Университет ИТМО

Кафедра вычислительной техники

Системы баз данных

**Лабораторная работа №1**

**Этап №1**

Выполнил:

Студент группы P3317

Пулич Алексей

Преподаватель:

Беликов Павел Андреевич

Санкт-Петербург

2016 г

**Выполнение**

1. Вывести содержимое всех столбцов таблицы Н\_СЕССИЯ, поля типа DATE представить в формате DD.MM.YYYY HH24:MI:SS

SELECT Н\_СЕССИЯ.\*,

TO\_CHAR(ДАТА, 'dd-mm-yyyy hh24:mi:ss') ДАТА, TO\_CHAR(ВРЕМЯ, 'dd-mm-yyyy hh24:mi:ss') ВРЕМЯ,

TO\_CHAR(ДАТА\_К, 'dd-mm-yyyy hh24:mi:ss') ДАТА\_К,

TO\_CHAR(ВРЕМЯ\_К, 'dd-mm-yyyy hh24:mi:ss') ВРЕМЯ\_К

FROM

Н\_СЕССИЯ;

1. Вывести неповторяющиеся наименования дисциплин из таблицы Н\_ДИСЦИПЛИНЫ

SELECT DISTINCT НАИМЕНОВАНИЕ

FROM Н\_ДИСЦИПЛИНЫ;

1. Вывести округленное значение разницы между днем рождения произвольной  персоны из Н\_ЛЮДИ и началом текущего учебного года.

SELECT ROUND(TO\_DATE('2016/09/01', 'yyyy/mm/dd') - ДАТА\_РОЖДЕНИЯ)

FROM Н\_ЛЮДИ

WHERE

ИД = 209788;

1. Получить фамилии и инициалы людей (в виде «Иванов И.И.»), которые родились в  том же месяце, что и произвольная персона из Н\_ЛЮДИ.

SELECT ФАМИЛИЯ || ' ' ||

SUBSTR(ИМЯ,1,1) || '.' ||

SUBSTR(ОТЧЕСТВО,1,1) || '.' ЧЕЛОВЕК

FROM Н\_ЛЮДИ

WHERE

TO\_CHAR(ДАТА\_РОЖДЕНИЯ,'month') =

(SELECT TO\_CHAR(ДАТА\_РОЖДЕНИЯ,'month')

FROM Н\_ЛЮДИ

WHERE ИД = 209788);

1. Вывести фамилии, имена, отчества и номера (ИД) людей, фамилии которых  начинаются на те же 2 буквы, что и у произвольной персоны из Н\_ЛЮДИ,  упорядочить по убыванию ФИО, вывести не более 75 строк.

SELECT ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО, ИД

FROM Н\_ЛЮДИ

WHERE

ФАМИЛИЯ LIKE(

SELECT SUBSTR((ФАМИЛИЯ),1,2) || '%'

FROM Н\_ЛЮДИ WHERE ИД = 209788

)

AND ROWNUM <= 75

ORDER BY ФАМИЛИЯ DESC,ИМЯ DESC, ОТЧЕСТВО DESC;

1. Вывести список персон (фамилия, имя, отчество, ИД), для которых инициалы не  равны ‘А’, ‘Б’, ‘З’ и ‘К’, ‘У’. Представить не менее 2х вариантов запроса.

SELECT ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО, ИД

FROM Н\_ЛЮДИ

WHERE

SUBSTR(ФАМИЛИЯ,1,1) NOT IN ('А','Б','З')

AND SUBSTR(ИМЯ,1,1) NOT IN ('К','У');

SELECT ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО, ИД

FROM Н\_ЛЮДИ

WHERE

SUBSTR(ФАМИЛИЯ,1,1) <> 'А'

AND SUBSTR(ФАМИЛИЯ,1,1) <> 'Б'

AND SUBSTR(ФАМИЛИЯ,1,1) <> 'З'

AND SUBSTR(ИМЯ,1,1) <> 'К'

AND SUBSTR(ИМЯ,1,1) <> 'У';

1. Вычислите количество персон, имя которых такое же, как у произвольной персоны  из Н\_ЛЮДИ, учесть возможность наличия дубликатов.

SELECT COUNT(DISTINCT ИД)

FROM Н\_ЛЮДИ

WHERE

ИМЯ = (

SELECT ИМЯ FROM Н\_ЛЮДИ

WHERE ИД = 105840

);

1. Вывести, используя таблицу Н\_ВЕДОМОСТИ, удвоенные (значение\_оценки\*2)  оценки произвольной персоны из Н\_ЛЮДИ. Использовать NOT IN и регулярные  выражения.

SELECT ОЦЕНКА\*2

FROM Н\_ВЕДОМОСТИ

WHERE

ЧЛВК\_ИД = 182872

AND REGEXP\_LIKE(ОЦЕНКА, '^[0-9]+$')

AND ОЦЕНКА NOT IN (3,5);

1. Вывести, используя таблицу Н\_ВЕДОМОСТИ, сумму оценок 7и произвольных  персон из Н\_ЛЮДИ. 7 персон задаются условием к таблице Н\_ЛЮДИ.

SELECT \* FROM

(

SELECT SUM(Н\_ВЕДОМОСТИ.ОЦЕНКА), Н\_ЛЮДИ.ФАМИЛИЯ

FROM Н\_ВЕДОМОСТИ

INNER JOIN Н\_ЛЮДИ

ON (Н\_ЛЮДИ.ИД=Н\_ВЕДОМОСТИ.ЧЛВК\_ИД)

WHERE

SUBSTR(Н\_ЛЮДИ.ФАМИЛИЯ,1,1) < 'П'

AND REGEXP\_LIKE(ОЦЕНКА, '^\d+$')

GROUP BY Н\_ЛЮДИ.ФАМИЛИЯ

)

WHERE

ROWNUM <=7;

1. Получить декартово соединение N таблиц, где N равно 3й цифре вашего  табельного номера.

*3-я цифра моего табельного номера равна единице, поэтому привожу пример для N=2*

SELECT Н\_ЛЮДИ.\*, Н\_ВЕДОМОСТИ.\*

FROM Н\_ЛЮДИ, Н\_ВЕДОМОСТИ;

1. Вывести, используя таблицу Н\_ВЕДОМОСТИ, среднюю оценку 7и произвольных  персон из Н\_ЛЮДИ , их фамилии, имена и отчества, при этом средняя оценка не  должна быть наибольшей среди всех персон с такой же фамилией.

SELECT t1.фамилия, ИМЯ, ОТЧЕСТВО, m, mmax

FROM(

SELECT ФАМИЛИЯ, MAX(m) as mmax

FROM (

SELECT ЧЛВК\_ИД, AVG(оценка) m

FROM Н\_ВЕДОМОСТИ

WHERE regexp\_like(оценка,'^\d+$')

GROUP BY ЧЛВК\_ИД

) avgs, Н\_ЛЮДИ

WHERE ЧЛВК\_ИД = Н\_ЛЮДИ.ИД

GROUP BY ФАМИЛИЯ

) t1,

(

SELECT ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО, m

FROM (

SELECT ЧЛВК\_ИД, AVG(ОЦЕНКА) m

FROM Н\_ВЕДОМОСТИ

WHERE regexp\_like(ОЦЕНКА,'^\d+$')

GROUP BY ЧЛВК\_ИД

) avgs, Н\_ЛЮДИ

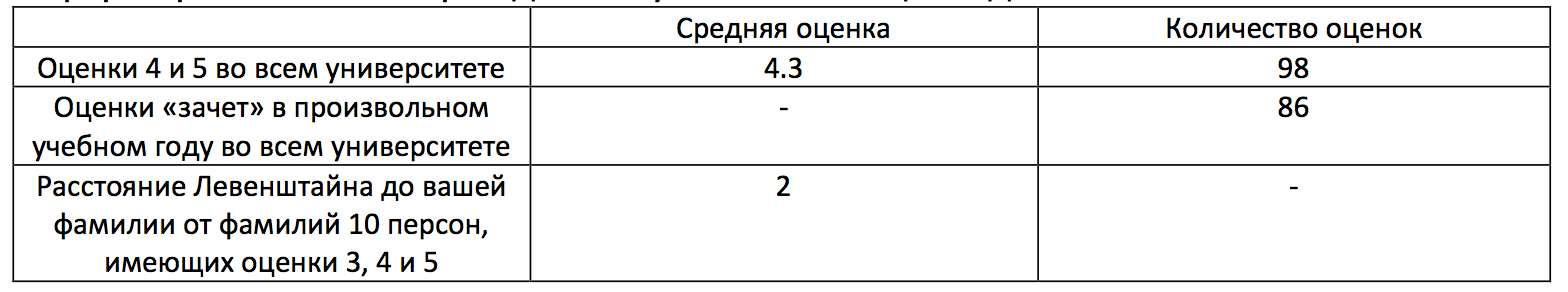
WHERE ЧЛВК\_ИД = Н\_ЛЮДИ.ИД

) t2

WHERE t1.ФАМИЛИЯ = t2.ФАМИЛИЯ

AND m < mmax AND ROWNUM <8;

1. Сформировать SQL-запрос для получения таблицы вида:



SELECT 'Оценки 4 и 5', AVG(ОЦЕНКА) Средняя\_оценка, COUNT(ОЦЕНКА) Количество\_оценок

FROM Н\_ВЕДОМОСТИ

WHERE

ОЦЕНКА IN('4','5')

UNION

SELECT 'Оценки ''зачет''',to\_number(NULL) Средняя\_оценка, COUNT(ОЦЕНКА) Количество\_оценок

FROM Н\_ВЕДОМОСТИ

WHERE

ДАТА BETWEEN TO\_DATE('01.09.2015')

AND TO\_DATE('30.07.2016')

AND ОЦЕНКА IN ('зачет')

UNION ALL

SELECT 'Расстояние Левенштайна', AVG(Средняя\_оценка),to\_number(NULL) Количество\_оценок

FROM (

SELECT utl\_match.edit\_distance('Пулич', ФАМИЛИЯ) Средняя\_оценка

FROM (

SELECT ФАМИЛИЯ

FROM Н\_ЛЮДИ

WHERE

ИД IN (

SELECT DISTINCT A.члвк\_ид

FROM н\_ведомости A, н\_ведомости B, н\_ведомости C

WHERE

A.члвк\_ид = B.члвк\_ид

AND A.члвк\_ид = C.члвк\_ид

AND A.оценка = '3'

AND B.оценка = '4'

AND C.оценка = '5'

)

AND ROWNUM <=10

)

);

1. Получить список персон, получивших оценки 3 и 4 с 01.09 прошлого календарного года по 20.07 текущего календарного года, упорядочить список по Ф.И.О. Использование объединений таблиц запрещено.

SELECT ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО

FROM Н\_ЛЮДИ

WHERE

ИД IN (

SELECT ЧЛВК\_ИД

FROM Н\_ВЕДОМОСТИ

WHERE ОЦЕНКА IN(3,4) AND ДАТА BETWEEN

TO\_DATE('2015/09/01', 'yyyy/mm/dd') AND

TO\_DATE('2016/07/20', 'yyyy/mm/dd')

GROUP BY ЧЛВК\_ИД

)

ORDER BY ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО;

1. Получить список людей с сочетаниями фамилии, имени и отчество, сумма оценок которых не превышает сумму цифр ИД произвольной персоны из таблицы Н\_ЛЮДИ.

WITH CHECKED AS (

SELECT k.ЧЛВК\_ИД, Н\_ЛЮДИ.ФАМИЛИЯ || ' ' || Н\_ЛЮДИ.ИМЯ || ' ' || Н\_ЛЮДИ.ОТЧЕСТВО as fio, k.summa

FROM Н\_ЛЮДИ,

(

SELECT ЧЛВК\_ИД, SUM(ОЦЕНКА) as summa

FROM Н\_ВЕДОМОСТИ

WHERE regexp\_like(оценка,'^\d+$')

GROUP BY ЧЛВК\_ИД) k

WHERE k.ЧЛВК\_ИД=Н\_ЛЮДИ.ИД

AND summa < (

(

SELECT 9 \* regexp\_count(a,'9') + 8 \* regexp\_count(a,'8') + 7 \*

regexp\_count(a,'7') + 6 \* regexp\_count(a,'6') + 5 \*

regexp\_count(a,'5') + 4 \* regexp\_count(a,'4') + 3 \*

regexp\_count(a,'3') + 2 \* regexp\_count(a,'2') + 1 \*

regexp\_count(a,'1')

FROM (

SELECT ИД AS a

FROM (

SELECT ИД

FROM Н\_ЛЮДИ

ORDER BY dbms\_random.random

)

WHERE rownum=1

)

)

)

)

SELECT CHECKED.fio, CHECKED.summa, cnt

FROM CHECKED,

(

SELECT fio, count (fio) as cnt

FROM CHECKED

GROUP BY FIO) f

WHERE CHECKED.fio=f.fio AND cnt > 1;