Министерство науки и высшего образования Российской Федерации  
федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

*Факультет программной инженерии и компьютерной техники*

*Направление подготовки: 09.03.04 – Нейротехнологии и программирование*

*Дисциплина «Информатика»*

Лабораторная работа по

Базам данных №2

Вариант 1366

Работу выполнил:

Мухин Никита Денисович

Группа:

P3123

Преподаватель:

Горбунов Михаил Витальевич

Санкт-Петербург,

2024

**Оглавление**

[**Текст задания** 2](#_Toc1)

[**Реализация запросов на SQL** 4](#_Toc2)

[**Вывод**............ 6](#_Toc3)

# **Текст задания**

По варианту, выданному преподавателем, составить и выполнить запросы к [базе данных "Учебный процесс"](https://se.ifmo.ru/documents/10180/733702/%D0%91%D0%94+%D0%A3%D1%87%D0%B5%D0%B1%D0%BD%D1%8B%D0%B9+%D0%9F%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81.pdf/2eae3fcd-ea34-4496-924b-6ee4e889a9e5).

Команда для подключения к базе данных ucheb:

*psql -h pg -d ucheb*

Отчёт по лабораторной работе должен содержать:

1. Текст задания.
2. Реализацию запросов на SQL.
3. Выводы по работе.

Темы для подготовки к защите лабораторной работы:

1. SQL
2. Соединение таблиц
3. Подзапросы
4. Представления
5. Последовательности

Составить запросы на языке SQL (пункты 1-7).

1. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:  
   Н\_ТИПЫ\_ВЕДОМОСТЕЙ, Н\_ВЕДОМОСТИ.  
    Вывести атрибуты: Н\_ТИПЫ\_ВЕДОМОСТЕЙ.ИД, Н\_ВЕДОМОСТИ.ЧЛВК\_ИД.  
    Фильтры (AND):   
    a) Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ > Экзаменационный лист.  
    b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 163249.  
    c) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД > 153285.  
    Вид соединения: LEFT JOIN.
2. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:  
   Таблицы: Н\_ЛЮДИ, Н\_ВЕДОМОСТИ, Н\_СЕССИЯ.  
    Вывести атрибуты: Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ВЕДОМОСТИ.ИД, Н\_СЕССИЯ.ЧЛВК\_ИД.  
    Фильтры (AND):   
    a) Н\_ЛЮДИ.ИМЯ = Владимир.  
    b) Н\_ВЕДОМОСТИ.ДАТА < 1998-01-05.  
    Вид соединения: RIGHT JOIN.
3. Вывести число названий дисциплин без учета повторений.  
   При составлении запроса нельзя использовать DISTINCT.
4. Выдать различные имена студентов и число людей с каждой из этих имен, ограничив список именами, встречающимися менее 50 раз на на заочной форме обучения.  
   Для реализации использовать подзапрос.
5. Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср\_оценка), у которых средняя оценка не больше минимальной оценк(е|и) в группе 1100.
6. Получить список студентов, зачисленных ровно первого сентября 2012 года на первый курс очной или заочной формы обучения. В результат включить:  
   номер группы;  
   номер, фамилию, имя и отчество студента;  
   номер и состояние пункта приказа;  
   Для реализации использовать подзапрос с IN.
7. Сформировать запрос для получения числа в СПбГУ ИТМО хорошистов.

# **Реализация запросов на SQL**

-- Задание №1:  
SELECT 'Запрос для получения атрибутов из указанных таблиц с LEFT JOIN:';  
SELECT Н\_ТИПЫ\_ВЕДОМОСТЕЙ.ИД, Н\_ВЕДОМОСТИ.ЧЛВК\_ИД  
FROM Н\_ТИПЫ\_ВЕДОМОСТЕЙ  
LEFT JOIN Н\_ВЕДОМОСТИ ON Н\_ТИПЫ\_ВЕДОМОСТЕЙ.ИД = Н\_ВЕДОМОСТИ.ТВ\_ИД  
WHERE Н\_ТИПЫ\_ВЕДОМОСТЕЙ.ИД > (SELECT ИД FROM Н\_ТИПЫ\_ВЕДОМОСТЕЙ WHERE НАИМЕНОВАНИЕ = 'Экзаменационный лист')  
AND Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 163249  
AND Н\_ВЕДОМОСТИ.ЧЛВК\_ИД > 153285;  
  
  
-- Задание №2:  
SELECT 'Запрос для получения атрибутов из указанных таблиц с RIGHT JOIN:';  
SELECT Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ВЕДОМОСТИ.ИД, Н\_СЕССИЯ.ЧЛВК\_ИД  
FROM Н\_ЛЮДИ  
RIGHT JOIN Н\_ВЕДОМОСТИ ON Н\_ЛЮДИ.ИД = Н\_ВЕДОМОСТИ.ЧЛВК\_ИД  
RIGHT JOIN Н\_СЕССИЯ ON Н\_ВЕДОМОСТИ.СЭС\_ИД = Н\_СЕССИЯ.ИД  
WHERE Н\_ЛЮДИ.ИМЯ = 'Владимир'  
AND DATE(Н\_ВЕДОМОСТИ.ДАТА) < '1998-01-05';  
  
SELECT 'Нету вывода т.к это минимальная дата:';  
SELECT MIN(DATE(Н\_ВЕДОМОСТИ.ДАТА)) AS "Минимальная дата начала"  
FROM "Н\_ВЕДОМОСТИ";  
  
  
-- Задание №3:  
SELECT 'Запрос для вычисления числа названий дисциплин без учета повторений:';  
SELECT COUNT(НАИМЕНОВАНИЕ) AS Число\_Дисциплин  
FROM (  
 SELECT НАИМЕНОВАНИЕ  
 FROM Н\_ДИСЦИПЛИНЫ  
 GROUP BY НАИМЕНОВАНИЕ  
) AS subquery;  
  
  
-- Задание №4:  
SELECT 'Запрос для вывода различных имен студентов и числа людей с каждым именем, у которых встречающихся менее 50 раз на заочной форме обучения:';  
SELECT Н\_ЛЮДИ.ИМЯ,  
 COUNT(Н\_ЛЮДИ.ИМЯ) AS Число\_Студентов  
FROM Н\_УЧЕНИКИ  
JOIN Н\_ЛЮДИ ON Н\_ЛЮДИ.ИД = Н\_УЧЕНИКИ.ЧЛВК\_ИД  
JOIN Н\_ПЛАНЫ ON Н\_УЧЕНИКИ.ПЛАН\_ИД = Н\_ПЛАНЫ.ИД  
JOIN Н\_ФОРМЫ\_ОБУЧЕНИЯ ON Н\_ПЛАНЫ.ФО\_ИД = Н\_ФОРМЫ\_ОБУЧЕНИЯ.ИД  
WHERE Н\_ФОРМЫ\_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ = 'Заочная'  
GROUP BY Н\_ЛЮДИ.ИМЯ  
HAVING COUNT(Н\_ЛЮДИ.ИМЯ) < 50;  
  
SELECT 'Вывод отстутсвует по причине отсутствия людей на заочной форме обучения:';  
SELECT \* FROM Н\_ПЛАНЫ WHERE ФО\_ИД = 3;  
  
  
-- Задание №5:  
SELECT 'Запрос для получения таблицы со средними оценками студентов группы 4100, у которых средняя оценка не больше минимальной оценки в группе 3100';  
SELECT Н\_ВЕДОМОСТИ.ЧЛВК\_ИД, AVG(CASE WHEN (ОЦЕНКА <> 'зачет' AND ОЦЕНКА <> 'осв' AND ОЦЕНКА <> 'незач' AND ОЦЕНКА <> 'неявка') THEN CAST(ОЦЕНКА AS INTEGER) END) AS СР\_ОЦЕНКА, Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ЛЮДИ.ИМЯ, Н\_ЛЮДИ.ОТЧЕСТВО  
FROM Н\_ВЕДОМОСТИ  
JOIN Н\_УЧЕНИКИ ON Н\_УЧЕНИКИ.ЧЛВК\_ИД = Н\_ВЕДОМОСТИ.ЧЛВК\_ИД  
JOIN Н\_ЛЮДИ ON Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = Н\_ЛЮДИ.ИД  
WHERE ГРУППА = '4100'  
GROUP BY Н\_ВЕДОМОСТИ.ЧЛВК\_ИД, ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО  
HAVING AVG(CASE  
 WHEN (ОЦЕНКА <> 'зачет'  
 AND ОЦЕНКА <> 'осв'  
 AND ОЦЕНКА <> 'незач'  
 AND ОЦЕНКА <> 'неявка')  
 THEN CAST(ОЦЕНКА AS INTEGER)  
 END) >= (  
 SELECT min(CASE WHEN (ОЦЕНКА <> 'зачет' AND ОЦЕНКА <> 'осв' AND ОЦЕНКА <> 'незач' AND ОЦЕНКА <> 'неявка') THEN CAST(ОЦЕНКА AS INTEGER) END)  
 FROM Н\_ВЕДОМОСТИ  
 JOIN Н\_УЧЕНИКИ on Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = Н\_УЧЕНИКИ.ЧЛВК\_ИД  
 WHERE Н\_УЧЕНИКИ.ГРУППА = '3100'  
);  
  
  
-- Задание №6:  
SELECT 'Запрос для получения списка студентов, зачисленных ровно первого сентября 2012 года на первый курс очной или заочной формы обучения';  
SELECT Н\_УЧЕНИКИ.ГРУППА,  
 Н\_УЧЕНИКИ.ИД,  
 Н\_ЛЮДИ.ФАМИЛИЯ,  
 Н\_ЛЮДИ.ИМЯ,  
 Н\_ЛЮДИ.ОТЧЕСТВО,  
 Н\_УЧЕНИКИ.СОСТОЯНИЕ  
FROM Н\_УЧЕНИКИ  
JOIN Н\_ЛЮДИ ON Н\_ЛЮДИ.ИД = Н\_УЧЕНИКИ.ЧЛВК\_ИД  
JOIN Н\_ПЛАНЫ ON Н\_УЧЕНИКИ.ПЛАН\_ИД = Н\_ПЛАНЫ.ИД  
JOIN Н\_ФОРМЫ\_ОБУЧЕНИЯ ON Н\_ПЛАНЫ.ФО\_ИД = Н\_ФОРМЫ\_ОБУЧЕНИЯ.ИД  
 AND (Н\_ФОРМЫ\_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ = 'Очная'  
 OR Н\_ФОРМЫ\_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ = 'Заочная')  
WHERE Н\_УЧЕНИКИ.ПРИЗНАК = 'обучен'  
 AND Н\_УЧЕНИКИ.СОСТОЯНИЕ = 'утвержден'  
AND DATE(Н\_УЧЕНИКИ.НАЧАЛО) = '2012-09-01';  
  
SELECT 'Нету вывода т.к нету такой даты начала обучения:';  
SELECT MAX(DATE("Н\_УЧЕНИКИ"."НАЧАЛО")) AS "Максимальная дата начала"  
FROM "Н\_УЧЕНИКИ";  
  
  
-- Задание №7:  
SELECT 'Запрос для получения числа хорошистов в СПбГУ ИТМО:';  
SELECT COUNT(DISTINCT Н\_ВЕДОМОСТИ.ЧЛВК\_ИД) AS "Количество отличников"  
FROM Н\_ВЕДОМОСТИ  
JOIN Н\_ОЦЕНКИ ON Н\_ВЕДОМОСТИ.ОЦЕНКА = Н\_ОЦЕНКИ.КОД  
JOIN Н\_УЧЕНИКИ ON Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = Н\_УЧЕНИКИ.ЧЛВК\_ИД  
JOIN Н\_ПЛАНЫ ON Н\_УЧЕНИКИ.ПЛАН\_ИД = Н\_ПЛАНЫ.ИД  
JOIN Н\_ОТДЕЛЫ ON Н\_ПЛАНЫ.ОТД\_ИД = Н\_ОТДЕЛЫ.ИД  
WHERE Н\_ОТДЕЛЫ.КОРОТКОЕ\_ИМЯ = 'СПБГУИТМО' AND Н\_ОЦЕНКИ.СОРТ >= 4;  
  
SELECT 'Нет вывода т.к. нет связанных с СПБГУИТМО записей:';  
SELECT Н\_ПЛАНЫ.\*  
FROM Н\_ПЛАНЫ  
JOIN Н\_ОТДЕЛЫ ON Н\_ПЛАНЫ.ОТД\_ИД = Н\_ОТДЕЛЫ.ИД  
WHERE КОРОТКОЕ\_ИМЯ = 'СПбГУИТМО';

## **Вывод**

При выполнении лабораторной работы я познакомился с основными функциями PostgreSQL. Научился писать запросы, получать, агрегировать, отсеивать и сортировать полученные данные с использованием различных синтаксических конструкций языка. В результате был освоен язык DML SQL, предназначенный для работы с данными, хранящимися внутри базы данных.