Федеральное государственное автономное образовательное учреждение высшего образования Национальный исследовательский университет ИТМО

Факультет программной инженерии и компьютерной техники

Базы данных

Лабораторная работа №2

Вариант № 716

Выполнил: студент группы P3108, Васильев Никита Алексеевич

Преподаватель: Афанасьев Дмитрий Борисович

Санкт-Петербург 2024

Оглавление

[Текст задания 3](#_Toc163139868)

[Реализация запросов на SQL 3](#_Toc163139869)

[Вывод 7](#_Toc163139870)

# Текст задания

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.НАИМЕНОВАНИЕ, Н\_ВЕДОМОСТИ.ИД.  
   Фильтры (AND):  
   a) Н\_ТИПЫ\_ВЕДОМОСТЕЙ.ИД = 2.  
   b) Н\_ВЕДОМОСТИ.ДАТА = 2010-06-18.  
   c) Н\_ВЕДОМОСТИ.ДАТА < 2022-06-08.  
   Вид соединения: RIGHT 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.ДАТА, Н\_СЕССИЯ.ЧЛВК\_ИД.  
   Фильтры (AND):  
   a) Н\_ЛЮДИ.ИД < 142095.  
   b) Н\_ВЕДОМОСТИ.ДАТА < 2010-06-18.  
   Вид соединения: LEFT JOIN.
3. Вывести число фамилий без учета повторений.  
   При составлении запроса нельзя использовать DISTINCT.
4. Выдать различные фамилии преподавателей и число людей с каждой из этих фамилий, ограничив список фамилиями, встречающимися ровно 10 раз на ФКТИУ.  
   Для реализации использовать подзапрос.
5. Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср\_оценка), у которых средняя оценка равна средней оценк(е|и) в группе 1100.
6. Получить список студентов, отчисленных до первого сентября 2012 года с очной формы обучения. В результат включить:  
   номер группы;  
   номер, фамилию, имя и отчество студента;  
   номер пункта приказа;  
   Для реализации использовать соединение таблиц.
7. Вывести список студентов, имеющих одинаковые имена, но не совпадающие ид.

# Реализация запросов на SQL

*-- Задание 1  
-- Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:  
-- Н\_ТИПЫ\_ВЕДОМОСТЕЙ, Н\_ВЕДОМОСТИ.  
-- Вывести атрибуты: Н\_ТИПЫ\_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ, Н\_ВЕДОМОСТИ.ИД.  
-- Фильтры (AND):  
-- a) Н\_ТИПЫ\_ВЕДОМОСТЕЙ.ИД = 2.  
-- b) Н\_ВЕДОМОСТИ.ДАТА = 2010-06-18.  
-- c) Н\_ВЕДОМОСТИ.ДАТА < 2022-06-08.  
-- Вид соединения: RIGHT JOIN.*SELECT "Н\_ТИПЫ\_ВЕДОМОСТЕЙ"."НАИМЕНОВАНИЕ", "Н\_ВЕДОМОСТИ"."ИД"  
FROM "Н\_ВЕДОМОСТИ"  
 RIGHT JOIN "Н\_ТИПЫ\_ВЕДОМОСТЕЙ"

ON "Н\_ВЕДОМОСТИ"."ТВ\_ИД" = "Н\_ТИПЫ\_ВЕДОМОСТЕЙ"."ИД"  
WHERE "Н\_ТИПЫ\_ВЕДОМОСТЕЙ"."ИД" = 2  
 AND "Н\_ВЕДОМОСТИ"."ДАТА" = '2010-06-18'  
 AND "Н\_ВЕДОМОСТИ"."ДАТА" < '2022-06-08';  
  
*-- Задание 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.ДАТА, Н\_СЕССИЯ.ЧЛВК\_ИД.  
-- Фильтры (AND):  
-- a) Н\_ЛЮДИ.ИД < 142095.  
-- b) Н\_ВЕДОМОСТИ.ДАТА < 2010-06-18.  
-- Вид соединения: LEFT JOIN.*SELECT "Н\_ЛЮДИ"."ОТЧЕСТВО", "Н\_ВЕДОМОСТИ"."ДАТА", "Н\_СЕССИЯ"."ЧЛВК\_ИД"  
FROM "Н\_ЛЮДИ"  
 LEFT JOIN "Н\_ВЕДОМОСТИ" ON "Н\_ЛЮДИ"."ИД" = "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД"  
 LEFT JOIN "Н\_СЕССИЯ" ON "Н\_ЛЮДИ"."ИД" = "Н\_СЕССИЯ"."ЧЛВК\_ИД"  
WHERE "Н\_ЛЮДИ"."ИД" > 142095  
 AND "Н\_ВЕДОМОСТИ"."ДАТА" < '2010-06-18';  
  
*-- Задание 3  
-- Вывести число фамилий без учета повторений.  
-- При составлении запроса нельзя использовать DISTINCT.*SELECT *count*("ФАМИЛИЯ") AS "Число фамилий без повторений"  
FROM (SELECT "Н\_ЛЮДИ"."ФАМИЛИЯ" AS "ФАМИЛИЯ"  
 FROM "Н\_ЛЮДИ"  
 GROUP BY "Н\_ЛЮДИ"."ФАМИЛИЯ") as НЛФ;  
  
*-- Задание 4  
-- Выдать различные фамилии преподавателей и число людей с каждой из этих фамилий, ограничив список фамилиями, встречающимися ровно 10 раз на ФКТИУ.  
-- Для реализации использовать подзапрос.*SELECT "ФАМИЛИЯ",  
 *count*(*\**) AS Количество  
FROM "Н\_ЛЮДИ"  
WHERE "ФАМИЛИЯ" IN(  
SELECT "ФАМИЛИЯ"  
FROM "Н\_ЛЮДИ"  
WHERE "ФАМИЛИЯ" IN (  
SELECT "ФАМИЛИЯ"  
FROM "Н\_ЛЮДИ"  
 INNER JOIN "Н\_УЧЕНИКИ" ON "Н\_ЛЮДИ"."ИД" = "Н\_УЧЕНИКИ"."ЧЛВК\_ИД"  
 INNER JOIN "Н\_ПЛАНЫ" ON "Н\_УЧЕНИКИ"."ПЛАН\_ИД" = "Н\_ПЛАНЫ"."ИД"  
 INNER JOIN "Н\_ОТДЕЛЫ" ON "Н\_ПЛАНЫ"."ОТД\_ИД" = "Н\_ОТДЕЛЫ"."ИД"  
WHERE "Н\_ОТДЕЛЫ"."КОРОТКОЕ\_ИМЯ" = 'КТиУ'  
GROUP BY "ФАМИЛИЯ"  
HAVING *count*(DISTINCT "Н\_ЛЮДИ"."ИД") = 10  
 AND *count*(*\**) = 10)  
 AND "ИД" NOT IN (  
SELECT "Н\_ЛЮДИ"."ИД"  
FROM "Н\_ЛЮДИ"  
 INNER JOIN "Н\_УЧЕНИКИ" ON "Н\_ЛЮДИ"."ИД" = "Н\_УЧЕНИКИ"."ЧЛВК\_ИД")  
GROUP BY "ФАМИЛИЯ")  
GROUP BY "ФАМИЛИЯ";  
*-- Задание 5  
-- Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср\_оценка), у которых средняя оценка равна средней оценк(е|и) в группе 1100.*WITH СР\_ОЦЕНКА\_3100 AS (  
 SELECT *avg*("Н\_ВЕДОМОСТИ"."ОЦЕНКА"::INTEGER)  
 FROM "Н\_ВЕДОМОСТИ"  
 JOIN "Н\_УЧЕНИКИ" ON "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД" = "Н\_УЧЕНИКИ"."ЧЛВК\_ИД"

AND "Н\_УЧЕНИКИ"."ГРУППА" = '3100'  
 WHERE "Н\_ВЕДОМОСТИ"."ОЦЕНКА" IN ('2', '3', '4', '5')  
)  
SELECT "Н\_ЛЮДИ"."ИД", "Н\_ЛЮДИ"."ФАМИЛИЯ", *avg*("Н\_ВЕДОМОСТИ"."ОЦЕНКА"::INTEGER)  
FROM "Н\_ЛЮДИ"  
 JOIN "Н\_ВЕДОМОСТИ" ON "Н\_ЛЮДИ"."ИД" = "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД"  
 AND "Н\_ВЕДОМОСТИ"."ОЦЕНКА" IN ('2', '3', '4', '5')  
 JOIN "Н\_УЧЕНИКИ" ON "Н\_ЛЮДИ"."ИД" = "Н\_УЧЕНИКИ"."ЧЛВК\_ИД"  
 AND "Н\_УЧЕНИКИ"."ГРУППА" = '4100'  
GROUP BY "Н\_ЛЮДИ"."ИД", "Н\_ЛЮДИ"."ИМЯ", "Н\_ЛЮДИ"."ФАМИЛИЯ", "Н\_ЛЮДИ"."ОТЧЕСТВО"  
HAVING *avg*("Н\_ВЕДОМОСТИ"."ОЦЕНКА"::INTEGER) = (SELECT *\** FROM СР\_ОЦЕНКА\_3100);  
  
*-- Задание 6  
-- Получить список студентов, отчисленных до первого сентября 2012 года с очной формы обучения. В результат включить:  
-- номер группы;  
-- номер, фамилию, имя и отчество студента;  
-- номер пункта приказа;  
-- Для реализации использовать соединение таблиц.*SELECT "Н\_УЧЕНИКИ"."ГРУППА", "Н\_УЧЕНИКИ"."ИД", "Н\_ЛЮДИ". "ФАМИЛИЯ", "Н\_ЛЮДИ"."ИМЯ", "Н\_ЛЮДИ"."ОТЧЕСТВО", "Н\_УЧЕНИКИ"."П\_ПРКОК\_ИД", "Н\_УЧЕНИКИ"."КОНЕЦ"  
FROM "Н\_УЧЕНИКИ"  
JOIN "Н\_ЛЮДИ" ON "Н\_ЛЮДИ"."ИД" = "Н\_УЧЕНИКИ"."ЧЛВК\_ИД"  
WHERE "Н\_УЧЕНИКИ"."ИД" IN (  
 SELECT "Н\_УЧЕНИКИ"."ИД"  
 FROM "Н\_УЧЕНИКИ"  
 JOIN "Н\_ПЛАНЫ" ON "Н\_ПЛАНЫ"."ИД" = "Н\_УЧЕНИКИ"."ПЛАН\_ИД"  
 JOIN "Н\_ФОРМЫ\_ОБУЧЕНИЯ" ON "Н\_ПЛАНЫ"."ФО\_ИД" = "Н\_ФОРМЫ\_ОБУЧЕНИЯ"."ИД"  
 WHERE "Н\_ФОРМЫ\_ОБУЧЕНИЯ"."НАИМЕНОВАНИЕ" = 'Очная'

AND "Н\_УЧЕНИКИ"."ПРИЗНАК" = 'отчисл'

AND "Н\_УЧЕНИКИ"."СОСТОЯНИЕ" = 'утвержден'

AND "Н\_УЧЕНИКИ"."КОНЕЦ" < '2012-09-01 00:00:00'  
 )  
ORDER BY "Н\_УЧЕНИКИ"."КОНЕЦ";  
  
*-- Задание 7  
-- Вывести список студентов, имеющих одинаковые имена, но не совпадающие ид.*SELECT "ИД", "ИМЯ"  
FROM "Н\_ЛЮДИ"  
WHERE "Н\_ЛЮДИ"."ИМЯ" IN (  
 SELECT "Н\_ЛЮДИ"."ИМЯ"  
 FROM "Н\_ЛЮДИ"  
 INNER JOIN "Н\_УЧЕНИКИ" ON "Н\_ЛЮДИ"."ИД" = "Н\_УЧЕНИКИ"."ЧЛВК\_ИД"  
 GROUP BY "Н\_ЛЮДИ"."ИМЯ"  
 HAVING *count*(DISTINCT "Н\_ЛЮДИ"."ИД") > 1  
 )  
ORDER BY "ИМЯ", "Н\_ЛЮДИ"."ИД";

# Вывод

При выполнении данной лабораторной работы я познакомился с реализацией запросов на SQL, и операторами, которые для этого используются, а также научился работать с полученными данными.