

Федеральное государственное автономное образовательное учреждение
высшего образования «Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Направление подготовки: 09.03.04 Программная инженерия

Отчет
по лабораторной работе №2
по дисциплине «Базы данных»

Вариант: 87846

Выполнила
Махнева Ирина Александровна,
Группа Р3117

Преподаватель
Чупанов Аликылыч Алибекович

Оглавление

Текст задания	3
Реализация запросов на SQL	4
Вывод.....	6

Текст задания

По варианту, выданному преподавателем, составить и выполнить запросы к базе данных "Учебный процесс".

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_ВЕДОМОСТИ.
Вывести атрибуты: Н_ЛЮДИ.ИД, Н_ВЕДОМОСТИ.ИД.
Фильтры (AND):
а) Н_ЛЮДИ.ОТЧЕСТВО > Владимирович.
б) Н_ВЕДОМОСТИ.ЧЛВК_ИД < 153285.
в) Н_ВЕДОМОСТИ.ЧЛВК_ИД > 153285.
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.ИМЯ, Н_ОБУЧЕНИЯ.НЗК, Н_УЧЕНИКИ.ГРУППА.
Фильтры: (AND)
а) Н_ЛЮДИ.ФАМИЛИЯ < Афанасьев.
б) Н_ОБУЧЕНИЯ.НЗК < 001000.
Вид соединения: RIGHT JOIN.
3. Вывести число студентов ФКТИУ, которые младше 20 лет.
Ответ должен содержать только одно число.
4. Найти группы, в которых в 2011 году было ровно 5 обучающихся студентов на ФКТИУ.
Для реализации использовать соединение таблиц.
5. Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср_оценка), у которых средняя оценка равна средней оценк(е|и) в группе 1100.
6. Получить список студентов, зачисленных до первого сентября 2012 года на первый курс очной или заочной формы обучения. 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 ФКТИУ (данные, о которых отсутствуют в таблице Н_УЧЕНИКИ). В запросе нельзя использовать DISTINCT.

Реализация запросов на SQL

```
-- 1 запрос
-- 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.ОТЧЕСТВО > Владимирович.
-- б) Н_ВЕДОМОСТИ.ЧЛВК_ИД < 153285.
-- с) Н_ВЕДОМОСТИ.ЧЛВК_ИД > 153285.
-- Вид соединения: INNER JOIN.
SELECT "Н_ЛЮДИ"."ИД", "Н_ВЕДОМОСТИ"."ИД" FROM "Н_ЛЮДИ"
    INNER JOIN "Н_ВЕДОМОСТИ" USING ("ИД")
    WHERE "Н_ЛЮДИ"."ОТЧЕСТВО" > 'Владимирович'
        AND "Н_ВЕДОМОСТИ"."ЧЛВК_ИД" < 153285
        AND "Н_ВЕДОМОСТИ"."ЧЛВК_ИД" > 153285;

-- 2 запрос
-- Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по
указанным условиям:
-- Таблицы: Н_ЛЮДИ, Н_ОБУЧЕНИЯ, Н_УЧЕНИКИ.
-- Вывести атрибуты: Н_ЛЮДИ.ИМЯ, Н_ОБУЧЕНИЯ.НЗК, Н_УЧЕНИКИ.ГРУППА.
-- Фильтры: (AND)
-- а) Н_ЛЮДИ.ФАМИЛИЯ < Афанасьев.
-- б) Н_ОБУЧЕНИЯ.НЗК < 001000.
-- Вид соединения: RIGHT JOIN.
SELECT "Н_ЛЮДИ"."ИМЯ", "Н_ОБУЧЕНИЯ"."НЗК", "Н_УЧЕНИКИ"."ГРУППА" 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"."ИД"
    WHERE "Н_ЛЮДИ"."ФАМИЛИЯ" < 'Афанасьев'
        AND "Н_ОБУЧЕНИЯ"."НЗК" < '001000';

-- 3 запрос
-- Вывести число студентов ФКТИУ, которые младше 20 лет.
-- Ответ должен содержать только одно число.
SELECT COUNT(*) 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" AND "Н_ОТДЕЛЫ"."КОРОТКОЕ_ИМЯ" = 'КТИУ')
    INNER JOIN "Н_ОБУЧЕНИЯ"
        USING ("ЧЛВК_ИД")
    INNER JOIN "Н_ЛЮДИ"
        ON ("Н_ЛЮДИ"."ИД" = "Н_ОБУЧЕНИЯ"."ЧЛВК_ИД")
    WHERE date_part('year', AGE("Н_ЛЮДИ"."ДАТА_РОЖДЕНИЯ")) < 20
;

-- 4 запрос
-- Найти группы, в которых в 2011 году было ровно 5 обучающихся студентов на ФКТИУ.
-- Для реализации использовать соединение таблиц.
SELECT "ГРУППА" 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" AND "Н_ОТДЕЛЫ"."КОРОТКОЕ_ИМЯ" = 'КТИУ')
    WHERE "Н_ПЛАНЫ"."УЧЕБНЫЙ_ГОД" = '2010/2011'
    GROUP BY "ГРУППА"
    HAVING COUNT("Н_УЧЕНИКИ"."ИД") = 5
;
```

```

-- 5 запрос
-- Выведите таблицу со средними оценками студентов группы 4100 (Номер, ФИО, Ср_оценка), у
которых средняя оценка равна средней оценк(е|и) в группе 1100.
WITH "Средняя оценка группы 1100" AS (
    SELECT AVG("Н_ВЕДОМОСТИ"."ОЦЕНКА"::INT) FROM "Н_УЧЕНИКИ"
    INNER JOIN "Н_ВЕДОМОСТИ"
        ON "Н_УЧЕНИКИ"."ЧЛВК_ИД" = "Н_ВЕДОМОСТИ"."ЧЛВК_ИД"
    WHERE "Н_ВЕДОМОСТИ"."ОЦЕНКА" ~ '^\\d+$'
        AND "Н_УЧЕНИКИ"."ГРУППА" = '1100'
    GROUP BY "ГРУППА"
)

SELECT "Н_УЧЕНИКИ"."ИД", "Н_ЛЮДИ"."ФАМИЛИЯ", "Н_ЛЮДИ"."ИМЯ", "Н_ЛЮДИ"."ОТЧЕСТВО",
AVG("Н_ВЕДОМОСТИ"."ОЦЕНКА"::INT) FROM "Н_УЧЕНИКИ"
    INNER JOIN "Н_ЛЮДИ"
        ON ("Н_ЛЮДИ"."ИД" = "Н_УЧЕНИКИ"."ЧЛВК_ИД")
    INNER JOIN "Н_ВЕДОМОСТИ" --у кого-то может не быть оценок
        ON "Н_ЛЮДИ"."ИД" = "Н_ВЕДОМОСТИ"."ЧЛВК_ИД"
    WHERE "Н_ВЕДОМОСТИ"."ОЦЕНКА" ~ '^\\d+$'
        AND "Н_УЧЕНИКИ"."ГРУППА" = '4100'
    GROUP BY "Н_УЧЕНИКИ"."ИД", "ФАМИЛИЯ", "ИМЯ", "ОТЧЕСТВО"
    HAVING AVG("Н_ВЕДОМОСТИ"."ОЦЕНКА"::INT) = (Select * from "Средняя оценка группы
1100")
;

-- 6 запрос
-- Получить список студентов, зачисленных до первого сентября 2012 года на первый курс
очной или заочной формы обучения. В результат включить:
-- номер группы;
-- номер, фамилию, имя и отчество студента;
-- номер и состояние пункта приказа;
-- Для реализации использовать подзапрос с EXISTS.
SELECT "Н_УЧЕНИКИ"."ГРУППА", "Н_УЧЕНИКИ"."ИД", "ФАМИЛИЯ", "ИМЯ", "ОТЧЕСТВО",
"П_ПРКОК_ИД", "СОСТОЯНИЕ" FROM "Н_УЧЕНИКИ"
    INNER JOIN "Н_ЛЮДИ"
        ON "Н_УЧЕНИКИ"."ЧЛВК_ИД" = "Н_ЛЮДИ"."ИД"
    WHERE ("Н_УЧЕНИКИ"."НАЧАЛО" < '1.09.2012' AND EXISTS (
        SELECT "Н_ПЛАНЫ"."ИД" FROM "Н_ПЛАНЫ"
            INNER JOIN "Н_ФОРМЫ_ОБУЧЕНИЯ" ON "Н_ФОРМЫ_ОБУЧЕНИЯ"."ИД" = "Н_ПЛАНЫ"."ФО_ИД"
            WHERE (("Н_ФОРМЫ_ОБУЧЕНИЯ")."НАИМЕНОВАНИЕ" = 'Очная' OR
("Н_ФОРМЫ_ОБУЧЕНИЯ")."НАИМЕНОВАНИЕ" = 'Заочная')
                AND "Н_ПЛАНЫ"."КУРС" = 1
                AND "Н_УЧЕНИКИ"."ПЛАН_ИД" = "Н_ПЛАНЫ"."ИД"
        )));

-- 7 запрос
-- 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"
        INNER JOIN "Н_ПЛАНЫ"
            ON "Н_УЧЕНИКИ"."ПЛАН_ИД" = "Н_ПЛАНЫ"."ИД"
        INNER 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"
);

```

Вывод

В ходе выполнения лабораторной работы я научилась использовать различные типы соединений таблиц, использовать подзапросы, фильтровать, группировать и сортировать данные.