-- Где, когда и по каким предметам будут проходить экзамены у
заданной группы?
SELECT Группа.Номер AS Группа, Название AS Предмет, Дата, Время,
Аудитория
FROM Группа
JOIN Поток ON Группа.Поток_id = Поток.Поток_id
JOIN Экзамен ON Поток.Поток_id = Экзамен.Поток_id
JOIN Предмет ON Экзамен.Предмет_id = Предмет.Предмет_id
WHERE Группа.Номер = 1303;
```

```
-- Конкурс на каждый факультет?
SELECT Факультет.Название, COUNT(Уникальный_номер) AS Конкурс
FROM Абитуриент
JOIN Кафедра ON Абитуриент.Кафедра_id = Кафедра.Кафедра_id
JOIN Факультет ON Кафедра.Факультет_id = Факультет.Факультет_id
GROUP BY Факультет.Название;
```

```

-- Средний балл по каждому предмету на каждом факультете?
SELECT Факультет.Название AS Факультет, Предмет.Название AS
Предмет, ROUND(AVG(Оценка), 2) AS Средний_балл
FROM Абитуриент
JOIN Кафедра ON Абитуриент.Кафедра_id = Кафедра.Кафедра_id
JOIN Факультет ON Кафедра.Факультет_id = Факультет.Факультет_id
JOIN      Ведомость      ON      Абитуриент.Уникальный_номер      =
Ведомость.Уникальный_номер
JOIN Экзамен ON Ведомость.Экзамен_id = Экзамен.Экзамен_id
JOIN Предмет ON Экзамен.Предмет_id = Предмет.Предмет_id
GROUP BY Факультет, Предмет
ORDER BY 1;

```