МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования  
**«Вятский государственный университет»**

Институт математики и информационных систем

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

**«Основы DML-запросов в PostgreSQL»**

Отчёт по лабораторной работе №2 дисциплины

«Базы данных»

Выполнил студент группы ИВТ-31\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Жеребцов К.А./

Проверил преподаватель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Клюкин В. Л./

Киров 2022

**Цели лабораторной работы:**

1) освоить основные варианты DML-запросов в PostgreSQL;  
2) научиться создавать SQL-скрипты для заполнения таблиц данными;   
3) научиться работать с представлениями.

**Задание на лабораторную работу:**   
При выполнении работы следует использовать базу данных, созданную

в лабораторной работе №1. Ниже приведены задания, которые необходимо

выполнить в ходе лабораторной работы:

1) Создать и выполнить SQL-скрипт, который будет заполнять таблицы

данными. Нужно добавить не менее 3-5 строк в каждую таблицу.

2) Создать представления для нескольких таблиц, в которых собираются

данные из самой таблицы и других, на которые она ссылается. Среди

представлений обязательно должно быть представление для таблицы,

которая используется для связи «многие-ко-многим». Выборка из

любого представления должна давать полную и осмысленную

информацию по сущностям. Хотя бы одно из представлений должно

быть сделано с использованием соединений (join) в запросе.

3) Для любой таблицы, содержащей столбец с числовыми данными,

создать представление следующего вида, отражающее информацию по

этому столбцу (в представлении ровно 3 столбца и 4 строки):

Изображение выглядит как стол

Автоматически созданное описание

**Ход работы**

**Скрипт INSERT:**

do language plpgsql $$

declare

-- Переменные, хранящие id новых записей

-- Хранит ID студентов

\_id\_students bigint;

--Хранит ID предмета 1

\_id\_subjects\_1 bigint;

--Хранит ID предмета 2

\_id\_subjects\_2 bigint;

--Хранит ID учителя 1

\_id\_teachers\_1 bigint;

--Хранит ID учителя 2

\_id\_teachers\_2 bigint;

--Хранит ID групп

\_id\_groups\_1 bigint;

\_id\_groups\_2 bigint;

\_id\_groups\_3 bigint;

\_id\_groups\_4 bigint;

\_id\_groups\_5 bigint;

begin

--Добавляем группу

insert into groups(name, number\_of\_students)

values ('ПИб',5)

returning id into \_id\_groups\_1;

insert into groups(name, number\_of\_students)

values ('ПА',5)

returning id into \_id\_groups\_2;

insert into groups(name, number\_of\_students)

values ('ОЛ',5)

returning id into \_id\_groups\_3;

insert into groups(name, number\_of\_students)

values ('ИТ',5)

returning id into \_id\_groups\_4;

insert into groups(name, number\_of\_students)

values ('ИВТ',5)

returning id into \_id\_groups\_5;

--Добавляем предмет 1

insert into subject(name\_subject, hours)

VALUES ('Программирование', 2)

returning id into \_id\_subjects\_1;

--Добавляем предмет 2

insert into subject(name\_subject, hours)

VALUES ('Математика', 1)

returning id into \_id\_subjects\_2;

--Добавляем успеваемость студентов

insert into student(first\_name, second\_name, middle\_name, id\_group, address, cource)

VALUES ('Макарова', 'Диана', 'Евгеньевна', \_id\_groups\_1, 'Киров, ул. Московская, д. 36', 2)

returning id into \_id\_students;

insert into progress (id\_student, id\_subject, mark)

values (\_id\_students, \_id\_subjects\_1, 4);

insert into progress (id\_student, id\_subject, mark)

values (\_id\_students, \_id\_subjects\_2, 5);

insert into student(first\_name, second\_name, middle\_name, id\_group, address, cource)

VALUES ('Двинянинова', 'Валерия', 'Андреевна', \_id\_groups\_2, 'Киров, ул. Павла Корчагина, д. 11', 1)

returning id into \_id\_students;

insert into progress (id\_student, id\_subject, mark)

values (\_id\_students, \_id\_subjects\_1, 4);

insert into progress (id\_student, id\_subject, mark)

values (\_id\_students, \_id\_subjects\_2, 5);

insert into student(first\_name, second\_name, middle\_name, id\_group, address, cource)

VALUES ('Юркина', 'Анастасия', 'Сергеевна', \_id\_groups\_3, 'Киров, ул. Ленина, д. 190, к. 1', 3)

returning id into \_id\_students;

insert into progress (id\_student, id\_subject, mark)

values (\_id\_students, \_id\_subjects\_1, 3);

insert into progress (id\_student, id\_subject, mark)

values (\_id\_students, \_id\_subjects\_2, 4);

insert into student(first\_name, second\_name, middle\_name, id\_group, address, cource)

VALUES ('Вологжанина', 'Александра', 'Алексеевна', \_id\_groups\_4, 'Киров, ул. Волкова, д. 10', 4)

returning id into \_id\_students;

insert into progress (id\_student, id\_subject, mark)

values (\_id\_students, \_id\_subjects\_1, 2);

insert into progress (id\_student, id\_subject, mark)

values (\_id\_students, \_id\_subjects\_2, 3);

insert into student(first\_name, second\_name, middle\_name, id\_group, address, cource)

VALUES ('Зудилова', 'Валерия', 'Ивановна', \_id\_groups\_5, 'Киров, ул. Ленина, д. 56', 3)

returning id into \_id\_students;

insert into progress (id\_student, id\_subject, mark)

values (\_id\_students, \_id\_subjects\_1, 4);

insert into progress (id\_student, id\_subject, mark)

values (\_id\_students, \_id\_subjects\_2, 5);

insert into student(first\_name, second\_name, middle\_name, id\_group, address, cource)

VALUES ('Петров', 'Валерий', 'Иванович', \_id\_groups\_2, 'Киров, ул. Ленина, д. 5', 1)

returning id into \_id\_students;

insert into progress (id\_student, id\_subject, mark)

values (\_id\_students, \_id\_subjects\_1, 2);

insert into progress (id\_student, id\_subject, mark)

values (\_id\_students, \_id\_subjects\_2, 4);

insert into student(first\_name, second\_name, middle\_name, id\_group, address, cource)

VALUES ('Зудилов', 'Андрей', 'Сереевич', \_id\_groups\_4, 'Киров, ул. Сормовская, д. 56', 4)

returning id into \_id\_students;

insert into progress (id\_student, id\_subject, mark)

values (\_id\_students, \_id\_subjects\_1, 4);

insert into progress (id\_student, id\_subject, mark)

values (\_id\_students, \_id\_subjects\_2, 4);

--Добавляем учителя и назначаем предмет

insert into teacher(first\_name, second\_name, middle\_name,position,phone\_number)

VALUES ('Петров', 'Алексей', 'Викторович', 'docent', NULL)

returning id into \_id\_teachers\_1;

insert into teacher(first\_name, second\_name, middle\_name,position,phone\_number)

VALUES ('Бузова', 'Наталья', 'Алексеевна', 'professors', NULL)

returning id into \_id\_teachers\_2;

insert into occupation(id\_subject, id\_teacher, position)

values(\_id\_subjects\_1, \_id\_teachers\_1, 'lecture'),

(\_id\_subjects\_2, \_id\_teachers\_2, 'seminar');

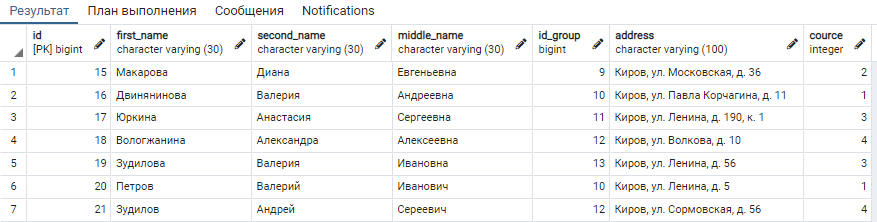
END;

$$;

**Скрипт SELECT:**

SELECT \*

FROM public.student;



**Скрипт DELETE:**

DELETE FROM public.progress

WHERE mark=2;

SELECT \* FROM public.progress;

Изображение выглядит как текст, мебель, внутренний, белый

Автоматически созданное описание Изображение выглядит как стол

Автоматически созданное описание

**Скрипт UPDATE:**

UPDATE public.progress

SET mark=5

WHERE id\_student=21;

Изображение выглядит как стол

Автоматически созданное описание Изображение выглядит как стол

Автоматически созданное описание

**Агрегатная функция:**

create or replace view groups\_v as

(select 'Минимальное значение', number\_of\_students, id from groups

ORDER BY number\_of\_students desc LIMIT 1)

union all

(select 'Максимальное значение', number\_of\_students, id from groups

ORDER BY number\_of\_students LIMIT 1)

union all

select 'Среднее значение', avg(g.number\_of\_students), null from groups g

union all

select 'Сумма значений', sum(number\_of\_students), null from groups;

SELECT \*

FROM public.groups\_v

Изображение выглядит как текст

Автоматически созданное описание

**Вывод:**

В ходе выполнения данной лабораторной работы были освоены основные варианты DML-запросов в PostgreSQL, а также как создавать SQL-скрипты для заполнения таблиц данными и работать с представлениями.