

"НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО"

Факультет «Инфокоммуникационных технологий»
Направление подготовки «09.03.03 Прикладная информатика»
Бакалаврская программа «Мобильные и сетевые технологии»

ОТЧЕТ
по лабораторной работе №4

по дисциплине «Проектирование и реализация баз данных»

Тема: «Запросы на выборку данных к БД PostgreSQL.
Представления в PostgreSQL»

Выполнил:

_____ / Шугинин Ю. А.
(гр. К3241)

Проверил:

_____ / Говорова М. М.

Дата: 05.06.2021

Санкт-Петербург
2021

Цель работы: овладение практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL и использования подзапросов при модификации данных.

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.
4. Просмотреть историю запросов.

Индивидуальное задание. Вариант 7. БД «Курсы»

Описание предметной области: Подразделение занимается организацией внебюджетного образования. Имеется несколько типов краткосрочных курсов, предназначенных для определенных специальностей, связанных с программным обеспечением ИТ. Каждый тип курсов имеет определенную длительность и свой перечень изучаемых дисциплин. На каждую программу может быть набрано несколько групп. По каждой дисциплине могут проводиться лекционные и лабораторные занятия. Подразделение обеспечивает следующие ресурсы: учебные классы, лекционные аудитории и преподавателей. Необходимо составить расписание занятий.

БД должна содержать следующий минимальный набор сведений: Фамилия слушателя. Имя слушателя. Паспортные данные. Контакты. Код программы. Программа. Тип программы. Объем часов. Номер группы. Количество человек в группе. Дата начала обучения. Дата окончания обучения. Название дисциплины. Количество часов. Дата занятий. Номер пары. Номер аудитории. Тип аудитории. Адрес площадки. Вид занятий (лекционные или практические). Фамилия преподавателя. Имя и отчество преподавателя. Должность преподавателя.

Выполнение:

Наименование БД: Courses

Схема логической модели базы данных:

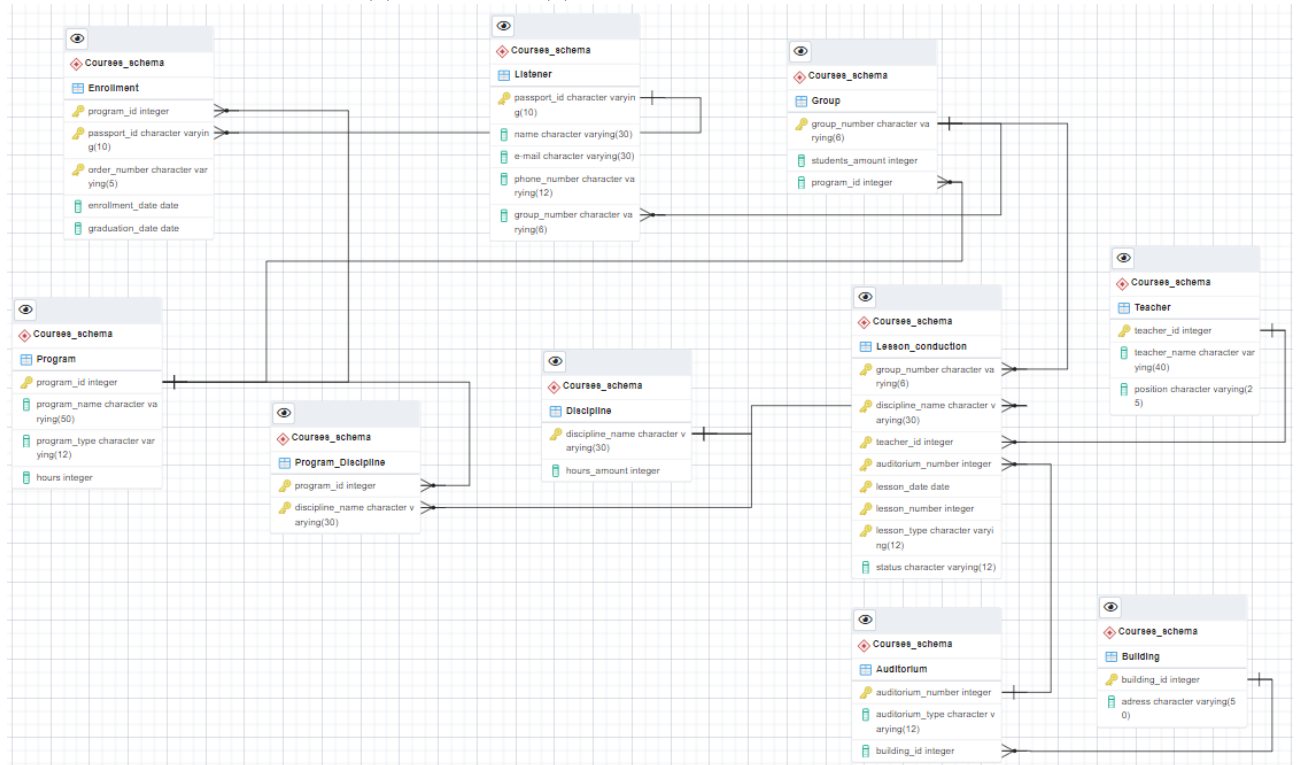


Рисунок 1 – Схема логической модели БД «Courses»

Запросы к базе данных:

1) Вывести все номера групп и программы, где количество слушателей меньше 10.

```
SELECT group_number, students_amount, program_name
FROM "Courses_schema"."Group", "Courses_schema"."Program"
WHERE "Group".program_id = "Program".program_id
AND "Group".students_amount < 10;
```

	group_number character varying (6)	students_amount integer	program_name character varying (50)
1	P3112	9	Program Engineering
2	U3411	8	Business Analytics

2) Вывести список преподавателей с указанием количества программ, где они преподавали за истекший год.

```
SELECT teacher_name, COUNT(program_id) AS programs_amount
FROM "Courses_schema"."Lesson_conduction"
NATURAL JOIN "Courses_schema"."Group"
NATURAL JOIN "Courses_schema"."Teacher"
WHERE lesson_date > '2020-09-01' AND lesson_date < '2021-06-09'
GROUP BY teacher_name;
```

	teacher_name character varying (40)	programs_amount bigint
1	Говорова Марина Михайловна	3
2	Сорокин Дмитрий Сергеевич	2
3	Хорошкева Ксения Александровна	1

3) Вывести список преподавателей, которые не проводят занятия на третьей паре ни в один из дней недели.

```
SELECT DISTINCT teacher_name FROM "Courses_schema"."Lesson_conduction"
    NATURAL JOIN "Courses_schema"."Teacher"
WHERE teacher_id NOT IN (SELECT teacher_id FROM
"Courses_schema"."Lesson_conduction" WHERE lesson_number = '3');
```

	teacher_name character varying (40)
1	Сорокин Дмитрий Сергеевич
2	Говорова Марина Михайловна

4) Вывести список свободных лекционных аудиторий на ближайший понедельник.

```
SELECT auditorium_number, adress
FROM "Courses_schema"."Auditorium"
    NATURAL JOIN "Courses_schema"."Building"
WHERE auditorium_number NOT IN (SELECT auditorium_number FROM
"Courses_schema"."Lesson_conduction"
    WHERE lesson_date >= CURRENT_DATE
    AND EXTRACT(DOW FROM lesson_date) = '1'
    AND lesson_date < (CURRENT_DATE + 7))
AND auditorium_type = 'Лекционная';
```

	auditorium_number integer	adress character varying (50)
1	1124	С-Пб, ул. Ломоносова 9

5) Вычислить общее количество обучающихся по каждой программе за последний год.

```
SELECT program_name, COUNT(*) AS students_total_amount
FROM "Courses_schema"."Program"
    NATURAL JOIN "Courses_schema"."Enrollment"
WHERE "Enrollment".enrollment_date < '2020-09-01'
    AND "Enrollment".graduation_date > '2021-06-08'
GROUP BY program_name
ORDER BY program_name;
```

	program_name character varying (50)	students_total_amount bigint
1	Applied Informatics	3
2	Business Analytics	2
3	Program Engineering	2

6) Вычислить среднюю загруженность компьютерных классов в неделю за последний месяц (в часах).

```
SELECT auditorium_number, adress, ROUND(COUNT(*)*1.5/4, 3) AS
average_workload
FROM "Courses_schema"."Lesson_conduction"
    NATURAL JOIN "Courses_schema"."Auditorium"
    NATURAL JOIN "Courses_schema"."Building"
WHERE lesson_date <= CURRENT_DATE
    AND lesson_date > (CURRENT_DATE - 31)
    AND auditorium_type = 'Компьютерный класс'
GROUP BY auditorium_number, adress;
```

	auditorium_number integer	adress character varying (50)	average_workload numeric
1	205	С-Пб, ул. Биржевая 14	1.125

7) Найти самые популярные программы за последние 3 года.

```
SELECT program_name, COUNT(*) AS total_popularity
FROM "Courses_schema"."Program"
    NATURAL JOIN "Courses_schema"."Enrollment"
WHERE enrollment_date < CURRENT_DATE
    AND graduation_date > '2018-09-01'
GROUP BY program_name
ORDER BY total_popularity DESC, program_name;
```

	program_name character varying (50)	total_popularity bigint
1	Applied Informatics	3
2	Business Analytics	2
3	Program Engineering	2

Представления:

1) Создать представление для потенциальных слушателей, содержащее перечень специальностей, изучаемых на них дисциплин и количество часов;

```
CREATE VIEW view_for_applicants AS
SELECT program_name, discipline_name AS disciplines, hours_amount
FROM "Courses_schema"."Program"
    NATURAL JOIN "Courses_schema"."Program_Discipline"
    NATURAL JOIN "Courses_schema"."Discipline"
ORDER BY program_name, hours_amount DESC;
```

Скриншот выполнения запроса:

Результат	План выполнения	Сообщения	Notifications
CREATE VIEW			
Запрос завершён успешно, время выполнения: 117 мсек.			

Просмотр содержимого представления view_for_applicants:

1	SELECT * FROM view_for_applicants		
---	---	--	--

[Query Editor](#)
[История запросов](#)
[Scratch Pad](#)

[Результат](#)
[План выполнения](#)
[Сообщения](#)
[Notifications](#)

	program_name character varying (50)	disciplines character varying (30)	hours_amount integer
1	Applied Informatics	Programming	144
2	Applied Informatics	Databases	108
3	Applied Informatics	Web design	72
4	Business Analytics	Databases	108
5	Business Analytics	Web design	72
6	Program Engineering	Programming	144
7	Program Engineering	Databases	108

2) Создать представление, содержащее сведения об общем доходе по каждой программе за последний год.

```

CREATE VIEW income_view AS
SELECT program_name, price AS price_per_year,
       COUNT(*) AS students_amount, SUM(price) AS total_income
FROM "Courses_schema"."Program"
     NATURAL JOIN "Courses_schema"."Enrollment"
WHERE "Enrollment".enrollment_date < CURRENT_DATE
     AND "Enrollment".graduation_date > CURRENT_DATE
GROUP BY program_name, price
ORDER BY total_income DESC;

```

Скриншот выполнения запроса:

Результат	План выполнения	Сообщения	Notifications
---------------------------	---------------------------------	---------------------------	-------------------------------

CREATE VIEW

Запрос завершён успешно, время выполнения: 95 msec.

Просмотр содержимого представления income_view:

1	SELECT * FROM income_view		
---	---	--	--

[Query Editor](#)
[История запросов](#)
[Scratch Pad](#)

[Результат](#)
[План выполнения](#)
[Сообщения](#)
[Notifications](#)

	program_name character varying (50)	price_per_year integer	students_amount bigint	total_income bigint
1	Applied Informatics	180000	3	540000
2	Program Engineering	220000	2	440000
3	Business Analytics	150000	2	300000

Запросы на модификацию данных:

- 1) **INSERT:** добавить в расписание практическое занятие по программированию у группы K3241 на 4-й паре 08.06.2021. В качестве аудитории выбрать свободный в это время компьютерный класс.

```
INSERT INTO "Courses_schema"."Lesson_conduction"
(group_number, discipline_name, teacher_id, auditorium_number,
lesson_date, lesson_number, lesson_type, status)
VALUES ('K3241', 'Programming', 4,
(SELECT auditorium_number FROM
"Courses_schema"."Lesson_conduction" NATURAL JOIN
"Courses_schema"."Auditorium"
WHERE auditorium_number NOT IN
(SELECT auditorium_number FROM
"Courses_schema"."Lesson_conduction" WHERE lesson_date = '2021-06-08'
AND lesson_number = 4)
AND auditorium_type = 'Компьютерный класс' LIMIT 1),
'2021-06-08', 4, 'Практика', 'Не проведено');
```

До:

	group_number [PK] character varying (6)	discipline_name [PK] character varying (30)	teacher_id [PK] integer	auditorium_number [PK] integer	lesson_date [PK] date	lesson_number [PK] integer	lesson_type [PK] character varying (12)	status character varying (12)
1	K3241	Databases		5	2021-05-11	5	Практика	Не проведено

После:

	group_number [PK] character varying (6)	discipline_name [PK] character varying (30)	teacher_id [PK] integer	auditorium_number [PK] integer	lesson_date [PK] date	lesson_number [PK] integer	lesson_type [PK] character varying (12)	status character varying (12)
1	K3241	Databases		5	2021-05-11	5	Практика	Не проведено
2	K3241	Programming		4	2021-06-08	4	Практика	Не проведено

- 2) **UPDATE:** известно, что у группы U3412 в мае 2021-го года было отменено 1 занятие. Его проведение компенсировали на третьей паре 8 июня в той же аудитории, где планировалось изначальное занятие. Отразить эти изменения в БД.

```
UPDATE "Courses_schema"."Lesson_conduction"
SET lesson_date = '2021-06-08', lesson_number = '3',
status = 'Проведено'
WHERE group_number = 'U3412'
AND lesson_date BETWEEN '2021-05-01' AND '2021-05-31'
AND status IN (SELECT status FROM
"Courses_schema"."Lesson_conduction" WHERE status = 'Отменено');
```

До:

	group_number [PK] character varying (6)	discipline_name [PK] character varying (30)	teacher_id [PK] integer	auditorium_number [PK] integer	lesson_date [PK] date	lesson_number [PK] integer	lesson_type [PK] character varying (12)	status character varying (12)
1	U3412	Web design		6	2021-05-10	4	Лекция	Отменено

После:

	group_number [PK] character varying (6)	discipline_name [PK] character varying (30)	teacher_id [PK] integer	auditorium_number [PK] integer	lesson_date [PK] date	lesson_number [PK] integer	lesson_type [PK] character varying (12)	status character varying (12)
1	U3412	Web design		6	2021-06-08	3	Лекция	Проведено

3) DELETE: удалить сведения об аудиториях, в которых не проводятся учебные занятия.

```
DELETE FROM "Courses_schema"."Auditorium"  
WHERE auditorium_number NOT IN (SELECT DISTINCT auditorium_number FROM  
"Courses_schema"."Lesson_conduction")
```

До:

	auditorium_number [PK] integer	auditorium_type character varying (18)	building_id integer
1	359	Практическая	1
2	1124	Лекционная	3
3	205	Компьютерный класс	2
4	312	Служебная	4

После:

	auditorium_number [PK] integer	auditorium_type character varying (18)	building_id integer
1	359	Практическая	1
2	1124	Лекционная	3
3	205	Компьютерный класс	2

Выводы: в процессе выполнения лабораторной работы были получены практические навыки создания представлений и запросов на выборку и модификацию данных к базе данных PostgreSQL с помощью инструмента запросов Query Tool в pgAdmin 4.