

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО
Факультет Программной инженерии и компьютерных технологий
Направление: Нейротехнологии и программная инженерия

Дисциплина: Базы данных
Лабораторная работа № 2
Вариант 373

Выполнил студент
Рязанов Демид Витальевич
Группа № Р3121

Преподаватель: Королёва Юлия Александровна

г. Санкт-Петербург

2023

Текст задания

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.ДАТА.
Фильтры (AND):
а) Н_ТИПЫ_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ = Экзаменационный лист.
б) Н_ВЕДОМОСТИ.ИД = 1457443.
с) Н_ВЕДОМОСТИ.ИД < 1250972.
Вид соединения: RIGHT JOIN.
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.ИМЯ > Николай.
б) Н_ОБУЧЕНИЯ.НЗК > 933232.
Вид соединения: INNER JOIN.
3. Вывести число студентов ФКТИУ, которые старше 25 лет.
Ответ должен содержать только одно число.
4. В таблице Н_ГРУППЫ_ПЛАНОВ найти номера планов, по которым обучается (обучалось) ровно 2 групп на кафедре вычислительной техники.
Для реализации использовать подзапрос.
5. Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср_оценка), у которых средняя оценка не больше средней оценк(е|и) в группе 1101.
6. Получить список студентов, отчисленных после первого сентября 2012 года с очной формы обучения (специальность: Программная инженерия). В результат включить:
номер группы;
номер, фамилию, имя и отчество студента;
номер пункта приказа;
Для реализации использовать соединение таблиц.
7. Сформировать запрос для получения числа на ФКТИУ троечников.

Реализация запросов

1.

```
SELECT TYPES.НАИМЕНОВАНИЕ, VED.ДАТА FROM  
H_ТИПЫ_ВЕДОМОСТЕЙ AS TYPES  
RIGHT JOIN H_ВЕДОМОСТИ AS VED  
ON VED.ВЕД_ИД = TYPES.ИД  
WHERE TYPES.НАИМЕНОВАНИЕ = 'Экзаменационный лист'  
AND VED.ИД = 1457443  
AND VED.ИД < 1250972;
```

2.

```
SELECT P.ФАМИЛИЯ, EDU.ЧЛВК_ИД, ST.ИД  
FROM H_ЛЮДИ AS P  
JOIN H_ОБУЧЕНИЯ AS EDU  
ON EDU.ЧЛВК_ИД = P.ИД  
JOIN H_УЧЕНИКИ AS ST  
USING (ЧЛВК_ИД, ВИД_ОБУЧ_ИД)  
WHERE P.ИМЯ > 'Николай'  
AND EDU.НЗК::int > 933232;
```

3.

```
SELECT COUNT(*)  
FROM H_ОТДЕЛЫ  
JOIN H_ПЛАНЫ  
ON H_ПЛАНЫ.ОТД_ИД = H_ОТДЕЛЫ.ИД  
JOIN H_УЧЕНИКИ  
ON H_ПЛАНЫ.ИД = H_УЧЕНИКИ.ПЛАН_ИД  
JOIN H_ЛЮДИ  
ON H_УЧЕНИКИ.ЧЛВК_ИД = H_ЛЮДИ.ИД  
WHERE H_ОТДЕЛЫ.КОРОТКОЕ_ИМЯ = 'КтиУ'  
AND H_УЧЕНИКИ.КОНЕЦ > now()  
AND H_УЧЕНИКИ.СОСТОЯНИЕ NOT IN ('отчислен')
```

AND date_part('year', age(Н_ЛЮДИ.ДАТА_РОЖДЕНИЯ)) > 25;

4.

```
SELECT HOMER
FROM (
    SELECT Н_ПЛАНЫ.ИД AS HOMER, COUNT(*) AS КОЛВО_ГРУПП,
    ОТД_ИД_ЗАКРЕПЛЕН_ЗА AS ОТД
    FROM Н_ГРУППЫ_ПЛАНОВ
    JOIN Н_ПЛАНЫ
    ON Н_ПЛАНЫ.ИД = Н_ГРУППЫ_ПЛАНОВ.ПЛАН_ИД
    GROUP BY (Н_ПЛАНЫ.ИД, ОТД_ИД_ЗАКРЕПЛЕН_ЗА)
) AS ПЛАНЫ
WHERE ОТД = 102
AND КОЛВО_ГРУПП = 2;
```

5.

```
WITH G AS (
    SELECT Н_ОБУЧЕНИЯ.НЗК, CONCAT (Н_ЛЮДИ.ФАМИЛИЯ, ' ', Н_ЛЮДИ.ИМЯ,
    ' ', Н_ЛЮДИ.ОТЧЕСТВО) AS ФИО, ВЕД.ЧЛВК_ИД, ВЕД.ОЦЕНКА,
    Н_УЧЕНИКИ.ГРУППА
    FROM (
        SELECT ЧЛВК_ИД, AVG(ОЦЕНКА) AS ОЦЕНКА
        FROM (
            SELECT ЧЛВК_ИД, (ОЦЕНКА::int) AS ОЦЕНКА
            FROM Н_ВЕДОМОСТИ
            WHERE ОЦЕНКА NOT IN ('осв', 'неявка', 'незач', 'зачет', '99')
        ) AS ВЕД GROUP BY (ЧЛВК_ИД)
    ) AS ВЕД
    JOIN Н_УЧЕНИКИ
    ON Н_УЧЕНИКИ.ЧЛВК_ИД = ВЕД.ЧЛВК_ИД
    JOIN Н_ОБУЧЕНИЯ
    ON Н_ОБУЧЕНИЯ.ЧЛВК_ИД = ВЕД.ЧЛВК_ИД
    JOIN Н_ЛЮДИ
```

```

        ON Н_ЛЮДИ.ИД = ВЕД.ЧЛВК_ИД
    )
    SELECT НЗК, ФИО, ОЦЕНКА
    FROM G
    WHERE ГРУППА = '4100'
    AND ОЦЕНКА <= (
        SELECT AVG(ОЦЕНКА)
        FROM G
        WHERE ГРУППА = '1101'
    );

```

6.

```

SELECT Н_УЧЕНИКИ.ГРУППА, CONCAT(Н_ЛЮДИ.ФАМИЛИЯ, ' ', Н_ЛЮДИ.ИМЯ, ' ',
Н_ЛЮДИ.ОТЧЕСТВО) AS ФИО
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
WHERE Н_УЧЕНИКИ.ПРИЗНАК = 'отчисл'
AND Н_УЧЕНИКИ.КОНЕЦ > '2012-09-01'
AND Н_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ IN ('Очная');

```

7.

```

SELECT Н_ВЕДОМОСТИ.ЧЛВК_ИД, CONCAT(Н_ЛЮДИ.ФАМИЛИЯ, ' ',
Н_ЛЮДИ.ИМЯ, ' ', Н_ЛЮДИ.ОТЧЕСТВО)
FROM Н_ВЕДОМОСТИ
JOIN Н_ЛЮДИ
ON Н_ЛЮДИ.ИД = Н_ВЕДОМОСТИ.ЧЛВК_ИД
JOIN Н_УЧЕНИКИ

```

```

ON Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД
JOIN Н_ПЛАНЫ
ON Н_ПЛАНЫ.ИД = Н_УЧЕНИКИ.ПЛАН_ИД
JOIN Н_ОТДЕЛЫ
ON Н_ОТДЕЛЫ.ИД = Н_ПЛАНЫ.ОТД_ИД
WHERE Н_ОТДЕЛЫ.КОРОТКОЕ_ИМЯ = 'КТиУ'
AND Н_ВЕДОМОСТИ.ОЦЕНКА = '3'
GROUP BY Н_ВЕДОМОСТИ.ЧЛВК_ИД, Н_ЛЮДИ.ФАМИЛИЯ, Н_ЛЮДИ.ИМЯ,
Н_ЛЮДИ.ОТЧЕСТВО;

```

Доп задания

1) Иерархия отделов

```

WITH RECURSIVE TEMP (ИД, ОТД_ИД, КОРОТКОЕ_ИМЯ, ПУТЬ) AS (
    SELECT T1.ИД, T1.ОТД_ИД, T1.КОРОТКОЕ_ИМЯ, CAST (T1.ИД AS
    VARCHAR(50)) AS ПУТЬ
    FROM Н_ОТДЕЛЫ AS T1
    WHERE T1.ОТД_ИД IS NULL
    UNION
    SELECT T2.ИД, T2.ОТД_ИД, T2.КОРОТКОЕ_ИМЯ, CAST (TEMP.ПУТЬ || ' -> ' ||
    T2.ИД AS VARCHAR(50))
    FROM Н_ОТДЕЛЫ AS T2
    JOIN TEMP ON TEMP.ИД = T2.ОТД_ИД
)
SELECT КОРОТКОЕ_ИМЯ, ПУТЬ
FROM TEMP;

```

2)

```

SELECT УЧ.ГРУППА,
CONCAT(Л.ФАМИЛИЯ, ' ', Л.ИМЯ, ' ', Л.ОТЧЕСТВО) AS ФИО, Д.КОРОТКОЕ_ИМЯ,
COALESCE(SUM(ВЕД.ОЦЕНКА::INT)
OVER (PARTITION BY Л.ИД ROWS BETWEEN UNBOUNDED PRECEDING AND
CURRENT ROW), 0) AS TOTAL
FROM

```

```

Н_ДИСЦИПЛИНЫ AS Д
JOIN Н_СТРОКИ_ПЛАНОВ AS T1
ON T1.ДИС_ИД=Д.ИД
JOIN Н_ЭЛЕМЕНТЫ_СТРОК AS T2
ON T2.СПЛ_ИД=T1.ИД
JOIN Н_СОДЕРЖАНИЯ_ЭЛЕМЕНТОВ_СТРОК AS T3
ON T3.ЭСТ_ИД=T2.ИД
JOIN Н_ВЕДОМОСТИ AS ВЕД
ON ВЕД.СЭС_ИД=T3.ИД
JOIN Н_ЛЮДИ AS Л
ON Л.ИД=ВЕД.ЧЛВК_ИД
JOIN Н_УЧЕНИКИ AS УЧ
ON УЧ.ЧЛВК_ИД=Л.ИД
WHERE ВЕД.ОЦЕНКА IN ('2', '3', '4', '5')
group by УЧ.ГРУППА, Д.КОРОТКОЕ_ИМЯ, ВЕД.ОЦЕНКА, Л.ИД, Л.ФАМИЛИЯ, Л.ИМЯ,
Л.ОТЧЕСТВО;

```

Вывод

В ходе выполнения данной лабораторной работы я научился реализовывать запросы к базе данных, с использованием соединения таблиц, подзапросов.