Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет Программной Инженерии и Компьютерной Техники

**Лабораторная работа по Базам данных №2**

Вариант 7535

Выполни л: Суворов Станислав Денисович

Проверил: Райла Мартин

Группа: Р3115

Санкт Петербург 2025 г

Содержание

[Ход работы 3](#__RefHeading___Toc2108_4178638284)

[Текст задания 3](#__RefHeading___Toc301_1751619282)

[Реализация запросов на SQL 4](#__RefHeading___Toc303_1751619282)

[Задание 1 4](#__RefHeading___Toc305_1751619282)

[Задание 2 4](#__RefHeading___Toc307_1751619282)

[Задание 3 4](#__RefHeading___Toc309_1751619282)

[Задание 4 5](#__RefHeading___Toc311_1751619282)

[Задание 5 5](#__RefHeading___Toc313_1751619282)

[Задание 6 7](#__RefHeading___Toc315_1751619282)

[Запрос с результатом: 7](#__RefHeading___Toc317_1751619282)

[Правильный запрос: 8](#__RefHeading___Toc319_1751619282)

[Задание 7 9](#__RefHeading___Toc321_1751619282)

[Выводы 9](#__RefHeading___Toc323_1751619282)

# **Ход работы**

Отчет на гитхабе доступен здесь – https://github.com/N0fckgway

## **Текст задания**

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

1. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:  
   Таблицы: Н\_ОЦЕНКИ, Н\_ВЕДОМОСТИ.  
   Вывести атрибуты: Н\_ОЦЕНКИ.КОД, Н\_ВЕДОМОСТИ.ДАТА.  
   Фильтры (AND):   
   a) Н\_ОЦЕНКИ.КОД < 2.  
   b) Н\_ВЕДОМОСТИ.ИД < 1250981.  
   Вид соединения: INNER 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.ОТЧЕСТВО, Н\_ВЕДОМОСТИ.ИД, Н\_СЕССИЯ.ИД.  
   Фильтры (AND):   
   a) Н\_ЛЮДИ.ИМЯ > Ярослав.  
   b) Н\_ВЕДОМОСТИ.ДАТА < 1998-01-05.  
   c) Н\_СЕССИЯ.УЧГОД < 2003/2004.  
   Вид соединения: LEFT JOIN.
3. Вывести число фамилий и имен без учета повторений.  
   При составлении запроса нельзя использовать DISTINCT.
4. В таблице Н\_ГРУППЫ\_ПЛАНОВ найти номера планов, по которым обучается (обучалось) ровно 2 групп на заочной форме обучения.  
   Для реализации использовать подзапрос.
5. Выведите таблицу со средним возрастом студентов во всех группах (Группа, Средний возраст), где средний возраст больше минимального возраста в группе 1101.
6. Получить список студентов, зачисленных после 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 EXISTS.
7. Сформировать запрос для получения числа в группе No 3100 троечников

## **Реализация запросов на SQL**

Задание 1

SELECT

Н\_ОЦЕНКИ.КОД,

Н\_ВЕДОМОСТИ.ДАТА

FROM

Н\_ВЕДОМОСТИ

INNER JOIN

Н\_ОЦЕНКИ

ON Н\_ОЦЕНКИ.КОД = Н\_ВЕДОМОСТИ.ОЦЕНКА

WHERE

Н\_ОЦЕНКИ.КОД < '2' AND

Н\_ВЕДОМОСТИ.ИД < 1250981;

Задание 2

SELECT

Н\_ЛЮДИ.ОТЧЕСТВО,

Н\_ВЕДОМОСТИ.ИД,

Н\_СЕССИЯ.ИД

FROM

Н\_ЛЮДИ

LEFT JOIN

Н\_ВЕДОМОСТИ

ON Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = Н\_ЛЮДИ.ИД

LEFT JOIN

Н\_СЕССИЯ

ON Н\_ВЕДОМОСТИ.СЭС\_ИД = Н\_СЕССИЯ.СЭС\_ИД

WHERE

Н\_ЛЮДИ.ИМЯ > 'Ярослав' AND

Н\_ВЕДОМОСТИ.ДАТА < '1998-01-05' AND

Н\_СЕССИЯ.УЧГОД < '2003/2004';

Задание 3

SELECT

COUNT(\*)

FROM

(SELECT ФАМИЛИЯ, ИМЯ FROM Н\_ЛЮДИ GROUP BY ФАМИЛИЯ, ИМЯ);

Задание 4

SELECT

ПЛАН\_ИД

FROM(SELECT

gr.ПЛАН\_ИД,

COUNT(gr.ГРУППА) AS КОЛИЧЕСТВО\_ГРУПП

FROM

Н\_ГРУППЫ\_ПЛАНОВ gr

FULL OUTER JOIN

Н\_ПЛАНЫ AS p

ON p.ПЛАН\_ИД = gr.ПЛАН\_ИД

WHERE p.ФО\_ИД = 3

GROUP BY gr.ПЛАН\_ИД)

WHERE КОЛИЧЕСТВО\_ГРУПП = 2

ORDER BY КОЛИЧЕСТВО\_ГРУПП;

Задание 5

SELECT

s.ГРУППА,

ROUND(AVG(DATE\_PART('year', AGE(CURRENT\_DATE, p.ДАТА\_РОЖДЕНИЯ)))) AS СРЕД\_ВОЗРАСТ

FROM

Н\_ЛЮДИ AS p

JOIN Н\_УЧЕНИКИ AS s

ON p.ИД = s.ЧЛВК\_ИД

GROUP BY

s.ГРУППА

HAVING

AVG(DATE\_PART('year', AGE(CURRENT\_DATE, p.ДАТА\_РОЖДЕНИЯ))) > (SELECT MIN(DATE\_PART('year', AGE(CURRENT\_DATE, p.ДАТА\_РОЖДЕНИЯ))) FROM Н\_ЛЮДИ AS p JOIN Н\_УЧЕНИКИ AS s ON p.ИД = s.ЧЛВК\_ИД WHERE s.ГРУППА = '1101')

ORDER BY

СРЕД\_ВОЗРАСТ;

Задание 6

SELECT

s.ГРУППА AS номер\_группы,

s.ИД AS номер\_ученика,

l.ФАМИЛИЯ AS фамилия\_студента,

l.ИМЯ AS имя\_студента,

l.ОТЧЕСТВО AS отчество\_студента,

s.ПЛАН\_ИД AS номер\_приказа,

s.СОСТОЯНИЕ AS состояние\_приказа

FROM Н\_ЛЮДИ AS l

JOIN Н\_УЧЕНИКИ AS s

ON l.ИД = s.ЧЛВК\_ИД

WHERE EXISTS(SELECT 1 FROM Н\_ПЛАНЫ AS p

JOIN

Н\_УЧЕБНЫЕ\_ГОДА AS y

ON p.УЧЕБНЫЙ\_ГОД = y.УЧЕБНЫЙ\_ГОД

WHERE

p.ФО\_ИД IN (1, 2)

AND p.КУРС = 1

AND y.УЧЕБНЫЙ\_ГОД = '2011/2012'

AND s.НАЧАЛО > '2011-09-01 00:00:00')

ORDER BY

s.ГРУППА,

l.ФАМИЛИЯ,

l.ИМЯ;

Задание 7

SELECT

COUNT(s.ИД)

FROM

Н\_УЧЕНИКИ AS s

LEFT JOIN

Н\_ВЕДОМОСТИ AS ved

ON ved.ЧЛВК\_ИД = s.ЧЛВК\_ИД

WHERE s.ГРУППА = '3101' AND ved.ОЦЕНКА = '3';

## **Выводы**

В ходе выполнения лабораторной работы были реализованы сложные запросы к учебной бд, которые включают в себя создание подзапросов, JOIN’ов или и того, и другого. Помимо этого были изучены способы фильтрации данных через операторы сравнения, LIKE, SIMILAR TO.