**Министерство науки и высшего образования Российской Федерации**

федеральное государственное автономное образовательное учреждение высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**Отчет**

по лабораторной работе номер 3

по дисциплине «**Информационные системы и базы данных**»

Автор: Дениченко Александр Олегович

Факультет: ПИиКТ

Группа: P3112

Преподаватель: Лисицина В.В



Санкт-Петербург, 2023

**Задание:**

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.ИД, Н\_СЕССИЯ.УЧГОД.

Фильтры (AND):

a) Н\_ЛЮДИ.ИД < 152862.

b) Н\_СЕССИЯ.ИД = 14.

Вид соединения: LEFT JOIN.

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.ИМЯ, Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД, Н\_УЧЕНИКИ.НАЧАЛО.

Фильтры: (AND)

a) Н\_ЛЮДИ.ОТЧЕСТВО = Александрович.

b) Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД > 163276.

c) Н\_УЧЕНИКИ.ИД = 1.

Вид соединения: INNER JOIN.

1. Вывести число студентов вечерней формы обучения, которые младше 20 лет.

Ответ должен содержать только одно число.

1. Выдать различные отчества студентов и число людей с каждой из этих отчеств, ограничив список отчествами, встречающимися более 10 раз на на заочной форме обучения.

Для реализации использовать подзапрос.

1. Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср\_оценка), у которых средняя оценка больше максимальной оценк(е|и) в группе 3100.
2. Получить список студентов, отчисленных после первого сентября 2012 года с очной или заочной формы обучения. В результат включить:

номер группы;

номер, фамилию, имя и отчество студента;

номер пункта приказа;

Для реализации использовать соединение таблиц.

1. Вывести список людей, не являющихся или не являвшихся студентами ФКТИУ (данные, о которых отсутствуют в таблице Н\_УЧЕНИКИ). В запросе нельзя использовать DISTINCT.

**Ход работы:**

-- Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:

-- Таблицы: Н\_ЛЮДИ, Н\_СЕССИЯ.

-- Вывести атрибуты: Н\_ЛЮДИ.ИД, Н\_СЕССИЯ.УЧГОД.

-- Фильтры (AND):

-- a) Н\_ЛЮДИ.ИД < 152862.

-- b) Н\_СЕССИЯ.ИД = 14.

-- Вид соединения: LEFT JOIN.

SELECT Н\_ЛЮДИ.ИД, Н\_СЕССИЯ.УЧГОД

FROM Н\_ЛЮДИ

LEFT JOIN Н\_СЕССИЯ ON Н\_СЕССИЯ.ЧЛВК\_ИД = Н\_ЛЮДИ.ИД

WHERE (Н\_ЛЮДИ.ИД < 152862 AND Н\_СЕССИЯ.ИД = 14);

-- Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:

-- Таблицы: Н\_ЛЮДИ, Н\_ОБУЧЕНИЯ, Н\_УЧЕНИКИ.

-- Вывести атрибуты: Н\_ЛЮДИ.ИМЯ, Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД, Н\_УЧЕНИКИ.НАЧАЛО.

-- Фильтры: (AND)

-- a) Н\_ЛЮДИ.ОТЧЕСТВО = Александрович.

-- b) Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД > 163276.

-- c) Н\_УЧЕНИКИ.ИД = 1.

-- Вид соединения: INNER JOIN.

SELECT Н\_ЛЮДИ.ИМЯ, Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД, Н\_УЧЕНИКИ.НАЧАЛО

FROM Н\_ЛЮДИ

INNER JOIN Н\_ОБУЧЕНИЯ ON Н\_ЛЮДИ.ИД = Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД

INNER JOIN Н\_УЧЕНИКИ ON Н\_ЛЮДИ.ИД = Н\_УЧЕНИКИ.ЧЛВК\_ИД

WHERE(Н\_ЛЮДИ.ОТЧЕСТВО = 'Александрович' AND Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД > 163276 AND Н\_УЧЕНИКИ.ИД = 1);

-- Вывести число студентов вечерней формы обучения, которые младше 20 лет.

-- Ответ должен содержать только одно число.

--(для проверки можно ввести очную форму обучения)

SELECT COUNT(\*) FROM Н\_ЛЮДИ

WHERE (date\_part('year', age("Н\_ЛЮДИ"."ДАТА\_РОЖДЕНИЯ" )) < 20

AND Н\_ЛЮДИ.ИД IN

(SELECT Н\_УЧЕНИКИ.ИД FROM Н\_УЧЕНИКИ

INNER JOIN Н\_ОБУЧЕНИЯ ON Н\_УЧЕНИКИ.ИД = Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД

WHERE(Н\_УЧЕНИКИ.ИД IN

(SELECT Н\_УЧЕНИКИ.ИД FROM Н\_УЧЕНИКИ

INNER JOIN Н\_ПЛАНЫ ON Н\_УЧЕНИКИ.ПЛАН\_ИД=Н\_ПЛАНЫ.ИД

WHERE(Н\_ПЛАНЫ.ИД IN (

SELECT Н\_ПЛАНЫ.ИД FROM Н\_ПЛАНЫ

INNER JOIN Н\_ФОРМЫ\_ОБУЧЕНИЯ ON Н\_ПЛАНЫ.ФО\_ИД=Н\_ФОРМЫ\_ОБУЧЕНИЯ.ИД

WHERE(Н\_ФОРМЫ\_ОБУЧЕНИЯ.ИМЯ\_В\_ИМИН\_ПАДЕЖЕ='вечерняя')))))));

-- Выдать различные отчества студентов и число людей с каждой из этих отчеств, ограничив список отчествами,

-- встречающимися более 10 раз на на "заочной" форме обучения.

-- Для реализации использовать подзапрос.

SELECT REPLACE(Н\_ЛЮДИ.ОТЧЕСТВО, 'вна', 'вич') AS ОТЧЕСТВО, COUNT(\*) AS КОЛИЧЕСТВО

FROM Н\_ЛЮДИ

WHERE (Н\_ЛЮДИ.ОТЧЕСТВО IN

(SELECT REPLACE(Н\_ЛЮДИ.ОТЧЕСТВО, 'вна', 'вич') AS ОТЧЕСТВО

FROM Н\_ЛЮДИ

WHERE (Н\_ЛЮДИ.ИД IN

(SELECT Н\_УЧЕНИКИ.ИД FROM Н\_УЧЕНИКИ

INNER JOIN Н\_ОБУЧЕНИЯ ON Н\_УЧЕНИКИ.ИД = Н\_ОБУЧЕНИЯ.ЧЛВК\_ИД

WHERE(Н\_УЧЕНИКИ.ИД IN

(SELECT Н\_УЧЕНИКИ.ИД FROM Н\_УЧЕНИКИ

INNER JOIN Н\_ПЛАНЫ ON Н\_УЧЕНИКИ.ПЛАН\_ИД=Н\_ПЛАНЫ.ИД

WHERE(Н\_ПЛАНЫ.ИД IN (

SELECT Н\_ПЛАНЫ.ИД FROM Н\_ПЛАНЫ

INNER JOIN Н\_ФОРМЫ\_ОБУЧЕНИЯ ON Н\_ПЛАНЫ.ФО\_ИД=Н\_ФОРМЫ\_ОБУЧЕНИЯ.ИД

WHERE(Н\_ФОРМЫ\_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ='Очная')))))))

GROUP BY REPLACE(Н\_ЛЮДИ.ОТЧЕСТВО, 'вна', 'вич')

HAVING count(\*) > 10

))

GROUP BY REPLACE(Н\_ЛЮДИ.ОТЧЕСТВО, 'вна', 'вич');

-- Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср\_оценка),

-- у которых средняя оценка больше максимальной оценк(е|и) в группе 3100.

-- (чтобы проверить правильность кода можно поменять группы местами и сделать чтобы средняя оценка была меньше максимальной оценк(е|и))

SELECT Н\_ЛЮДИ.ИД, Н\_ЛЮДИ.ИМЯ, Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ЛЮДИ.ОТЧЕСТВО,

SUM(Н\_ОЦЕНКИ.СОРТ)/COUNT(Н\_ОЦЕНКИ.СОРТ) AS ОЦЕНКА

FROM Н\_ЛЮДИ

INNER JOIN Н\_ВЕДОМОСТИ ON Н\_ЛЮДИ.ИД=Н\_ВЕДОМОСТИ.ЧЛВК\_ИД

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

WHERE Н\_ЛЮДИ.ИД IN (

SELECT Н\_ЛЮДИ.ИД FROM Н\_ЛЮДИ

WHERE Н\_ЛЮДИ.ИД IN (

SELECT Н\_УЧЕНИКИ.ИД FROM Н\_УЧЕНИКИ

WHERE Н\_УЧЕНИКИ.ГРУППА= '4100'

)

)

GROUP BY Н\_ЛЮДИ.ИД, Н\_ЛЮДИ.ИМЯ, Н\_ЛЮДИ.ФАМИЛИЯ, Н\_ЛЮДИ.ОТЧЕСТВО

HAVING CAST((SUM(Н\_ОЦЕНКИ.СОРТ)/COUNT(Н\_ОЦЕНКИ.СОРТ)) AS INT) > (

SELECT MAX(Н\_ОЦЕНКИ.СОРТ) FROM Н\_ЛЮДИ

INNER JOIN Н\_ВЕДОМОСТИ ON Н\_ЛЮДИ.ИД=Н\_ВЕДОМОСТИ.ЧЛВК\_ИД

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

WHERE (Н\_ЛЮДИ.ИД IN(

SELECT Н\_ЛЮДИ.ИД FROM Н\_ЛЮДИ

WHERE (Н\_ЛЮДИ.ИД IN

(SELECT Н\_УЧЕНИКИ.ИД FROM Н\_УЧЕНИКИ

WHERE(Н\_УЧЕНИКИ.ГРУППА= '3100'))))));

-- Получить список студентов, отчисленных после первого сентября 2012 года

-- с очной или заочной формы обучения.

-- В результат включить:

-- номер группы;

-- номер, фамилию, имя и отчество студента;

-- номер пункта приказа;

-- Для реализации использовать соединение таблиц.

-- (чтобы проверить правильность кода можно ввести не 2012, а 2000 год)

SELECT Н\_УЧЕНИКИ.НАЧАЛО, Н\_УЧЕНИКИ.ГРУППА, Н\_УЧЕНИКИ.ИД as ИД\_СТУДЕНТА, concat(Н\_ЛЮДИ.ФАМИЛИЯ, ' ', Н\_ЛЮДИ.ИМЯ, ' ', Н\_ЛЮДИ.ОТЧЕСТВО) as ФИО, concat(Н\_УЧЕНИКИ.ПРИЗНАК) as ПРИКАЗ FROM Н\_УЧЕНИКИ

JOIN Н\_ЛЮДИ 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

WHERE((Н\_ФОРМЫ\_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ = 'Очная' OR Н\_ФОРМЫ\_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ = 'Заочная') AND (Н\_УЧЕНИКИ.ПРИЗНАК = 'отчисл') AND (DATE(Н\_УЧЕНИКИ.НАЧАЛО) > '2012-09-01'));

-- Вывести список людей, не являющихся или не являвшихся студентами ФКТИУ

-- (данные, о которых отсутствуют в таблице Н\_УЧЕНИКИ). В запросе нельзя использовать DISTINCT.

SELECT concat(Н\_ЛЮДИ.ИМЯ, ' ', Н\_ЛЮДИ.ФАМИЛИЯ)

FROM Н\_ЛЮДИ

WHERE Н\_ЛЮДИ.ИД != ANY (SELECT Н\_УЧЕНИКИ.ЧЛВК\_ИД

FROM Н\_УЧЕНИКИ

JOIN Н\_ПЛАНЫ ON Н\_УЧЕНИКИ.ПЛАН\_ИД = Н\_ПЛАНЫ.ИД

JOIN Н\_ОТДЕЛЫ ON Н\_ОТДЕЛЫ.ИД = Н\_ПЛАНЫ.ОТД\_ИД

AND Н\_ОТДЕЛЫ.КОРОТКОЕ\_ИМЯ = 'КТиУ')

GROUP BY Н\_ЛЮДИ.ИМЯ, Н\_ЛЮДИ.ФАМИЛИЯ;

**Выводы:** Составлены запросы на языке SQL, изучено соединение таблиц, реализованы подзапросы.