Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования  
**"Национальный Исследовательский Университет ИТМО"**

Факультет Программной Инженерии и Компьютерных Технологий

**Лабораторная работа №3**

по дисциплине

**«Информационные системы и базы данных»**

Вариант 87169

Выполнил:

Студент группы P33302

Тюрин Святослав Вячеславович

Преподаватель

Николаев Владимир Вячеславович

Санкт Петербург

2023

**Изображение выглядит как текст, План, диаграмма, черно-белый

Автоматически созданное описание**

**Ход работы:**

1. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:  
   Таблицы: Н\_ЛЮДИ, Н\_ВЕДОМОСТИ.  
   Вывести атрибуты: Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ВЕДОМОСТИ.ЧЛВК\_ИД.  
   Фильтры (AND):  
   a) Н\_ЛЮДИ.ОТЧЕСТВО = Георгиевич.  
   b) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 142390.  
   c) Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = 142390.  
   Вид соединения: RIGHT JOIN.

select Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ВЕДОМОСТИ.ЧДВК

from Н\_ЛЮДИ right join Н\_ВЕДОМОСТИ on Н\_ЛЮДИ.ИД = Н\_ВЕДОМОСТИ.ЧЛВК\_ИД

where Н\_ЛЮДИ.ОТЧЕСТВО = Георгиевич

and Н\_ВЕДОМОСТИ.ЧЛВК\_ИД < 142390

and Н\_ВЕДОМОСТИ.ЧЛВК\_ИД = 142390;

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

1. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:  
   Таблицы: Н\_ЛЮДИ, Н\_ОБУЧЕНИЯ, Н\_УЧЕНИКИ.  
   Вывести атрибуты: Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД, Н\_УЧЕНИКИ.ИД.  
   Фильтры: (AND)  
   a) Н\_ЛЮДИ.ИД > 142095.  
   b) Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД > 112514.  
   Вид соединения: LEFT JOIN.

select Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД, Н\_УЧЕНИКИ.ИД

from Н\_ОБУЧЕНИЯ left join Н\_ЛЮДИ on Н\_ЛЮДИ.ИД = Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД

left join Н\_УЧЕНИКИ on Н\_УЧЕНИКИ.ИД = Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД

where Н\_ЛЮДИ.ИД > 142095

and Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД > 112514;

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

1. Составить запрос, который ответит на вопрос, есть ли среди студентов группы 3102 люди без ИНН.

select count(Н\_УЧЕНИКИ.ГРУППА)

from Н\_ОБУЧЕНИЯ left join Н\_ЛЮДИ on Н\_ЛЮДИ.ИД = Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД

left join Н\_УЧЕНИКИ on Н\_УЧЕНИКИ.ИД = Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД

where Н\_УЧЕНИКИ.ГРУПППА = 3102

and Н\_ЛЮДИ.ИНН is NULL

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

1. Выдать различные имена студентов и число людей с каждой из этих имен, ограничив список именами, встречающимися менее 10 раз на заочной форме обучения.  
   Для реализации использовать подзапрос.

select Н\_ЛЮДИ.ИМЯ, count(Н\_ЛЮДИ.ИМЯ) as COUNT\_NAME

from Н\_ЛЮДИ

where Н\_ЛЮДИ.ИД in (select Н\_УЧЕНИКИ.ИД

from Н\_УЧЕНИКИ

left join Н\_ОБУЧЕНИЯ on Н\_УЧЕНИКИ.ИД = Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД

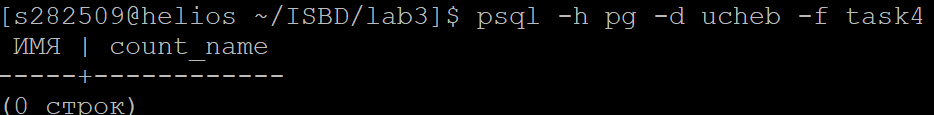
left join Н\_ЛЮДИ on Н\_ЛЮДИ.ИД = Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД

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\_NAME < 10



Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

1. Выведите таблицу со средним возрастом студентов во всех группах (Группа, Средний возраст), где средний возраст больше максимального возраста в группе 1100.

select ГРУППА, avg(date\_part('year', age(LEAST(now(), ДАТА\_СМЕРТИ), ДАТА\_РОЖДЕНИЯ))) as СРЕДНИЙ\_ВОЗРАСТ

from Н\_ЛЮДИ

inner join Н\_УЧЕНИКИ on Н\_УЧЕНИКИ.ЧЛВК\_ИД = Н\_ЛЮДИ.ИД

group by ГРУППА

having avg(date\_part('year', age(LEAST(now(), ДАТА\_СМЕРТИ), ДАТА\_РОЖДЕНИЯ))) >

(

select max(date\_part('year', age(LEAST(now(), ДАТА\_СМЕРТИ), ДАТА\_РОЖДЕНИЯ)))

from Н\_ЛЮДИ

inner join Н\_УЧЕНИКИ on Н\_УЧЕНИКИ.ЧЛВК\_ИД = Н\_ЛЮДИ.ИД

where ГРУППА = '1100'

group by ГРУППА

);

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

1. Получить список студентов, зачисленных после первого сентября 2012 года на первый курс очной или заочной формы обучения (специальность: 230101). В результат включить:  
   номер группы;  
   номер, фамилию, имя и отчество студента;  
   номер и состояние пункта приказа;  
   Для реализации использовать соединение таблиц.

select Н\_УЧЕНИКИ.ГРУППА,  
 Н\_ЛЮДИ.ИД,  
 Н\_ЛЮДИ.ФАМИЛИЯ,  
 Н\_ЛЮДИ.ИМЯ,  
 Н\_ЛЮДИ.ОТЧЕСТВО,  
 Н\_УЧЕНИКИ.П\_ПРКОК\_ИД,  
 Н\_УЧЕНИКИ.СОСТОЯНИЕ  
from Н\_УЧЕНИКИ  
 inner join Н\_ПЛАНЫ on Н\_УЧЕНИКИ.ПЛАН\_ИД = Н\_ПЛАНЫ.ИД  
 inner join Н\_ФОРМЫ\_ОБУЧЕНИЯ on Н\_ПЛАНЫ.ФО\_ИД = Н\_ФОРМЫ\_ОБУЧЕНИЯ.ИД  
 inner join Н\_ЛЮДИ on Н\_УЧЕНИКИ.ЧЛВК\_ИД = Н\_ЛЮДИ.ИД  
where Н\_УЧЕНИКИ.НАЧАЛО >= '2012-09-01'  
 and Н\_ПЛАНЫ.КУРС = 1  
 and Н\_УЧЕНИКИ.СОСТОЯНИЕ = 'утвержден'  
 and (Н\_ФОРМЫ\_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ = 'Очная' or Н\_ФОРМЫ\_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ = 'Заочная');

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

1. Вывести список студентов, имеющих одинаковые фамилии, но не совпадающие ид.

select Н\_ЛЮДИ.ИМЯ, Н\_ЛЮДИ.ФАМИЛИЯ, Н\_УЧЕНИКИ.ИД

from "Н\_УЧЕНИКИ"

join "Н\_ЛЮДИ" on "Н\_ЛЮДИ"."ИД" = "Н\_УЧЕНИКИ"."ЧЛВК\_ИД"

where "ЧЛВК\_ИД" in (select "ИД"

from "Н\_ЛЮДИ"

where "ФАМИЛИЯ" in (select "ФАМИЛИЯ" from "Н\_ЛЮДИ" group by "ФАМИЛИЯ" having count("ОТЧЕСТВО") > 1)

and "ИД" in (select "ИД" from "Н\_ЛЮДИ" group by "ИД" having count("ИД") = 1))

order by Н\_ЛЮДИ.ФАМИЛИЯ;

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

**Выводы по работе:**

Основная сложность лабораторной работы отнюдь не в написании команд, а в работе с чужой базой данных, и попытке понять концепцию их распределения по таблицам. В целом, база спроектирована понятно и удобно, из минусов – названия полей и таблиц прописными буквами и на русском. Также расстроило то, что на многие запросы ничего не выводится, потому что условия к запросам генерируются автоматически. Сам же язык SQL весьма удобен и очевиден, что видимо и сделало его столь популярным.