

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение высшего
образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Отчет

по лабораторной работе №4 «Запросы на выборку данных к БД PostgreSQL.
Представления. Работа с индексами»

по дисциплине **«Проектирование и реализация баз данных»**

Автор: Кузнецов А. Г.

Факультет: ИКТ

Группа: K3240

Преподаватель: Говорова М.М.



Санкт-Петербург 2023

Оглавление

Цель работы	3
Практическое задание	3
Выполнение	4
Запросы к базе данных.....	5
Представления	16
Запросы на модификацию данных	18
Создание индексов	21
Вывод.....	22

Цель работы

Овладеть практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL, использования подзапросов при модификации данных и индексов.

Практическое задание

1. Создать запросы и представления на выборку данных к базе данных PostgreSQL (согласно индивидуальному заданию, часть 2 и 3).
2. Составить 3 запроса на модификацию данных (INSERT, UPDATE, DELETE) с использованием подзапросов.
3. Изучить графическое представление запросов и просмотреть историю запросов.
4. Создать простой и составной индексы для двух произвольных запросов и сравнить время выполнения запросов без индексов и с индексами. Для получения плана запроса использовать команду EXPLAIN.

Выполнение

Вариант 12. БД «Сессия»

Схема логической модели базы данных, сгенерированная в Generate ERD указана на рисунке 1.

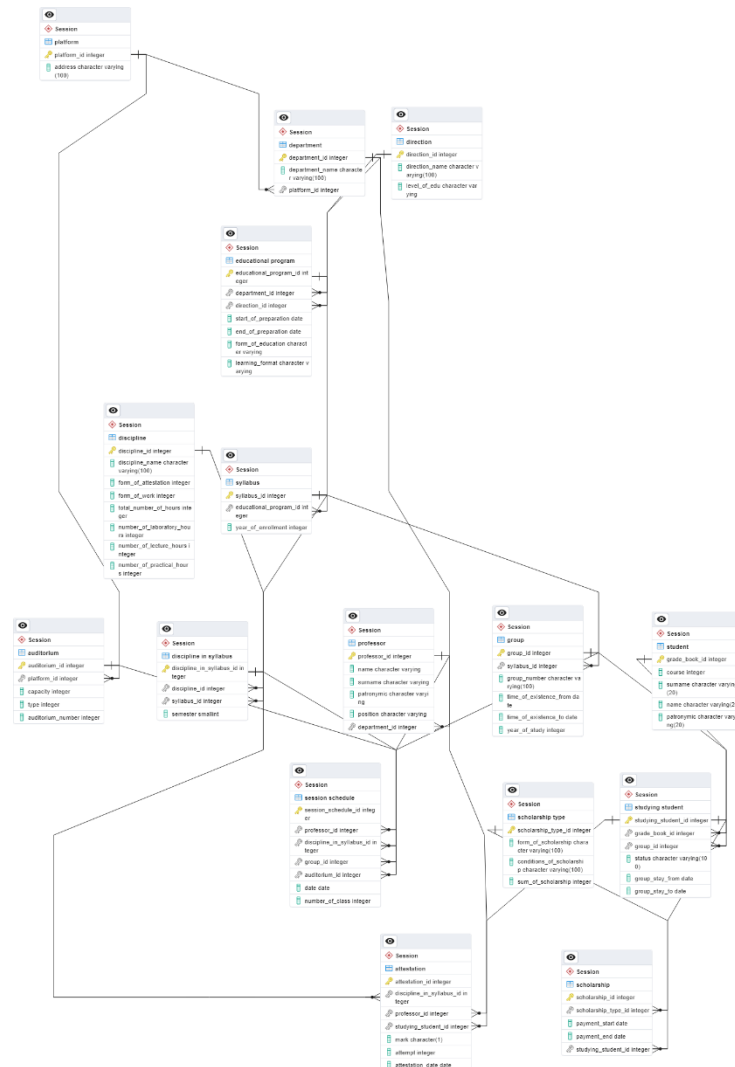


Рисунок 1 – Схема логической модели базы данных.

Запросы к базе данных

1. Составить список дисциплин, которые должны быть сданы заданной группой с указанием дат сдачи и фамилий преподавателей.

```
SELECT
    "Session"."discipline".discipline_name,
    "Session"."session schedule".attestation_date,
    "Session"."professor".professor_surname
FROM
    "Session"."session schedule"
    JOIN "Session"."group" ON "Session"."session schedule".group_id = "Session"."group".group_id
    JOIN "Session"."discipline in syllabus" ON "Session"."session schedule".discipline_in_syllabus_id
= "Session"."discipline in syllabus".discipline_in_syllabus_id
    JOIN "Session"."discipline" ON "Session"."discipline in syllabus".discipline_id =
"Session"."discipline".discipline_id
    JOIN "Session".professor ON "Session"."session schedule".professor_id =
"Session".professor.professor_id
WHERE
    "Session"."group".group_id = 22 AND (
    (
        EXTRACT(MONTH FROM CURRENT_DATE) = 1
        AND (
            (EXTRACT(MONTH FROM "Session"."session schedule".attestation_date) = 1 AND EXTRACT(YEAR
FROM "Session"."session schedule".attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
            OR
            (EXTRACT(MONTH FROM "Session"."session schedule".attestation_date) = 12 AND EXTRACT(YEAR
FROM "Session"."session schedule".attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) - 1)
        )
    ) OR (
        EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 7 AND 12
        AND (
            (EXTRACT(MONTH FROM "Session"."session schedule".attestation_date) = 12 AND EXTRACT(YEAR
FROM "Session"."session schedule".attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
            OR
            (EXTRACT(MONTH FROM "Session"."session schedule".attestation_date) = 1 AND EXTRACT(YEAR
FROM "Session"."session schedule".attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) + 1)
        )
    ) OR (
        EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 2 AND 6
        AND EXTRACT(MONTH FROM "Session"."session schedule".attestation_date) BETWEEN 5 AND 6
        AND EXTRACT(YEAR FROM "Session"."session schedule".attestation_date) = EXTRACT(YEAR FROM
CURRENT_DATE)
    )
    )
)
```

	discipline_name character varying (100)	attestation_date date	professor_surname character varying
1	Объектно-ориентированное программирование	2024-01-14	Петрова
2	Математический анализ	2023-12-26	Иванов

2. Вывести список студентов, получивших двойки на первой попытке с указанием фамилии преподавателя, которым они должны пересдать экзамен.

```
SELECT
    "Session"."student".student_surname,
    "Session".attestation.mark,
    "Session"."professor".professor_surname
```

```


FROM
  "Session".attestation
  JOIN "Session"."studying student" ON "Session".attestation.studying_student_id =
"Session"."studying student".studying_student_id
  JOIN "Session".student ON "Session"."studying student".grade_book_id =
"Session".student.grade_book_id
  JOIN "Session".professor ON "Session".attestation.professor_id = "Session".professor.professor_id
WHERE
  "Session".attestation.attempt = 1 AND "Session".attestation.mark = 'F' AND (
    (
      EXTRACT(MONTH FROM CURRENT_DATE) = 12
      AND (
        (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) + 1)
        OR
        (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
      )
    ) OR (
      EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 1 AND 4
      AND (
        (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) - 1)
        OR
        (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
      )
    ) OR (
      EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 5 AND 11
      AND EXTRACT(MONTH FROM "Session".attestation.attestation_date) BETWEEN 5 AND 6
      AND EXTRACT(YEAR FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM
CURRENT_DATE)
    )
  )

```

	student_surname character varying (20) 🔒	mark character 🔒	professor_surname character varying 🔒
1	Горшкова	F	Иванов
2	Степанов	F	Агатьева
3	Сергеев	F	Семёнов
4	Одинцов	F	Носик
5	Куликов	F	Яналова
6	Евсеева	F	Зорина

3. Вывести фамилии студентов, получивших оценки по дисциплине, которые выше среднего балла по этой дисциплине.

```
SELECT
    "Session".student.student_surname
FROM
    "Session".student
    JOIN "Session"."studying student" ON "Session".student.grade_book_id = "Session"."studying
student".grade_book_id
    JOIN "Session".attestation ON "Session"."studying student".studying_student_id =
"Session".attestation.studying_student_id
    JOIN "Session"."discipline in syllabus" ON "Session".attestation.discipline_in_syllabus_id =
"Session"."discipline in syllabus".discipline_in_syllabus_id
    JOIN "Session".discipline ON "Session"."discipline in syllabus".discipline_id =
"Session".discipline.discipline_id
WHERE
    CASE "Session".attestation.mark
        WHEN 'A' THEN 5
        WHEN 'B' THEN 4
        WHEN 'C' THEN 4
        WHEN 'D' THEN 3
        WHEN 'E' THEN 3
        WHEN 'F' THEN 2
    END > (
        SELECT AVG(
            CASE "Session".attestation.mark
                WHEN 'A' THEN 5
                WHEN 'B' THEN 4
                WHEN 'C' THEN 4
                WHEN 'D' THEN 3
                WHEN 'E' THEN 3
                WHEN 'F' THEN 2
            END
        )
        FROM "Session".attestation
        WHERE "Session".attestation.discipline_in_syllabus_id = "Session"."discipline in
syllabus".discipline_in_syllabus_id
        ) AND (
            (
                EXTRACT(MONTH FROM CURRENT_DATE) = 12
                AND (
                    (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) + 1)
                    OR
                    (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
                )
            ) OR (
                EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 1 AND 4
                AND (
                    (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) - 1)
                    OR
                    (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
                )
            ) OR (
                EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 5 AND 11
                AND EXTRACT(MONTH FROM "Session".attestation.attestation_date) BETWEEN 5 AND 6
                AND EXTRACT(YEAR FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM
CURRENT_DATE)
            )
        )
    )
```

	student_surname character varying (20) 
1	Морозов
2	Суханов
3	Суханов
4	Назаров
5	Назаров
6	Куликов
7	Куликов
8	Стрелков
9	Стрелков
10	Сазонов
11	Сазонов
12	Савельев
13	Савельев
14	Чернов
15	Маслов
16	Фролов
17	Аксёнова
18	Аксёнова
19	Боброва
20	Рябова
21	Казакова

22	Казакова
23	Титова
24	Титова
25	Рожкова
26	Кулагина
27	Кулагина
28	Владимирова
29	Владимирова
30	Евсеева
31	Панфилова
32	Шубина
33	Абрамова
34	Абрамова
35	Лобанов
36	Данилов
37	Кондратьев
38	Щербаков
39	Куликов
40	Сергеев
41	Ефремов



42	Ефремов
43	Шестаков
44	Никифоров
45	Фролов
46	Фролов
47	Тимофеев
48	Евсеев
49	Евсеев
50	Устинов
51	Соболев

4. Создать рейтинговый список групп по заданному направлению по результатам сдачи сессии, упорядочить его по убыванию.

```

SELECT
    "Session"."group".group_number,
    ROUND(AVG(CASE "Session".attestation.mark
        WHEN 'A' THEN 5
        WHEN 'B' THEN 4
        WHEN 'C' THEN 4
        WHEN 'D' THEN 3
        WHEN 'E' THEN 3
        WHEN 'F' THEN 2
    END), 2) AS average_mark
FROM
    "Session"."group"
    JOIN "Session"."studying student" ON "Session"."group".group_id = "Session"."studying
student".group_id
    JOIN "Session"."attestation" ON "Session"."studying student".studying_student_id =
"Session"."attestation".studying_student_id
    JOIN "Session"."discipline in syllabus" ON "Session"."attestation".discipline_in_syllabus_id =
"Session"."discipline in syllabus".discipline_in_syllabus_id
    JOIN "Session"."syllabus" ON "Session"."discipline in syllabus".syllabus_id =
"Session"."syllabus".syllabus_id
    JOIN "Session"."educational program" ON "Session"."syllabus".educational_program_id =
"Session"."educational program".educational_program_id
    JOIN "Session"."direction" ON "Session"."educational program".direction_id =
"Session"."direction".direction_id
WHERE
    "Session"."direction".direction_id = 276 AND (
    (
        EXTRACT(MONTH FROM CURRENT_DATE) = 12
        AND (
            (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) + 1)
            OR
            (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
        )
    ) OR (
        EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 1 AND 4
        AND (
            (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) - 1)
            OR
            (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
        )
    ) OR (
        EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 5 AND 11
        AND EXTRACT(MONTH FROM "Session".attestation.attestation_date) BETWEEN 5 AND 6
        AND EXTRACT(YEAR FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM
CURRENT_DATE)
    )
)
GROUP BY
    "Session"."group".group_number
ORDER BY
    average_mark DESC;

```

	group_number character varying (100) 	average_mark numeric 
1	F6594	5.0000000000000000
2	C2041	4.3333333333333333

5. Создайте списки студентов, упорядоченные по группам и фамилиям студентов, содержащие данные о средних баллах и назначении на стипендии. Студент получает стипендию, если он сдал сессию без троек. Если студент не назначен на стипендию, указать 0, если назначен – 1.

```
SELECT
    "Session"."group".group_number,
    "Session".student.student_surname,
    ROUND(AVG(CASE "Session".attestation.mark
        WHEN 'A' THEN 5
        WHEN 'B' THEN 4
        WHEN 'C' THEN 4
        WHEN 'D' THEN 3
        WHEN 'E' THEN 3
        WHEN 'F' THEN 2
    END), 2) AS average_mark,
    CASE
        WHEN COUNT(CASE WHEN "Session".attestation.mark IN ('F', 'E', 'D') THEN 1 END) = 0 THEN 1
        ELSE 0
    END AS scholarship_assigned
FROM
    "Session"."studying student"
    JOIN "Session".student ON "Session"."studying student".grade_book_id =
"Session".student.grade_book_id
    JOIN "Session"."group" ON "Session"."studying student".group_id = "Session"."group".group_id
    JOIN "Session".attestation ON "Session"."studying student".studying_student_id =
"Session".attestation.studying_student_id
WHERE
    (
        EXTRACT(MONTH FROM CURRENT_DATE) = 12
        AND (
            (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) + 1)
            OR
            (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
        )
    ) OR (
        EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 1 AND 4
        AND (
            (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) - 1)
            OR
            (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
        )
    ) OR (
        EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 5 AND 11
        AND EXTRACT(MONTH FROM "Session".attestation.attestation_date) BETWEEN 5 AND 6
        AND EXTRACT(YEAR FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM
CURRENT_DATE)
    )
GROUP BY
    "Session"."group".group_number,
    "Session".student.student_surname
ORDER BY
    "Session"."group".group_number,
    "Session".student.student_surname;
```

	group_number character varying (100) 🔒	student_surname character varying (20) 🔒	average_mark numeric 🔒	scholarship_assigned integer 🔒
1	C2041	Суханов	5.00	1
2	C2041	Чернов	4.00	0
3	C2041	Шубина	4.00	0
4	C2513	Абрамова	5.00	1
5	C2513	Горшкова	2.50	0
6	C2513	Королёв	3.00	0
7	C2513	Рожкова	3.50	0
8	C2513	Третьякова	3.50	0
9	C2513	Фролов	4.00	0
10	C3415	Алексеев	2.50	0
11	C3415	Моисеева	2.50	0
12	C3415	Молчанов	3.00	0
13	C3415	Никифоров	3.50	0
14	C3415	Стрелков	5.00	1
15	C3415	Устинов	3.50	0
16	C3415	Шестаков	4.00	0
17	C9593	Титова	5.00	1
18	F6594	Казакова	5.00	1
19	F6594	Лобанов	4.50	1
20	F6594	Фролов	5.00	1
21	F9115	Афанасьев	3.50	0
22	F9115	Карпова	3.50	0
23	F9115	Куликов	5.00	1
24	F9115	Рябова	3.50	0
25	F9115	Сергеев	3.50	0
26	F9216	Кудряшова	3.50	0
27	F9216	Носова	3.00	0
28	F9216	Шарапов	3.00	0
29	K3933	Кондратьев	3.50	0
30	K6170	Зимин	2.50	0
31	K6170	Соболев	4.50	1
32	M4016	Боброва	4.50	1
33	M4016	Гаврилова	3.00	0
34	M4016	Евсеев	5.00	1
35	M4016	Калашников	3.50	0
36	M4016	Карпова	3.00	0
37	M6066	Ефремов	5.00	1
38	M6066	Тимофеев	4.00	0
39	M8812	Владимирова	4.50	1
40	M8812	Евсеева	3.00	0
41	M8812	Маслов	4.00	0
42	M8812	Фокин	3.00	0
43	N3658	Куликов	3.00	0
44	N3658	Морозов	3.50	0
45	N3658	Савельев	4.50	1
46	N3658	Сазонов	5.00	1
47	N3658	Шестаков	2.50	0
48	N3658	Щербаков	3.00	0
49	N7290	Аксёнова	5.00	1
50	N7290	Гуляева	3.00	0
51	N7290	Назаров	4.50	1
52	N7290	Рогов	2.00	0
53	N7515	Потапов	2.50	0
54	N8944	Самойлов	3.00	0
55	P2534	Герасимова	3.00	0
56	P2534	Игнатова	3.50	0
57	P2534	Кулагина	5.00	1
58	P2534	Одинцов	2.50	0
59	P6589	Данилов	4.00	0
60	P6589	Панфилова	4.00	0
61	P6589	Степанов	2.50	0

6. Вывести список студентов, сдавших все положенные экзамены.

```
SELECT
    "Session"."student".grade_book_id,
    "Session"."student".student_surname,
    "Session"."student".student_name,
    "Session"."student".student_patronymic
FROM
    "Session"."student"
    JOIN "Session"."studying student" ON "Session"."student".grade_book_id = "Session"."studying
student".grade_book_id
    JOIN "Session"."session schedule" ON "Session"."studying student".group_id = "Session"."session
schedule".group_id
    JOIN "Session".attestation ON "Session"."studying student".studying_student_id =
"Session".attestation.studying_student_id
WHERE
    (
        (
            EXTRACT(MONTH FROM CURRENT_DATE) = 12
            AND (
                (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) + 1)
                OR
                (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
            )
        ) OR (
            EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 1 AND 4
            AND (
                (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) - 1)
                OR
                (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
            )
        ) OR (
            EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 5 AND 11
            AND EXTRACT(MONTH FROM "Session".attestation.attestation_date) BETWEEN 5 AND 6
            AND EXTRACT(YEAR FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM
CURRENT_DATE)
        )
    )
    AND (
        (
            EXTRACT(MONTH FROM CURRENT_DATE) = 12
            AND (
                (EXTRACT(MONTH FROM "Session"."session schedule".attestation_date) = 1 AND
EXTRACT(YEAR FROM "Session"."session schedule".attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) +
1)
                OR
                (EXTRACT(MONTH FROM "Session"."session schedule".attestation_date) = 12 AND
EXTRACT(YEAR FROM "Session"."session schedule".attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
            )
        ) OR (
            EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 1 AND 4
            AND (
                (EXTRACT(MONTH FROM "Session"."session schedule".attestation_date) = 12 AND
EXTRACT(YEAR FROM "Session"."session schedule".attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) -
1)
                OR
                (EXTRACT(MONTH FROM "Session"."session schedule".attestation_date) = 1 AND
EXTRACT(YEAR FROM "Session"."session schedule".attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
            )
        ) OR (
            EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 5 AND 11
            AND EXTRACT(MONTH FROM "Session"."session schedule".attestation_date) BETWEEN 5 AND 6
            AND EXTRACT(YEAR FROM "Session"."session schedule".attestation_date) = EXTRACT(YEAR FROM
CURRENT_DATE)
        )
    )
)
```

GROUP BY

```
"Session"."student".grade_book_id,
"Session"."student".student_surname,
"Session"."student".student_name,
"Session"."student".student_patronymic
```

HAVING

```
COUNT(DISTINCT "Session".attestation.attestation_id) = COUNT(DISTINCT "Session"."session
schedule".session_schedule_id)
```

	grade_book_id [PK] integer	student_surname character varying (20)	student_name character varying (20)	student_patronymic character varying (20)
1	706939	Кудряшова	Юнона	Протасьевна
2	264239	Морозов	Игорь	Онисимович
3	669646	Панфилова	Владислава	Геласьевна
4	793977	Савельев	Ипполит	Ильяович
5	164932	Зимин	Юстин	Фролович
6	788349	Евсеев	Петр	Витальевич
7	669647	Калашников	Филипп	Васильевич
8	793980	Степанов	Аввакуум	Ростиславович
9	593771	Боброва	Сабрина	Артемовна
10	985248	Маслов	Лука	Максимович
11	494414	Назаров	Владлен	Данилович
12	689100	Стрелков	Святослав	Семёнович
13	734628	Тимофеев	Евгений	Георгиевич
14	491385	Куликов	Ипполит	Георгьевич
15	689103	Никифоров	Сергей	Петрович
16	494417	Одинцов	Дмитрий	Валерьянович
17	985250	Афанасьев	Вальтер	Созонович
18	706941	Фролов	Парамон	Валерьевич
19	593770	Потапов	Герасим	Леонидович
20	706938	Сазонов	Георгий	Степанович
21	793979	Данилов	Василий	Русланович
22	182992	Фролов	Михаил	Феликсович
23	706940	Абрамова	Милослава	Михайловна
24	669645	Герасимова	Селена	Михайловна
25	264241	Кулагина	Иоланта	Федосеевна
26	894006	Чернов	Ермолай	Борисович
27	985251	Алексеев	Архип	Наумович
28	788347	Карпова	Глория	Витальевна
29	894007	Титова	Богдана	Мэлоровна
30	536284	Шестаков	Роман	Лукьевич
31	678309	Соболев	Пантелей	Наумович
32	788348	Шарапов	Арнольд	Куприянович
33	604509	Куликов	Артем	Кириллович
34	734627	Лобанов	Аскольд	Витальевич
35	593773	Сергеев	Кирилл	Александрович
36	734626	Карпова	Светлана	Авксентьевна
37	602823	Ефремов	Нинель	Борисович
38	494416	Горшкова	Марьяна	Даниловна
39	164934	Рожкова	Мэри	Рубеновна
40	604512	Шестаков	Игнатий	Эльдарович
41	264240	Рогов	Мечеслав	Данилович
42	602821	Игнатова	Анжелика	Иосифовна
43	823779	Казакова	Эмилия	Максовна
44	894009	Устинов	Семен	Аркадьевич
45	689101	Гуляева	Дина	Владиславовна
46	894008	Кондратьев	Филипп	Яковлевич
47	536282	Аксёнова	Триана	Геннадьевна
48	182991	Королёв	Давид	Максович
49	264242	Фокин	Юлий	Антонинович
50	182994	Щербakov	Владислав	Тарасович
51	164933	Самойлов	Флор	Лукьевич
52	678308	Носова	Кармелитта	Рудольфовна
53	182993	Гаврилова	Глория	Оскаровна
54	491382	Молчанов	Май	Игоревич
55	689102	Шубина	Санда	Лукьяновна
56	602822	Третьякова	Эльвира	Игнатьевна
57	491384	Владимирова	Илона	Васильевна
58	604511	Евсеева	Роксалана	Степановна
59	793978	Рябова	Артемиды	Куприяновна
60	491383	Моисеева	Евдокия	Пантелеймоновна
61	440851	Суханов	Рубен	Альбертович

7. Вывести список студентов, получивших максимальный средний балл в своей группе.

```
WITH StudentAverage AS (  
  SELECT  
    "Session".student.grade_book_id,  
    ROUND(AVG(CASE "Session".attestation.mark  
      WHEN 'A' THEN 5  
      WHEN 'B' THEN 4  
      WHEN 'C' THEN 4  
      WHEN 'D' THEN 3  
      WHEN 'E' THEN 3  
      WHEN 'F' THEN 2  
    END), 2) AS average_mark  
  FROM  
    "Session".student  
    JOIN "Session"."studying student" ON "Session".student.grade_book_id = "Session"."studying  
student".grade_book_id  
    JOIN "Session".attestation ON "Session"."studying student".studying_student_id =  
"Session".attestation.studying_student_id  
  WHERE  
    (  
      (  
        EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 5 AND 11  
        AND (  
          (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 5 AND EXTRACT(YEAR  
FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))  
          OR  
          (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 6 AND EXTRACT(YEAR  
FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))  
        )  
      ) OR (  
        EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 1 AND 4  
        AND (  
          (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR  
FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))  
          OR  
          (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR  
FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) - 1)  
        )  
      ) OR (  
        EXTRACT(MONTH FROM CURRENT_DATE) = 12  
        AND (  
          (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR  
FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) + 1)  
          OR  
          (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR  
FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))  
        )  
      )  
    )  
  GROUP BY  
    "Session".student.grade_book_id  
,  
MaxAveragePerGroup AS (  
  SELECT  
    "Session"."studying student".group_id,  
    MAX(StudentAverage.average_mark) AS max_average_mark  
  FROM  
    StudentAverage  
    JOIN "Session"."studying student" ON StudentAverage.grade_book_id = "Session"."studying  
student".grade_book_id  
  GROUP BY  
    "Session"."studying student".group_id  
)  
SELECT  
  "Session".student.grade_book_id,  
  "Session".student.student_surname,  
  "Session".student.student_name,  
  "Session".student.student_patronymic,  
  "Session"."studying student".group_id,
```

```

StudentAverage.average_mark
FROM
StudentAverage
JOIN "Session".student ON StudentAverage.grade_book_id = "Session".student.grade_book_id
JOIN "Session"."studying student" ON StudentAverage.grade_book_id = "Session"."studying
student".grade_book_id
JOIN MaxAveragePerGroup ON "Session"."studying student".group_id = MaxAveragePerGroup.group_id AND
StudentAverage.average_mark = MaxAveragePerGroup.max_average_mark;

```

	grade_book_id integer	student_surname character varying (20)	student_name character varying (20)	student_patronymic character varying (20)	group_id integer	average_mark numeric
1	164933	Самойлов	Флор	Лукьевич	218	3.00
2	264241	Кулагина	Иоланта	Федосеевна	169	5.00
3	440851	Суханов	Рубен	Альбертович	208	5.00
4	491384	Владимирова	Илона	Васильевна	19	4.50
5	536282	Аксёнова	Триана	Геннадьевна	22	5.00
6	593770	Потапов	Герасим	Леонидович	93	2.50
7	602823	Ефремов	Нинель	Борисович	164	5.00
8	604509	Куликов	Артем	Кириллович	132	5.00
9	669646	Панфилова	Владислава	Геласьевна	277	4.00
10	678309	Соболев	Пантелей	Наумович	103	4.50
11	689100	Стрелков	Святослав	Семёнович	176	5.00
12	706938	Сазонов	Георгий	Степанович	73	5.00
13	706939	Кудряшова	Юнона	Протасьевна	211	3.50
14	706940	Абрамова	Милослава	Михайловна	203	5.00
15	706941	Фролов	Парамон	Валерьевич	117	5.00
16	788349	Евсеев	Петр	Витальевич	231	5.00
17	793979	Данилов	Василий	Русланович	277	4.00
18	823779	Казакова	Эмилия	Максовна	117	5.00
19	894007	Титова	Богдана	Мэлоровна	119	5.00
20	894008	Кондратьев	Филипп	Яковлевич	185	3.50

Представления

1. Список студентов, получивших двойки на первой попытке с указанием фамилии преподавателя, которым они должны пересдать экзамен;

```
CREATE OR REPLACE VIEW "Session".students_with_failed_attempts AS
SELECT
    "Session".student.student_surname,
    "Session".attestation.mark,
    "Session".professor.professor_surname
FROM
    "Session".attestation
JOIN "Session"."studying student" ON "Session".attestation.studying_student_id =
"Session"."studying student".studying_student_id
JOIN "Session".student ON "Session"."studying student".grade_book_id =
"Session".student.grade_book_id
JOIN "Session".professor ON "Session".attestation.professor_id = "Session".professor.professor_id
WHERE
    "Session".attestation.attempt = 1
    AND "Session".attestation.mark = 'F'
    AND (
        (
            EXTRACT(MONTH FROM CURRENT_DATE) = 12
            AND (
                (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) + 1)
                OR
                (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
            )
        )
        OR (
            EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 1 AND 4
            AND (
                (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) - 1)
                OR
                (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
            )
        )
        OR (
            EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 5 AND 11
            AND EXTRACT(MONTH FROM "Session".attestation.attestation_date) BETWEEN 5 AND 6
            AND EXTRACT(YEAR FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM
CURRENT_DATE)
        )
    );
```

1 SELECT * FROM "Session".students_with_failed_attempts							
Data Output Messages Notifications							
	grade_book_id integer	student_surname character varying (20)	student_name character varying (20)	student_patronymic character varying (20)	mark character	attempt integer	professor_surname character varying
1	494416	Горшкова	Марьяна	Данииловна	F	1	Иванов
2	604511	Евсеева	Роксалана	Степановна	F	1	Зорина
3	491385	Куликов	Ипполит	Георгьевич	F	1	Яналова
4	494417	Одинцов	Дмитрий	Валерьянович	F	1	Носик
5	593773	Сергеев	Кирилл	Александрович	F	1	Семёнов
6	793980	Степанов	Аввакуум	Ростиславович	F	1	Агатьева

2. Данных о студентах при получении ими хотя бы одной оценки 2 (после 3-й попытки).

```
CREATE OR REPLACE VIEW "Session"."students_with_low_grades" AS
SELECT
    "Session".student.grade_book_id,
    "Session".student.student_surname,
    "Session".student.student_name,
    "Session".student.student_patronymic,
    "Session".attestation.mark,
    "Session".attestation.attempt
FROM
    "Session".student
    JOIN "Session"."studying student" ON "Session".student.grade_book_id = "Session"."studying
student".grade_book_id
    JOIN "Session".attestation ON "Session"."studying student".studying_student_id =
"Session".attestation.studying_student_id
WHERE
    "Session".attestation.mark = 'F'
    AND "Session".attestation.attempt = 3
    AND (
        (
            EXTRACT(MONTH FROM CURRENT_DATE) = 12
            AND (
                (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) + 1)
                OR
                (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
            )
        )
        OR (
            EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 1 AND 4
            AND (
                (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) - 1)
                OR
                (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR FROM
"Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
            )
        )
        OR (
            EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 5 AND 11
            AND EXTRACT(MONTH FROM "Session".attestation.attestation_date) BETWEEN 5 AND 6
            AND EXTRACT(YEAR FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM
CURRENT_DATE)
        )
    );
```

1

SELECT * FROM "Session".students_with_low_grades

2

Data Output

Messages

Notifications

	grade_book_id integer	course integer	student_surname character varying (20)	student_name character varying (20)	student_patronymic character varying (20)	mark character	attempt integer
1	182991	5	Королёв	Давид	Максович	F	3
2	264239	3	Морозов	Игорь	Онисимович	F	3
3	985251	4	Алексеев	Архип	Наумович	F	3

Запросы на модификацию данных

1. INSERT: Добавление новой дисциплины "Методы борьбы с ошибками в коде".

```
INSERT INTO
  "Session".discipline (discipline_id, discipline_name, total_number_of_hours,
number_of_laboratory_hours, number_of_lecture_hours, number_of_practical_hours, form_of_attestation,
form_of_work)
VALUES
  ((SELECT COALESCE(MAX(discipline_id), 0) + 1 FROM "Session".discipline), 'Методы борьбы с ошибками
в коде', 404, 200, 100, 100, 'Зачёт', 'Заочно');
```

20	999	Линейная алгебра	200	64	70	64	Экзамен	Очно-заоч...
21	1000	Методы борьбы с ошибками в коде	404	200	100	100	Зачёт	Заочно

2. UPDATE: Повышение одного счастливого младшего преподавателя, который провёл больше всех аттестаций.

```
UPDATE "Session".professor
SET professor_position = 'Преподаватель'
WHERE professor_id = (
  SELECT att.professor_id
  FROM "Session".attestation
  JOIN "Session".professor ON "Session".attestation.professor_id =
"Session".professor.professor_id
  WHERE "Session".professor.professor_position = 'Младший преподаватель'
  AND (
    (
      EXTRACT(MONTH FROM CURRENT_DATE) = 12
      AND (
        (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR
FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) + 1)
        OR
        (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR
FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
      )
    )
    OR (
      EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 1 AND 4
      AND (
        (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR
FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) - 1)
        OR
        (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR
FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
      )
    )
    OR (
      EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 5 AND 11
      AND EXTRACT(MONTH FROM "Session".attestation.attestation_date) BETWEEN 5 AND 6
      AND EXTRACT(YEAR FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM
CURRENT_DATE)
    )
  )
  GROUP BY "Session".attestation.professor_id
  ORDER BY COUNT("Session".attestation.attestation_id) DESC
  LIMIT 1
);
```

2	60	Михаил	Буданцев	Сергеевич	Преподаватель	253
3	132	Алина	Носик	Викторовна	Младший преподаватель	316
4	146	Алексей	Громов	Алексеевич	Преподаватель	634
3	132	Алина	Носик	Викторовна	Преподаватель	316
4	146	Алексей	Громов	Алексеевич	Преподаватель	634

3. DELETE: Увольнение всех сотрудников, которые не участвовали в принятии студентов на аттестации

```
DELETE FROM "Session".professor
WHERE professor_id NOT IN (
    SELECT DISTINCT "Session"."session schedule".professor_id
    FROM "Session"."session schedule"
    WHERE "Session"."session schedule".professor_id IN (
        SELECT DISTINCT "Session".attestation.professor_id
        FROM "Session".attestation
        WHERE (
            (
                EXTRACT(MONTH FROM CURRENT_DATE) = 12
                AND (
                    (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR
FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) + 1)
                    OR
                    (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR
FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
                )
            )
            OR (
                EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 1 AND 4
                AND (
                    (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 12 AND EXTRACT(YEAR
FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE) - 1)
                    OR
                    (EXTRACT(MONTH FROM "Session".attestation.attestation_date) = 1 AND EXTRACT(YEAR
FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM CURRENT_DATE))
                )
            )
            OR (
                EXTRACT(MONTH FROM CURRENT_DATE) BETWEEN 5 AND 11
                AND EXTRACT(MONTH FROM "Session".attestation.attestation_date) BETWEEN 5 AND 6
                AND EXTRACT(YEAR FROM "Session".attestation.attestation_date) = EXTRACT(YEAR FROM
CURRENT_DATE)
            )
        )
    )
);
```

	professor_id [PK] integer	professor_name character varying	professor_surname character varying	professor_patronymic character varying	professor_position character varying	department_id integer
1	36	Анжела	Петрова	Юрьевна	Старший преподаватель	898
2	60	Михаил	Буданцев	Сергеевич	Преподаватель	253
3	132	Алина	Носик	Викторовна	Преподаватель	316
4	146	Алексей	Громов	Алексеевич	Преподаватель	634
5	155	Алла	Храбова	Сергеевна	Преподаватель	762
6	333	Елена	Григорьева	Андреева	Преподаватель	320
7	372	Александр	Пушкин	Сергеевич	Преподаватель	284
8	458	Иосиф	Белых	Иванович	Преподаватель	864
9	477	Эмилия	Зорина	Родионовна	Младший преподаватель	533
10	556	Лидия	Петрова	Геннадьевна	Старший преподаватель	616
11	650	Арина	Агатьева	Юрьевна	Старший преподаватель	661
12	727	Максим	Бушманов	Викторович	Старший преподаватель	675
13	836	Николай	Дрозд	Ильич	Преподаватель	115
14	870	Иван	Иванов	Иванович	Старший преподаватель	456
15	872	Анна	Сидорова	Борисовна	Старший преподаватель	884
16	936	Арина	Яналова	Виталиевна	Преподаватель	582
17	949	Мария	Трофимова	Павловна	Младший преподаватель	736
18	965	Павел	Врубель	Станиславович	Преподаватель	86
19	976	Лев	Толстой	Николаевич	Преподаватель	482
20	989	Константин	Семёнов	Витальевич	Преподаватель	477

	professor_id [PK] integer	professor_name character varying	professor_surname character varying	professor_patronymic character varying	professor_position character varying	department_id integer
1	36	Анжела	Петрова	Юрьевна	Старший преподаватель	898
2	60	Михаил	Буданцев	Сергеевич	Преподаватель	253
3	132	Алина	Носик	Викторовна	Преподаватель	316
4	155	Алла	Храбова	Сергеевна	Преподаватель	762
5	458	Иосиф	Белых	Иванович	Преподаватель	864
6	477	Эмилия	Зорина	Родионовна	Младший преподаватель	533
7	556	Лидия	Петрова	Геннадьевна	Старший преподаватель	616
8	650	Арина	Агатьева	Юрьевна	Старший преподаватель	661
9	727	Максим	Бушманов	Викторович	Старший преподаватель	675
10	836	Николай	Дрозд	Ильич	Преподаватель	115
11	870	Иван	Иванов	Иванович	Старший преподаватель	456
12	872	Анна	Сидорова	Борисовна	Старший преподаватель	884
13	936	Арина	Яналова	Виталиевна	Преподаватель	582
14	949	Мария	Трофимова	Павловна	Младший преподаватель	736
15	965	Павел	Врубель	Станиславович	Преподаватель	86
16	976	Лев	Толстой	Николаевич	Преподаватель	482
17	989	Константин	Семёнов	Витальевич	Преподаватель	477

Создание индексов

```
EXPLAIN ANALYZE
SELECT * FROM "Session".attestation
WHERE professor_id = 132;
```

	QUERY PLAN	
	text	🔒
1	Seq Scan on attestation (cost=0.00..6.51 rows=16 width=26) (actual time=0.104..0.147 rows=15 loops...	
2	Filter: (professor_id = 132)	
3	Rows Removed by Filter: 266	
4	Planning Time: 0.318 ms	
5	Execution Time: 0.169 ms	
Total rows: 5 of 5 Query complete 00:00:00.253		

```
CREATE INDEX professor_id_index ON "Session".attestation (professor_id);
EXPLAIN ANALYZE
SELECT * FROM "Session".attestation
WHERE professor_id = 132;
```

	QUERY PLAN	
	text	🔒
1	Seq Scan on attestation (cost=0.00..6.51 rows=16 width=26) (actual time=0.044..0.060 rows=15 loops...	
2	Filter: (professor_id = 132)	
3	Rows Removed by Filter: 266	
4	Planning Time: 1.128 ms	
5	Execution Time: 0.090 ms	
Total rows: 5 of 5 Query complete 00:00:00.123		

```
EXPLAIN ANALYZE
SELECT * FROM "Session".attestation
WHERE professor_id = 132 OR attempt = 1;
```

	QUERY PLAN	
	text	🔒
1	Seq Scan on attestation (cost=0.00..7.21 rows=130 width=26) (actual time=0.065..0.123 rows=144 loops...	
2	Filter: ((professor_id = 132) OR (attempt = 1))	
3	Rows Removed by Filter: 137	
4	Planning Time: 0.272 ms	
5	Execution Time: 0.142 ms	
Total rows: 5 of 5 Query complete 00:00:00.206		

```
CREATE INDEX professor_id_attempt_index ON "Session".attestation
(professor_id, attempt);
EXPLAIN ANALYZE
SELECT * FROM "Session".attestation
WHERE professor_id = 132 OR attempt = 1;
```

	QUERY PLAN	
	text	🔒
1	Seq Scan on attestation (cost=0.00..7.21 rows=130 width=26) (actual time=0.066..0.099 rows=144 loops...	
2	Filter: ((professor_id = 132) OR (attempt = 1))	
3	Rows Removed by Filter: 137	
4	Planning Time: 1.918 ms	
5	Execution Time: 0.123 ms	
Total rows: 5 of 5 Query complete 00:00:00.127		

Вывод

В ходе лабораторной работы удалось овладеть практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL, использования подзапросов при модификации данных и индексов. Также при выполнении запросов без индексов и с индексами, удалось выяснить, что с использованием индексов схема сработала на ~55% быстрее.