МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«Национальный исследовательский университет ИТМО»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И  
КОМПЬЮТЕРНОЙ ТЕХНИКИ

**ЛАБОРАТОРНАЯ РАБОТА №2**

по дисциплине

«Базы данных»

Вариант № 2834923

***Выполнил:***Студент группы P3107  
Пшеничников Артём Дмитриевич

***Преподаватель:***  
Байрамова Хумай Бахруз Кызы

**Содержание**

[Задание 3](#_gjdgxs)

[Реализация запросов 3](#_69g569z0kesh)

[Вывод](#_3znysh7) 8

# **Задание**

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

# **Реализация запросов**

1. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:  
Таблицы: Н\_ОЦЕНКИ, Н\_ВЕДОМОСТИ.  
Вывести атрибуты: Н\_ОЦЕНКИ.ПРИМЕЧАНИЕ, Н\_ВЕДОМОСТИ.ЧЛВК\_ИД.  
Фильтры (AND):   
a) Н\_ОЦЕНКИ.КОД < 4.  
b) Н\_ВЕДОМОСТИ.ДАТА < 2022-06-08.  
c) Н\_ВЕДОМОСТИ.ДАТА = 2010-06-18.  
Вид соединения: LEFT JOIN.

SELECT  
 "Н\_ОЦЕНКИ"."ПРИМЕЧАНИЕ",  
 "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД"  
FROM  
 "Н\_ОЦЕНКИ"  
 LEFT JOIN  
 "Н\_ВЕДОМОСТИ" ON "Н\_ОЦЕНКИ"."КОД" = "Н\_ВЕДОМОСТИ"."ОЦЕНКА"  
WHERE  
 "Н\_ОЦЕНКИ"."КОД" IN ('3', '2')  
 AND "Н\_ВЕДОМОСТИ"."ДАТА" < DATE '2022-06-08'  
 AND "Н\_ВЕДОМОСТИ"."ДАТА" = DATE '2010-06-18';

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.ЧЛВК\_ИД, Н\_СЕССИЯ.ДАТА.  
Фильтры (AND):   
a) Н\_ЛЮДИ.ФАМИЛИЯ > Ёлкин.  
b) Н\_ВЕДОМОСТИ.ИД < 1490007.  
Вид соединения: INNER JOIN.

SELECT  
 "Н\_ЛЮДИ"."ОТЧЕСТВО",  
 "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД",  
 "Н\_СЕССИЯ"."ДАТА"  
FROM  
 "Н\_ЛЮДИ"  
 INNER JOIN  
 "Н\_ВЕДОМОСТИ" ON "Н\_ЛЮДИ"."ИД" = "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД"  
 INNER JOIN  
 "Н\_СЕССИЯ" ON "Н\_ВЕДОМОСТИ"."СЭС\_ИД" = "Н\_СЕССИЯ"."ИД"  
WHERE  
 "Н\_ЛЮДИ"."ФАМИЛИЯ" > 'Ёлкин'  
 AND "Н\_ВЕДОМОСТИ"."ИД" < 1490007;

3. Вывести число имен без учета повторений.  
При составлении запроса нельзя использовать DISTINCT.

SELECT *count*("ИМЯ") AS "Число имен без учета повторений"  
FROM (SELECT "Н\_ЛЮДИ"."ИМЯ"  
 FROM "Н\_ЛЮДИ"  
 GROUP BY "Н\_ЛЮДИ"."ИМЯ")  
 AS "table";

4. Выдать различные отчества преподавателей и число людей с каждой из этих отчеств, ограничив список отчествами, встречающимися менее 50 раз на ФКТИУ.  
Для реализации использовать соединение таблиц.

SELECT "Н\_ЛЮДИ"."ОТЧЕСТВО",  
 *COUNT*("Н\_ЛЮДИ"."ИД") AS "количество\_людей"  
FROM "Н\_ЛЮДИ"  
 LEFT JOIN  
 "Н\_ОБУЧЕНИЯ" ON "Н\_ЛЮДИ"."ИД" = "Н\_ОБУЧЕНИЯ"."ЧЛВК\_ИД"  
 LEFT JOIN  
 "Н\_УЧЕНИКИ" ON "Н\_ОБУЧЕНИЯ"."ЧЛВК\_ИД" = "Н\_УЧЕНИКИ"."ЧЛВК\_ИД"  
 LEFT JOIN  
 "Н\_ПЛАНЫ" ON "Н\_УЧЕНИКИ"."ПЛАН\_ИД" = "Н\_ПЛАНЫ"."ИД"  
 LEFT JOIN  
 "Н\_ОТДЕЛЫ" ON "Н\_ПЛАНЫ"."ОТД\_ИД" = "Н\_ОТДЕЛЫ"."ИД"  
WHERE "Н\_ОТДЕЛЫ"."КОРОТКОЕ\_ИМЯ" = 'КТиУ' -- с ФКТИУ нет результатов  
 AND "Н\_ЛЮДИ"."ОТЧЕСТВО" IN (SELECT "Н\_ЛЮДИ"."ОТЧЕСТВО"  
 FROM "Н\_ЛЮДИ"  
 LEFT JOIN  
 "Н\_ОБУЧЕНИЯ" ON "Н\_ЛЮДИ"."ИД" = "Н\_ОБУЧЕНИЯ"."ЧЛВК\_ИД"  
 WHERE "Н\_ОБУЧЕНИЯ"."ЧЛВК\_ИД" IS NULL  
 AND "Н\_ЛЮДИ"."ОТЧЕСТВО" IS NOT NULL  
 GROUP BY "Н\_ЛЮДИ"."ОТЧЕСТВО") -- отчества преподов  
GROUP BY "Н\_ЛЮДИ"."ОТЧЕСТВО"  
HAVING *COUNT*("Н\_ЛЮДИ"."ИД") < 50;

5. Выведите таблицу со средним возрастом студентов во всех группах (Группа, Средний возраст), где средний возраст больше среднего возраста в группе 3100.

SELECT "Н\_УЧЕНИКИ"."ГРУППА",  
 *DATE\_PART*('year', *AVG*(*AGE*("Н\_ЛЮДИ"."ДАТА\_РОЖДЕНИЯ"))) "СРЕДНИЙ\_ВОЗРАСТ"  
  
FROM "Н\_УЧЕНИКИ"  
 JOIN "Н\_ЛЮДИ" ON "Н\_УЧЕНИКИ"."ЧЛВК\_ИД" = "Н\_ЛЮДИ"."ИД"  
GROUP BY "Н\_УЧЕНИКИ"."ГРУППА"  
HAVING *DATE\_PART*('year', *AVG*(*AGE*("Н\_ЛЮДИ"."ДАТА\_РОЖДЕНИЯ"))) >   
 (SELECT *DATE\_PART*('year', *AVG*(*AGE*("Н\_ЛЮДИ"."ДАТА\_РОЖДЕНИЯ")))

"СРЕДНИЙ\_ВОЗРАСТ"  
  
 FROM "Н\_УЧЕНИКИ"  
 JOIN "Н\_ЛЮДИ" ON "Н\_УЧЕНИКИ"."ЧЛВК\_ИД" = "Н\_ЛЮДИ"."ИД"  
 WHERE *CAST*("Н\_УЧЕНИКИ"."ГРУППА" AS INTEGER) = 3100  
 GROUP BY "Н\_УЧЕНИКИ"."ГРУППА");

6. Получить список студентов, отчисленных после первого сентября 2012 года с очной или заочной формы обучения (специальность: Программная инженерия). В результат включить:  
номер группы;  
номер, фамилию, имя и отчество студента;  
номер пункта приказа;  
Для реализации использовать подзапрос с IN.

SELECT "Н\_УЧЕНИКИ"."ГРУППА",  
 "Н\_УЧЕНИКИ"."ИД",  
 "Н\_ЛЮДИ"."ФАМИЛИЯ",  
 "Н\_ЛЮДИ"."ИМЯ",  
 "Н\_ЛЮДИ"."ОТЧЕСТВО",  
 "Н\_УЧЕНИКИ"."ПРИЗНАК",  
 "Н\_ОТДЕЛЫ"."КОРОТКОЕ\_ИМЯ"  
  
from "Н\_УЧЕНИКИ"  
 LEFT JOIN  
 "Н\_ОБУЧЕНИЯ" ON "Н\_УЧЕНИКИ"."ИД" = "Н\_ОБУЧЕНИЯ"."ЧЛВК\_ИД"  
 LEFT JOIN  
 "Н\_ЛЮДИ" ON "Н\_ОБУЧЕНИЯ"."ЧЛВК\_ИД" = "Н\_ЛЮДИ"."ИД"  
 LEFT JOIN  
 "Н\_ПЛАНЫ" ON "Н\_УЧЕНИКИ"."ПЛАН\_ИД" = "Н\_ПЛАНЫ"."ИД"  
 LEFT JOIN  
 "Н\_ОТДЕЛЫ" ON "Н\_ПЛАНЫ"."ОТД\_ИД" = "Н\_ОТДЕЛЫ"."ИД"  
WHERE -- очная и заочная => все  
 "Н\_ОБУЧЕНИЯ"."ЧЛВК\_ИД" IS NOT NULL  
 AND "Н\_ОТДЕЛЫ"."КОРОТКОЕ\_ИМЯ" = 'КТиУ' -- Программная инженерия нет  
 AND "Н\_УЧЕНИКИ"."ИД" IN (SELECT "ИД"  
 FROM "Н\_УЧЕНИКИ"  
 WHERE "ПРИЗНАК" = 'отчисл'  
 AND "КОНЕЦ" > DATE '2012-09-01');

7. Сформировать запрос для получения числа на ФКТИУ хорошистов.

SELECT *COUNT*(avg\_rating)  
  
FROM (SELECT *AVG*(*CAST*("Н\_ОЦЕНКИ"."КОД" AS INTEGER)) AS "avg\_rating"  
 FROM "Н\_УЧЕНИКИ"  
 JOIN  
 "Н\_ЛЮДИ" ON "Н\_УЧЕНИКИ"."ЧЛВК\_ИД" = "Н\_ЛЮДИ"."ИД"  
 LEFT JOIN  
 "Н\_ВЕДОМОСТИ" ON "Н\_ЛЮДИ"."ИД" = "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД"  
 LEFT JOIN  
 "Н\_ОЦЕНКИ" ON "Н\_ВЕДОМОСТИ"."ОЦЕНКА" = "Н\_ОЦЕНКИ"."КОД"  
 LEFT JOIN  
 "Н\_ПЛАНЫ" ON "Н\_УЧЕНИКИ"."ПЛАН\_ИД" = "Н\_ПЛАНЫ"."ИД"  
 LEFT JOIN  
 "Н\_ОТДЕЛЫ" ON "Н\_ПЛАНЫ"."ОТД\_ИД" = "Н\_ОТДЕЛЫ"."ИД"  
 WHERE "Н\_ОТДЕЛЫ"."КОРОТКОЕ\_ИМЯ" = 'КТиУ' -- с ФКТИУ нет результатов  
 AND "Н\_ОЦЕНКИ"."КОД" IN ('5', '4', '3', '2')  
 GROUP BY "Н\_ЛЮДИ"."ИД"  
 HAVING *AVG*(*CAST*("Н\_ОЦЕНКИ"."КОД" AS INTEGER)) >= 3.5) as grades;

# **Заключение**

В ходе лабораторной работы я освоил написание SQL-запросов, включая JOIN, подзапросы и агрегатные функции. Также научился фильтровать данные и обрабатывать ошибки.