

Факультет программной инженерной и компьютерной техники
Базы Данных

Лабораторная работа №2
Вариант 10053

Выполнил: Алхимовици Арсений 408138
Принял: Бострикова Дарья Константиновна
Р3110

Санкт-Петербург, 2023

Текст Задания

Введите вариант:

Внимание! У разных вариантов разный текст задания!

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.ОТЧЕСТВО, Н_СЕССИЯ.УЧГОД.

Фильтры (AND):

а) Н_ЛЮДИ.ОТЧЕСТВО = Сергеевич.

б) Н_СЕССИЯ.ЧЛВК_ИД > 106059.

в) Н_СЕССИЯ.ЧЛВК_ИД > 106059.

Вид соединения: INNER 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.НЗК, Н_УЧЕНИКИ.НАЧАЛО.

Фильтры: (AND)

а) Н_ЛЮДИ.ИД < 152862.

б) Н_ОБУЧЕНИЯ.ЧЛВК_ИД > 163484.

Вид соединения: RIGHT JOIN.

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

4. В таблице Н_ГРУППЫ_ПЛАНОВ найти номера планов, по которым обучается (обучалось) более 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 меньше средней оценк(е)и в группе 3100.

6. Получить список студентов, зачисленных до первого сентября 2012 года на первый курс очной 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.

7. Вывести список людей, не являющихся или не являвшихся студентами СПбГУ ИТМО (данные, о которых отсутствуют в таблице Н_УЧЕНИКИ). В запросе нельзя использовать DISTINCT.

Реализация запросов на SQL

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

Таблицы: Н_ЛЮДИ, Н_СЕССИЯ.

Вывести атрибуты: Н_ЛЮДИ.ОТЧЕСТВО, Н_СЕССИЯ.УЧГОД.

Фильтры (AND):

а) Н_ЛЮДИ.ОТЧЕСТВО = Сергеевич.

б) Н_СЕССИЯ.ЧЛВК_ИД > 106059.

в) Н_СЕССИЯ.ЧЛВК_ИД > 106059.

Вид соединения: INNER JOIN.

```
SELECT Н_ЛЮДИ.ОТЧЕСТВО, Н_СЕССИЯ.УЧГОД
```

```
FROM Н_ЛЮДИ JOIN Н_СЕССИЯ ON Н_ЛЮДИ.ИД = Н_СЕССИЯ.ЧЛВК_ИД
```

```
WHERE Н_ЛЮДИ.ОТЧЕСТВО = 'Сергеевич' AND Н_СЕССИЯ.ЧЛВК_ИД > 106059;
```

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.НЗК, Н_УЧЕНИКИ.НАЧАЛО.

Фильтры: (AND)

а) Н_ЛЮДИ.ИД < 152862.

б) Н_ОБУЧЕНИЯ.ЧЛВК_ИД > 163484.

Вид соединения: RIGHT JOIN.

```

SELECT Н_ЛЮДИ.ФАМИЛИЯ, Н_ОБУЧЕНИЯ.НЗК, Н_УЧЕНИКИ.НАЧАЛО
FROM Н_ЛЮДИ RIGHT JOIN Н_ОБУЧЕНИЯ ON Н_ЛЮДИ.ИД = Н_ОБУЧЕНИЯ.ЧЛВК_ИД
RIGHT JOIN Н_УЧЕНИКИ ON Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД
WHERE Н_ЛЮДИ.ИД < 152862 AND Н_ОБУЧЕНИЯ.ЧЛВК_ИД > 163484;

```

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

```

SELECT
CASE
    WHEN COUNT(Н_ЛЮДИ.ОТЧЕСТВО IS NULL)>0 THEN 'да'
    ELSE 'нет'
END AS ответ
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_ПЛАНЫ.ФО_ИД=2 AND Н_ЛЮДИ.ОТЧЕСТВО IS NULL;

```

4. В таблице Н_ГРУППЫ_ПЛАНОВ найти номера планов, по которым обучается (обучалось) более 2 групп на кафедре вычислительной техники. Для реализации использовать подзапрос.

```

SELECT Н_ГРУППЫ_ПЛАНОВ.ПЛАН_ИД AS Номер_плана FROM Н_ГРУППЫ_ПЛАНОВ
WHERE Н_ГРУППЫ_ПЛАНОВ.ПЛАН_ИД IN (
    SELECT Н_ПЛАНЫ.ИД FROM Н_ПЛАНЫ
    JOIN Н_ОТДЕЛЫ ON Н_ПЛАНЫ.ОТД_ИД = Н_ОТДЕЛЫ.ИД
    WHERE Н_ОТДЕЛЫ.КОРОТКОЕ_ИМЯ = 'ВТ')
GROUP BY Номер_плана HAVING COUNT(Н_ГРУППЫ_ПЛАНОВ.ГРУППА) > 2;

```

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 3100.

```
SELECT Н_ЛЮДИ.ИД AS НОМЕР, Н_ЛЮДИ.ИМЯ, Н_ЛЮДИ.ФАМИЛИЯ, Н_ЛЮДИ.ОТЧЕСТВО,  
  
ROUND(AVG(Н_ВЕДОМОСТИ.ОЦЕНКА::INTEGER), 3) AS "средняя отметка" FROM  
Н_УЧЕНИКИ  
  
JOIN Н_ЛЮДИ ON Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД  
  
JOIN Н_ВЕДОМОСТИ ON Н_ЛЮДИ.ИД = Н_ВЕДОМОСТИ.ЧЛВК_ИД  
  
WHERE CAST(Н_УЧЕНИКИ.ГРУППА AS INTEGER) = 4100 AND Н_ВЕДОМОСТИ.ОЦЕНКА  
NOT IN ('зачет', 'незач', 'осв', 'неявка')  
  
GROUP BY Н_ЛЮДИ.ИД, Н_ЛЮДИ.ИМЯ, Н_ЛЮДИ.ФАМИЛИЯ, Н_ЛЮДИ.ОТЧЕСТВО  
  
HAVING ROUND(AVG(Н_ВЕДОМОСТИ.ОЦЕНКА::INTEGER), 3) > (  
  
    SELECT ROUND(AVG(Н_ВЕДОМОСТИ.ОЦЕНКА::INTEGER), 3) FROM Н_ВЕДОМОСТИ  
  
    JOIN Н_ЛЮДИ ON Н_ВЕДОМОСТИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД  
  
    JOIN Н_УЧЕНИКИ ON Н_ЛЮДИ.ИД = Н_УЧЕНИКИ.ЧЛВК_ИД  
  
    WHERE CAST(Н_УЧЕНИКИ.ГРУППА AS INTEGER) = 3100 AND  
    Н_ВЕДОМОСТИ.ОЦЕНКА NOT IN ('зачет', 'незач', 'осв', 'неявка')  
  
);
```

6. Получить список студентов, зачисленных до первого сентября 2012 года на первый курс очной 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 УЧЕНИКИ.ГРУППА AS группа, УЧЕНИКИ.ИД AS номер,  
  
    Н_ЛЮДИ.ФАМИЛИЯ, Н_ЛЮДИ.ИМЯ, Н_ЛЮДИ.ОТЧЕСТВО,  
  
    УЧЕНИКИ.П_ПРКОК_ИД AS приказ, УЧЕНИКИ.ПРИЗНАК AS приуф  
  
FROM Н_УЧЕНИКИ AS УЧЕНИКИ  
  
JOIN Н_ЛЮДИ ON УЧЕНИКИ.ЧЛВК_ИД = Н_ЛЮДИ.ИД  
  
JOIN Н_ПЛАНЫ ON УЧЕНИКИ.ПЛАН_ИД = Н_ПЛАНЫ.ИД
```

```
JOIN Н_ФОРМЫ_ОБУЧЕНИЯ ON Н_ПЛАНЫ.ФО_ИД = Н_ФОРМЫ_ОБУЧЕНИЯ.ИД AND  
Н_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ = 'Очная'
```

```
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 Н_УЧЕНИКИ.ПРИЗНАК, Н_УЧЕНИКИ.КОНЕЦ, Н_УЧЕНИКИ.ИД
```

```
    FROM Н_УЧЕНИКИ
```

```
    WHERE Н_УЧЕНИКИ.ПРИЗНАК = 'отчисл'
```

```
        AND Н_УЧЕНИКИ.ИД = УЧЕНИКИ.ИД
```

```
        AND DATE(Н_УЧЕНИКИ.КОНЕЦ) < '2012-09-01');
```

7. Вывести список людей, не являющихся или не являвшихся студентами СПбГУ ИТМО (данные, о которых отсутствуют в таблице Н_УЧЕНИКИ). В запросе нельзя использовать DISTINCT.

```
SELECT Не_Студенты.ИД, Не_Студенты.ИМЯ, Не_Студенты.ФАМИЛИЯ,  
Не_Студенты.ОТЧЕСТВО
```

```
FROM Н_ЛЮДИ AS Не_Студенты
```

```
WHERE NOT EXISTS (
```

```
    SELECT Н_ОТДЕЛЫ.КОРОТКОЕ_ИМЯ, Н_УЧЕНИКИ.ИД
```

```
    FROM Н_УЧЕНИКИ
```

```
    JOIN Н_ПЛАНЫ ON Н_УЧЕНИКИ.ПЛАН_ИД = Н_ПЛАНЫ.ИД
```

```
    JOIN Н_ОТДЕЛЫ ON Н_ПЛАНЫ.ОТД_ИД = Н_ОТДЕЛЫ.ИД
```

```
    WHERE Н_ОТДЕЛЫ.КОРОТКОЕ_ИМЯ = 'СПбГУИТМО' AND Н_УЧЕНИКИ.ЧЛВК_ИД =  
    Не_Студенты.ИД
```

```
);
```

Вывод

В ходе данной лабораторной работы я научился составлять SQL запросы, познакомился с разными видами соединения (RIGHT JOIN, LEFT JOIN, INNER JOIN). Так же поработал с подзапросами. Посмотрел учебную базу данных *“Учебный процесс”*.