

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский университет ИТМО»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ

ЛАБОРАТОРНАЯ РАБОТА №3

по дисциплине

‘Информационные системы и базы данных’

Вариант: 3331230

Выполнил:

Студент группы Р33312

Абульфатов Руслан

Мехтиевич

Преподаватель:

Наумова Надежда

Александровна

Санкт-Петербург, 2023

Задание:

По варианту, выданному преподавателем составить и выполнить запросы к базе данных “Учебный процесс”.

Команда для подключения к базе данных usheb:

```
psql -h pg -d usheb
```

Реализация запросов на SQL:

-- 1.

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

-- Таблицы: Н_ЛЮДИ, Н_ВЕДОМОСТИ.

-- Вывести атрибуты: Н_ЛЮДИ.ОТЧЕСТВО, Н_ВЕДОМОСТИ.ЧЛВК_ИД.

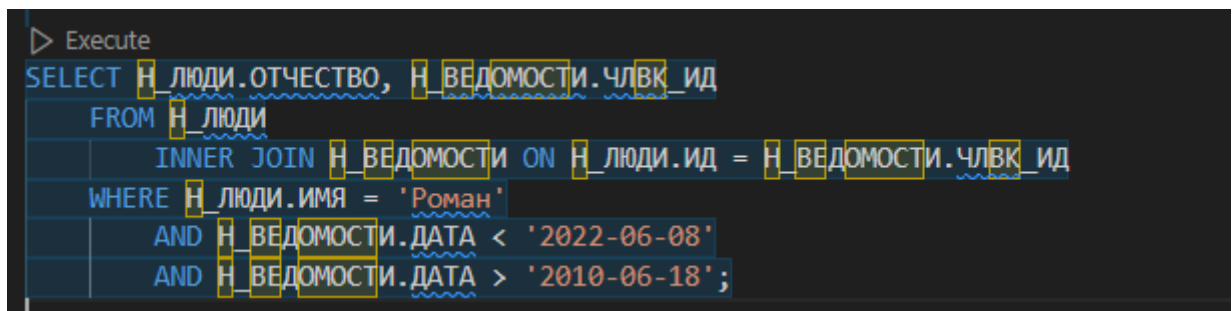
-- Фильтры (AND):

-- а) Н_ЛЮДИ.ИМЯ = Роман.

-- б) Н_ВЕДОМОСТИ.ДАТА < 2022-06-08.

-- с) Н_ВЕДОМОСТИ.ДАТА < 2010-06-18.

-- Вид соединения: INNER JOIN.



```
Execute
SELECT Н_ЛЮДИ.ОТЧЕСТВО, Н_ВЕДОМОСТИ.ЧЛВК_ИД
FROM Н_ЛЮДИ
INNER JOIN Н_ВЕДОМОСТИ ON Н_ЛЮДИ.ИД = Н_ВЕДОМОСТИ.ЧЛВК_ИД
WHERE Н_ЛЮДИ.ИМЯ = 'Роман'
AND Н_ВЕДОМОСТИ.ДАТА < '2022-06-08'
AND Н_ВЕДОМОСТИ.ДАТА > '2010-06-18';
```

Вывод:

ОТЧЕСТВО	ЧЛВК_ИД
Валерьевич	153329
Валерьевич	153329
Александрович	154935
Геннадьевич	142202
Викторович	142391
Алексеевич	142400
Геннадьевич	142202
Викторович	142391
Валерьевич	153329
Валерьевич	153329
Владимирович	144882
Викторович	142391
Геннадьевич	142202
Викторович	142391

-- 2.

-- 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)

-- а) Н_ЛЮДИ.ИД < 100012.

-- б) Н_ОБУЧЕНИЯ.НЗК < 933232.

-- Вид соединения: INNER JOIN.

```
SELECT Н_ЛЮДИ.ФАМИЛИЯ, Н_ОБУЧЕНИЯ.ЧЛВК_ИД, Н_УЧЕНИКИ.ИД
FROM Н_ЛЮДИ
    INNER JOIN Н_УЧЕНИКИ ON Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ИД
    INNER JOIN Н_ОБУЧЕНИЯ ON Н_УЧЕНИКИ.ИД = Н_ОБУЧЕНИЯ.ЧЛВК_ИД
WHERE Н_ЛЮДИ.ИД < '100012'
    AND Н_ОБУЧЕНИЯ.НЗК < '933232';
```

Вывод:

```
ФАМИЛИЯ | ЧЛВК_ИД | ИД
-----+-----+-----
(0 строк)
```

-- 3.

-- Вывести число студентов вечерней формы обучения, которые не имеют отчества.

-- Ответ должен содержать только одно число.

```
SELECT COUNT(*)
FROM Н_люди
JOIN Н_ОБУЧЕНИЯ ON Н_люди.ИД = Н_ОБУЧЕНИЯ.ЧЛВК_ИД
JOIN Н_УЧЕНИКИ ON Н_ОБУЧЕНИЯ.ЧЛВК_ИД = Н_УЧЕНИКИ.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_люди.ОТЧЕСТВО IS NULL
AND Н_ФОРМЫ_ОБУЧЕНИЯ.ИД = 2;
```

Вывод:

```
count
-----
0
(1 строка)
```

-- 4.

-- В таблице Н_ГРУППЫ_ПЛАНОВ найти номера планов, по которым обучается (обучалось) ровно 2 групп на заочной форме обучения.

-- Для реализации использовать подзапрос.

```
SELECT Н_ПЛАНЫ.НОМЕР
FROM Н_ПЛАНЫ
JOIN Н_ФОРМЫ_ОБУЧЕНИЯ ON Н_ПЛАНЫ.ФО_ИД = Н_ФОРМЫ_ОБУЧЕНИЯ.ИД
JOIN Н_ГРУППЫ_ПЛАНОВ ON Н_ПЛАНЫ.ИД = Н_ГРУППЫ_ПЛАНОВ.ПЛАН_ИД
WHERE Н_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ = 'Очно-заочная(вечерняя)'
GROUP BY Н_ПЛАНЫ.НОМЕР HAVING COUNT(Н_ГРУППЫ_ПЛАНОВ.ГРУППА) = 2;
```

Вывод:

НОМЕР

(0 строк)

-- 5.

-- Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср_оценка), у которых средняя оценка не меньше средней оценк(е|и) в группе 1101.

```
SELECT Н_УЧЕНИКИ.ЧЛВК_ИД,  
       Н_ЛЮДИ.ФАМИЛИЯ,  
       Н_ЛЮДИ.ИМЯ,  
       Н_ЛЮДИ.ОТЧЕСТВО,  
       AVG(CAST(Н_ВЕДОМОСТИ.ОЦЕНКА AS DECIMAL)) AS Средняя_оценка  
FROM Н_УЧЕНИКИ  
     JOIN Н_ОБУЧЕНИЯ ON Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ОБУЧЕНИЯ.ЧЛВК_ИД  
     JOIN Н_ЛЮДИ ON Н_ЛЮДИ.ИД = Н_ОБУЧЕНИЯ.ЧЛВК_ИД  
     JOIN Н_ВЕДОМОСТИ ON Н_ЛЮДИ.ИД = Н_ВЕДОМОСТИ.ЧЛВК_ИД  
WHERE Н_ВЕДОМОСТИ.ОЦЕНКА ~ '^[0-9]+$'  
      AND Н_УЧЕНИКИ.ГРУППА = '4100'  
GROUP BY (Н_УЧЕНИКИ.ЧЛВК_ИД, Н_ЛЮДИ.ФАМИЛИЯ, Н_ЛЮДИ.ИМЯ, Н_ЛЮДИ.ОТЧЕСТВО)  
HAVING AVG(CAST(Н_ВЕДОМОСТИ.ОЦЕНКА AS DECIMAL)) >= (SELECT AVG(CAST(Н_ВЕДОМОСТИ.ОЦЕНКА AS DECIMAL))  
                                                         FROM Н_ВЕДОМОСТИ  
                                                         JOIN Н_ЛЮДИ ON Н_ЛЮДИ.ИД = Н_ВЕДОМОСТИ.ЧЛВК_ИД  
                                                         JOIN Н_ОБУЧЕНИЯ ON Н_ЛЮДИ.ИД = Н_ОБУЧЕНИЯ.ЧЛВК_ИД  
                                                         JOIN Н_УЧЕНИКИ ON Н_ОБУЧЕНИЯ.ЧЛВК_ИД = Н_УЧЕНИКИ.ЧЛВК_ИД  
                                                         WHERE ГРУППА = '1101'  
                                                         AND Н_ВЕДОМОСТИ.ОЦЕНКА ~ '^[0-9]+$'))  
;
```

Вывод:

ЧЛВК_ИД	ФАМИЛИЯ	ИМЯ	ОТЧЕСТВО	Средняя_оценка
121863	Петров	Евгений	Владимирович	6.7200000000000000
119495	Шарков	Николай	Александрович	4.2941176470588235
149131	Наумова	Дарья	Александровна	4.0294117647058824
129388	Горбунова	Елена	Павловна	3.9056603773584906
129343	Ступников	Станислав	Владимирович	4.0338983050847458
121043	Семенова	Мария	Александровна	4.2692307692307692
148760	Дымов	Евгений	Александрович	4.4375000000000000
142187	Першенко	Елена	Станиславовна	4.1836734693877551
142156	Завадис	Клара	Евгеньевна	3.9354838709677419
129404	Панасик	Оксана	Андреевна	4.3137254901960784
119497	Шеповалов	Владимир	Николаевич	4.0000000000000000
125633	Смирнов	Сергей	Владимирович	4.5714285714285714
129495	Кукарских	Сергей	Сергеевич	4.6862745098039216
119464	Уфимцева	Елена	Валерьевна	4.3921568627450980
142192	Смирнов	Николай	Викторович	4.4375000000000000
129482	Максимов	Андрей	Николаевич	6.5357142857142857

-- 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.

```

SELECT ГРУППА,
       ЧЛВК_ИД,
       ФАМИЛИЯ,
       ИМЯ,
       ОТЧЕСТВО,
       СОСТОЯНИЕ,
       Н_УЧЕНИКИ.П_ПРКОК_ИД
FROM Н_УЧЕНИКИ
     JOIN Н_ПЛАНЫ ON Н_УЧЕНИКИ.ПЛАН_ИД = Н_ПЛАНЫ.ИД
     JOIN Н_ЛЮДИ ON Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД
     JOIN Н_ФОРМЫ_ОБУЧЕНИЯ ON Н_ПЛАНЫ.ФО_ИД = Н_ФОРМЫ_ОБУЧЕНИЯ.ИД AND Н_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ = 'Очно-заочная(вечерняя)'
     JOIN Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ ON Н_ПЛАНЫ.НАПС_ИД = Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.ИД
     JOIN Н_НАПР_СПЕЦ ON Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.НС_ИД = Н_НАПР_СПЕЦ.ИД AND Н_НАПР_СПЕЦ.КОД_НАПРСПЕЦ = '230101'
WHERE EXISTS (SELECT ИД
              FROM Н_ФОРМЫ_ОБУЧЕНИЯ
              WHERE НАЧАЛО = '2012-09-01'::TIMESTAMP);

```

Вывод:

```

ГРУППА | ЧЛВК_ИД | ФАМИЛИЯ | ИМЯ | ОТЧЕСТВО | СОСТОЯНИЕ | П_ПРКОК_ИД
-----+-----+-----+-----+-----+-----+-----
(0 строк)

```

```

-- 7.
-- Сформировать запрос для получения числа в группе No 3100 отличников.
WITH УЧЕНИКИ_3100 AS
(
    FROM Н_УЧЕНИКИ
    WHERE Н_УЧЕНИКИ.ГРУППА = '3100'
)
SELECT COUNT(*) FROM
(
    SELECT УЧЕНИКИ_3100.ИД FROM УЧЕНИКИ_3100
    JOIN Н_ВЕДОМОСТИ USING(ЧЛВК_ИД)
    WHERE ОЦЕНКА = '5' OR ОЦЕНКА = 'зачет'
    GROUP BY УЧЕНИКИ_3100.ИД
)
AS ОТЛИЧНИКИ WHERE ОТЛИЧНИКИ.ИД NOT IN
(
    SELECT УЧЕНИКИ_3100.ИД FROM УЧЕНИКИ_3100
    JOIN Н_ВЕДОМОСТИ USING(ЧЛВК_ИД)
    WHERE ОЦЕНКА != '5' AND ОЦЕНКА != 'зачет'
    GROUP BY УЧЕНИКИ_3100.ИД
);

```

Вывод:

```

count
-----
0
(1 строка)

```

Выводы по работе:

В ходе выполнения данной лабораторной работы я узнал про view и materialized view, также узнал, что можно последовательность запросов заключать в одну переменную с помощью as и как производить рекурсии в запросах. По сути, это лабораторная расширила мои знания, которые мной были получены в процессе выполнения первой лабораторной работы.