**ОТЧЕТ**

**По лабораторной работе №3 «Запросы»**

**Вариант 8**

Студентка группы 13  
Врублевская Е. А.

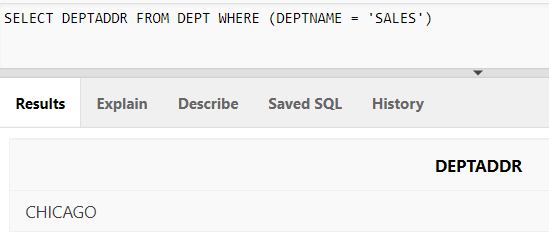
Исправить with, case константный и еще коррелирующий запрос

**Задание 3.1**

**ПРОСТЕЙШИЕ ЗАПРОСЫ:**

1. Выдать информацию о местоположении отдела продаж (SALES) компании.

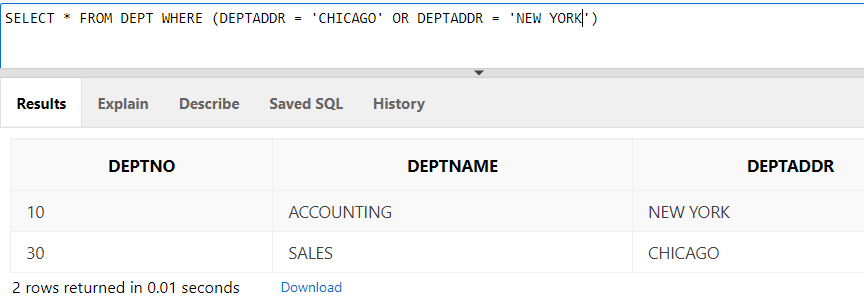
SELECT DEPTADDR FROM DEPT WHERE (DEPTNAME = 'SALES')

Результат: 

1. Выдать информацию об отделах, расположенных в Chicago и New York.

SELECT \* FROM DEPT WHERE (DEPTADDR = 'CHICAGO' OR DEPTADDR = 'NEW YORK')

Результат:

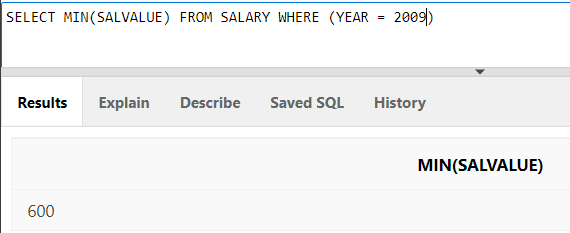


**ФУНКЦИИ:**

1. Найти минимальную заработную плату, начисленную в 2009 году.

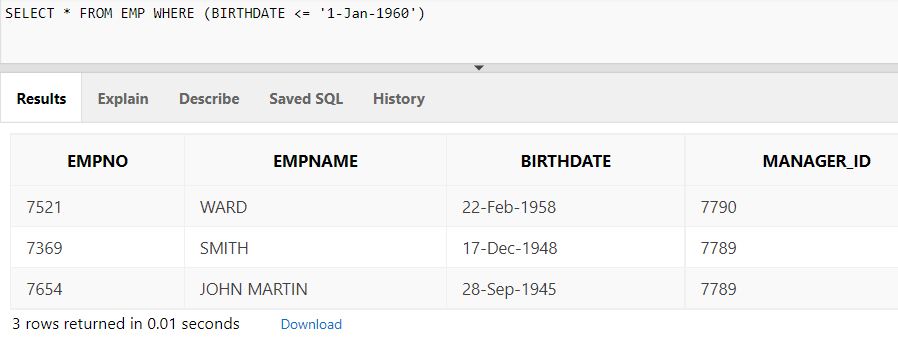
SELECT MIN(SALVALUE) FROM SALARY WHERE (YEAR = 2009)

Результат:



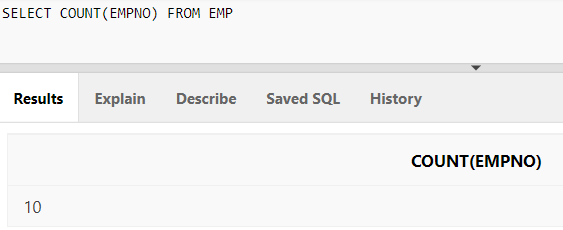
1. Выдать информацию обо всех работниках, родившихся не позднее 1 января 1960 года.

SELECT \* FROM EMP WHERE (BIRTHDATE <= '1-Jan-1960')



1. Подсчитать число работников, сведения о которых имеются в базе данных .

SELECT COUNT(EMPNO) FROM EMP



1. Найти работников, чьё имя состоит из одного слова. Имена выдать на нижнем регистре, с удалением стоящей справа буквы t.

SELECT

CASE WHEN LOWER(EMPNAME) LIKE '%t'

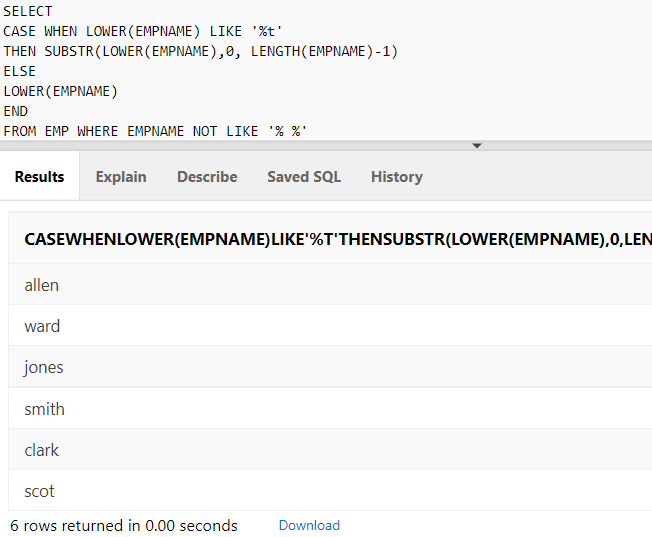
THEN SUBSTR(LOWER(EMPNAME),0, LENGTH(EMPNAME)-1)

ELSE

LOWER(EMPNAME)

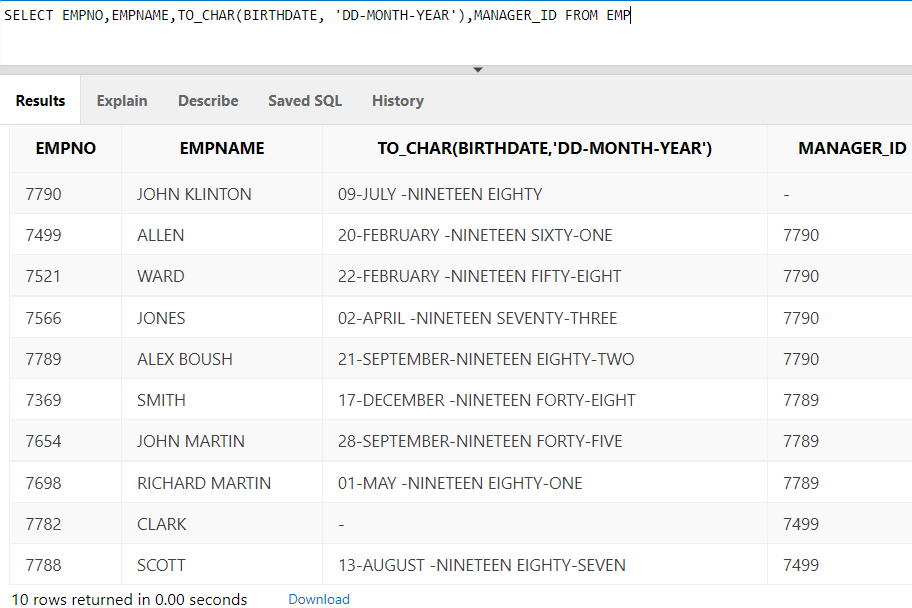
END

FROM EMP WHERE EMPNAME NOT LIKE '% %'



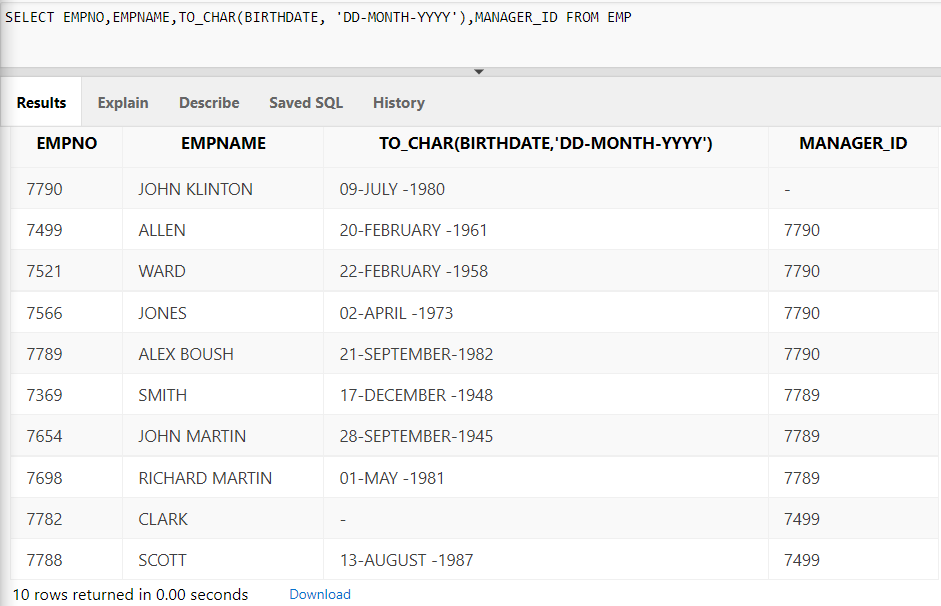
1. Выдать информацию о работниках, указав дату рождения в формате день(число), месяц(название), год(название).

SELECT EMPNO,EMPNAME,TO\_CHAR(BIRTHDATE, 'DD-MONTH-YEAR'), MANAGER\_ID FROM EMP



Тоже, но год числом.

SELECT EMPNO,EMPNAME,TO\_CHAR(BIRTHDATE, 'DD-MONTH-YYYY'), MANAGER\_ID FROM EMP



1. Выдать информацию о должностях, изменив названия должности “CLERK” и “DRIVER” на “WORKER”.

SELECT

CASE WHEN (JOBNAME = 'CLERK' OR JOBNAME = 'DRIVER')

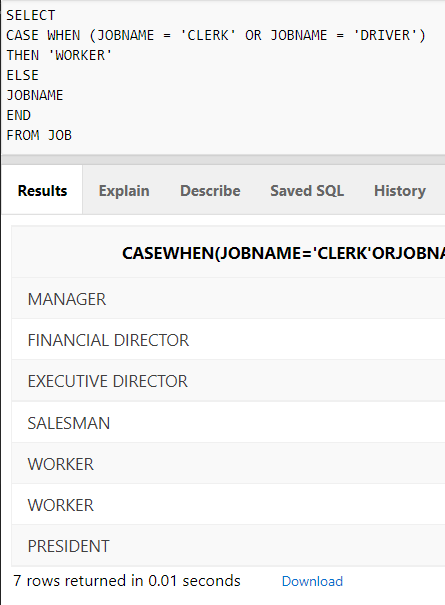
THEN 'WORKER'

ELSE

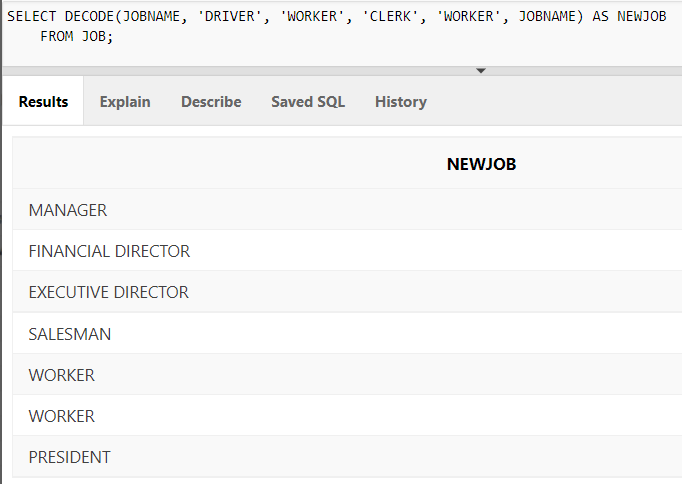
JOBNAME

END

FROM JOB



Или SELECT DECODE(JOBNAME, 'DRIVER', 'WORKER', 'CLERK', 'WORKER', JOBNAME) AS NEWJOB FROM JOB;



**HAVING**:

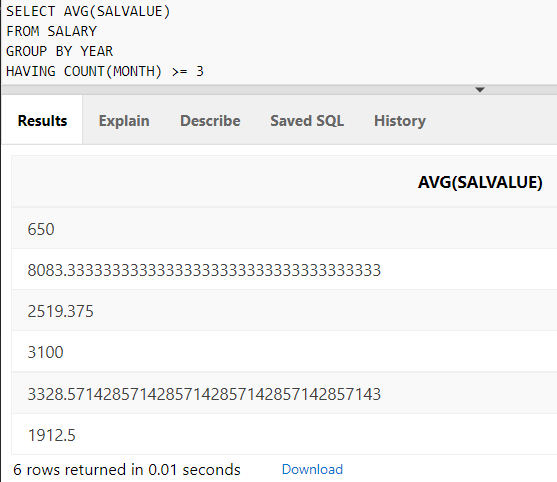
1. Определите среднюю зарплату за годы, в которые были начисления не менее чем за три месяца.

SELECT AVG(SALVALUE)

FROM SALARY

GROUP BY YEAR

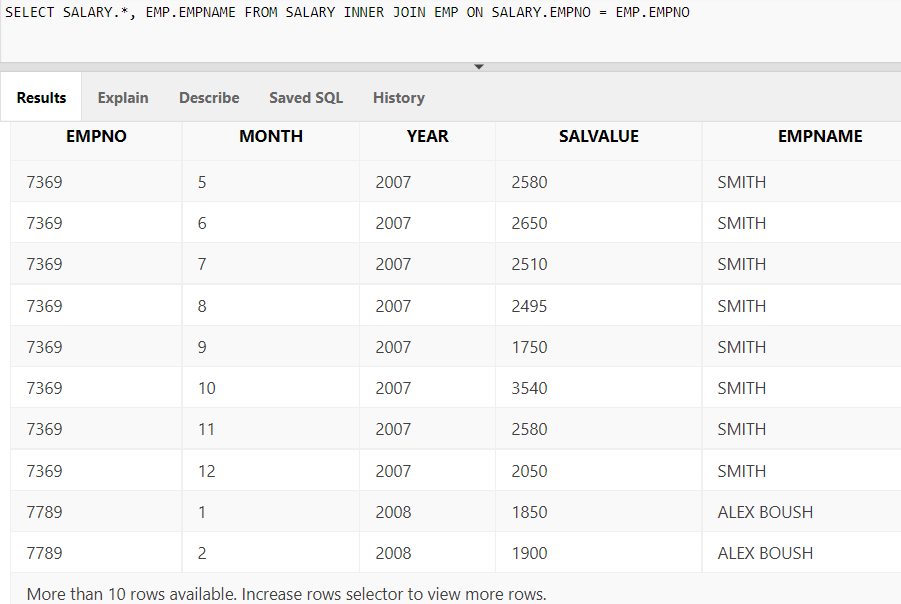
HAVING COUNT(MONTH) >= 3



**СОЕДИНЕНИЕ ПО РАВЕНСТВУ:**

1. Выведите ведомость получения зарплаты с указанием имен служащих.

SELECT SALARY.\*, EMP.EMPNAME FROM SALARY INNER JOIN EMP ON SALARY.EMPNO = EMP.EMPNO



**СОЕДИНЕНИЕ НЕ ПО РАВЕНСТВУ**:

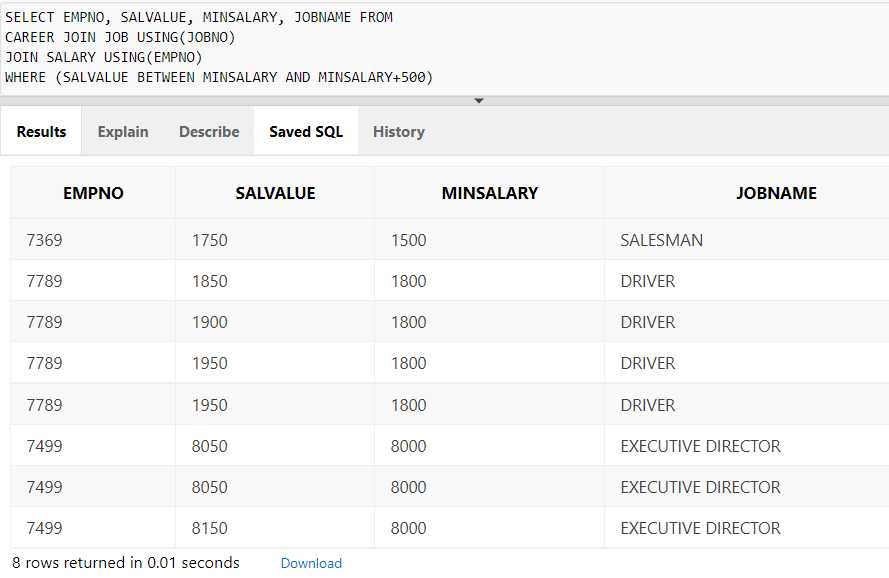
1. Укажите  сведения о начислении сотрудникам зарплаты, попадающей в вилку: минимальный оклад по должности - минимальный оклад по должности плюс пятьсот. Укажите соответствующую вилке  должность.

SELECT EMPNO, SALVALUE, MINSALARY, JOBNAME FROM

CAREER JOIN JOB USING(JOBNO)

JOIN SALARY USING(EMPNO)

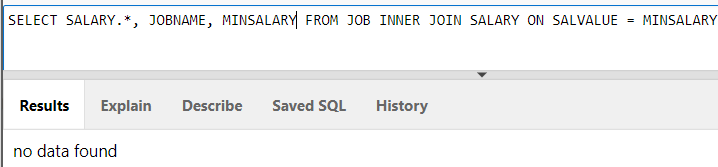
WHERE (SALVALUE BETWEEN MINSALARY AND MINSALARY+500)



**ОБЪЕДИНЕНИЕ ТАБЛИЦ:**

**ВНУТРЕННЕЕ:**

1. Укажите сведения о заработной плате, совпадающей с минимальными окладами по должностям (с указанием этих должностей).

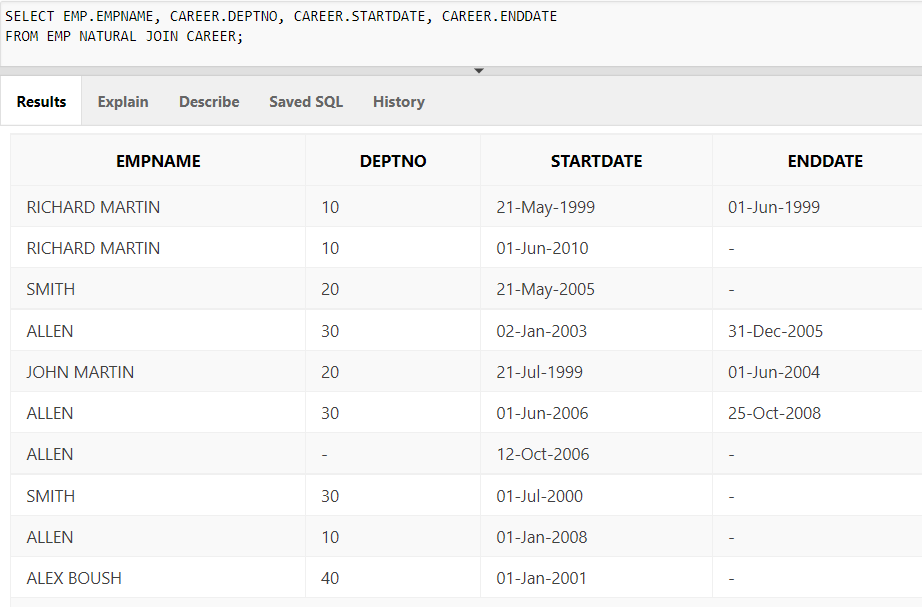
SELECT SALARY.\*, JOBNAME, MINSALARY FROM JOB INNER JOIN SALARY ON SALVALUE = MINSALARY

Ничего не найдено, т.к. ни одна зарплата не совпадает с минимальным окладом любой из должностей.

**ЕСТЕСТВЕННОЕ:**

1. Найдите  сведения о карьере сотрудников с указанием вместо номера сотрудника его имени.

SELECT EMP.EMPNAME, CAREER.DEPTNO, CAREER.STARTDATE, CAREER.ENDDATE FROM EMP NATURAL JOIN CAREER

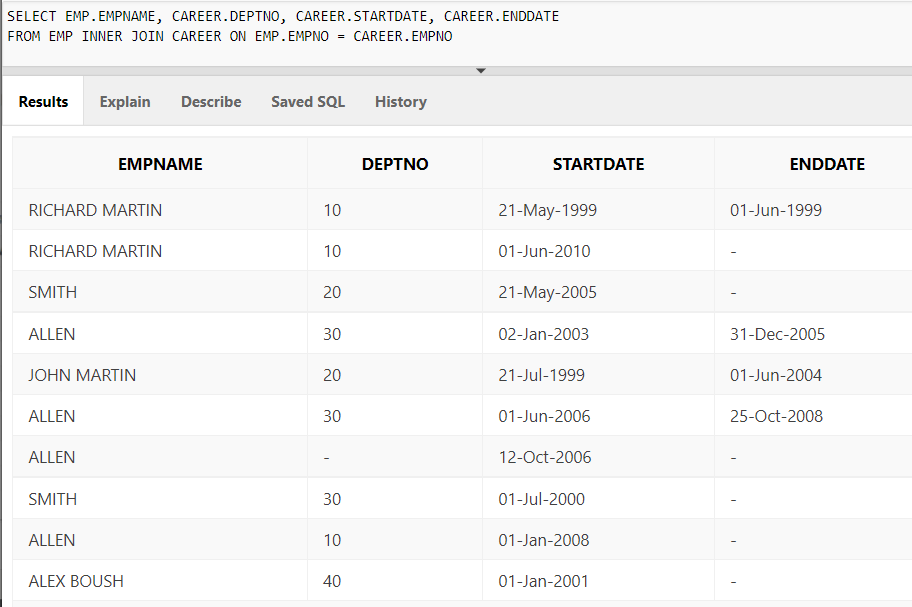


**ПРОСТОЕ ВНУТРЕННЕЕ СОЕДИНЕНИЕ:**

1. Найдите  сведения о карьере сотрудников с указанием вместо номера сотрудника его имени.

SELECT EMP.EMPNAME, CAREER.DEPTNO, CAREER.STARTDATE, CAREER.ENDDATE

FROM EMP INNER JOIN CAREER ON EMP.EMPNO = CAREER.EMPNO



**ОБЪЕДИНЕНИЕ ТРЁХ И БОЛЬШЕГО ЧИСЛА ТАБЛИЦ:**

1. Выдайте сведения о карьере сотрудников с указанием их имён, наименования должности, и названия отдела.

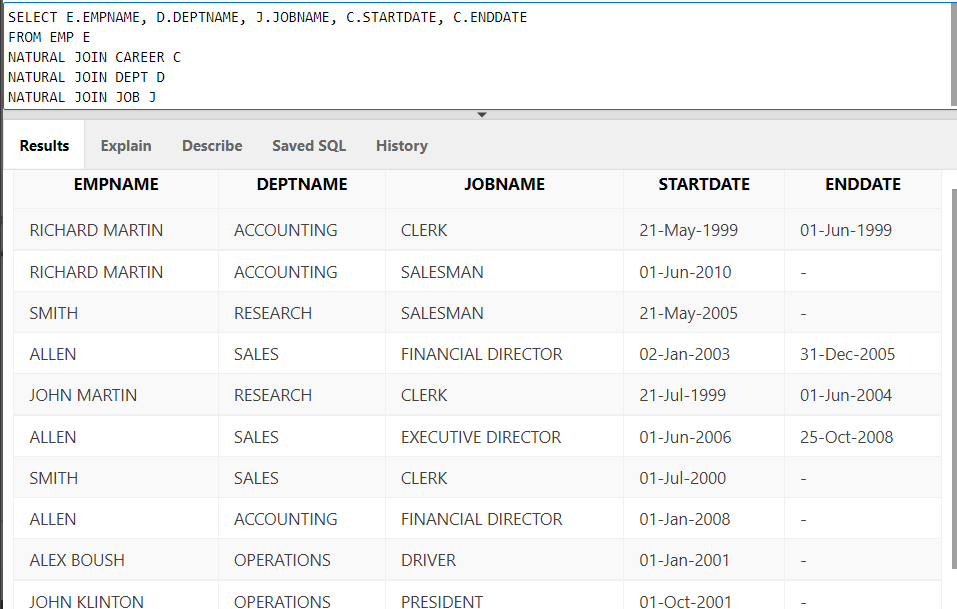
SELECT E.EMPNAME, D.DEPTNAME, J.JOBNAME, C.STARTDATE, C.ENDDATE

FROM EMP E

NATURAL JOIN CAREER C

NATURAL JOIN DEPT D

NATURAL JOIN JOB J



**ВНЕШНЕЕ ОБЪЕДИНЕНИЕ:**

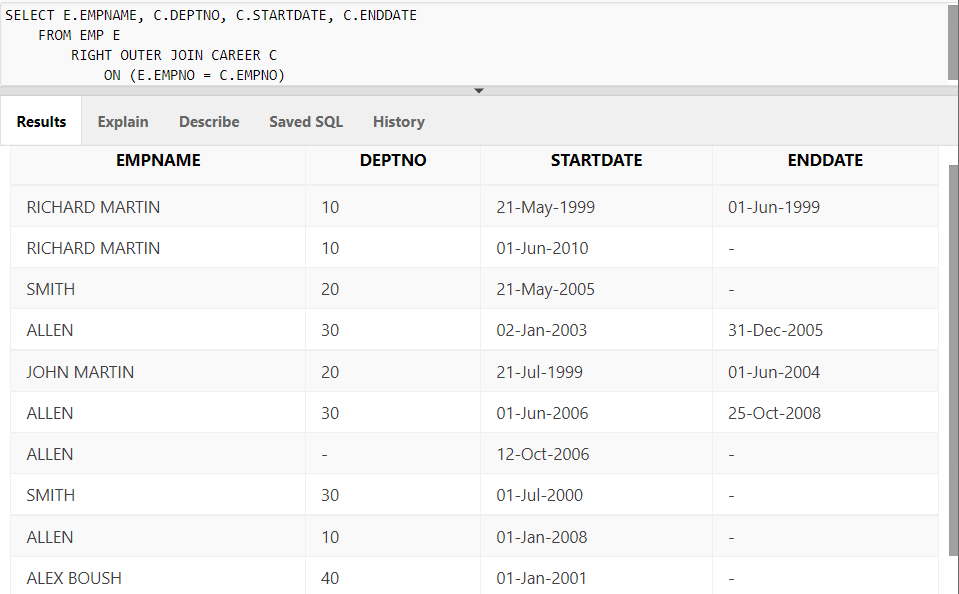
1. Выдайте сведения о карьере сотрудников с указанием их имён. Какой вид внешнего объединения Вы использовали? Составьте запрос с использованием противоположного вида соединения. Составьте запрос с использованием полного внешнего соединения.

SELECT E.EMPNAME, C.DEPTNO, C.STARTDATE, C.ENDDATE

FROM EMP E

RIGHT OUTER JOIN CAREER C

ON (E.EMPNO = C.EMPNO)



Использовала правое внешнее объединение.

Противоположное – левое внешнее:

SELECT E.EMPNAME, C.DEPTNO, C.STARTDATE, C.ENDDATE

FROM EMP E

LEFT OUTER JOIN CAREER C

ON (E.EMPNO = C.EMPNO)

Полное внешнее соединение:

SELECT E.EMPNAME, C.DEPTNO, C.STARTDATE, C.ENDDATE

FROM EMP E

FULL OUTER JOIN CAREER C

ON (E.EMPNO = C.EMPNO)

**Задание 3.2**

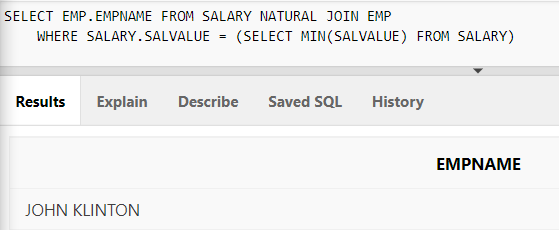
**ВЛОЖЕННЫЕ ПОДЗАПРОСЫ:**

**ПОДЗАПРОСЫ, ВЫБИРАЮЩИЕ ОДНУ СТРОКУ**

1. Найти имена сотрудников, получивших за годы начисления зарплаты минимальную зарплату.

SELECT EMP.EMPNAME FROM SALARY NATURAL JOIN EMP

WHERE SALARY.SALVALUE = (SELECT MIN(SALVALUE) FROM SALARY)



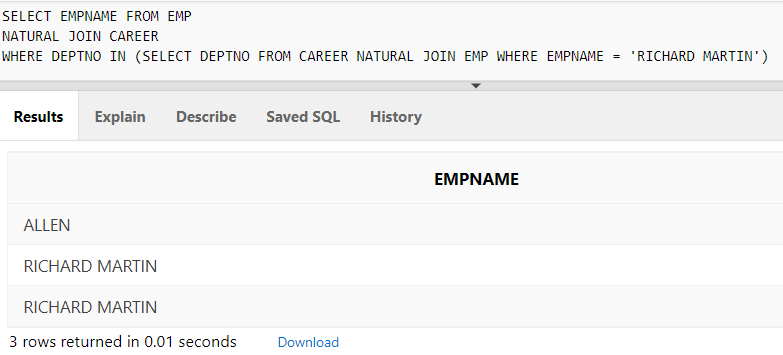
**ПОДЗАПРОСЫ, ВОЗВРАЩАЮЩИЕ БОЛЕЕ ОДНОЙ СТРОКИ**

1. Найти имена сотрудников, работавших или работающих в тех же отделах, в которых работал или работает сотрудник с именем RICHARD MARTIN.

SELECT EMPNAME FROM EMP

NATURAL JOIN CAREER

WHERE DEPTNO IN (SELECT DEPTNO FROM CAREER NATURAL JOIN EMP WHERE EMPNAME = 'RICHARD MARTIN')



Если не учитывать самого RICHARD MARTIN, то запрос будет выглядеть так:

SELECT EMPNAME FROM EMP

NATURAL JOIN CAREER

WHERE DEPTNO IN (SELECT DEPTNO FROM CAREER NATURAL JOIN EMP WHERE EMPNAME = 'RICHARD MARTIN')

AND NOT EMPNAME = 'RICHARD MARTIN'

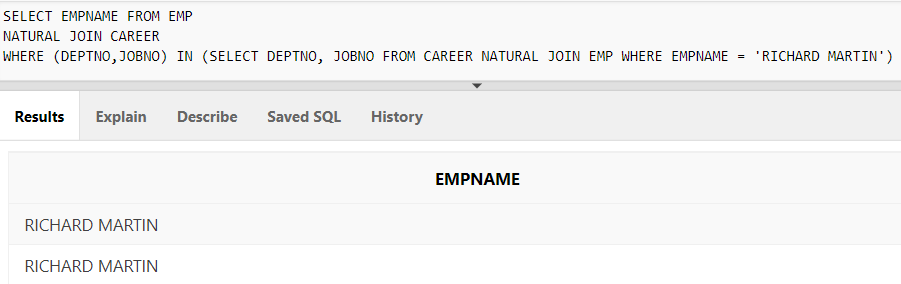
**СРАВНЕНИЕ БОЛЕЕ ЧЕМ ПО ОДНОМУ ЗНАЧЕНИЮ**

1. Найти имена сотрудников, работавших или работающих в тех же отделах и должностях, что и сотрудник 'RICHARD MARTIN'.

SELECT EMPNAME FROM EMP

NATURAL JOIN CAREER

WHERE (DEPTNO,JOBNO) IN (SELECT DEPTNO, JOBNO FROM CAREER NATURAL JOIN EMP WHERE EMPNAME = 'RICHARD MARTIN')



Если не учитывать самого RICHARD MARTIN, то запрос будет выглядеть так:

SELECT EMPNAME FROM EMP

NATURAL JOIN CAREER

WHERE (DEPTNO,JOBNO) IN (SELECT DEPTNO, JOBNO FROM CAREER NATURAL JOIN EMP WHERE EMPNAME = 'RICHARD MARTIN')

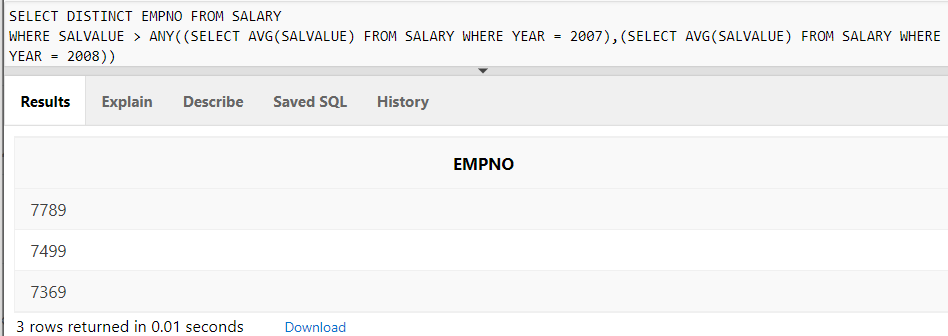
AND NOT EMPNAME = 'RICHARD MARTIN'

**ОПЕРАТОРЫ ANY/ALL**

4. Найти сведения о номерах сотрудников, получивших за какой-либо месяц зарплату большую, чем средняя зарплата за 2007 г. или большую чем средняя зарплата за 2008г.

SELECT DISTINCT EMPNO FROM SALARY

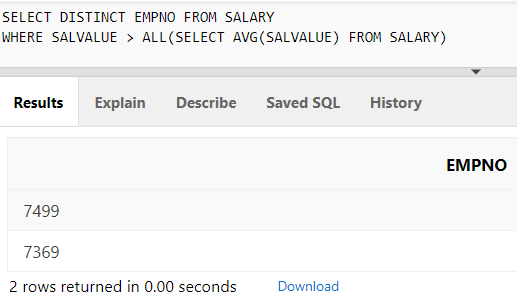
WHERE SALVALUE > ANY((SELECT AVG(SALVALUE) FROM SALARY WHERE YEAR = 2007),(SELECT AVG(SALVALUE) FROM SALARY WHERE YEAR = 2008))



1. Найти сведения о номерах сотрудников, получивших зарплату за какой-либо месяц большую, чем средние зарплаты за все годы начислений.

SELECT DISTINCT EMPNO FROM SALARY

WHERE SALVALUE > ALL(SELECT AVG(SALVALUE) FROM SALARY)



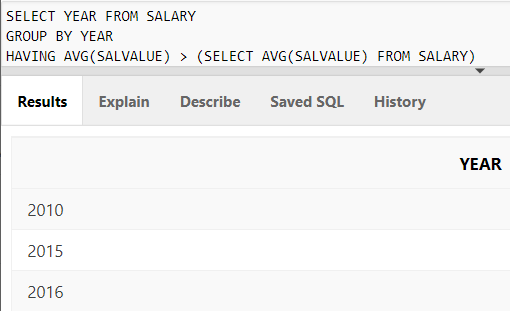
**ИСПОЛЬЗОВАНИЕ HAVING С ВЛОЖЕННЫМИ ПОДЗАПРОСАМИ**

1. Определить годы, в которые начисленная средняя зарплата была больше средней зарплаты за все годы начислений.

SELECT YEAR FROM SALARY

GROUP BY YEAR

HAVING AVG(SALVALUE) > (SELECT AVG(SALVALUE) FROM SALARY)

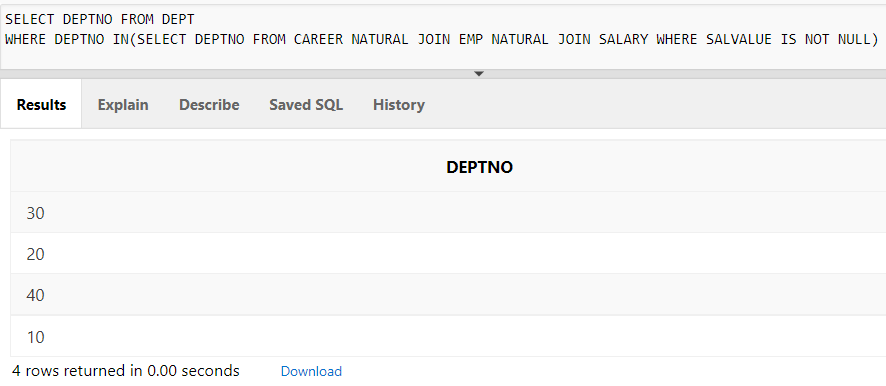


**КОРРЕЛИРУЮЩИЕ ПОДЗАПРОСЫ**

7. Определить номера отделов, в которых работали или работают сотрудники, имеющие начисления зарплаты.

SELECT DEPTNO FROM DEPT

WHERE DEPTNO IN(SELECT DEPTNO FROM CAREER NATURAL JOIN EMP NATURAL JOIN SALARY WHERE SALVALUE IS NOT NULL)

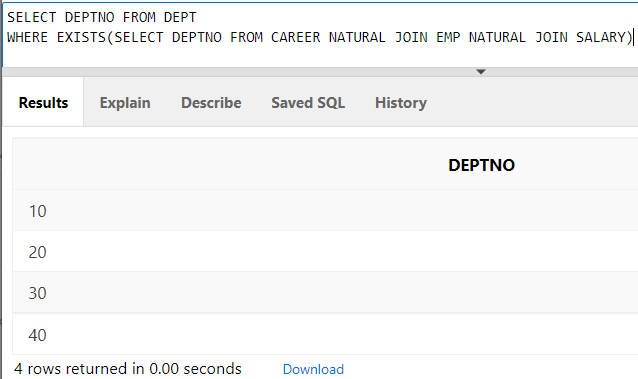


**ОПЕРАТОР EXISTS**

1. Определить номера отделов, в которых работали или работают сотрудники, имеющие начисления зарплаты.

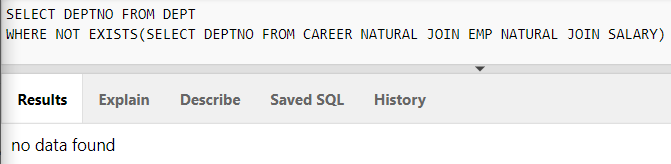
SELECT DEPTNO FROM DEPT

WHERE EXISTS(SELECT DEPTNO FROM CAREER NATURAL JOIN EMP NATURAL JOIN SALARY)



**ОПЕРАТОР NOT EXISTS**

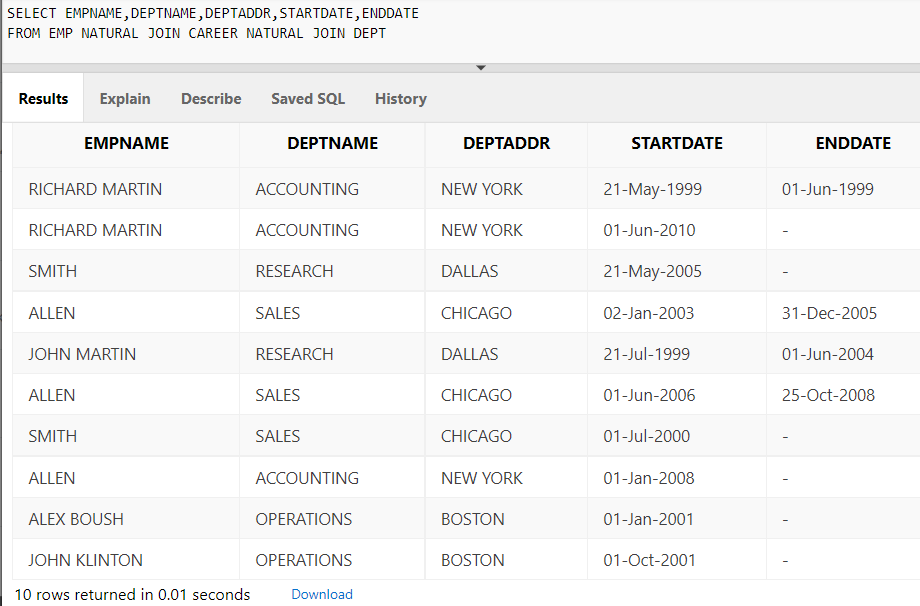
1. Определить номера отделов, для сотрудников которых не начислялась зарплата.



**СОСТАВНЫЕ ЗАПРОСЫ**

1. Вывести сведения о карьере сотрудников с указанием названий и адресов отделов вместо номеров отделов.

SELECT EMPNAME,DEPTNAME,DEPTADDR,STARTDATE,ENDDATE

FROM EMP NATURAL JOIN CAREER NATURAL JOIN DEPT 

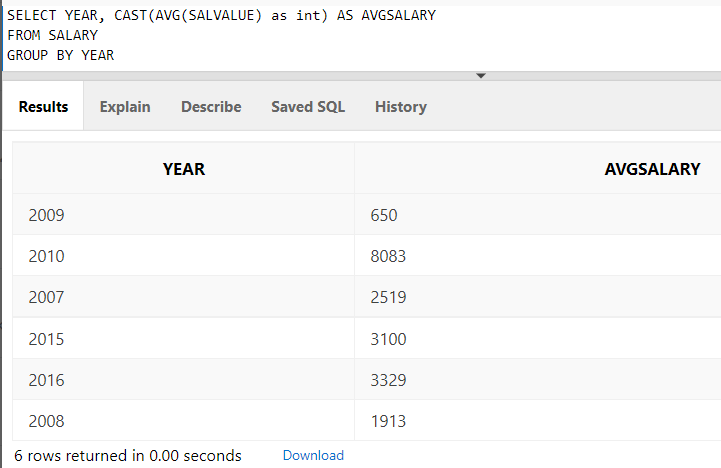
**ОПЕРАТОР CAST**

1. Определить целую часть средних зарплат,  по годам начисления.

SELECT YEAR, CAST(AVG(SALVALUE) as int) AS AVGSALARY

FROM SALARY

GROUP BY YEAR



**ОПЕРАТОР CASE**

1. Разделите сотрудников на возрастные группы: A) возраст 20-30 лет; B) 31-40 лет; C) 41-50;    D) 51-60 или возраст не определён.

SELECT EMPNO,EMPNAME,

CASE

WHEN MONTHS\_BETWEEN (SYSDATE, BIRTHDATE) / 12 BETWEEN 20 AND 30

THEN 'A'

WHEN MONTHS\_BETWEEN (SYSDATE, BIRTHDATE) / 12 BETWEEN 31 AND 40

THEN 'B'

WHEN MONTHS\_BETWEEN (SYSDATE, BIRTHDATE) / 12 BETWEEN 41 AND 50

THEN 'C'

WHEN MONTHS\_BETWEEN (SYSDATE, BIRTHDATE) / 12 BETWEEN 51 AND 60

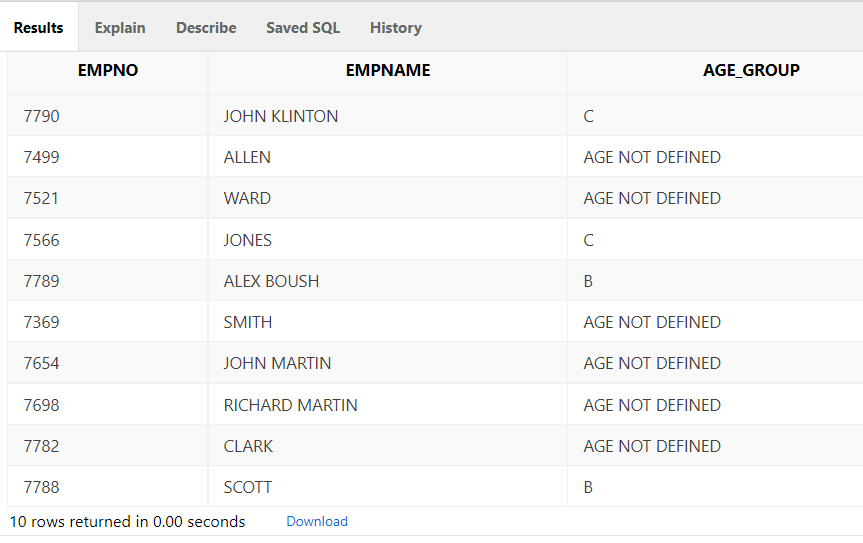
THEN 'D'

ELSE 'AGE NOT DEFINED'

END

AS AGE\_GROUP

FROM EMP



1. Перекодируйте номера отделов, добавив перед номером отдела буквы BI для номеров <=20,  буквы  LN для номеров >=30.

SELECT DEPTNO,

CASE

WHEN DEPTNO <= 20

THEN CONCAT('BI',DEPTNO)

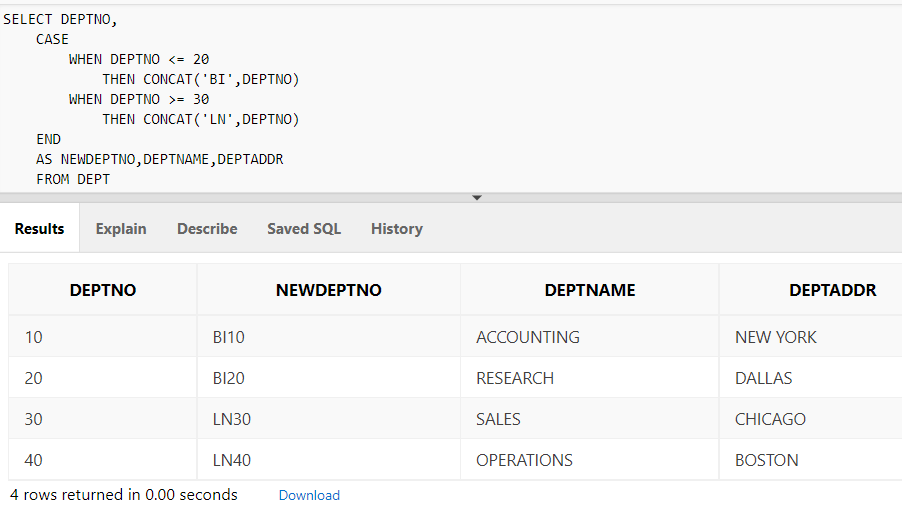
WHEN DEPTNO >= 30

THEN CONCAT('LN',DEPTNO)

END

AS NEWDEPTNO,DEPTNAME,DEPTADDR

FROM DEPT

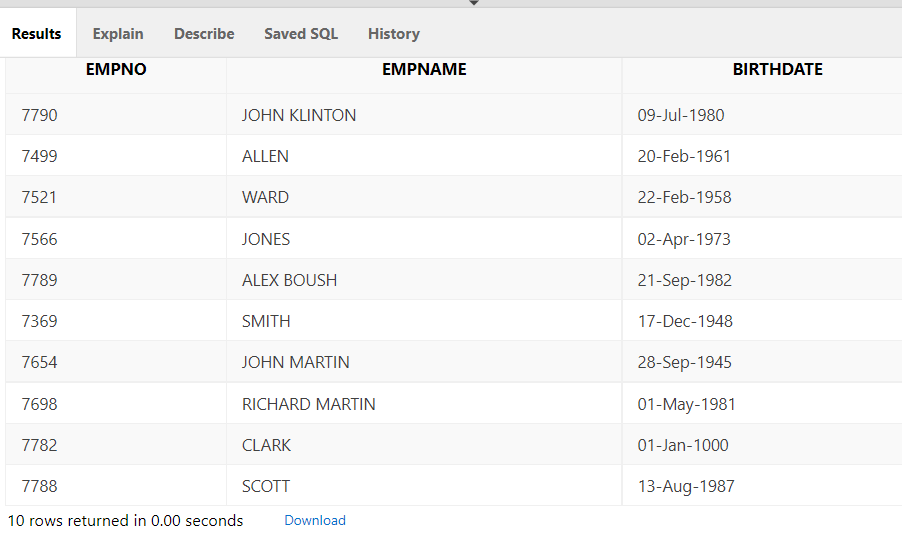


**ОПЕРАТОР COALESCE (объединяться)**

1. Выдать информацию о сотрудниках из таблицы EMP, заменив отсутствие данного о дате рождения  датой '01-01-1000'.

SELECT EMPNO,EMPNAME,COALESCE(BIRTHDATE, to\_date('01-01-1000', 'dd-mm-yyyy')) AS BIRTHDATE

FROM EMP



**Задание 3.3  Иерархические запросы**

1. **Рефлексивное соединение. Представление отношений родитель-потомок**

**Задача**

Требуется представить имя каждого сотрудника таблицы EMP, а также имя его руководителя.

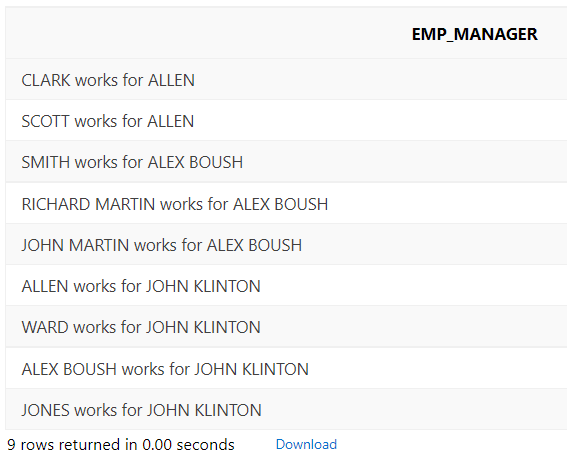
SELECT (A.EMPNAME || ' works for ' || B.EMPNAME) AS EMP\_MANAGER

FROM EMP A

JOIN EMP B

ON A.MANAGER\_ID = B.EMPNO

ORDER BY A.MANAGER\_ID;



1. **Иерархический запрос**

**Задача**

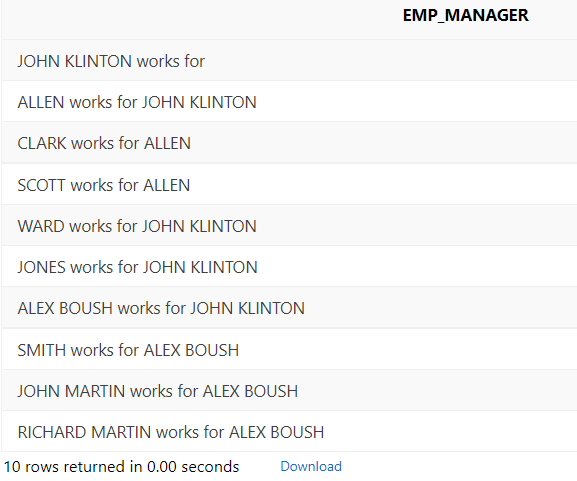
Требуется представить имя каждого сотрудника таблицы EMP (даже сотрудника, которому не назначен руководитель) и имя его руководителя.

SELECT (EMPNAME || ' works for ' || PRIOR EMPNAME) AS EMP\_MANAGER

FROM EMP

START WITH MANAGER\_ID IS NULL

CONNECT BY PRIOR EMPNO = MANAGER\_ID



1. **Представление отношений потомок-родитель-прародитель**

**Задача**

Требуется показать иерархию от CLARK до JOHN KLINTON.

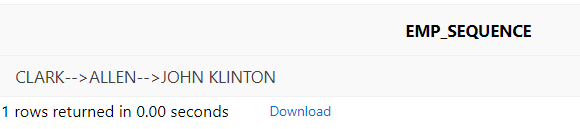
SELECT LTRIM(SYS\_CONNECT\_BY\_PATH(EMPNAME,'-->'),'-->') AS EMP\_Sequence

FROM EMP

WHERE EMPNAME = 'JOHN KLINTON'

START WITH EMPNAME = 'CLARK'

CONNECT BY EMPNO = PRIOR MANAGER\_ID



1. **Иерархическое представление таблицы**

**Задача**

Требуется получить результирующее множество, описывающее иерархию всей таблицы.

SELECT LTRIM(SYS\_CONNECT\_BY\_PATH(EMPNAME,'-->'),'-->') AS EMP\_Sequence

FROM EMP

START WITH MANAGER\_ID IS NULL

CONNECT BY PRIOR EMPNO = MANAGER\_ID

ORDER BY LEVEL



ORDER BY LEVEL – сортировка по уровню иерархии.

1. **Представление уровня иерархии**

**Задача**

Требуется показать уровень иерархии каждого сотрудника.

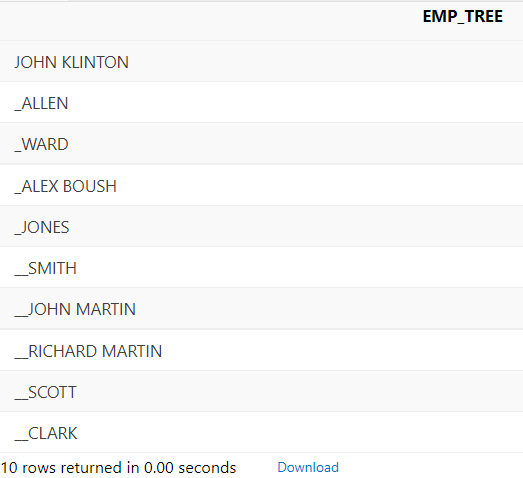
SELECT LPAD('\_', LEVEL-1, '\_') || EMPNAME AS EMP\_TREE

FROM EMP

START WITH MANAGER\_ID IS NULL

CONNECT BY PRIOR EMPNO = MANAGER\_ID

ORDER BY LEVEL



1. **Выбор всех дочерних строк для заданной строки**

**Задача**

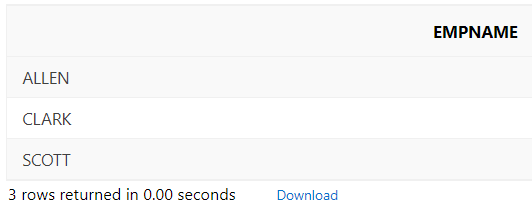
Требуется найти всех служащих, которые явно или  неявно подчиняются ALLEN.

SELECT EMPNAME

FROM EMP

START WITH EMPNAME = 'ALLEN'

CONNECT BY PRIOR EMPNO = MANAGER\_ID



**Задание 3.4  Работа с датами**

1. **Добавление, вычитание дней, месяцев, лет**

**Задача**

Требуется используя значения столбца START\_DATE получить дату за десять дней до и после приема на работу, пол года до и после приема на работу, год до и после приема на работу сотрудника JOHN KLINTON.

SELECT STARTDATE, ENDDATE,

(STARTDATE - 10) AS BEFORE\_10\_DAYS,

(ENDDATE + 10) AS AFTER\_10\_DAYS,

ADD\_MONTHS(STARTDATE, -6) AS BEFORE\_6\_MONTHS,

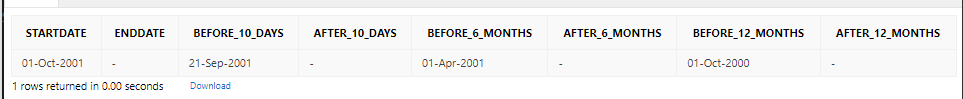
ADD\_MONTHS(ENDDATE, 6) AS AFTER\_6\_MONTHS,

ADD\_MONTHS(STARTDATE, -12) AS BEFORE\_12\_MONTHS,

ADD\_MONTHS(ENDDATE, 12) AS AFTER\_12\_MONTHS

FROM CAREER

WHERE EMPNO = (SELECT EMPNO FROM EMP WHERE EMPNAME = 'JOHN KLINTON');



1. **Определение количества дней между двумя датами.**

**Задача**

Требуется найти разность между двумя датами и представить результат в днях. Вычислите разницу в днях между датами приема на работу сотрудников JOHN MARTIN и ALEX BOUSH.

SELECT

ABS((SELECT STARTDATE

FROM CAREER

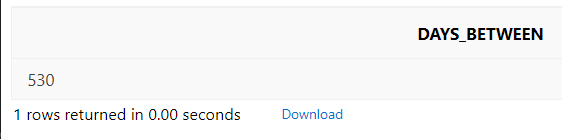
WHERE EMPNO = (SELECT EMPNO FROM EMP WHERE EMPNAME = 'JOHN MARTIN')) -

(SELECT STARTDATE

FROM CAREER

WHERE EMPNO = (SELECT EMPNO FROM EMP WHERE EMPNAME = 'ALEX BOUSH'))) AS DAYS\_BETWEEN

FROM DUAL



1. **Определение количества месяцев или лет между датами**

**Задача**

Требуется найти разность между двумя датами в месяцах и в годах.

SELECT

MONTHS\_BETWEEN(SYSDATE, TO\_DATE('29-04-2000', 'DD-MM-YYYY')) AS MONTHS\_BETWEEN,

(MONTHS\_BETWEEN(SYSDATE, TO\_DATE('29-04-2000', 'DD-MM-YYYY')) / 12) AS YEARS\_BETWEEN

FROM DUAL



1. **Определение интервала времени между текущей и следующей записями**

**Задача**

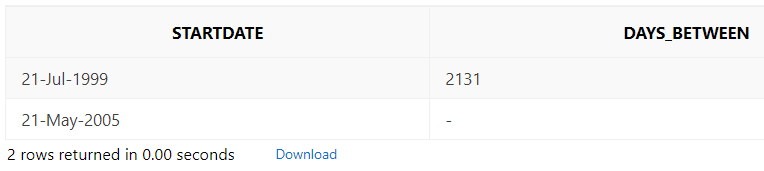
Требуется определить интервал времени в днях между двумя датами. Для каждого сотрудника 20-го отдела найти сколько дней прошло между датой его приема на работу и датой приема на работу следующего сотрудника.

SELECT STARTDATE,

((LEAD(STARTDATE, 1) OVER (ORDER BY STARTDATE)) - STARTDATE) AS DAYS\_BETWEEN

FROM CAREER

WHERE DEPTNO = '20'



1. **Определение количества дней в году**

**Задача**

Требуется подсчитать количество дней в году по столбцу START\_DATE.

**Рекомендации**

Используйте функцию TRUNC для нахождения начала года, а ADD\_MONTHS – для нахождения начала следующего года.

SELECT EXTRACT(YEAR FROM STARTDATE) AS YEAR,

(ADD\_MONTHS(TRUNC(STARTDATE, 'y'), 12) - TRUNC(STARTDATE, 'y')) AS DAYS

FROM CAREER



1. **Извлечение единиц времени из даты**

**Задача**

Требуется разложить текущую дату на день, месяц, год, секунды, минуты, часы. Результаты вернуть в численном виде.

SELECT TO\_CHAR(SYSDATE, 'DD.MM.YYYY HH24:MI:SS') AS DATE\_CHAR,

TO\_NUMBER(TO\_CHAR(SYSDATE, 'DD')) AS day,

TO\_NUMBER(TO\_CHAR(SYSDATE, 'MM')) AS month,

TO\_NUMBER(TO\_CHAR(SYSDATE, 'YYYY')) AS year,

TO\_NUMBER(TO\_CHAR(SYSDATE, 'HH24')) AS hour,

TO\_NUMBER(TO\_CHAR(SYSDATE, 'MI')) AS minute,

TO\_NUMBER(TO\_CHAR(SYSDATE, 'SS')) AS second

FROM DUAL



1. **Определение первого и последнего дней месяца**

**Задача**

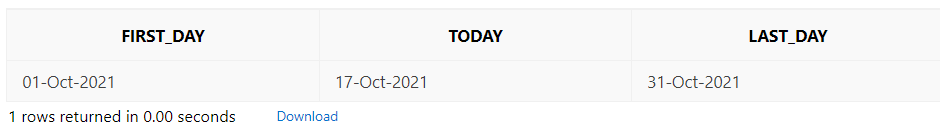
Требуется получить первый и последний дни текущего месяца.

SELECT TRUNC(LAST\_DAY(SYSDATE) - 1, 'mm') AS FIRST\_DAY,

SYSDATE AS TODAY,

LAST\_DAY(SYSDATE) AS LAST\_DAY

FROM DUAL



1. **Выбор всех дат года, выпадающих на определенный день недели**

**Задача**

Требуется возвратить даты начала и конца каждого из четырех кварталов данного года.

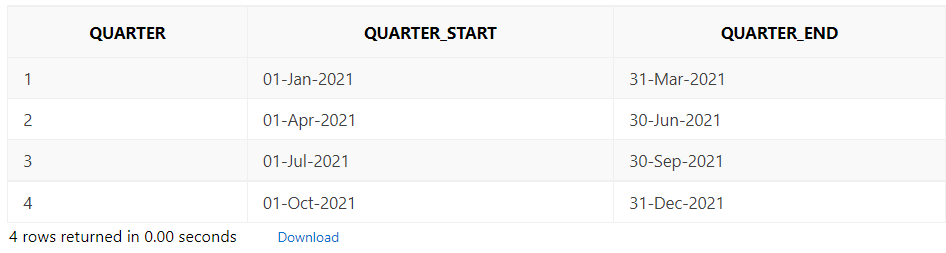
SELECT LEVEL AS QUARTER,

ADD\_MONTHS(trunc(sysdate, 'YEAR'), (LEVEL - 1) \* 3) AS QUARTER\_START,

ADD\_MONTHS(trunc(sysdate, 'YEAR'), LEVEL \* 3) - 1 AS QUARTER\_END

FROM DUAL

CONNECT BY LEVEL <= 4;



1. **Выбор всех дат года, выпадающих на определенный день недели**

**Задача**

Требуется найти все даты года, соответствующие заданному дню недели. Сформируйте список понедельников текущего года.

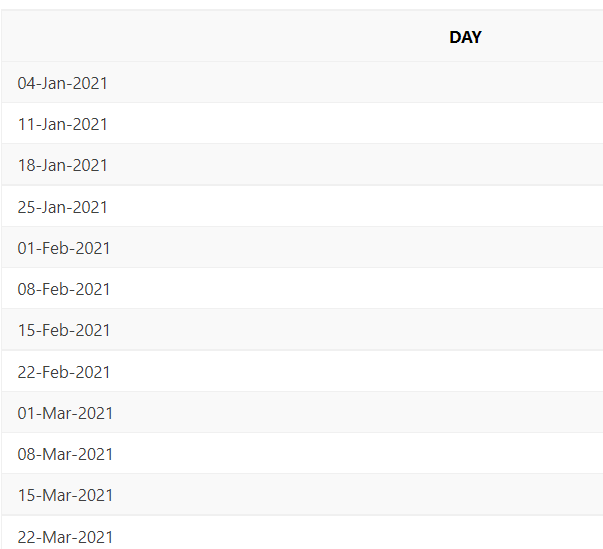
SELECT \*

FROM (SELECT (trunc(sysdate, 'YEAR') + LEVEL - 1) AS DAY

FROM DUAL

CONNECT BY LEVEL <= 366)

WHERE TO\_CHAR(DAY, 'fmday') = 'monday'



1. **Создание календаря**

**Задача**

Требуется создать календарь на текущий месяц. Календарь должен иметь семь столбцов в ширину и пять строк вниз.

WITH X AS (SELECT \* FROM (

TO\_CHAR(TRUNC(SYSDATE, 'MM') + LEVEL - 1, 'IW') WEEK\_NUMBER,

TO\_CHAR(TRUNC(SYSDATE, 'MM') + LEVEL - 1, 'DD') DAY\_NUMBER,

TO\_NUMBER(TO\_CHAR(TRUNC(SYSDATE, 'MM')+ LEVEL - 1, 'D')) WEEK\_DAY,

TO\_CHAR(TRUNC(SYSDATE, 'MM') + LEVEL - 1, 'MM') CURR\_MONTH,

TO\_CHAR(SYSDATE, 'MM') MONTH

FROM DUAL

CONNECT BY LEVEL <= 31)

WHERE CURR\_MONTH = MONTH)

SELECT MAX(CASE WEEK\_DAY WHEN 2 THEN DAY\_NUMBER END) AS MONDAY,

MAX(CASE WEEK\_DAY WHEN 3 THEN DAY\_NUMBER END) AS TUESDAY,

MAX(CASE WEEK\_DAY WHEN 4 THEN DAY\_NUMBER END) AS WEDNESDAY,

MAX(CASE WEEK\_DAY WHEN 5 THEN DAY\_NUMBER END) AS THURSDAY,

MAX(CASE WEEK\_DAY WHEN 6 THEN DAY\_NUMBER END) AS FRIDAY,

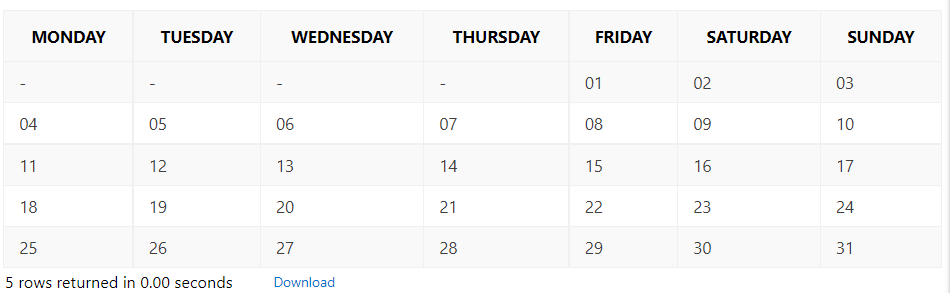
MAX(CASE WEEK\_DAY WHEN 7 THEN DAY\_NUMBER END) AS SATURDAY,

MAX(CASE WEEK\_DAY WHEN 1 THEN DAY\_NUMBER END) AS SUNDAY

FROM X

GROUP BY WEEK\_NUMBER

ORDER BY WEEK\_NUMBER

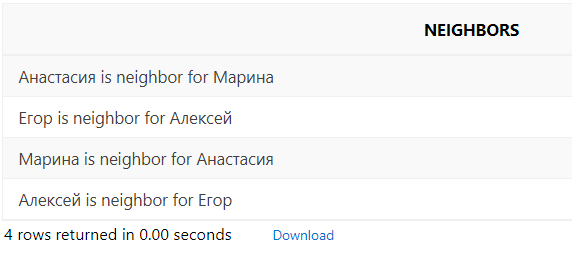


**Задание 3.5 индивидуальное**

1. Составьте запросы на выборку данных с использованием рефлексивного соединения для таблицы из задания 5 лабораторной работы №2.

SELECT (G.Имя ||' is neighbor for '|| Gu.Имя) as Neighbors FROM Гость G join Гость Gu

on G.сосед\_ID = Gu.id



Составьте запросы на выборку данных с использованием следующих операторов, конструкций и функций языка SQL:

1. простого оператора CASE;

SELECT id, Площадь,

CASE Сектор\_id

WHEN 1

THEN 'рекреационный'

WHEN 2

THEN 'хозяйственный'

WHEN 3

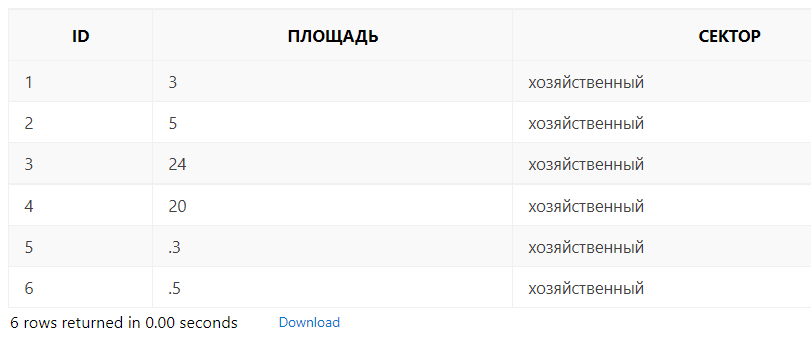
THEN 'жилой'

ELSE 'неизвестный'

END

AS сектор

FROM Участок



1. поискового оператора CASE;

SELECT id, Площадь,

(CASE

WHEN Площадь < 1

THEN 'маленький'

WHEN Площадь BETWEEN 1 AND 20

THEN 'средний'

WHEN Площадь BETWEEN 21 AND 40

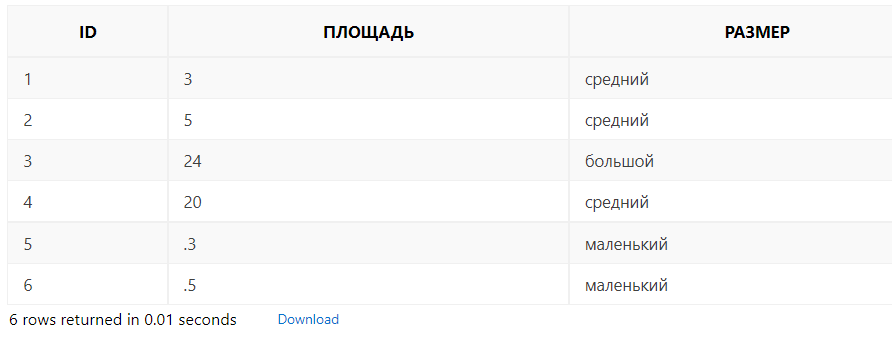
THEN 'большой'

ELSE 'огромный'

END)

AS размер

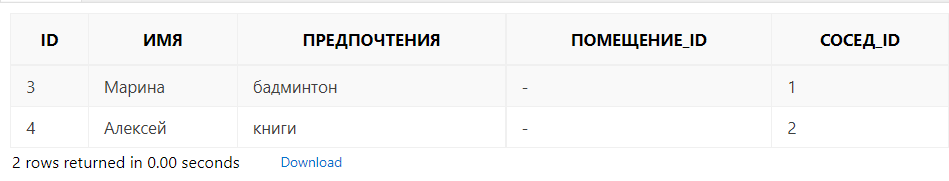
FROM Участок



1. оператора WITH;

with people as (select \* from Гость where Предпочтения not in ('шашки','шахматы'))

select \* from people where (сосед\_id is not null)



1. встроенного представления;

SELECT тип as сектор, id as id\_сектора, площадь as площадь\_участка, культура\_id, СТОИМОСТЬ

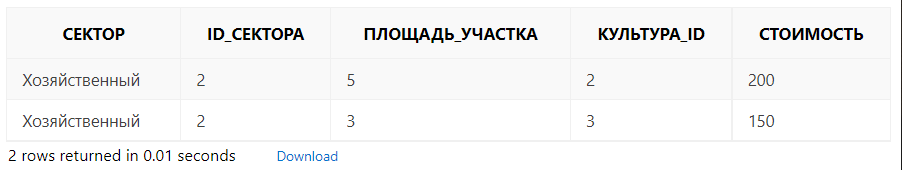
(SELECT \* FROM Культура k

INNER JOIN Участок u on u.культура\_id = k.id

INNER JOIN Сектор s on u.сектор\_id = s.id

WHERE k.стоимость < s.затраты)

WHERE площадь <= 20



1. некоррелированного запроса;

SELECT \* FROM Жилое\_помещение

WHERE id IN

(SELECT Помещение\_id

FROM Гость

GROUP BY Помещение\_id)

ORDER BY id



1. коррелированного запроса;

SELECT \* FROM Сектор S INNER JOIN Жилое\_помещение P ON S.ID = P.Сектор\_ID

WHERE P.id IN

(SELECT Помещение\_id

FROM Гость

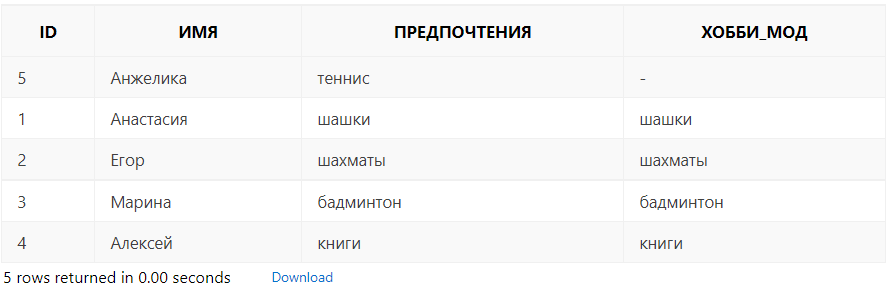
WHERE P.ПЛОЩАДЬ / P.СПАЛЬНЫХ\_МЕСТ >= 20)

ORDER BY p.id



1. функции NULLIF;

SELECT id, Имя, предпочтения, NULLIF(Предпочтения, 'теннис') as Хобби\_мод FROM Гость

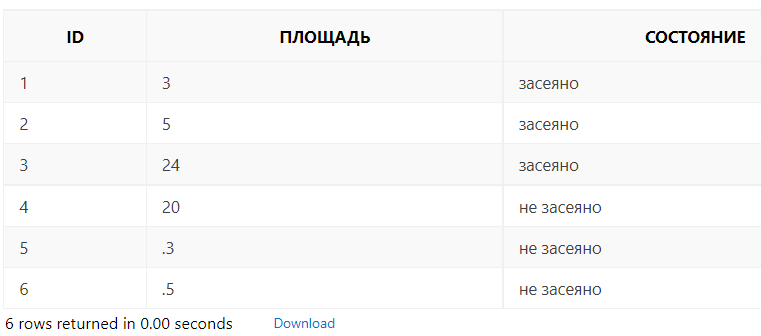


1. функции NVL2;

SELECT id,площадь, NVL2(культура\_id, 'засеяно', 'не засеяно') AS Состояние

FROM Участок

ORDER BY ID



1. TOP-N анализа;

SELECT ROWNUM as Num, id, СТОИМОСТЬ, КОЛВО\_УДОБРЕНИЙ, НОРМА\_ПОСЕВА, РЕЖИМ\_ПОЛИВА, ВЕГЕТ\_ПЕРИОД,

СРОК\_УБОРКИ, ВОДА\_ПОЛИВ

FROM (SELECT \*

FROM Культура

ORDER BY СТОИМОСТЬ DESC,КОЛВО\_УДОБРЕНИЙ DESC)

WHERE ROWNUM<=3



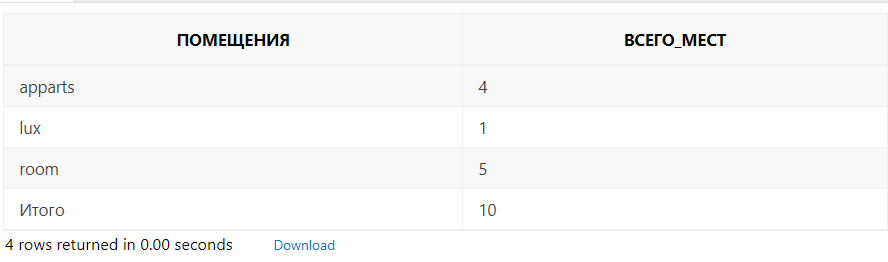
1. функции ROLLUP.

SELECT coalesce(тип, 'Итого') AS Помещения,

SUM(спальных\_мест) AS Всего\_мест

FROM Жилое\_помещение

GROUP BY ROLLUP(тип)



1. Составьте запрос на использование оператора MERGE языка манипулирования данными.

MERGE INTO Сектор1 Target

USING Сектор Source

ON (Target.id = Source.id)

WHEN MATCHED THEN

UPDATE SET Target.затраты = Source.затраты, Target.тип = Source.тип

WHEN NOT MATCHED THEN

INSERT VALUES (Source.id, Source.затраты, Source.тип);