

Санкт-Петербургский Национальный Исследовательский Университет
Информационных Технологий, Механики и Оптики

Лабораторные работы по Системам баз данных на тему
"Запросы SQL"

Выполнил студент
Группы 3121
Шестель Владислав

Простая выборка.

1. Выдать содержимое всех столбцов таблицы Н_ЦИКЛЫ_ДИСЦИПЛИН

```
select * from Н_ЦИКЛЫ_ДИСЦИПЛИН;
```

2. Выдать содержимое столбцов АББРЕВИАТУРА и НАИМЕНОВАНИЕ той же таблицы.

```
select АББРЕВИАТУРА, НАИМЕНОВАНИЕ from Н_ЦИКЛЫ_ДИСЦИПЛИН;
```

3. Получить перечень квалификаций, присваиваемых выпускникам нашего университета.

```
select * from Н_КВАЛИФИКАЦИИ;
```

Исключение дубликатов.

4. Выдать неповторяющиеся имена людей из таблицы Н_ЛЮДИ.

```
select distinct ИМЯ from Н_ЛЮДИ;
```

5. Какие состояния студентов (признаки) используются в таблице Н_УЧЕНИКИ.

```
select distinct ПРИЗНАК from Н_УЧЕНИКИ;
```

Выборка вычисляемых значений.

6. Из таблицы Н_УЧЕБНЫЕ_ГОДА получить названия и продолжительность каждого учебного года (напомним, что единица измерения дат - одни сутки).

```
select УЧЕБНЫЙ_ГОД, (КОНЕЦ - НАЧАЛО) from Н_УЧЕБНЫЕ_ГОДА;
```

7. Используя сведения из таблицы Н_ИЗМ_ЛЮДИ, определить:

а) целое число дней, прошедших с момента изменения фамилии до текущей даты (SYSDATE);

```
select round(SYSDATE - КОГДА_ИЗМЕНИЛ) from Н_ИЗМ_ЛЮДИ;
```

б) целое число месяцев, прошедших с момента изменения фамилии до текущей даты.

```
select round(months_between(SYSDATE, КОГДА_ИЗМЕНИЛ)) from Н_ИЗМ_ЛЮДИ;
```

Выборка с использованием фразы WHERE и упорядочением.

8. Выдать фамилию и инициалы людей (получить в результате один столбец)

```
select ФАМИЛИЯ || ' ' || substr(ИМЯ, 0, 1) || '.' || substr(ОТЧЕСТВО, 0, 1) || '.' from Н_ЛЮДИ where ROWNUM <= 50;
```

9. Выдать номер, фамилию и инициалы людей (оформить в виде "104567 Сидоров С.С.").

```
select TO_CHAR(ИД) || ' ' || ФАМИЛИЯ || ' ' || substr(ИМЯ, 0, 1) || '.' || substr(ОТЧЕСТВО, 0, 1) || '.' from Н_ЛЮДИ where ROWNUM <= 50;
```

10. Выдать номер (ИД) Короткое_имя и название (имя_в_имин_падеже) всех отделов факультета Компьютерных технологий и управления.

```
select ИД, КОРОТКОЕ_ИМЯ, ИМЯ_В_ИМИН_ПАДЕЖЕ from Н_ОТДЕЛЫ where ОТД_ИД = (select ИД from Н_ОТДЕЛЫ where КОРОТКОЕ_ИМЯ='КТИУ');
```

11. Выдать "Фамилия, Имя, Отчество" всех людей, фамилии которых начинаются на "Яков".

```
select ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО from Н_ЛЮДИ where substr(ФАМИЛИЯ, 0, 4) = 'Яков';
```

12. Выдать "Фамилия, Имя, Отчество" всех людей из запроса номер 11, отсортировав строки по:

1) имени: select ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО from Н_ЛЮДИ where substr(ФАМИЛИЯ, 0, 4) = 'Яков' order by ИМЯ;

2) имени и отчеству: select ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО from Н_ЛЮДИ where substr(ФАМИЛИЯ, 0, 4) = 'Яков' order by ИМЯ, ОТЧЕСТВО;

3) фамилии, имени и отчеству:

```
select ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО from Н_ЛЮДИ where substr(ФАМИЛИЯ, 0, 4) = 'Яков' order by ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО;
```

13. Повторить предыдущие запросы задавая сортировку позициями, а не именами столбцов.

13.1. `select ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО from Н_ЛЮДИ where substr(ФАМИЛИЯ, 0, 4) = 'Яков' order by 2;`

13.2. `select ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО from Н_ЛЮДИ where substr(ФАМИЛИЯ, 0, 4) = 'Яков' order by 2, 3;`

13.3. `select ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО from Н_ЛЮДИ where substr(ФАМИЛИЯ, 0, 4) = 'Яков' order by 1, 2, 3;`

14. Выдать " фамилия, имя, отчество" всех людей с фамилиями, начинающимися на "Э", "Ю" и "Я", с упорядочением по фамилии (составить не менее двух вариантов запроса).

`select ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО from Н_ЛЮДИ where substr(ФАМИЛИЯ, 0, 1) in ('Э', 'Ю', 'Я') order by ФАМИЛИЯ;`

`select ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО from Н_ЛЮДИ where ФАМИЛИЯ like 'Э%' OR ФАМИЛИЯ like 'Ю%' OR ФАМИЛИЯ like 'Я%' order by 1;`

15. Выдать номера, короткие имена и названия отделов с номерами 102, 111 и 212.

`select ИД, КОРОТКОЕ_ИМЯ, ИМЯ_В_ИМИН_ПАДЕЖЕ from Н_ОТДЕЛЫ where ИД in (102, 111, 212);`

16. Получить из таблицы `н_напр_спец` код и наименование специализаций специальности 220100

(напомним, что номера специализаций отличаются от номера специальности двумя последними цифрами).

`select КОД_НАПРСПЕЦ, НАИМЕНОВАНИЕ from Н_НАПР_СПЕЦ where КОД_НАПРСПЕЦ like '2201__';`

17. По таблице `Н_УЧЕНИКИ` познакомиться с особенностями процесса обучения студентов с номерами 114052,114746,115264,116483,117127,118705

`select DISTINCT Человек(ЧЛВК_ИД, 'И'), СОСТОЯНИЕ, НАЧАЛО, КОНЕЦ from Н_УЧЕНИКИ where ЧЛВК_ИД in ('114052', '114746', '115264', '116483', '117127', '118705') order by 1;`

18. Вывести из таблицы `Н_ВЕДОМОСТЬ` ваши оценки.

`select ОЦЕНКА, DECODE(ОЦЕНКА, '5', 'отлично', '4', 'хорошо', '3', 'удовлетворительно', '2', 'неудовлетворительно', 'зачет', 'зачет', 'незачет', 'незачет', 'осв', 'освобождение', 'неявка', 'неявка', '99', 'диплом с отличием', '.', 'выдача экзаменационного листа', 'отсутствие данных') AS Эквивалент, ДАТА, СЭС_ИД, (select ИМЯ_ДИСЦ from Н_СОДЕРЖИМОЕ_ЭЛЕМЕНТОВ_СТРОК JOIN Н_ВЕДОМОСТЬ USING (СЭС_ИД)) as НАЗВАНИЕ from Н_ВЕДОМОСТЬ where ЧЛВК_ИД = '162866' order by СЭС_ИД, ДАТА;`

19. Вывести список тех ваших оценок, которые заданы цифрами 5, 4, 3, 2. Список должен содержать оценку, её удвоенный цифровой эквивалент (псевдоним - Оценка*2), дату получения оценки и ссылку (сэс_ид) на идентификатор таблицы `Н_СОДЕРЖИМОЕ_ЭЛЕМЕНТОВ_СТРОК`. Зная сэс_ид можно получить имя дисциплины, по которой получена оценка. Список упорядочить по сэс_ид и дате.

`select ОЦЕНКА, TO_NUMBER(ОЦЕНКА) * 2 AS "Оценка*2", ДАТА, СЭС_ИД, (select ИМЯ_ДИСЦ from Н_СОДЕРЖИМОЕ_ЭЛЕМЕНТОВ_СТРОК JOIN Н_ВЕДОМОСТЬ USING (СЭС_ИД)) as Название from Н_ВЕДОМОСТЬ where ОЦЕНКА IN ('5', '4', '3', '2') AND ЧЛВК_ИД = '162866' order by СЭС_ИД, ДАТА;`

20. Выполнить запрос 4, удалив из результата пробелы и точки (отсутствие имени у некоторых студентов), а также однобуквенные имена, получившиеся при вводе инициалов

`select distinct ИМЯ from Н_ЛЮДИ where Length(ИМЯ) > 1 AND ИМЯ NOT LIKE '._';`

21. Сколько "Алексеев" в таблице `Н_люди`?

`select "Алексей - " || count(ИМЯ) from Н_ЛЮДИ where ИМЯ = "Алексей";`

22. Определить количество различных имен людей в таблице `Н_ЛЮДИ`, удалив из результата пробелы, точки и однобуквенные имена

`select count(distinct ИМЯ) from Н_ЛЮДИ where Length(ИМЯ) > 1 AND ИМЯ NOT LIKE '._';`

23. Определите вашу среднюю оценку. Создайте два запроса, в которых средняя оценки определяется с помощью функции среднее значение (функция `AVG`) и путем деления суммы (функция `SUM`) оценок на их количество (функция `COUNT`). В результате необходимо оставить два десятичных знака после запятой.

`select ROUND(AVG(ОЦЕНКА), 2) from Н_ВЕДОМОСТЬ where ОЦЕНКА IN ('5', '4', '3', '2') AND ЧЛВК_ИД = '162866';`

`select ROUND(SUM(ОЦЕНКА) / COUNT(ОЦЕНКА), 2) from Н_ВЕДОМОСТЬ where ОЦЕНКА IN ('5', '4', '3', '2') AND ЧЛВК_ИД = '162866';`

24. Сколько всего людей с фамилией Иванов и сколько у них различных имен и различных отчеств. Для получения результата использовать один запрос со следующими псевдонимами столбцов: Всего, Разных_имен, Разных_отчеств. В результат не должны включаться имена (отчества) в виде пробелов, точек и однобуквенных имен

```
select COUNT(Фамилия) as Всего, COUNT(distinct ИМЯ) as Разных_имен, COUNT(distinct ОТЧЕСТВО) as Разных_отчеств from Н_ЛЮДИ where Фамилия = 'Иванов' AND Length(ИМЯ) > 1 AND ИМЯ NOT LIKE '_' AND length(Отчество) > 1 AND Отчество NOT LIKE '.';
```

25. Выдать различные

а) фамилии людей и число людей с каждой из этих фамилий, ограничив список фамилиями, встречающимися не менее 50 раз

```
select ФАМИЛИЯ, count(ФАМИЛИЯ) from Н_ЛЮДИ group by ФАМИЛИЯ having count(ФАМИЛИЯ) >= 50 order by ФАМИЛИЯ desc;
```

б) имена людей и число людей с каждым из этих имен, ограничив список именами, встречающимися не менее 300 раз

```
select ИМЯ, count(ИМЯ) from Н_ЛЮДИ group by ИМЯ having count(ИМЯ) >= 300 order by ИМЯ desc;
```

в) отчества людей и число людей с каждым из этих отчеств, ограничив список отчествами, встречающимися не менее 300 раз

```
select ОТЧЕСТВО, count(ОТЧЕСТВО) from Н_ЛЮДИ group by ОТЧЕСТВО having count(ОТЧЕСТВО) >= 300 order by ОТЧЕСТВО desc;
```

26. Найти группы, в которых 4.7.2004 было менее десяти обучающихся студентов

```
select ГРУППА, count(ИД) from Н_УЧЕНИКИ where НАЧАЛО < to_date('04.07.2004') and КОНЕЦ > to_date('04.07.2004') group by ГРУППА having count(ИД) < 10 order by count(ИД);
```

27. В таблице Н_ГРУППЫ_ПЛАНОВ найти номера планов, по которым обучается (обучалось) более 4 групп.

```
select ПЛАН_ИД from Н_ГРУППЫ_ПЛАНОВ group by ПЛАН_ИД having count(ГРУППА) > 4;
```

28. Используя опыт, полученный при реализации запроса 23, выведите таблицу со средними оценками студентов вашей группы (Номер, ФИО, Ср_оценка).

```
select Н_УЧЕНИКИ.ЧЛВК_ИД as Номер, Человек(Н_УЧЕНИКИ.ЧЛВК_ИД, 'И') as ФИО, ROUND(AVG(Н_ВЕДОМОСТИ.ОЦЕНКА), 2) as Ср_оценка from Н_УЧЕНИКИ, Н_ВЕДОМОСТИ where Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ВЕДОМОСТИ.ЧЛВК_ИД and ГРУППА = '3121' and НАЧАЛО < SYSDATE and КОНЕЦ > SYSDATE and ОЦЕНКА in ('2', '3', '4', '5') group by Н_УЧЕНИКИ.ЧЛВК_ИД;
```

29. Выполнить предыдущий запрос используя для получения номера, фамилии, имени и отчества студента не функцию Человек, а текстовую строку, полученную из таблицы Н_ЛЮДИ примерно так, как формировался запрос 8.

```
select Н_УЧЕНИКИ.ЧЛВК_ИД as Номер, Н_ЛЮДИ.ФАМИЛИЯ||' '||Н_ЛЮДИ.ИМЯ||' '||Н_ЛЮДИ.ОТЧЕСТВО||' ' AS ФИО, ROUND(AVG(TO_NUMBER(Н_ВЕДОМОСТИ.ОЦЕНКА)),1) AS Ср_оценка from Н_УЧЕНИКИ, Н_ВЕДОМОСТИ, Н_ЛЮДИ where Н_УЧЕНИКИ.ЧЛВК_ИД = Н_ВЕДОМОСТИ.ЧЛВК_ИД and НАЧАЛО < SYSDATE and КОНЕЦ > SYSDATE and ГРУППА = '3121' and ОЦЕНКА in ('2','3','4','5') group by Н_УЧЕНИКИ.ЧЛВК_ИД, Н_ЛЮДИ.ФАМИЛИЯ, Н_ЛЮДИ.ИМЯ, Н_ЛЮДИ.ОТЧЕСТВО;
```

30. Составить запрос, позволяющий получить таблицу направлений (специальностей и специализаций), имеющую следующий вид:

```
select Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.ИД, Н_НАПРАВЛЕНИЕ_СПЕЦИАЛ.УРОВЕНЬ, Н_НАПРАВЛЕНИЕ_СПЕЦИАЛ.ДАТА_ГОС, Н_НАПР_СПЕЦ.КОД_НАПРСПЕЦ ||' '|| Н_НАПР_СПЕЦ.НАИМЕНОВАНИЕ as 'Направление/Специальность', Н_ТИПЫ_СТАНДАРТОВ.НАИМЕНОВАНИЕ as 'ТИП СТАНДАРТА' from Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ join Н_НАПР_СПЕЦ on Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.НС_ИД = Н_НАПР_СПЕЦ.ИД join Н_ТИПЫ_СТАНДАРТОВ on Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.ТС_ИД = Н_ТИПЫ_СТАНДАРТОВ.ИД;
```

31. Преобразовать предыдущий запрос так, чтобы в получаемой таблице перед столбцом «Тип стандарта» появился столбец «Квалификация»

```
select Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.ИД, Н_НАПРАВЛЕНИЕ_СПЕЦИАЛ.УРОВЕНЬ, Н_НАПРАВЛЕНИЕ_СПЕЦИАЛ.ДАТА_ГОС, Н_НАПР_СПЕЦ.КОД_НАПРСПЕЦ ||' '|| Н_НАПР_СПЕЦ.НАИМЕНОВАНИЕ as 'Направление/Специальность', Н_КВАЛИФИКАЦИИ.НАИМЕНОВАНИЕ as Квалификация, Н_ТИПЫ_СТАНДАРТОВ.НАИМЕНОВАНИЕ as 'Тип стандарта' from Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ join Н_НАПР_СПЕЦ on Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.НС_ИД = Н_НАПР_СПЕЦ.ИД join Н_ТИПЫ_СТАНДАРТОВ on Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.ТС_ИД = Н_ТИПЫ_СТАНДАРТОВ.ИД join Н_КВАЛИФИКАЦИИ on ВАЛИФИКАЦИИ.ИД = Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.КВАЛ_ИД;
```

32. Получить список студентов, зачисленных первого сентября позапрошлого учебного года на первый курс очной формы обучения специальности 220100. В результат включить: номер группы, номер, фамилию, имя и отчество студента, номер и состояние пункта приказа, признак, характеризующий состояние студента, дату конца действия этого пункта, Результат упорядочить по номеру группы и фамилии.

```
select Н_УЧЕНИКИ.ГРУППА, Н_ЛЮДИ.ИД, Н_ЛЮДИ.ФАМИЛИЯ, Н_ЛЮДИ.ИМЯ, Н_ЛЮДИ.ОТЧЕСТВО, Н_УЧЕНИКИ.П_ПРКОК_ПРИКАЗ,
Н_УЧЕНИКИ.СОСТОЯНИЕ, Н_УЧЕНИКИ.ПРИЗНАК, Н_УЧЕНИКИ.КОНЕЦ from Н_УЧЕНИКИ, Н_ЛЮДИ, Н_ПЛАНЫ, Н_НАПР_СПЕЦ,
Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ, Н_ФОРМЫ_ОБУЧЕНИЯ where Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД and Н_ПЛАНЫ.ИД = Н_УЧЕНИКИ.ПЛАН_ИД
and Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.ИД = Н_НАПР_СПЕЦ.ИД and Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.ИД = Н_ПЛАНЫ.ИД and Н_ПЛАНЫ.ФО_ИД =
Н_ФОРМЫ_ОБУЧЕНИЯ.ИД and upper(Н_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ) = 'ОЧНАЯ' and Н_ПЛАНЫ.КУРС = 1 and
Н_ПЛАНЫ.УЧЕБНЫЙ_ГОД = '2011/2012' and to_char(Н_НАПР_СПЕЦ.КОД_НАПРСПЕЦ) = '220100' and
to_char(Н_УЧЕНИКИ.НАЧАЛО)='1.9.2011' order by Н_УЧЕНИКИ.ГРУППА, Н_ЛЮДИ.ФАМИЛИЯ;
```

33. Получить список студентов, по условиям п. 32, сохранив в нем только те строки, в которых состояние пунктами приказа равно 'утвержден'

```
select Н_УЧЕНИКИ.ГРУППА, Н_ЛЮДИ.ИД, Н_ЛЮДИ.ФАМИЛИЯ, Н_ЛЮДИ.ИМЯ, Н_ЛЮДИ.ОТЧЕСТВО, Н_УЧЕНИКИ.П_ПРКОК_ПРИКАЗ,
Н_УЧЕНИКИ.СОСТОЯНИЕ, Н_УЧЕНИКИ.ПРИЗНАК, Н_УЧЕНИКИ.КОНЕЦ from Н_УЧЕНИКИ, Н_ЛЮДИ, Н_ПЛАНЫ, Н_НАПР_СПЕЦ,
Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ, Н_ФОРМЫ_ОБУЧЕНИЯ where Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД and Н_ПЛАНЫ.ИД = Н_УЧЕНИКИ.ПЛАН_ИД
and Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.ИД = Н_НАПР_СПЕЦ.ИД and Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.ИД = Н_ПЛАНЫ.ИД and Н_ПЛАНЫ.ФО_ИД =
Н_ФОРМЫ_ОБУЧЕНИЯ.ИД and upper(Н_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ) = 'ОЧНАЯ' and Н_ПЛАНЫ.КУРС = 1 and
Н_ПЛАНЫ.УЧЕБНЫЙ_ГОД = '2011/2012' and to_char(Н_НАПР_СПЕЦ.КОД_НАПРСПЕЦ) = '220100' and
to_char(Н_УЧЕНИКИ.НАЧАЛО)='1.9.2011' and upper(Н_УЧЕНИКИ.СОСТОЯНИЕ) = 'УТВЕРЖДЕН' order by Н_УЧЕНИКИ.ГРУППА,
Н_ЛЮДИ.ФАМИЛИЯ;
```

34. Получить список студентов по условиям п. 33, но только на конец учебного года (31 августа).

```
select Н_УЧЕНИКИ.ГРУППА, Н_ЛЮДИ.ИД, Н_ЛЮДИ.ФАМИЛИЯ, Н_ЛЮДИ.ИМЯ, Н_ЛЮДИ.ОТЧЕСТВО, Н_УЧЕНИКИ.П_ПРКОК_ПРИКАЗ,
Н_УЧЕНИКИ.СОСТОЯНИЕ, Н_УЧЕНИКИ.ПРИЗНАК, Н_УЧЕНИКИ.КОНЕЦ from Н_УЧЕНИКИ, Н_ЛЮДИ, Н_ПЛАНЫ, Н_НАПР_СПЕЦ,
Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ, Н_ФОРМЫ_ОБУЧЕНИЯ where Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД and Н_ПЛАНЫ.ИД = Н_УЧЕНИКИ.ПЛАН_ИД
and Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.ИД = Н_НАПР_СПЕЦ.ИД and Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.ИД = Н_ПЛАНЫ.ИД and Н_ПЛАНЫ.ФО_ИД =
Н_ФОРМЫ_ОБУЧЕНИЯ.ИД and upper(Н_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ) = 'ОЧНАЯ' and Н_ПЛАНЫ.КУРС = 1 and
Н_ПЛАНЫ.УЧЕБНЫЙ_ГОД = '2011/2012' and to_char(Н_НАПР_СПЕЦ.КОД_НАПРСПЕЦ) = '220100' and
to_char(Н_УЧЕНИКИ.НАЧАЛО)='1.9.2011' and upper(Н_УЧЕНИКИ.СОСТОЯНИЕ) = 'УТВЕРЖДЕН' and Н_УЧЕНИКИ.КОНЕЦ > to_date('31.8.2011')
order by Н_УЧЕНИКИ.ГРУППА, Н_ЛЮДИ.ФАМИЛИЯ;
```

35. Выявить людей с одинаковыми фамилиями, именами и отчествами, но разными номерами.

Список должен содержать упорядоченные строки с уникальными сочетаниями номера, фамилии, имени и отчества.

```
select Н_ЛЮДИ.ИД, Н_ЛЮДИ.ФАМИЛИЯ, Н_ЛЮДИ.ИМЯ, Н_ЛЮДИ.ОТЧЕСТВО from Н_ЛЮДИ, Н_ЛЮДИ as temp where Н_ЛЮДИ.ФАМИЛИЯ
= temp.ФАМИЛИЯ and Н_ЛЮДИ.ИМЯ = temp.ИМЯ and Н_ЛЮДИ.ОТЧЕСТВО = temp.ОТЧЕСТВО and Н_ЛЮДИ.ИД != temp.ИД order by
Н_ЛЮДИ.ФАМИЛИЯ, Н_ЛЮДИ.ИМЯ, Н_ЛЮДИ.ОТЧЕСТВО;
```

36. Преобразовать запрос п. 32 так, чтобы во фразе FROM осталось соединение только тех таблиц, столбцы которых входят в списки фраз SELECT и ORDER BY. Остальные таблицы, данные из которых нужны для отбора нужных строк результата, необходимо разместить во фразе WHERE, не используя их соединений. Для организации подзапросов использовать предикат IN (проверка на принадлежность).

```
select Н_УЧЕНИКИ.ГРУППА, Н_ЛЮДИ.ИД, Н_ЛЮДИ.ФАМИЛИЯ, Н_ЛЮДИ.ИМЯ, Н_ЛЮДИ.ОТЧЕСТВО, Н_УЧЕНИКИ.П_ПРКОК_ПРИКАЗ,
Н_УЧЕНИКИ.СОСТОЯНИЕ, Н_УЧЕНИКИ.ПРИЗНАК, Н_УЧЕНИКИ.КОНЕЦ from Н_УЧЕНИКИ, Н_ЛЮДИ, Н_ПЛАНЫ, Н_НАПР_СПЕЦ,
Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ, Н_ФОРМЫ_ОБУЧЕНИЯ where Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД and
to_char(Н_УЧЕНИКИ.НАЧАЛО)='1.9.2011' and Н_УЧЕНИКИ.ПЛАН_ИД in (select Н_ПЛАНЫ.ИД from Н_ПЛАНЫ where Н_ПЛАНЫ.КУРС = 1
and Н_ПЛАНЫ.УЧЕБНЫЙ_ГОД = '2011/2012' and Н_ПЛАНЫ.ФО_ИД in (select Н_ФОРМЫ_ОБУЧЕНИЯ.ИД where
upper(Н_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ) = 'ОЧНАЯ') and Н_ПЛАНЫ.НАПС_ИД in (select Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.ИД from
Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ where Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.ИД in (select Н_НАПР_СПЕЦ.ИД from Н_НАПР_СПЕЦ where
to_char(Н_НАПР_СПЕЦ.КОД_НАПРСПЕЦ) = '220100')))) order by Н_УЧЕНИКИ.ГРУППА, Н_ЛЮДИ.ФАМИЛИЯ;
```

37. Преобразовать предыдущий запрос, используя для организации подзапросов предикат EXISTS (проверка на существование)

```
select Н_УЧЕНИКИ.ГРУППА, Н_ЛЮДИ.ИД, Н_ЛЮДИ.ФАМИЛИЯ, Н_ЛЮДИ.ИМЯ, Н_ЛЮДИ.ОТЧЕСТВО, Н_УЧЕНИКИ.П_ПРКОК_ПРИКАЗ,
Н_УЧЕНИКИ.СОСТОЯНИЕ, Н_УЧЕНИКИ.ПРИЗНАК, Н_УЧЕНИКИ.КОНЕЦ from Н_УЧЕНИКИ, Н_ЛЮДИ, Н_ПЛАНЫ, Н_НАПР_СПЕЦ,
Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ, Н_ФОРМЫ_ОБУЧЕНИЯ where Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД and
to_char(Н_УЧЕНИКИ.НАЧАЛО)='1.9.2011' and exist (select * from Н_ПЛАНЫ where Н_ПЛАНЫ.КУРС = 1 and Н_ПЛАНЫ.УЧЕБНЫЙ_ГОД =
'2011/2012' and exist (select Н_ФОРМЫ_ОБУЧЕНИЯ.ИД where upper(Н_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ) = 'ОЧНАЯ') and exist (select
```

Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ.ИД from Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ where exist (select Н_НАПР_СПЕЦ.ИД from Н_НАПР_СПЕЦ where to_char(Н_НАПР_СПЕЦ.КОД_НАПРСПЕЦ) = '220100')) order by Н_УЧЕНИКИ.ГРУППА, Н_ЛЮДИ.ФАМИЛИЯ;

38. Преобразовать запрос п. 29 так, чтобы во фразе FROM осталось соединение только тех таблиц, столбцы которых входят в списки фраз SELECT и GROUP BY. Остальные таблицы, данные из которых нужны для отбора нужных строк результата, необходимо разместить во фразе WHERE.

```
select Н_ЛЮДИ.ИД as Номер, Н_ЛЮДИ.ФАМИЛИЯ||' '||Н_ЛЮДИ.ИМЯ||' '||Н_ЛЮДИ.ОТЧЕСТВО||' ' AS ФИО,
ROUND(AVG(TO_NUMBER(Н_ВЕДОМОСТИ.ОЦЕНКА)),1) AS Ср_оценка from Н_ВЕДОМОСТИ, Н_ЛЮДИ where Н_ЛЮДИ.ИД =
Н_ВЕДОМОСТИ.ЧЛВК_ИД and НАЧАЛО < SYSDATE and КОНЕЦ > SYSDATE and ОЦЕНКА in ('2','3','4','5') and Н_ЛЮДИ.ИД in (select
Н_УЧЕНИКИ.ЧЛВК_ИД from _УЧЕНИКИ where ГРУППА = '3121') group by Н_УЧЕНИКИ.ЧЛВК_ИД, Н_ЛЮДИ.ФАМИЛИЯ, Н_ЛЮДИ.ИМЯ,
Н_ЛЮДИ.ОТЧЕСТВО;
```

39. Вывести упорядоченный по ФИО список людей (с фамилиями, начинающимися на "До"), не являющихся или не являвшихся студентами СПбГУИТМО (т.е. данные о которых отсутствуют в таблице Н_УЧЕНИКИ). Составить не менее двух вариантов запросов с условиями IN и EXISTS.

```
select * from Н_ЛЮДИ where Фамилия LIKE 'До%' and not exist (select * from Н_УЧЕНИКИ where Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД) order
by Фамилия, Имя, Отчество;
```

```
select * from Н_ЛЮДИ where Фамилия LIKE 'До%' and ИД not in (select Н_УЧЕНИКИ.ЧЛВК_ИД from Н_УЧЕНИКИ) order by Фамилия, Имя,
Отчество;
```

40. Выполнить запрос п. 35, используя для его реализации вложенный подзапрос, а не соединение таблицы со своей копией (в запросе нельзя использовать DISTINCT).

```
select ИД, ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО from Н_ЛЮДИ where exist (select * from Н_ЛЮДИ as Л where Л.ФАМИЛИЯ = Н_ЛЮДИ.ФАМИЛИЯ
and Л.ИМЯ = Н_ЛЮДИ.ИМЯ and Л.ОТЧЕСТВО = Н_ЛЮДИ.ОТЧЕСТВО and Л.ИД != Н_ЛЮДИ.ИД) order by ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО;
```

41. Сформировать запрос (единственный) для получения таблицы вида:

```
Select '1. Круглых отличников', count(distinct члвк_ид) as Кол_во from н_ведомости where члвк_ид not in (select члвк_ид from
н_ведомости where оценка not in ('5','зачет','осв'))
```

```
Union select '2. Учеников без троек', count(distinct члвк_ид) Кол_во from н_ведомости where члвк_ид not in (select члвк_ид from
н_ведомости where оценка='3')
```

```
Union select '3. Круглых троечников', count(distinct члвк_ид) Кол_во from н_ведомости where члвк_ид not in (select члвк_ид from
н_ведомости where оценка not in ('3', 'зачет'));
```