

Составить запросы на языке SQL:

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

Таблицы: Н_ЛЮДИ, Н_ВЕДОМОСТИ.

Вывести атрибуты: Н_ЛЮДИ.ИД, Н_ВЕДОМОСТИ.ИД.

Фильтры (AND):

а) Н_ЛЮДИ.ФАМИЛИЯ > Иванов.

б) Н_ВЕДОМОСТИ.ЧЛВК_ИД > 153285.

Вид соединения: RIGHT JOIN.

```
SELECT Н_ЛЮДИ.ИД AS ЛЮДИ_ИД, Н_ВЕДОМОСТИ.ИД AS ВЕДОМОСТИ_ИД
FROM Н_ЛЮДИ
RIGHT JOIN Н_ВЕДОМОСТИ ON Н_ЛЮДИ.ИД = Н_ВЕДОМОСТИ.ЧЛВК_ИД
WHERE Н_ЛЮДИ.ФАМИЛИЯ > 'Иванов' AND Н_ВЕДОМОСТИ.ЧЛВК_ИД > 153285;
```

	ЛЮДИ_ИД integer	ВЕДОМОСТИ_ИД integer
1	153942	1210783
2	153355	1218044
3	153344	1218045
4	153412	1209261
5	153411	1209262
6	153409	1209263
7	153408	1209264
8	153366	1209265
9	153316	1209266
10	153313	1209267
11	153312	1209268
12	153311	1209269
13	153310	1209270
14	153308	1209271
15	153305	1209272
16	153303	1209273
Total rows: 7051 of 7051		Query comp

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.ФАМИЛИЯ, Н_ВЕДОМОСТИ.ИД, Н_СЕССИЯ.УЧГОД.

Фильтры (AND):

- а) Н_ЛЮДИ.ИМЯ < Николай.
 б) Н_ВЕДОМОСТИ.ИД < 1250972.
 с) Н_СЕССИЯ.ДАТА < 2004-01-17.
 Вид соединения: RIGHT JOIN.

```
SELECT Н_ЛЮДИ.ФАМИЛИЯ, Н_ВЕДОМОСТИ.ИД, Н_СЕССИЯ.УЧГОД
FROM Н_ЛЮДИ
RIGHT JOIN Н_ВЕДОМОСТИ ON Н_ЛЮДИ.ИД = Н_ВЕДОМОСТИ.ЧЛВК_ИД
RIGHT JOIN Н_СЕССИЯ ON Н_ВЕДОМОСТИ.ЧЛВК_ИД = Н_СЕССИЯ.ЧЛВК_ИД
WHERE Н_ЛЮДИ.ИМЯ < 'Николай' AND Н_ВЕДОМОСТИ.ИД < 1250972 AND
Н_СЕССИЯ.ДАТА < '2004-01-17';
```

	ФАМИЛИЯ character varying (25) 🔒	ИД integer 🔒	УЧГОД character varying (9) 🔒
1	Оголюк	88677	2002/2003
2	Оголюк	88678	2002/2003
3	Оголюк	88679	2002/2003
Total rows: 70 of 70		Query complete 00:00:00.106	

3. Составить запрос, который ответит на вопрос, есть ли среди студентов вечерней формы обучения те, кто не имеет отчества.

```
SELECT * FROM
(Н_ЛЮДИ INNER JOIN Н_УЧЕНИКИ ON Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД) AS СПИСОК
WHERE СПИСОК.ОТЧЕСТВО IS NULL AND СПИСОК.ВИД_ОБУЧ_ИД = 2;
```

ИД integer 🔒

Total rows: 0 of 0

4. Найти группы, в которых в 2011 году было менее 10 обучающихся студентов на ФКТИУ.

Для реализации использовать соединение таблиц.

```
WITH ЛИСТ AS
(
    SELECT ГРУППА, COUNT(ЧЛВК_ИД) AS ВСЕГО_СТУДЕНТОВ FROM Н_УЧЕНИКИ
        INNER JOIN Н_ПЛАНЫ ON Н_УЧЕНИКИ.ПЛАН_ИД = Н_ПЛАНЫ.ИД
        INNER JOIN Н_ОТДЕЛЫ ON Н_ОТДЕЛЫ.ИД = Н_ПЛАНЫ.ОТД_ИД
    WHERE Н_ОТДЕЛЫ.КОРОТКОЕ_ИМЯ = 'КТИУ' AND НАЧАЛО = '2010-09-01' AND КОНЕЦ
    = '2011-06-29' GROUP BY ГРУППА)
SELECT * FROM ЛИСТ WHERE ВСЕГО_СТУДЕНТОВ < 10
```

	ГРУППА character varying (4)	ВСЕГО_СТУДЕНТОВ bigint
1	4125	7
2	6113	7
3	6114	2
Total rows: 3 of 3 Query complete 00:00:00.133		

5. Выведите таблицу со средним возрастом студентов во всех группах (Группа, Средний возраст), где средний возраст меньше максимального возраста в группе 3100.

```
WITH КОНТРОЛЬНАЯ_ГРУППА AS (
    SELECT Н_УЧЕНИКИ.ГРУППА, EXTRACT(YEAR FROM AGE(ДАТА_РОЖДЕНИЯ)) AS ВОЗРАСТ
    FROM Н_УЧЕНИКИ
        INNER JOIN Н_ЛЮДИ ON Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД
    WHERE Н_УЧЕНИКИ.ГРУППА = '3100'
)
SELECT Н_УЧЕНИКИ.ГРУППА, AVG(EXTRACT(YEAR FROM AGE(ДАТА_РОЖДЕНИЯ))) AS
СРЕДНИЙ_ВОЗРАСТ
FROM Н_УЧЕНИКИ
        INNER JOIN Н_ЛЮДИ ON Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД
GROUP BY Н_УЧЕНИКИ.ГРУППА
HAVING AVG(EXTRACT(YEAR FROM AGE(ДАТА_РОЖДЕНИЯ))) < (SELECT MAX(ВОЗРАСТ)
FROM КОНТРОЛЬНАЯ_ГРУППА)
```

	ГРУППА character varying (4) 🔒	СРЕДНИЙ_ВОЗРАСТ numeric 🔒
1	354	25.7761904761904762
2	2508	33.8850574712643678
3	5509	35.5454545454545455
4	551	24.5428571428571429
5	5114	29.1250000000000000
6	395	25.4585987261146497
7	250	31.7175925925925926
8	3104	31.2168674698795181
9	219	36.1285714285714286
Total rows: 162 of 162		Query complete 00:00:00

6. Получить список студентов, зачисленных ровно 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 подзапрос с EXISTS.

```

WITH СПИСОК AS (
SELECT ГРУППА, ЧЛВК_ИД, ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО, Н_УЧЕНИКИ.ПЛАН_ИД,
Н_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ, НАЧАЛО, Н_ПЛАНЫ.КУРС FROM Н_УЧЕНИКИ
JOIN Н_ПЛАНЫ ON Н_УЧЕНИКИ.ПЛАН_ИД = Н_ПЛАНЫ.ПЛАН_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
)
SELECT ГРУППА, ЧЛВК_ИД, ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО, ПЛАН_ИД, НАЧАЛО, КУРС FROM
СПИСОК
WHERE EXISTS (SELECT 1 FROM Н_ФОРМЫ_ОБУЧЕНИЯ
WHERE
Н_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ=СПИСОК.НАИМЕНОВАНИЕ AND
(Н_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ='Очная' OR
Н_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ='Заочная') AND
НАЧАЛО='2011-09-01' AND
КУРС=1
)

```


???

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

```

SELECT люди.ИД FROM Н_люди AS люди, Н_ученики WHERE люди.ИД=Н_уче-
ники.ЧЛВК_ИД AND EXISTS (
SELECT 1 FROM Н_люди
JOIN Н_ученики ON Н_люди.ИД = Н_ученики.ЧЛВК_ИД
WHERE Н_люди.ИД != люди.ИД AND люди.ОТЧЕСТВО=Н_люди.ОТЧЕСТВО)

```

	ИД integer 
1	130685
2	130665
3	153820
4	130664
5	153821
6	153822
7	153823
8	153824
9	114092
Total rows: 1000 of 4850	