

Министерство высшего образования и науки Российской Федерации
Национальный научно-исследовательский университет ИТМО
Факультет программной инженерии и компьютерной техники

ЛАБОРАТОРНАЯ РАБОТА №2

по дисциплине

«БАЗЫ ДАННЫХ».

Вариант №686.

Выполнил:

Петров Вячеслав Маркович,

Студент группы Р3108.

Преподаватель:

Афанасьев Дмитрий Борисович

Санкт-Петербург, 2024

Оглавление

Текст задания	3
Реализация запросов на SQL.....	4
Выводы по работе	7

Текст задания

Составить запросы на языке SQL (пункты 1-7).

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.ИД.
Фильтры (AND):
а) Н_ТИПЫ_ВЕДОМОСТЕЙ.ИД < 1.
б) Н_ВЕДОМОСТИ.ЧЛВК_ИД = 117219.
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.НЗК, Н_УЧЕНИКИ.ГРУППА.
Фильтры: (AND)
а) Н_ЛЮДИ.ОТЧЕСТВО < Александрович.
б) Н_ОБУЧЕНИЯ.ЧЛВК_ИД > 112514.
с) Н_УЧЕНИКИ.НАЧАЛО > 2009-02-09.
Вид соединения: INNER JOIN.
3. Вывести число дней без учета повторений.
При составлении запроса нельзя использовать DISTINCT.
4. Выдать различные фамилии преподавателей и число людей с каждой из этих фамилий, ограничив список фамилиями, встречающимися менее 10 раз на очной форме обучения.
Для реализации использовать подзапрос.
5. Выведите таблицу со средним возрастом студентов во всех группах (Группа, Средний возраст), где средний возраст меньше минимального возраста в группе 1100.
6. Получить список студентов, отчисленных до первого сентября 2012 года с заочной формы обучения (специальность: 230101). В результат включить:
номер группы;
номер, фамилию, имя и отчество студента;
номер пункта приказа;
Для реализации использовать подзапрос с IN.
7. Вывести список людей, не являющихся или не являвшихся студентами ФКТИУ (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:

-- Н_ТИПЫ_ВЕДОМОСТЕЙ, Н_ВЕДОМОСТИ.

-- Вывести атрибуты: Н_ТИПЫ_ВЕДОМОСТЕЙ.НАИМЕНОВАНИЕ, Н_ВЕДОМОСТИ.ИД.

-- Фильтры (AND):

-- а) Н_ТИПЫ_ВЕДОМОСТЕЙ.ИД < 1.

-- б) Н_ВЕДОМОСТИ.ЧЛВК_ИД = 117219.

-- Вид соединения: RIGHT JOIN.

SELECT "Н_ТИПЫ_ВЕДОМОСТЕЙ"."НАИМЕНОВАНИЕ", "Н_ВЕДОМОСТИ"."ИД"

FROM "Н_ТИПЫ_ВЕДОМОСТЕЙ"

RIGHT JOIN "Н_ВЕДОМОСТИ" ON "Н_ТИПЫ_ВЕДОМОСТЕЙ"."ИД" =
"Н_ВЕДОМОСТИ"."ВЕД_ИД"

WHERE "Н_ТИПЫ_ВЕДОМОСТЕЙ"."ИД" < 1

AND "Н_ВЕДОМОСТИ"."ЧЛВК_ИД" = 117219;

-- 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.ГРУППА.

-- Фильтры: (AND)

-- а) Н_ЛЮДИ.ОТЧЕСТВО < Александрович.

-- б) Н_ОБУЧЕНИЯ.ЧЛВК_ИД > 112514.

-- с) Н_УЧЕНИКИ.НАЧАЛО > 2009-02-09.

-- Вид соединения: INNER JOIN.

SELECT "Н_ЛЮДИ"."ФАМИЛИЯ", "Н_ОБУЧЕНИЯ"."НЗК", "Н_УЧЕНИКИ"."ГРУППА"

FROM "Н_ЛЮДИ"

JOIN "Н_ОБУЧЕНИЯ" ON "Н_ОБУЧЕНИЯ"."ЧЛВК_ИД" = "Н_ЛЮДИ"."ИД"

JOIN "Н_УЧЕНИКИ" ON "Н_УЧЕНИКИ"."ЧЛВК_ИД" = "Н_ОБУЧЕНИЯ"."ЧЛВК_ИД"

WHERE "Н_ЛЮДИ"."ОТЧЕСТВО" < 'Александрович'

AND "Н_ОБУЧЕНИЯ"."ЧЛВК_ИД" > 112514

AND DATE("Н_УЧЕНИКИ"."НАЧАЛО") > '2009-02-09';

-- 3. Вывести число дней без учета повторений. При составлении запроса нельзя использовать DISTINCT.

```
SELECT count("ДАТА")
FROM (SELECT "Н_ВЕДОМОСТИ"."ДАТА" AS "ДАТА"
      FROM "Н_ВЕДОМОСТИ"
      GROUP BY "Н_ВЕДОМОСТИ"."ДАТА") as foo;
```

-- 4. Выдать различные фамилии преподавателей и число людей с каждой из этих фамилий, ограничив список фамилиями, встречающимися менее 10 раз на очной форме обучения.

-- Для реализации использовать подзапрос.

```
SELECT "Н_ЛЮДИ"."ФАМИЛИЯ", count("Н_ЛЮДИ"."ИД")
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"
      AND (Н_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ = 'Очная')
WHERE "Н_ЛЮДИ"."ФАМИЛИЯ" IN (SELECT Н_ЛЮДИ.ФАМИЛИЯ
                              FROM Н_ЛЮДИ
                              JOIN Н_СЕССИЯ ON Н_ЛЮДИ.ИД = Н_СЕССИЯ.ЧЛВК_ИД)
GROUP BY "Н_ЛЮДИ"."ФАМИЛИЯ"
HAVING count("Н_ЛЮДИ"."ИД") < 10;
```

-- 5. Выведите таблицу со средним возрастом студентов во всех группах (Группа, Средний возраст), где средний возраст меньше минимального возраста в группе 1100.

```
SELECT "Н_УЧЕНИКИ"."ГРУППА", avg(date_part('year', age("Н_ЛЮДИ"."ДАТА_РОЖДЕНИЯ")))
FROM "Н_ЛЮДИ"
      JOIN "Н_УЧЕНИКИ" ON "Н_УЧЕНИКИ"."ЧЛВК_ИД" = "Н_ЛЮДИ"."ИД"
GROUP BY "Н_УЧЕНИКИ"."ГРУППА"
HAVING avg(date_part('year', age("Н_ЛЮДИ"."ДАТА_РОЖДЕНИЯ"))) <
      (SELECT min(date_part('year', age("Н_ЛЮДИ"."ДАТА_РОЖДЕНИЯ")))
      FROM "Н_ЛЮДИ"
      JOIN "Н_УЧЕНИКИ" ON "Н_УЧЕНИКИ"."ЧЛВК_ИД" = "Н_ЛЮДИ"."ИД"
      WHERE "Н_УЧЕНИКИ"."ГРУППА" = '1100');
```

-- 6. Получить список студентов, отчисленных до первого сентября 2012 года с заочной формы обучения (специальность: 230101). В результат включить:

-- номер группы;

-- номер, фамилию, имя и отчество студента;

-- номер пункта приказа;

-- Для реализации использовать подзапрос с IN.

```
SELECT "ВСЕ_УЧЕНИКИ"."ГРУППА",
       "ВСЕ_УЧЕНИКИ"."ИД",
       "Н_ЛЮДИ"."ФАМИЛИЯ",
       "Н_ЛЮДИ"."ИМЯ",
       "Н_ЛЮДИ"."ОТЧЕСТВО",
       "ВСЕ_УЧЕНИКИ"."П_ПРКОК_ИД"
FROM "Н_УЧЕНИКИ" AS "ВСЕ_УЧЕНИКИ"
      JOIN "Н_ЛЮДИ" ON "Н_ЛЮДИ"."ИД" = "ВСЕ_УЧЕНИКИ"."ЧЛВК_ИД"
      JOIN "Н_ПЛАНЫ" ON "ВСЕ_УЧЕНИКИ"."ПЛАН_ИД" = "Н_ПЛАНЫ"."ИД"
      JOIN "Н_ФОРМЫ_ОБУЧЕНИЯ" ON "Н_ПЛАНЫ"."ФО_ИД" =
"Н_ФОРМЫ_ОБУЧЕНИЯ"."ИД"
      AND (Н_ФОРМЫ_ОБУЧЕНИЯ.НАИМЕНОВАНИЕ = 'Заочная')
      JOIN "Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ" ON "Н_ПЛАНЫ"."НАПС_ИД" =
"Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ"."ИД"
      JOIN "Н_НАПР_СПЕЦ" ON "Н_НАПР_СПЕЦ"."ИД" =
"Н_НАПРАВЛЕНИЯ_СПЕЦИАЛ"."НС_ИД"
      AND "Н_НАПР_СПЕЦ"."КОД_НАПРСПЕЦ" = '230101'
WHERE "ВСЕ_УЧЕНИКИ"."ИД" IN (SELECT "ОТЧИСЛ_УЧЕНИКИ"."ИД"
      FROM "Н_УЧЕНИКИ" AS "ОТЧИСЛ_УЧЕНИКИ"
      WHERE "ОТЧИСЛ_УЧЕНИКИ"."ПРИЗНАК" = 'отчисл'
      AND "ОТЧИСЛ_УЧЕНИКИ"."СОСТОЯНИЕ" = 'утвержден'
      AND "ОТЧИСЛ_УЧЕНИКИ"."ИД" = "ВСЕ_УЧЕНИКИ"."ИД"
      AND DATE("ОТЧИСЛ_УЧЕНИКИ"."КОНЕЦ") < '2012-09-01');
```

-- 7. Вывести список людей, не являющихся или не являвшихся студентами ФКТИУ

-- (данные, о которых отсутствуют в таблице Н_УЧЕНИКИ). В запросе нельзя

-- использовать DISTINCT.

```
SELECT "Н_ЛЮДИ"."ИД",
       "Н_ЛЮДИ"."ФАМИЛИЯ",
```

```
"Н_ЛЮДИ"."ИМЯ",  
"Н_ЛЮДИ"."ОТЧЕСТВО"  
FROM "Н_ЛЮДИ"  
WHERE NOT EXISTS (SELECT *  
    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"  
        AND "Н_ОТДЕЛЫ"."КОРОТКОЕ_ИМЯ" = 'КТиУ'  
    WHERE "Н_УЧЕНИКИ"."ЧЛВК_ИД" = "Н_ЛЮДИ"."ИД");
```

Выводы по работе

При выполнении лабораторной работы я познакомился с основными функциями языка SQL и диалекта PostgreSQL. Научился писать запросы, получать, отсеивать и сортировать полученные данные с использованием различных синтаксических конструкций языка. В результате был освоен язык DML SQL, предназначенный для работы с данными, хранящимися внутри базы данных.