**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,   
МЕХАНИКИ И ОПТИКИ**

***Изображение выглядит как черный, темнота

Автоматически созданное описание***

**Факультет Программной Инженерии и Компьютерной Техники**

**Дисциплина:**

***Информационные системы и базы данных***

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ ЛАБОТЕ № 3**

**Выполнил:**

Студент гр. P33151 *Соловьев Артемий Александрович*

**Проверила:**

*Байрамова Хумай Бахруз Кызы*

Санкт-Петербург

2023г.

**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\_ЛЮДИ, Н\_СЕССИЯ.  
   Вывести атрибуты: Н\_ЛЮДИ.ФАМИЛИЯ, Н\_СЕССИЯ.ИД.  
   Фильтры (AND):   
   a) Н\_ЛЮДИ.ОТЧЕСТВО < Александрович.  
   b) Н\_СЕССИЯ.ЧЛВК\_ИД = 100012.  
   c) Н\_СЕССИЯ.ЧЛВК\_ИД < 100012.  
   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.ЧЛВК\_ИД, Н\_СЕССИЯ.ИД.  
   Фильтры (AND):   
   a) Н\_ЛЮДИ.ФАМИЛИЯ = Петров.  
   b) Н\_ВЕДОМОСТИ.ДАТА > 2022-06-08.  
   Вид соединения: LEFT JOIN.
3. Вывести число отчеств 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 средними оценками студентов группы 4100 (Номер, ФИО, Ср\_оценка), у которых средняя оценка меньше минимальной оценк(е|и) в группе 1101.
6. Получить список студентов, зачисленных до первого сентября 2012 года на первый курс заочной формы обучения (специальность: 230101). В результат включить:  
   номер группы;  
   номер, фамилию, имя и отчество студента;  
   номер и состояние пункта приказа;  
   Для реализации использовать подзапрос с EXISTS.
7. Вывести список студентов, имеющих одинаковые отчества, но не совпадающие ид.

**Реализация запросов на SQL**

**Задание 1.**

Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:  
Таблицы: Н\_ЛЮДИ, Н\_СЕССИЯ.  
Вывести атрибуты: Н\_ЛЮДИ.ФАМИЛИЯ, Н\_СЕССИЯ.ИД.  
Фильтры (AND):   
a) Н\_ЛЮДИ.ОТЧЕСТВО < Александрович.  
b) Н\_СЕССИЯ.ЧЛВК\_ИД = 100012.  
c) Н\_СЕССИЯ.ЧЛВК\_ИД < 100012.  
Вид соединения: RIGHT JOIN.

*Запрос:*  
SELECT Н\_ЛЮДИ.ФАМИЛИЯ, Н\_СЕССИЯ.ИД FROM Н\_ЛЮДИ RIGHT JOIN Н\_СЕССИЯ ON (Н\_ЛЮДИ.ИД = Н\_СЕССИЯ.ЧЛВК\_ИД) WHERE (Н\_ЛЮДИ.ОТЧЕСТВО < 'Александрович' AND Н\_СЕССИЯ.ЧЛВК\_ИД = 100012 AND Н\_СЕССИЯ.ЧЛВК\_ИД < 100012);

*Вывод:*  
**Изображение выглядит как текст, снимок экрана, Шрифт, черный

Автоматически созданное описание**

**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.ЧЛВК\_ИД, Н\_СЕССИЯ.ИД.  
Фильтры (AND):   
a) Н\_ЛЮДИ.ФАМИЛИЯ = Петров.  
b) Н\_ВЕДОМОСТИ.ДАТА > 2022-06-08.  
Вид соединения: LEFT JOIN.

*Запрос:*  
SELECT Н\_ЛЮДИ.ОТЧЕСТВО, Н\_ВЕДОМОСТИ.ЧЛВК\_ИД, Н\_СЕССИЯ.ИД FROM Н\_ЛЮДИ LEFT JOIN Н\_ВЕДОМОСТИ ON (Н\_ЛЮДИ.ИД = Н\_ВЕДОМОСТИ.ЧЛВК\_ИД) LEFT JOIN Н\_СЕССИЯ ON (Н\_ВЕДОМОСТИ.СЭС\_ИД = Н\_СЕССИЯ.СЭС\_ИД) WHERE (Н\_ЛЮДИ.ФАМИЛИЯ = 'Петров' AND Н\_ВЕДОМОСТИ.ДАТА > '2022-06-08');

*Вывод:*  
**Изображение выглядит как текст, снимок экрана, Шрифт, черный

Автоматически созданное описание**

**Задание 3.**

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

*Запрос:*  
SELECT Н\_ЛЮДИ.ОТЧЕСТВО, COUNT(\*) AS КОЛИЧЕСТВО FROM Н\_ЛЮДИ GROUP BY Н\_ЛЮДИ.ОТЧЕСТВО;

*Вывод:*  
Изображение выглядит как текст, снимок экрана, меню

Автоматически созданное описание

**Задание 4.**

В таблице Н\_ГРУППЫ\_ПЛАНОВ найти номера планов, по которым обучается (обучалось) ровно 2 групп на заочной форме обучения.  
Для реализации использовать подзапрос.

*Запрос:*  
SELECT "Н\_ГРУППЫ\_ПЛАНОВ"."ПЛАН\_ИД" FROM "Н\_ГРУППЫ\_ПЛАНОВ"

join (SELECT COUNT(\*) AS cnt, "ГРУППА" FROM "Н\_ГРУППЫ\_ПЛАНОВ" GROUP BY "ГРУППА") as groups ON "Н\_ГРУППЫ\_ПЛАНОВ"."ГРУППА" = groups."ГРУППА"

join "Н\_ПЛАНЫ" on "Н\_ГРУППЫ\_ПЛАНОВ"."ПЛАН\_ИД" = "Н\_ПЛАНЫ"."ПЛАН\_ИД"

join "Н\_ФОРМЫ\_ОБУЧЕНИЯ" on "Н\_ФОРМЫ\_ОБУЧЕНИЯ"."ИД" = "Н\_ПЛАНЫ"."ФО\_ИД" and "НАИМЕНОВАНИЕ"= 'Заочная'  
WHERE cnt = 2;

*Вывод:*  
Изображение выглядит как текст, снимок экрана, Шрифт, черный

Автоматически созданное описание

**Задание 5.**

Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср\_оценка), у которых средняя оценка меньше минимальной оценк(е|и) в группе 1101.

*Запрос*:  
SELECT "ЧЛВК\_ИД", "ФАМИЛИЯ", "ИМЯ", "ОТЧЕСТВО", AVG("ОЦЕНКА"::int) AS "AVG" FROM "Н\_ВЕДОМОСТИ"

join "Н\_УЧЕНИКИ" USING ("ЧЛВК\_ИД")

join "Н\_ЛЮДИ" ON "Н\_ЛЮДИ"."ИД" = "Н\_ВЕДОМОСТИ"."ЧЛВК\_ИД"

WHERE "ГРУППА" = '4100' AND "ОЦЕНКА" SIMILAR TO '1|2|3|4|5$'

GROUP BY "ЧЛВК\_ИД", "ФАМИЛИЯ", "ИМЯ", "ОТЧЕСТВО"

HAVING AVG("ОЦЕНКА"::int) >(SELECT MIN("ОЦЕНКА"::int) FROM "Н\_ВЕДОМОСТИ"  
JOIN "Н\_УЧЕНИКИ" USING ("ЧЛВК\_ИД")  
WHERE "ГРУППА" = '1101' AND "ОЦЕНКА" SIMILAR TO '1|2|3|4|5');

*Вывод:  
Изображение выглядит как текст, снимок экрана, Шрифт, черный

Автоматически созданное описание*

**Задание 6.**

Получить список студентов, зачисленных до первого сентября 2012 года на первый курс заочной формы обучения (специальность: 230101). В результат включить:  
номер группы;  
номер, фамилию, имя и отчество студента;  
номер и состояние пункта приказа;  
Для реализации использовать подзапрос с EXISTS.

*Запрос:*SELECT "Н\_УЧЕНИКИ"."ГРУППА", "Н\_ЛЮДИ"."ИД", "Н\_ЛЮДИ"."ФАМИЛИЯ", "Н\_ЛЮДИ"."ИМЯ", "Н\_ЛЮДИ"."ОТЧЕСТВО", "Н\_УЧЕНИКИ"."В\_СВЯЗИ\_С", "Н\_УЧЕНИКИ"."СОСТОЯНИЕ" FROM "Н\_ЛЮДИ"  
join "Н\_УЧЕНИКИ" 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"  
join "Н\_НАПРАВЛЕНИЯ\_СПЕЦИАЛ" on "Н\_НАПРАВЛЕНИЯ\_СПЕЦИАЛ"."НАПС\_ИД" = "Н\_ПЛАНЫ"."НАПС\_ИД"  
join "Н\_НАПР\_СПЕЦ" on "Н\_НАПРАВЛЕНИЯ\_СПЕЦИАЛ"."НС\_ИД" = "Н\_НАПР\_СПЕЦ"."ИД"  
and ("Н\_ФОРМЫ\_ОБУЧЕНИЯ"."НАИМЕНОВАНИЕ" = 'Заочная' and "Н\_НАПР\_СПЕЦ"."КОД\_НАПРСПЕЦ" = '230101')  
WHERE EXISTS(SELECT \* FROM "Н\_ПЛАНЫ" WHERE "КУРС" = 1 and "НАЧАЛО" < '2012-09-01' );

*Вывод:  
Изображение выглядит как текст, снимок экрана, Шрифт, черный

Автоматически созданное описание*

**Задание 7.**

Вывести список студентов, имеющих одинаковые отчества, но не совпадающие ид.

*Запрос:*SELECT "Н\_ЛЮДИ"."ИД", "ФАМИЛИЯ", "ИМЯ", "ОТЧЕСТВО" FROM "Н\_УЧЕНИКИ"  
join "Н\_ЛЮДИ" ON "Н\_УЧЕНИКИ"."ЧЛВК\_ИД" = "Н\_ЛЮДИ"."ИД"  
WHERE "ОТЧЕСТВО" IN (SELECT "ОТЧЕСТВО" FROM "Н\_ЛЮДИ" GROUP BY "ОТЧЕСТВО" HAVING count("ОТЧЕСТВО") > 1)  
ORDER BY "ОТЧЕСТВО";

*Вывод:  
Изображение выглядит как текст, снимок экрана

Автоматически созданное описание*