**ОТЧЕТ**

**По лабораторной работе №4**

**«Язык SQL. Операторы манипулирования данными»**

**Вариант 8**

Студентка группы 13  
Врублевская Е. А.

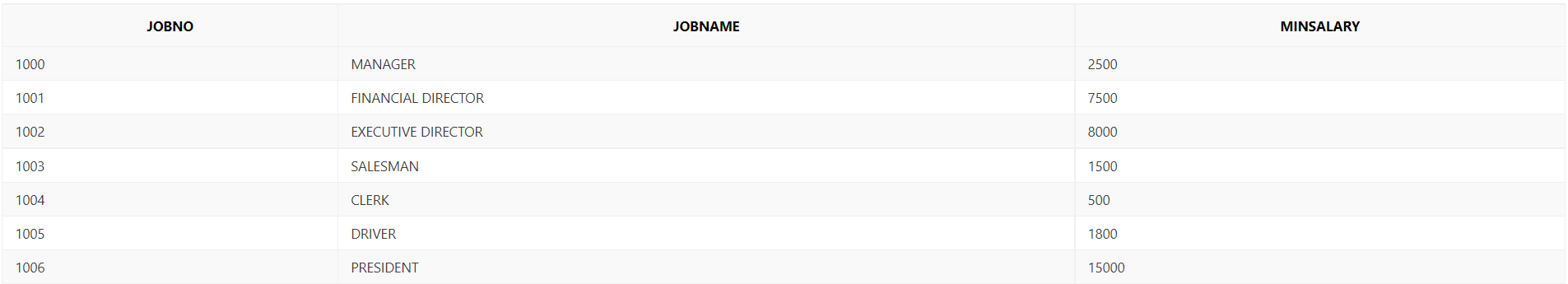
**Задание 1**

1. Поднимите нижнюю границу минимальной заработной платы в таблице JOB до 1000$.

Выполнение.

Исходное состояние таблицы

SELECT \* FROM JOB;



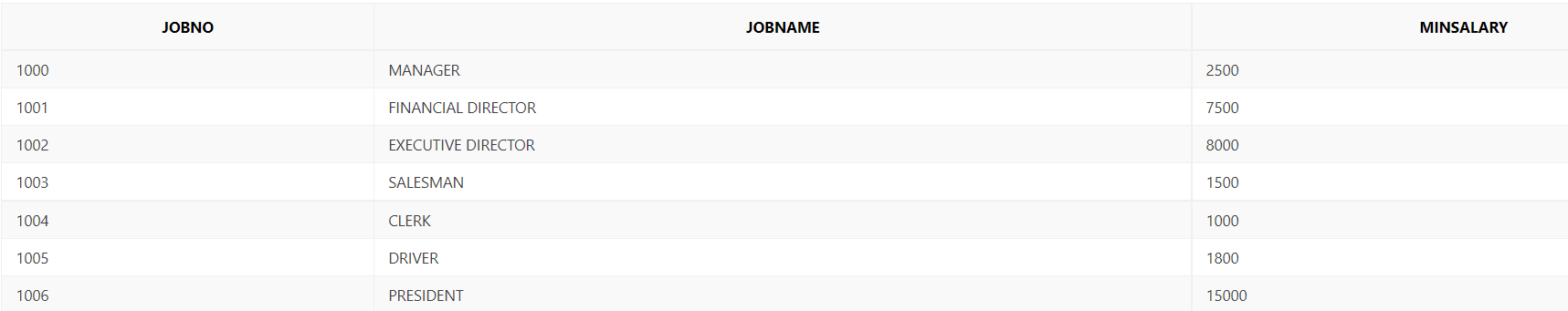
Решение задачи

**UPDATE JOB**

**SET MINSALARY = 1000**

**WHERE MINSALARY < 1000;**

Состояние таблицы после изменения

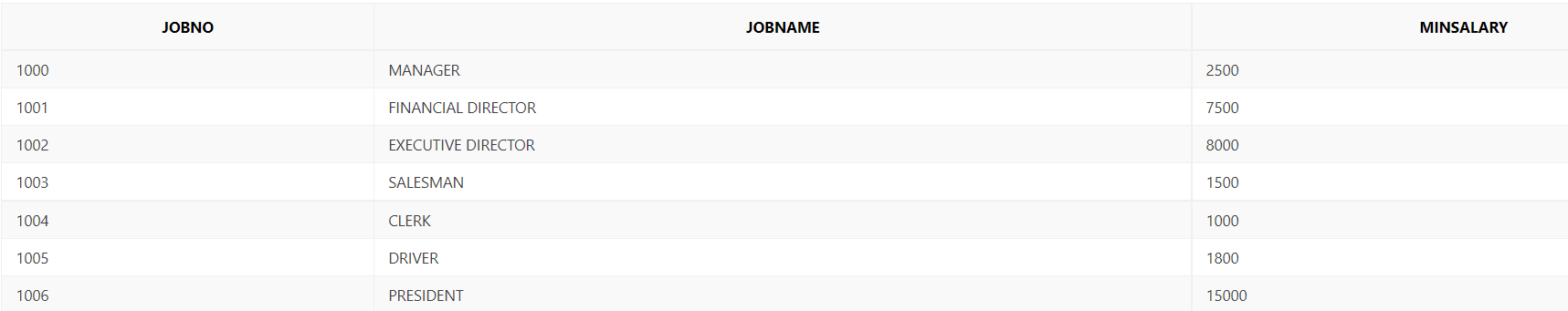


1. Поднимите минимальную зарплату в таблице JOB на 10% для всех специальностей,

кроме финансового директора.

Исходное состояние таблицы

SELECT \* FROM JOB;



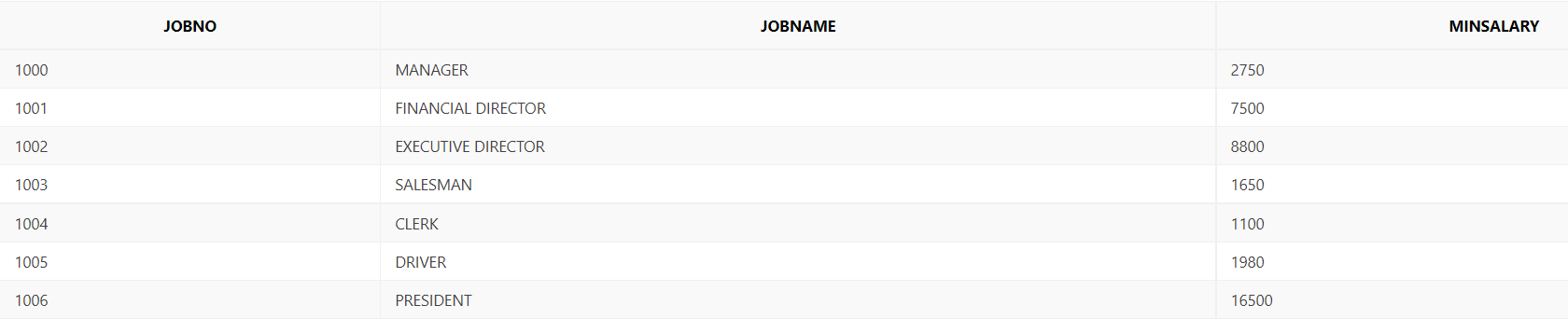
Решение задачи

**UPDATE JOB**

**SET MINSALARY = 1.1 \* MINSALARY**

**WHERE JOBNAME != 'FINANCIAL DIRECTOR';**

Состояние таблицы после изменения

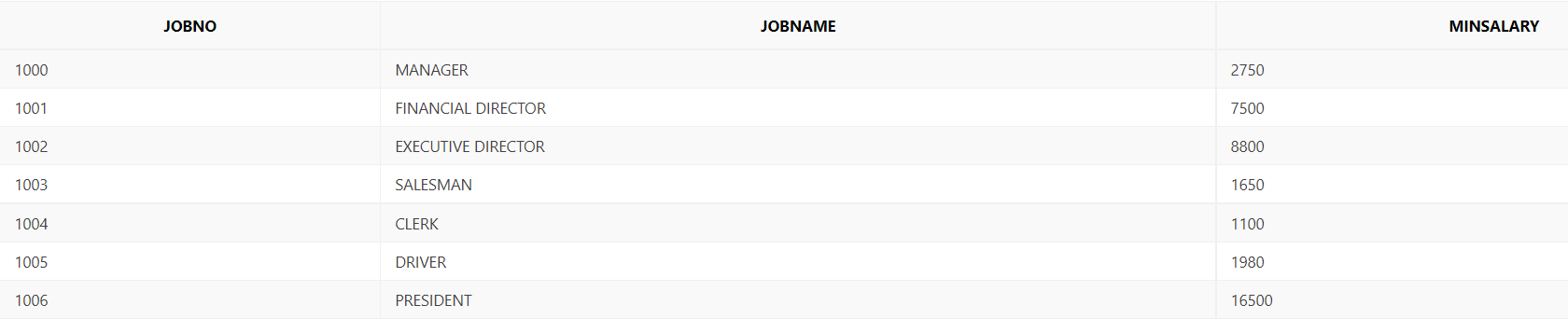


1. Поднимите минимальную зарплату в таблице JOB на 10% для клерков и на 20% для

финансового директора (одним оператором).

Исходное состояние таблицы

SELECT \* FROM JOB;



Решение задачи

**UPDATE JOB**

**SET MINSALARY = MINSALARY \* (**

**CASE WHEN JOBNAME = 'CLERK'**

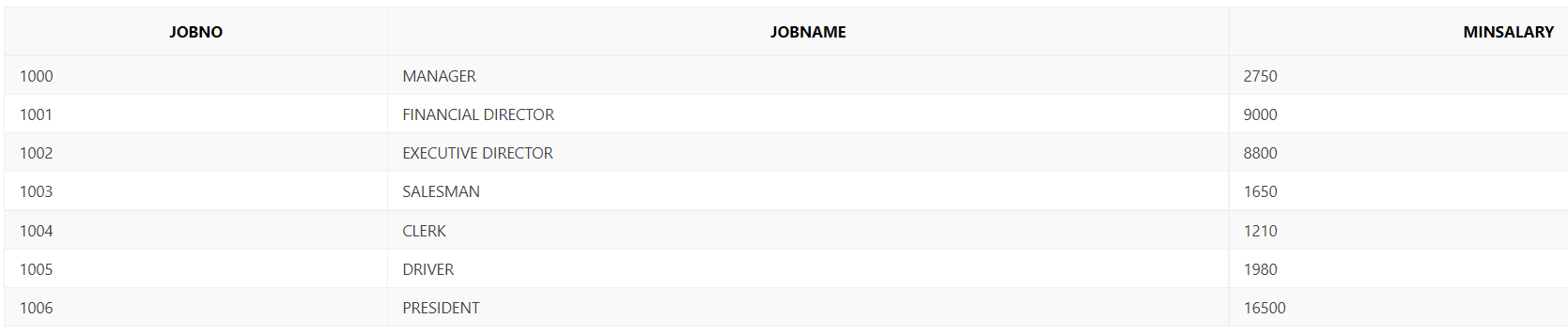
**THEN 1.1**

**ELSE 1.2**

**END)**

**WHERE JOBNAME IN ('FINANCIAL DIRECTOR', 'CLERK');**

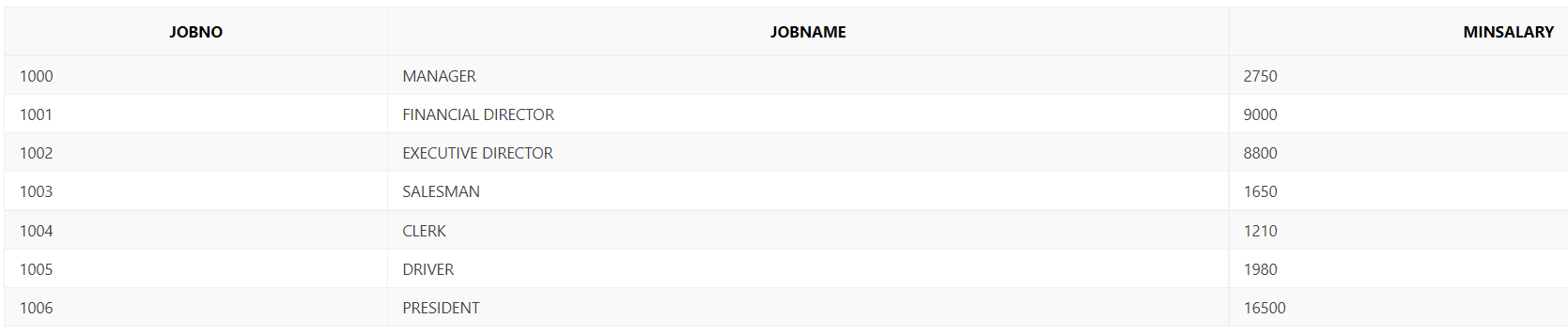
Состояние таблицы после изменения



1. Установите минимальную зарплату финансового директора равной 90% от зарплаты исполнительного директора.

Исходное состояние таблицы

SELECT \* FROM JOB;



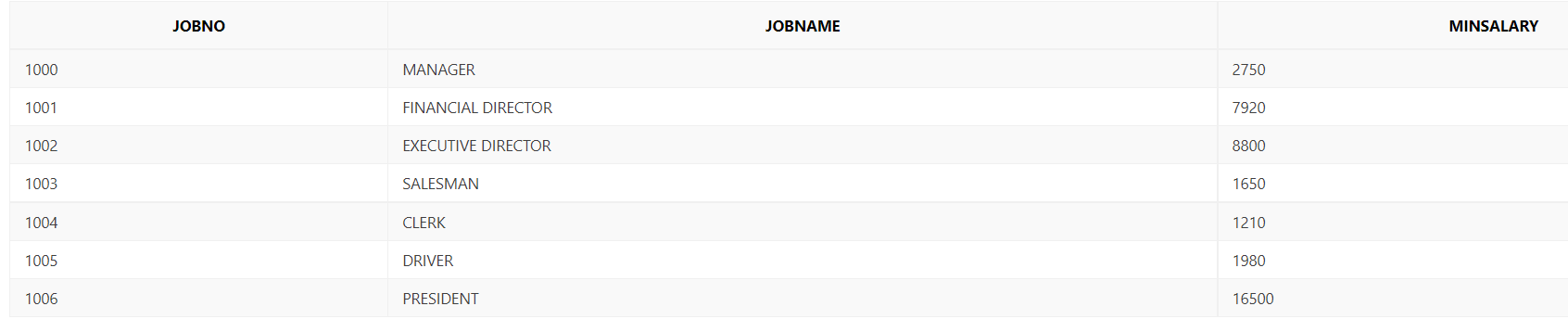
Решение задачи

**UPDATE JOB**

**SET MINSALARY = 0.9\*(SELECT MINSALARY FROM JOB WHERE JOBNAME = 'EXECUTIVE DIRECTOR')**

**WHERE JOBNAME = 'FINANCIAL DIRECTOR';**

Состояние таблицы после изменения



1. Приведите в таблице EMP имена служащих, начинающиеся на букву ‘J’, к нижнему регистру.

Исходное состояние таблицы

SELECT \* FROM EMP;



Решение задачи

**UPDATE EMP**

**SET EMPNAME = LOWER(EMPNAME)**

**WHERE EMPNAME LIKE 'J%';**

Состояние таблицы после изменения



1. Измените в таблице EMP имена служащих, состоящие из двух слов, так, чтобы оба слова в имени начинались с заглавной буквы, а продолжались прописными.

Исходное состояние таблицы

SELECT \* FROM EMP;



Решение задачи

**UPDATE EMP**

**SET EMPNAME = INITCAP(EMPNAME)**

**WHERE EMPNAME LIKE '% %';**

Состояние таблицы после изменения



1. Приведите в таблице EMP имена служащих к верхнему регистру.

Исходное состояние таблицы

SELECT \* FROM EMP;

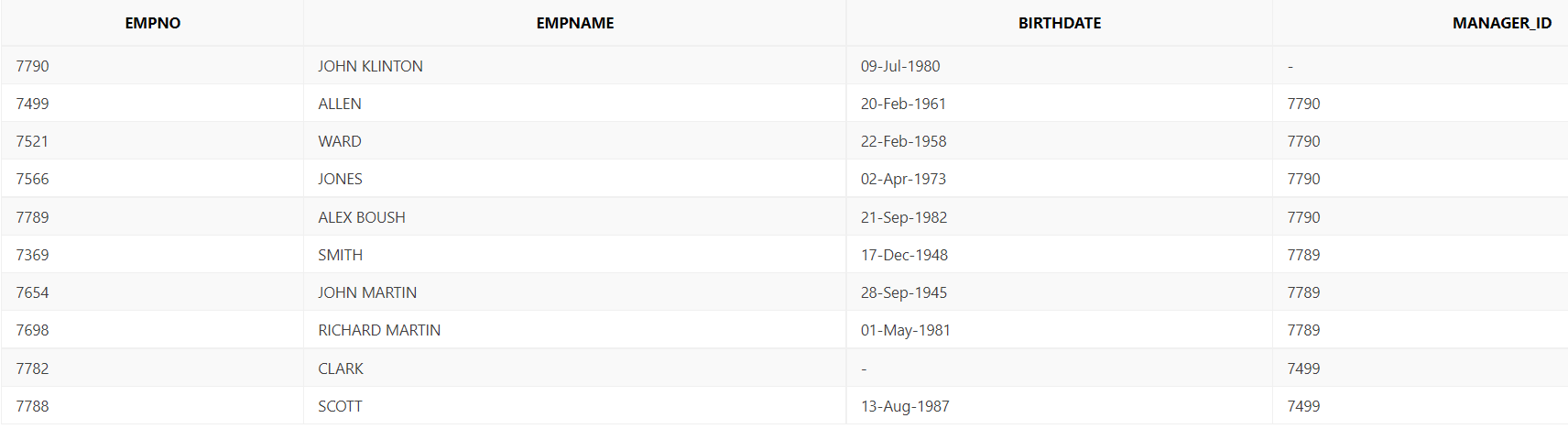


Решение задачи

**UPDATE EMP**

**SET EMPNAME = UPPER(EMPNAME);**

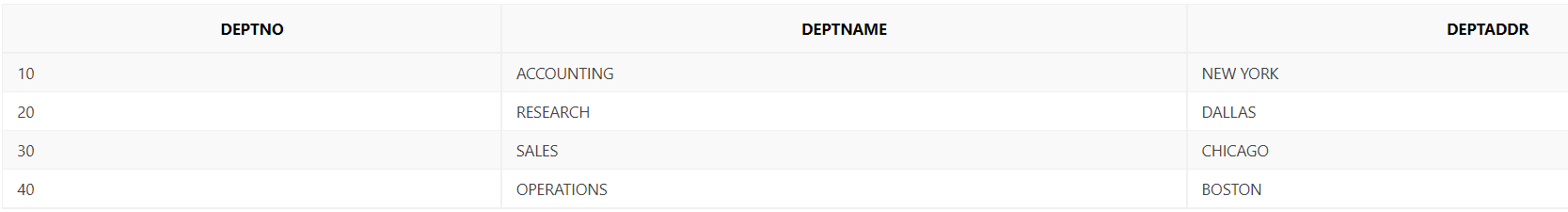
Состояние таблицы после изменения



1. Перенесите отдел исследований (RESEARCH) в тот же город, в котором расположен отдел продаж (SALES).

Исходное состояние таблицы

SELECT \* FROM DEPT;



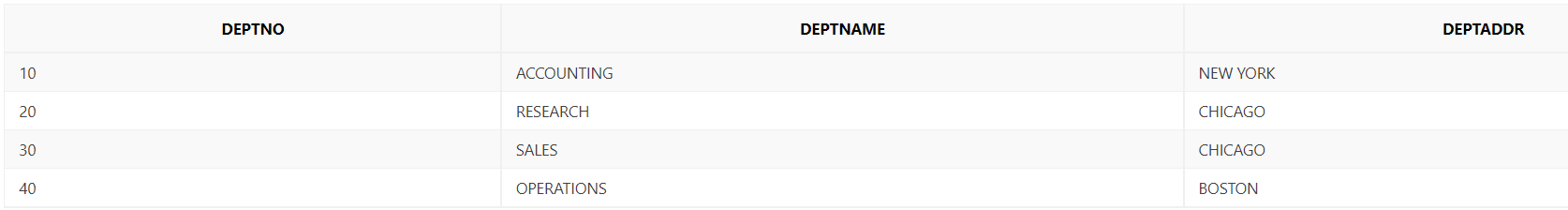
Решение задачи

**UPDATE DEPT**

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

**WHERE DEPTNAME = 'RESEARCH';**

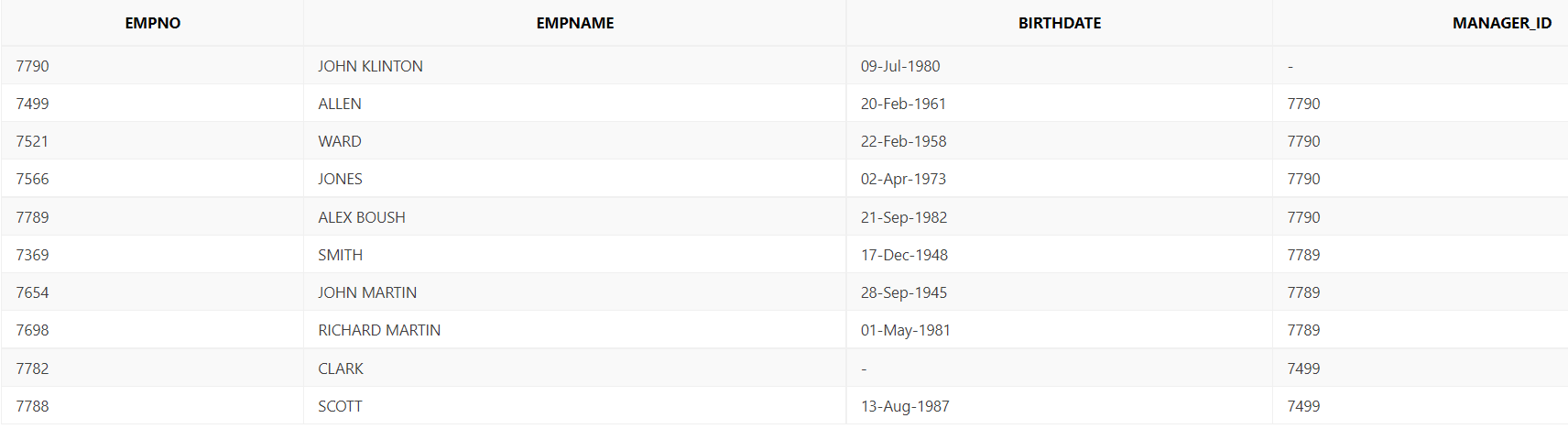
Состояние таблицы после изменения



1. Добавьте нового сотрудника в таблицу EMP. Его имя и фамилия должны совпадать с Вашими, записанными латинскими буквами согласно паспорту, дата рождения также совпадает с Вашей.

Исходное состояние таблицы

SELECT \* FROM EMP;



Решение задачи

**INSERT INTO EMP VALUES ((SELECT MAX(EMPNO) FROM EMP) + 1, 'EKATERINA VROUBLEVSKAIA', TO\_DATE('29.07.2002', 'dd.mm.yyyy'),7789);**

Состояние таблицы после изменения



1. Определите нового сотрудника (см. предыдущее задание) на работу в бухгалтерию (отдел ACCOUNTING) начиная с текущей даты.

Исходное состояние таблицы

SELECT \* FROM EMP;



Решение задачи

**INSERT INTO CAREER VALUES (**

**(SELECT JOBNO FROM JOB WHERE JOBNAME = 'CLERK'),**

**(SELECT EMPNO FROM EMP WHERE EMPNAME = 'EKATERINA VROUBLEVSKAIA'),**

**(SELECT DEPTNO FROM DEPT WHERE DEPTNAME = 'ACCOUNTING'),**

**SYSDATE,**

**NULL);**

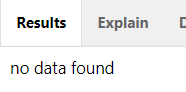
Состояние таблицы после изменения



1. Удалите все записи из таблицы TMP\_EMP. Добавьте в нее информацию о сотрудниках, которые работают клерками в настоящий момент.

Исходное состояние таблицы (после удаления всех записей)

SELECT \* FROM TMP\_EMP;



Решение задачи

Удаление:

**DELETE FROM TMP\_EMP;**

Добавление:

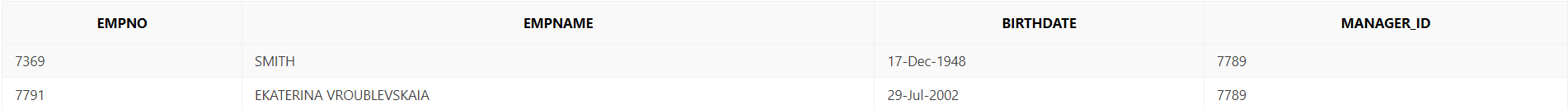
**INSERT INTO TMP\_EMP**

**(SELECT \* FROM EMP WHERE EMPNO IN**

**(SELECT EMPNO FROM CAREER C NATURAL JOIN JOB J**

**WHERE J.JOBNAME = 'CLERK' AND C.ENDDATE IS NULL));**

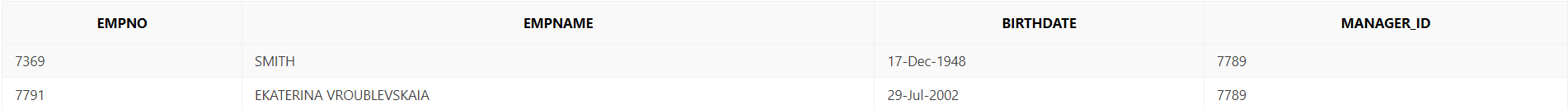
Состояние таблицы после изменения



1. Добавьте в таблицу TMP\_EMP информацию о тех сотрудниках, которые уже не работают на предприятии, а в период работы занимали только одну должность.

Исходное состояние таблицы

SELECT \* FROM TMP\_EMP;



Решение задачи

**INSERT INTO TMP\_EMP**

**SELECT \* FROM EMP**

**WHERE EMPNO IN**

**(SELECT EMPNO FROM CAREER WHERE EMPNO IN (SELECT EMPNO FROM (SELECT DISTINCT JOBNO, EMPNO FROM CAREER) T**

**GROUP BY EMPNO**

**HAVING COUNT(EMPNO) = 1)**

**AND ENDDATE IS NOT NULL AND ENDDATE < CURRENT\_DATE);**

Состояние таблицы после изменения



1. Выполните тот же запрос для тех сотрудников, которые никогда не приступали к работе на предприятии.

Исходное состояние таблицы

SELECT \* FROM TMP\_EMP;



Решение задачи

**INSERT INTO TMP\_EMP**

**SELECT \* FROM EMP**

**WHERE EMPNO NOT IN (SELECT EMPNO FROM CAREER);**

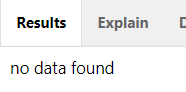
Состояние таблицы после изменения



1. Удалите все записи из таблицы TMP\_JOB и добавьте в нее информацию по тем специальностям, которые не используются в настоящий момент на предприятии.

Исходное состояние таблицы (после удаления всех записей)

SELECT \* FROM TMP\_JOB;



Решение задачи

Удаление:

**DELETE FROM TMP\_JOB;**

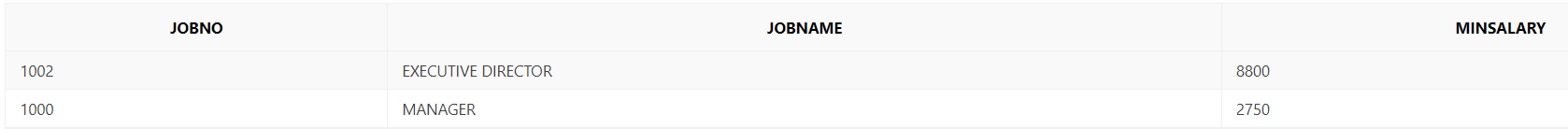
Добавление:

**INSERT INTO TMP\_JOB**

**SELECT \* FROM JOB**

**WHERE JOBNO NOT IN (SELECT JOBNO FROM CAREER WHERE ENDDATE IS NULL);**

Состояние таблицы после изменения



1. Начислите зарплату в размере 120% минимального должностного оклада всем сотрудникам, работающим на предприятии. Зарплату начислять по должности, занимаемой сотрудником в настоящий момент и отнести ее на прошлый месяц относительно текущей даты.

Исходное состояние таблицы

SELECT \* FROM SALARY;



Решение задачи

**INSERT INTO SALARY**

**SELECT EMPNO, EXTRACT(MONTH FROM ADD\_MONTHS(sysdate, -1)) AS MONTH,**

**EXTRACT(YEAR FROM ADD\_MONTHS(sysdate, -1)) AS YEAR, 1.2 \* MINSALARY AS SALVALUE**

**FROM CAREER**

**NATURAL JOIN JOB**

**WHERE ENDDATE IS NULL;**

Состояние таблицы после изменения



1. Удалите данные о зарплате за прошлый год.

Исходное состояние таблицы

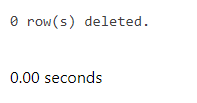
SELECT \* FROM SALARY;



Решение задачи

**DELETE FROM SALARY**

**WHERE YEAR = EXTRACT(YEAR FROM SYSDATE) – 1;**

****

Так как за прошлый год данных не было.

1. Удалите информацию о карьере сотрудников, которые в настоящий момент уже не работают на предприятии, но когда-то работали.

Исходное состояние таблицы

SELECT \* FROM CAREER;

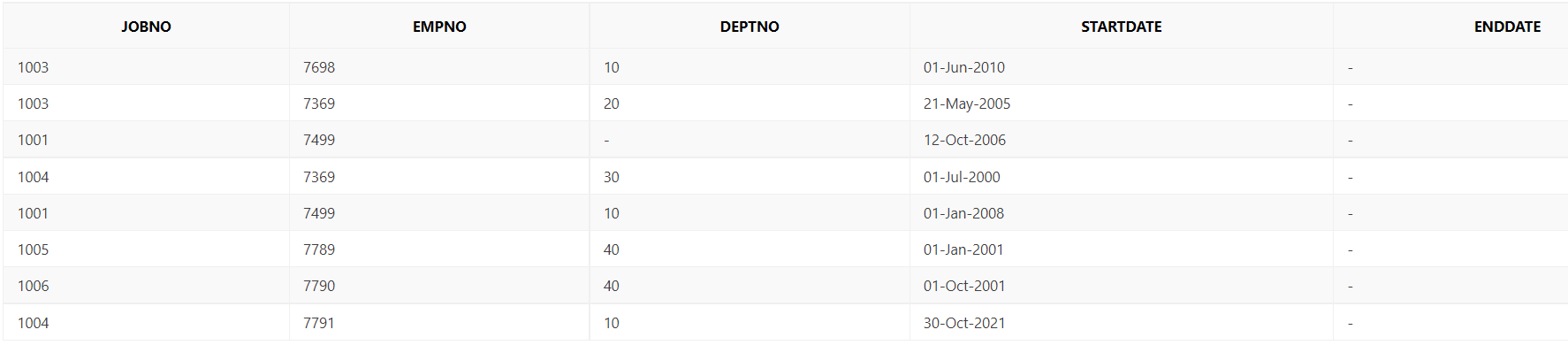


Решение задачи

**DELETE FROM CAREER**

**WHERE ENDDATE IS NOT NULL AND ENDDATE < SYSDATE;**

Состояние таблицы после изменения



1. Удалите информацию о начисленной зарплате сотрудников, которые в настоящий момент уже не работают на предприятии (можно использовать результаты работы предыдущего запроса)

Исходное состояние таблицы

SELECT \* FROM SALARY;

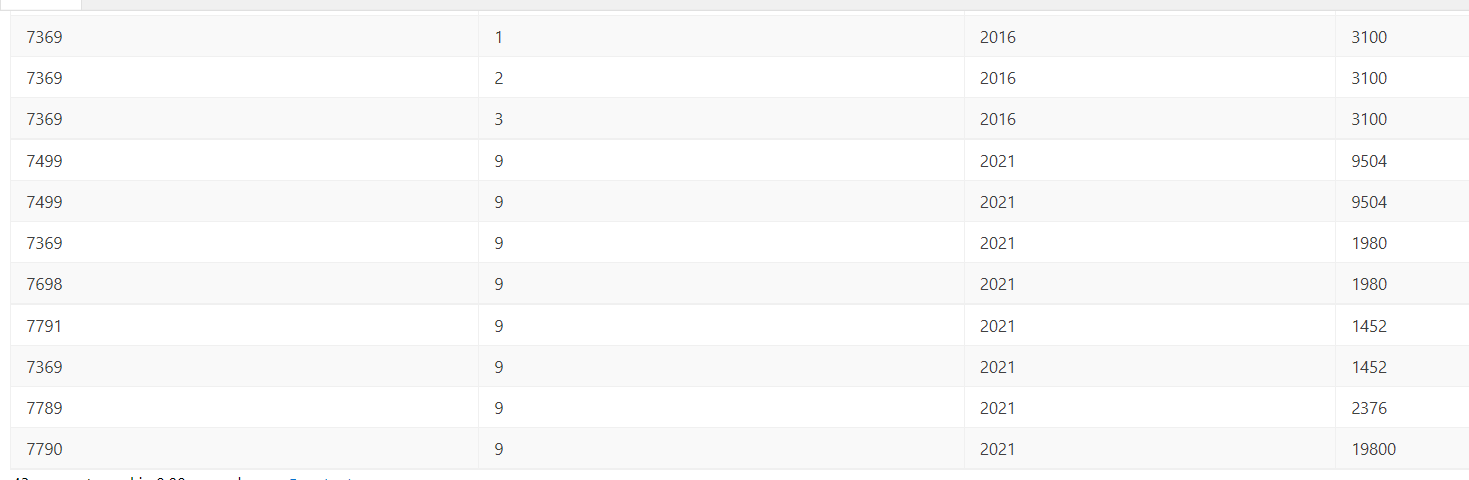


Решение задачи

**DELETE FROM SALARY**

**WHERE EMPNO NOT IN (SELECT EMPNO FROM CAREER);**

Состояние таблицы после изменения



1. Удалите записи из таблицы EMP для тех сотрудников, которые никогда не приступали к работе на предприятии.

Исходное состояние таблицы

SELECT \* FROM EMP;



Решение задачи

**DELETE FROM EMP**

**WHERE EMPNO NOT IN (SELECT EMPNO FROM CAREER);**

Состояние таблицы после изменения



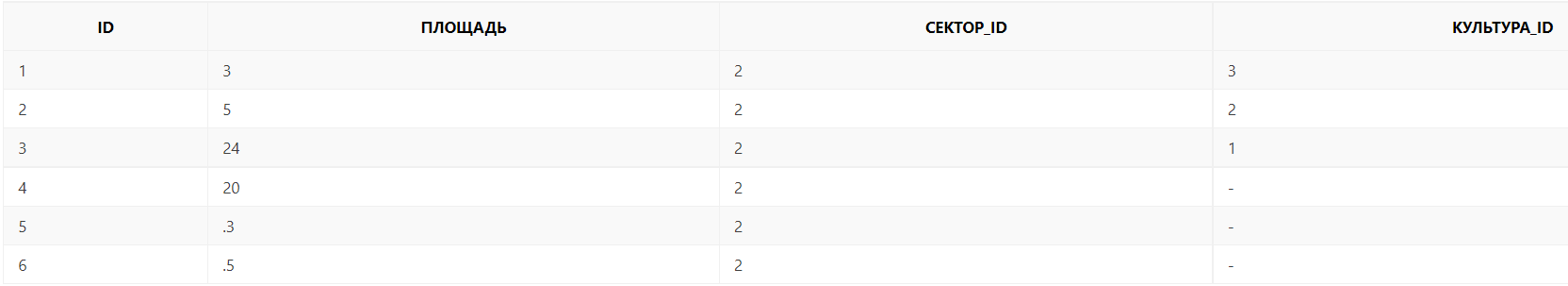
**Задание 2.**

**Вариант 8. Управление дачным хозяйством.**

1. Добавить сведения о новой культуре. Высадить культуру на ближайшем свободном участке.

Исходное состояние таблицы

SELECT \* FROM УЧАСТОК;



Добавляю сведения о новой культуре:

**INSERT INTO КУЛЬТУРА VALUES (6,115,44,'15',3,'27-Aug-2022',99,119)**

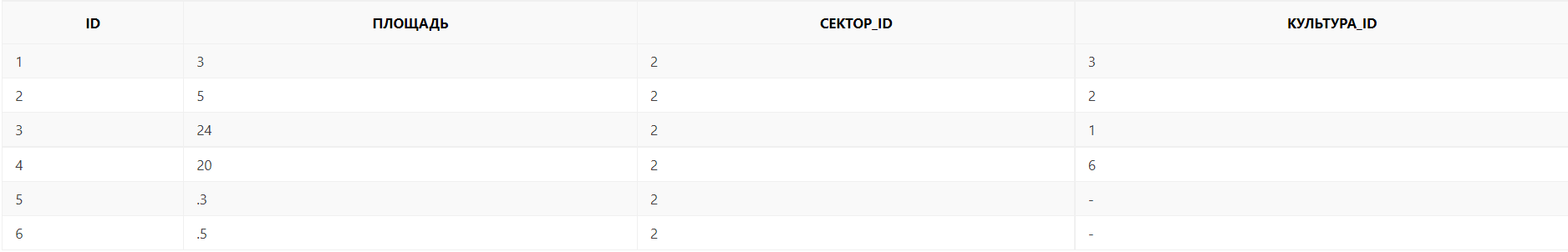
Высаживаю культуру на ближайшем свободном участке:

**UPDATE УЧАСТОК**

**SET КУЛЬТУРА\_ID = (SELECT MAX(ID) FROM КУЛЬТУРА)**

**WHERE КУЛЬТУРА\_ID IS NULL AND ID = (SELECT MIN(ID) FROM УЧАСТОК WHERE КУЛЬТУРА\_ID IS NULL)**

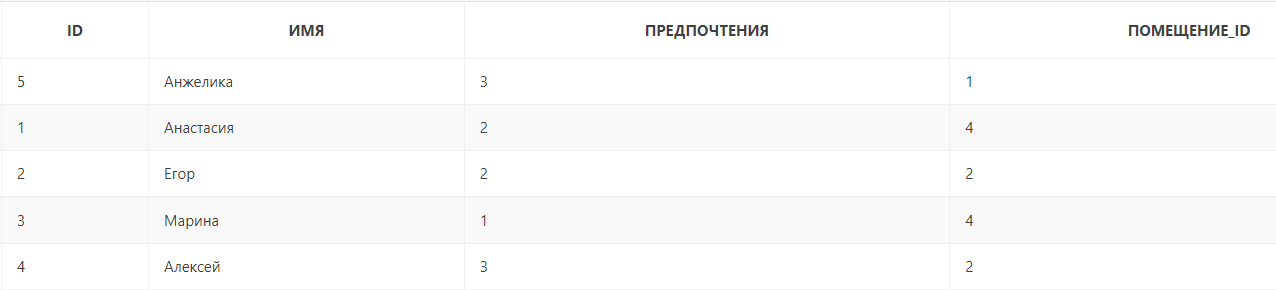
Состояние таблицы после изменения:



1. Переселить гостей из аварийного помещения в другие жилые помещения в соответствии с их предпочтениями.

Исходное состояние таблицы

SELECT \* FROM ГОСТЬ;



Решение задачи.

**UPDATE ГОСТЬ g**

-- выбираю свободное помещение по интересам гостя

-- нахожу минимальный айди неаварийного помещения со свободным местом и по предпочтениям данного гостя

**SET g.ПОМЕЩЕНИЕ\_ID = (SELECT MIN(ID) FROM ЖИЛОЕ\_ПОМЕЩЕНИЕ**

**WHERE (АВАРИЙНОЕ = 'N' AND СПАЛЬНЫХ\_МЕСТ > (SELECT COUNT(\*) FROM ГОСТЬ GROUP BY ПОМЕЩЕНИЕ\_ID HAVING ПОМЕЩЕНИЕ\_ID = g.ПОМЕЩЕНИЕ\_ID)**

**AND g.ПРЕДПОЧТЕНИЯ = СПАЛЬНЫХ\_МЕСТ))**

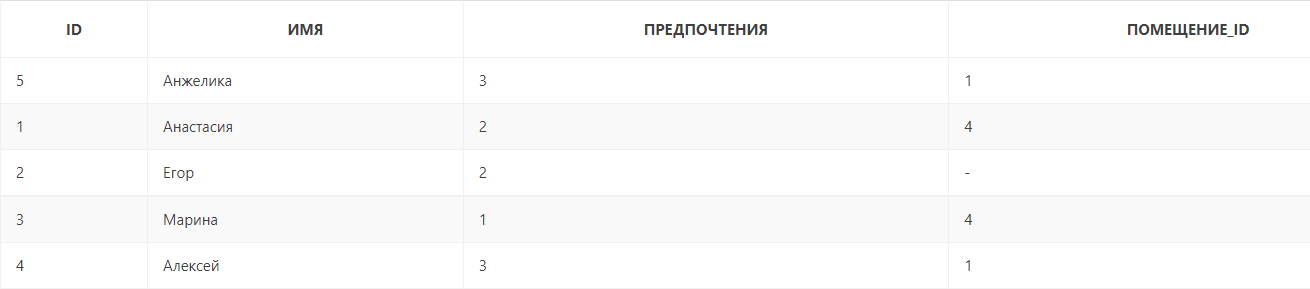
**--** ищу гостей в аварийных помещениях

**WHERE ПОМЕЩЕНИЕ\_ID IN (SELECT ID FROM ЖИЛОЕ\_ПОМЕЩЕНИЕ WHERE АВАРИЙНОЕ = 'Y')**

Состояние таблицы после изменения

SELECT \* FROM ГОСТЬ;

**(аварийное помещение №2)**

****

1. Вывести из землевладения заброшенные участки. Сохранить архивные данные: номер участка, площадь участка, период использования и общий объем культуры выращиваемой на этом участке.

Изначально архив пуст.

Таблица Участок:



-- сохраняю данные в архив

**INSERT INTO АРХИВ**

**SELECT U.ID, U.ПЛОЩАДЬ, K.ВЕГЕТ\_ПЕРИОД, K.НОРМА\_ПОСЕВА\*U.ПЛОЩАДЬ FROM УЧАСТОК U JOIN КУЛЬТУРА K ON U.КУЛЬТУРА\_ID=K.ID**

**WHERE K.СРОК\_УБОРКИ < SYSDATE**

**--**вывожу из землевладения заброшенные участки

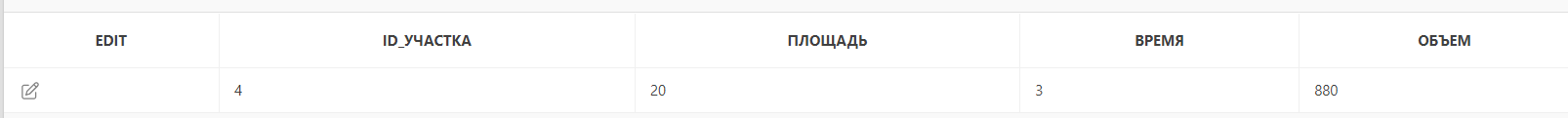
**DELETE FROM УЧАСТОК**

**WHERE ID IN (SELECT U.ID FROM УЧАСТОК U JOIN КУЛЬТУРА K ON U.КУЛЬТУРА\_ID=K.ID**

**WHERE K.СРОК\_УБОРКИ < SYSDATE)**

Результат:

Архив

****

Участок

