Пороскун Олена Олегівна ПМ-81

Лабораторна робота 12

Завдання

1. Створіть тригер, який відстежує зміну зарплати користувача і заносить в таблицю звітності

**-- drop table emp11;**

**-- drop table emp22;**

**create table emp11(name char(10), salary integer);**

**create table emp22(name char(10), old\_salary integer, new\_salary integer);**

**insert into emp11 values('Damon', 9800);**

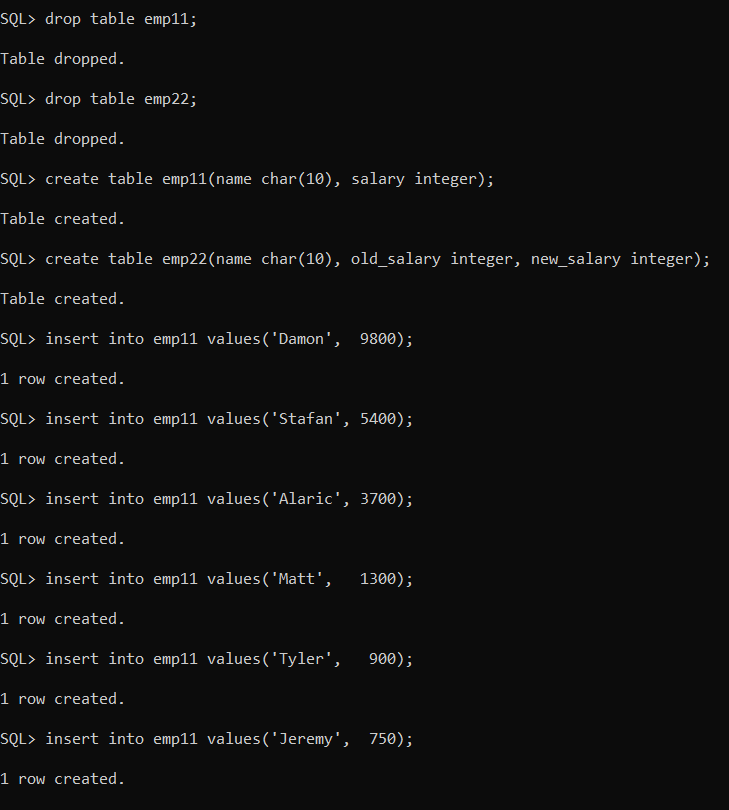
**insert into emp11 values('Stafan', 5400);**

**insert into emp11 values('Alaric', 3700);**

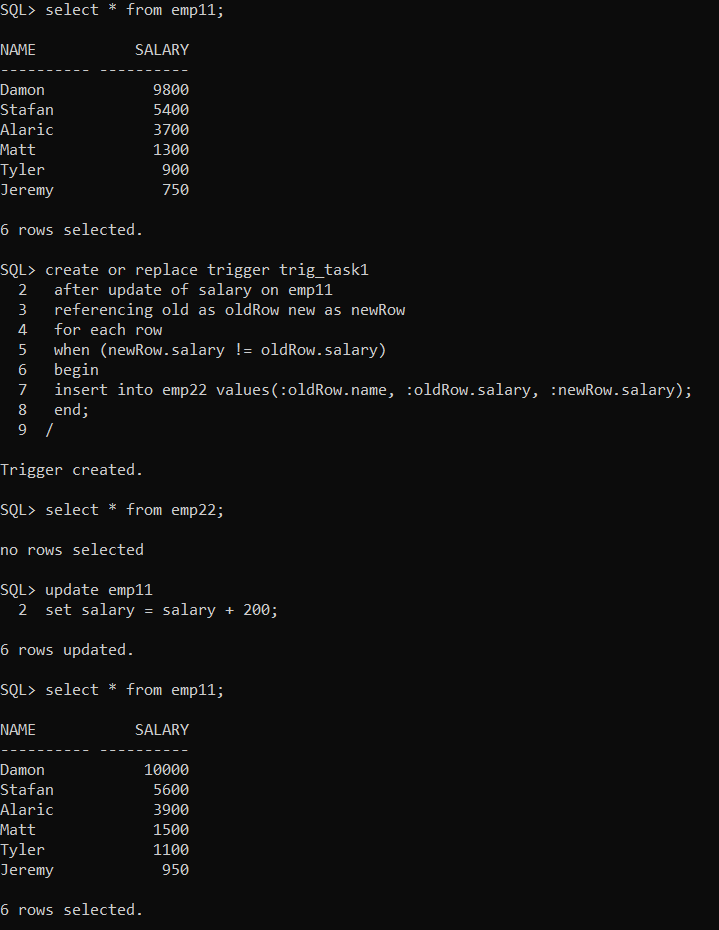
**insert into emp11 values('Matt', 1300);**

**insert into emp11 values('Tyler', 900);**

**insert into emp11 values('Jeremy', 750);**

****

**select \* from emp11;**

****

**create or replace trigger trig\_task1**

**after update of salary on emp11**

**referencing old as oldRow new as newRow**

**for each row**

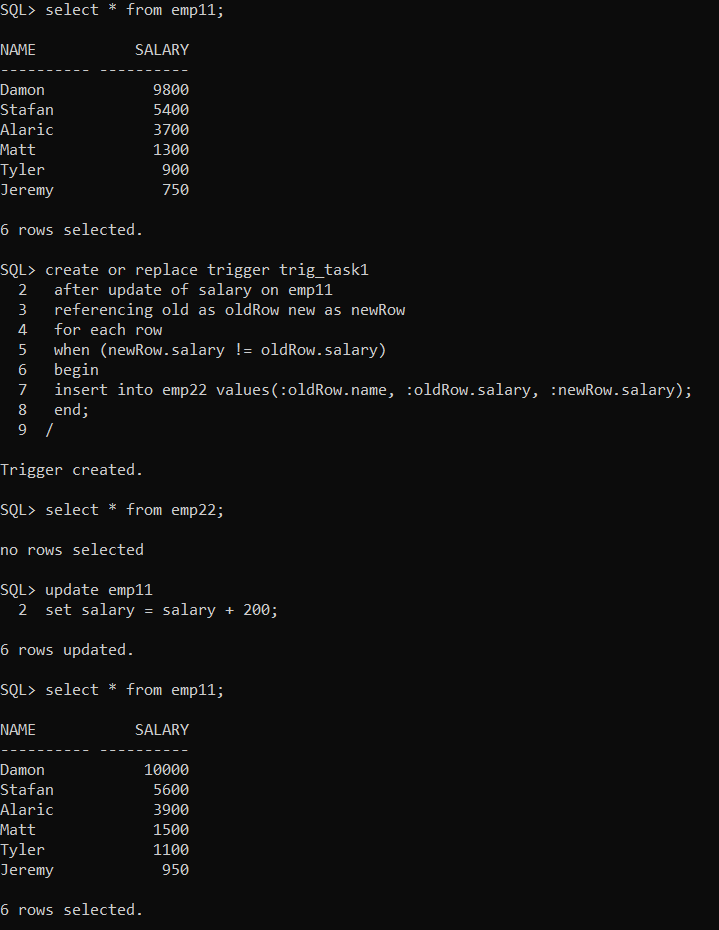
**when (newRow.salary != oldRow.salary)**

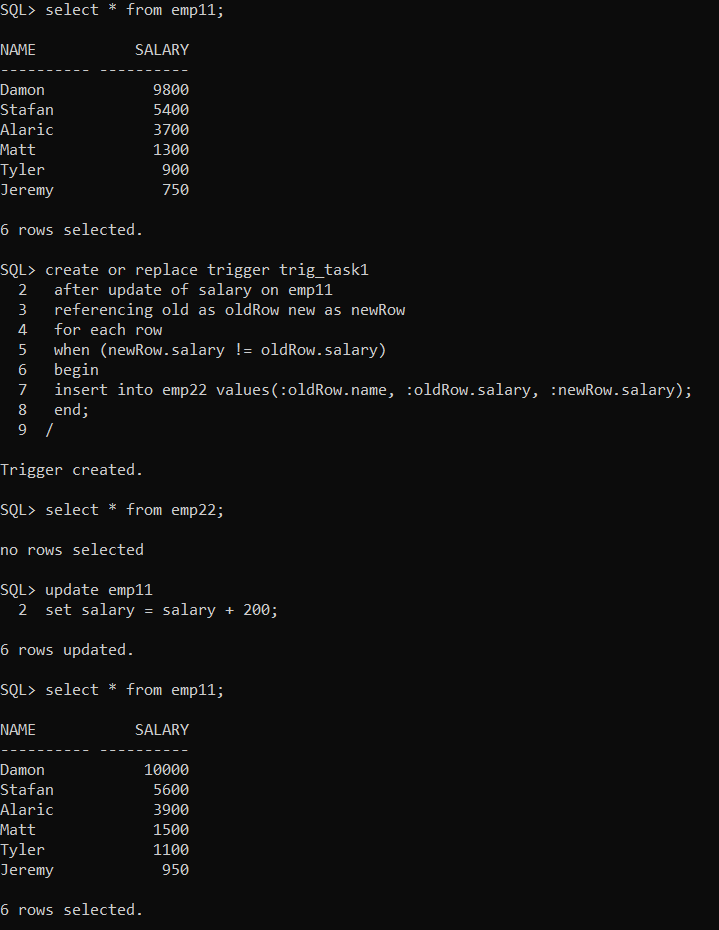
**begin**

**insert into emp22 values(:oldRow.name, :oldRow.salary, :newRow.salary);**

**end;**

**/**

****

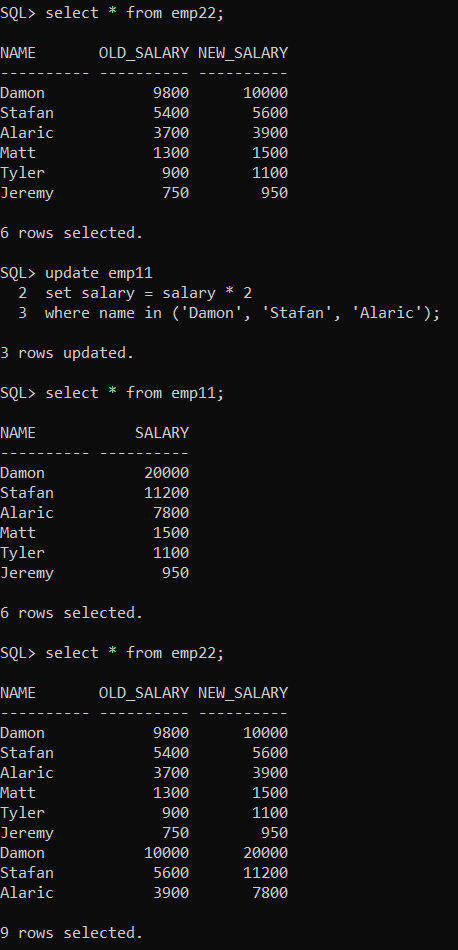
**select \* from emp22;**

**update emp11**

**set salary = salary + 200;**

**select \* from emp11;**

**select \* from emp22;**

****

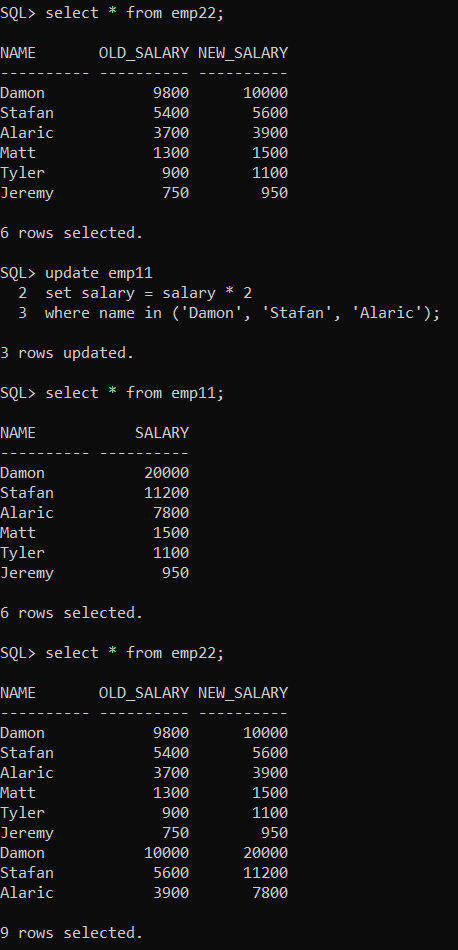
**update emp11**

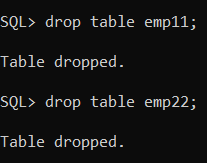
**set salary = salary \* 2**

**where name in ('Damon', 'Stafan', 'Alaric');**

**select \* from emp11;**

**select \* from emp22;**

****

**drop table emp11;**

**drop table emp22;**

2. \* Створіть тригер, який відстежує зміну посади користувача і записує її в журнал. Журнал зберігає стару і нову посаду і час зміни. У разі видалення користувача він вважається звільненим, про що заноситься запис в журнал. У разі додавання користувача він вважається прийнятим на роботу, про що заноситься запис у журнал.

**--drop table emp2;**

**--drop table register;**

**create table emp2(name char(10), job char(10));**

**create table register(name char(10), old\_job char(10), new\_job char(10), time\_of\_change date);**

**create or replace trigger trig\_task2**

**after insert or update or delete on emp2**

**referencing old as oldRow new as newRow**

**for each row**

**begin**

**if inserting then**

**insert into register values(:newRow.name, 'HIRED', :newRow.job, sysdate);**

**elsif updating then**

**insert into register values(:oldRow.name, :oldRow.job, :newRow.job, sysdate);**

**elsif deleting then**

**insert into register values(:oldRow.name, :oldRow.job, 'DISMISSED', sysdate);**

**end if;**

**end;**

**/**

**insert into emp2 values('Damon', 'MANAGER');**

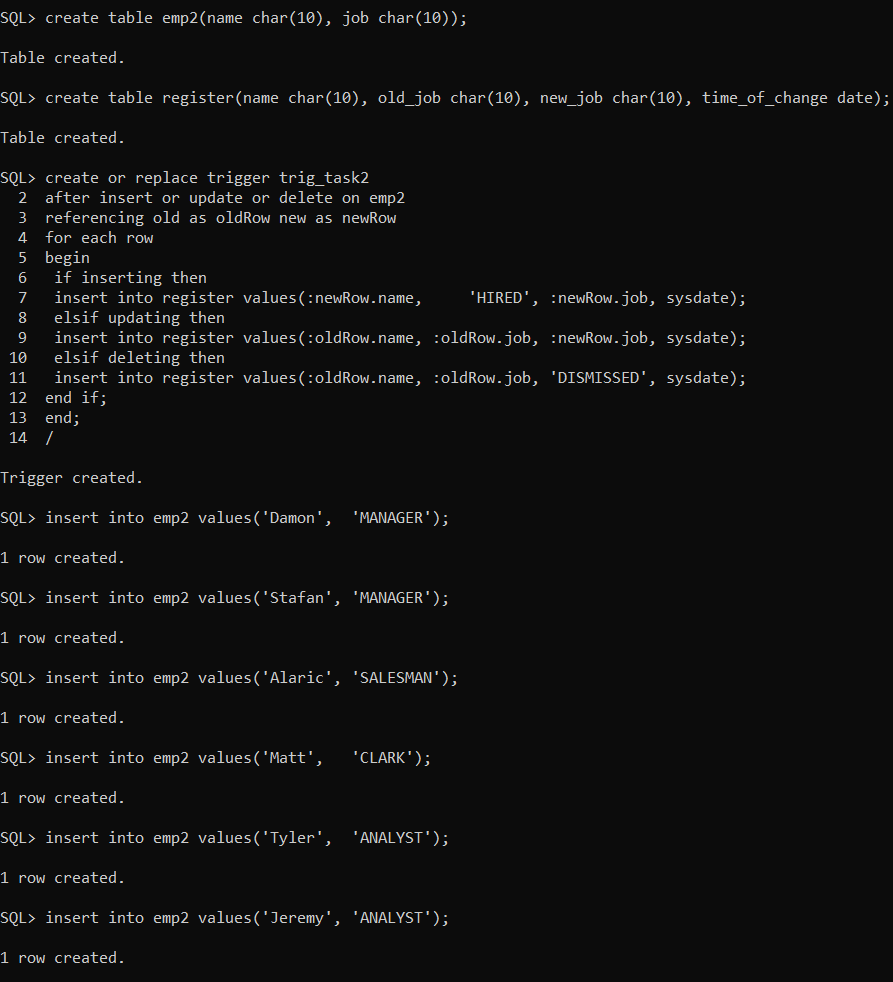
**insert into emp2 values('Stafan', 'MANAGER');**

**insert into emp2 values('Alaric', 'SALESMAN');**

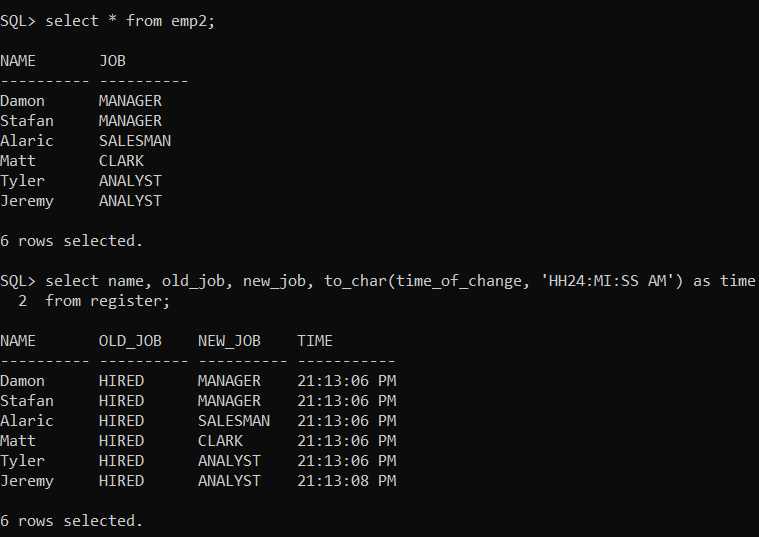
**insert into emp2 values('Matt', 'CLARK');**

**insert into emp2 values('Tyler', 'ANALYST');**

**insert into emp2 values('Jeremy', 'ANALYST');**

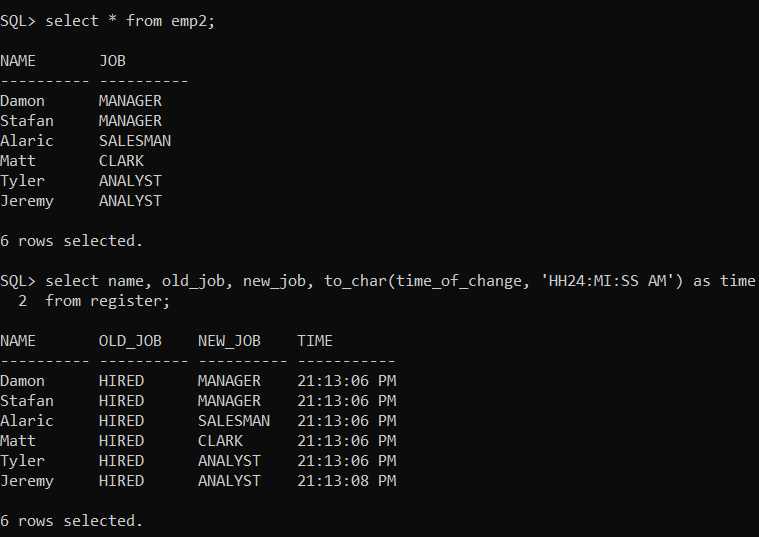


**select \* from emp2;**

****

**select name, old\_job, new\_job, to\_char(time\_of\_change, 'HH24:MI:SS AM') as time**

**from register;**

****

**update emp2**

**set job = 'PRESIDENT'**

**where name = 'Damon';**

**update emp2**

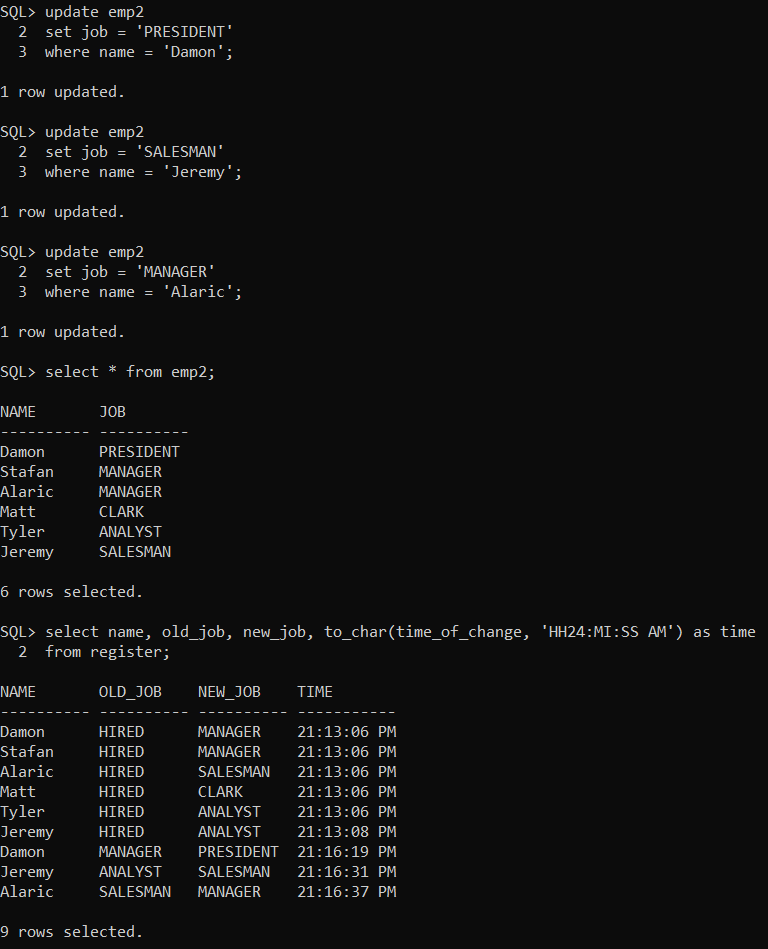
**set job = 'SALESMAN'**

**where name = 'Jeremy';**

**update emp2**

**set job = 'MANAGER'**

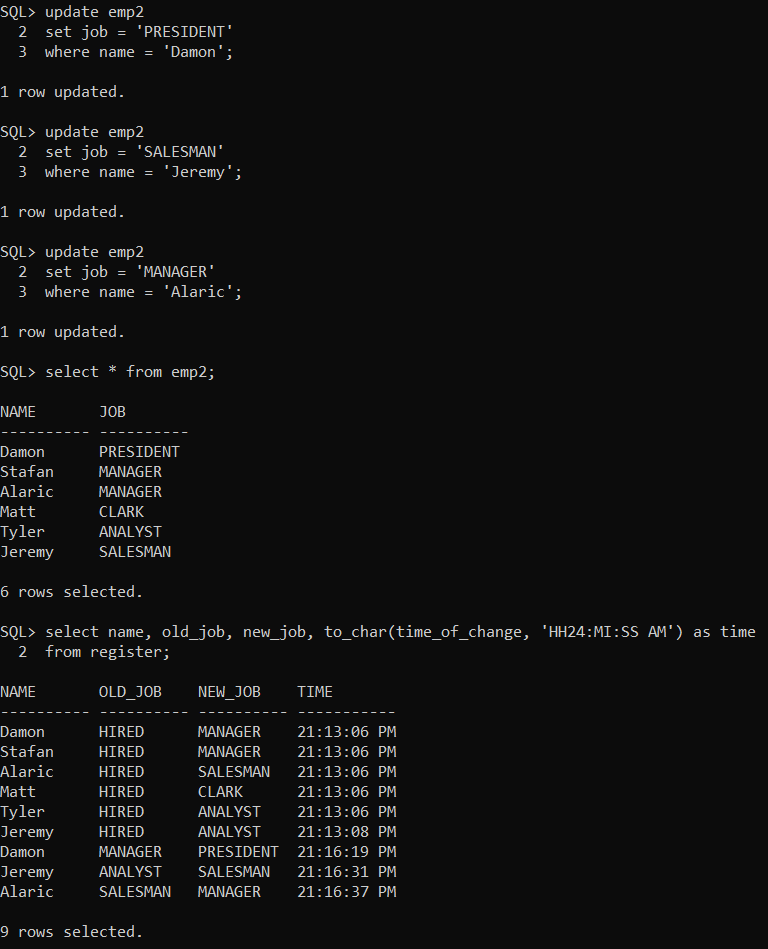
**where name = 'Alaric';**

****

**select \* from emp2;**

**select name, old\_job, new\_job, to\_char(time\_of\_change, 'HH24:MI:SS AM') as time**

**from register;**

****

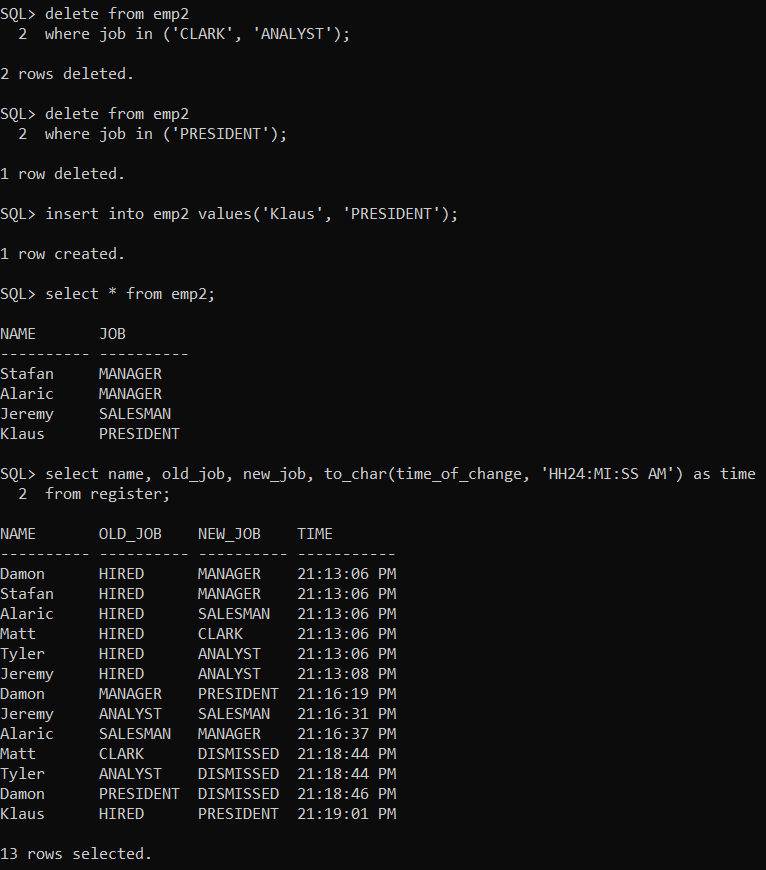
**delete from emp2**

**where job in ('CLARK', 'ANALYST');**

**delete from emp2**

**where job in ('PRESIDENT');**

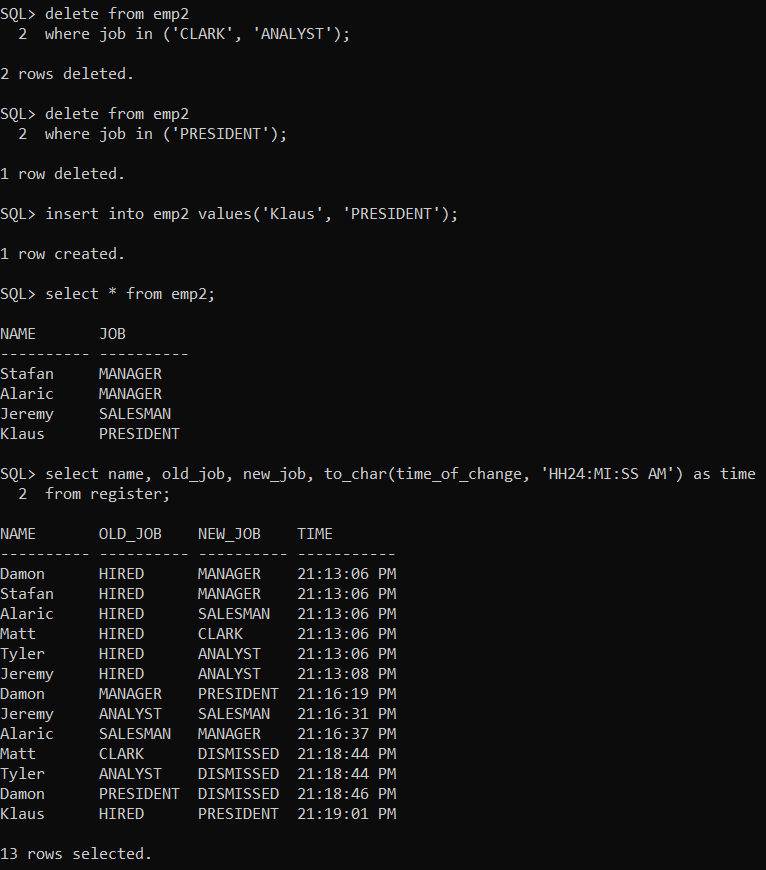
**insert into emp2 values('Klaus', 'PRESIDENT');**

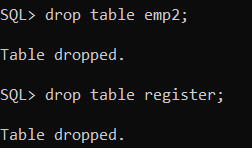
****

**select \* from emp2;**

**select name, old\_job, new\_job, to\_char(time\_of\_change, 'HH24:MI:SS AM') as time**

**from register;**

****

**drop table emp2;**

**drop table register;**

Завдання

3. \*Візьміть з тексту лекції приклад тригера реалізації каскадного видалення. Реалізуйте тригер, що видаляє всіх співробітників, які працюють у відділі, при видаленні відділу.

**--drop table emp3;**

**--drop table dept3;**

**CREATE TABLE dept3( -- описание подразделения**

**deptno NUMBER(3) NOT NULL, -- № подразделения**

**dname VARCHAR(14) NOT NULL, -- название подразделения**

**loc VARCHAR(13), -- местоположение подразделения**

**CONSTRAINT dept\_pk3 PRIMARY KEY (deptno),**

**CONSTRAINT dept\_naem\_uk3 UNIQUE (dname)**

**);**

**CREATE TABLE emp3 ( -- описание сотрудников подразделений**

**empno NUMBER(4) NOT NULL, -- № сотрудника**

**ename VARCHAR(20), -- имя сотрудника**

**job VARCHAR(9), -- должность сотрудника**

**sal NUMBER(7, 2), -- з/п сотрудника**

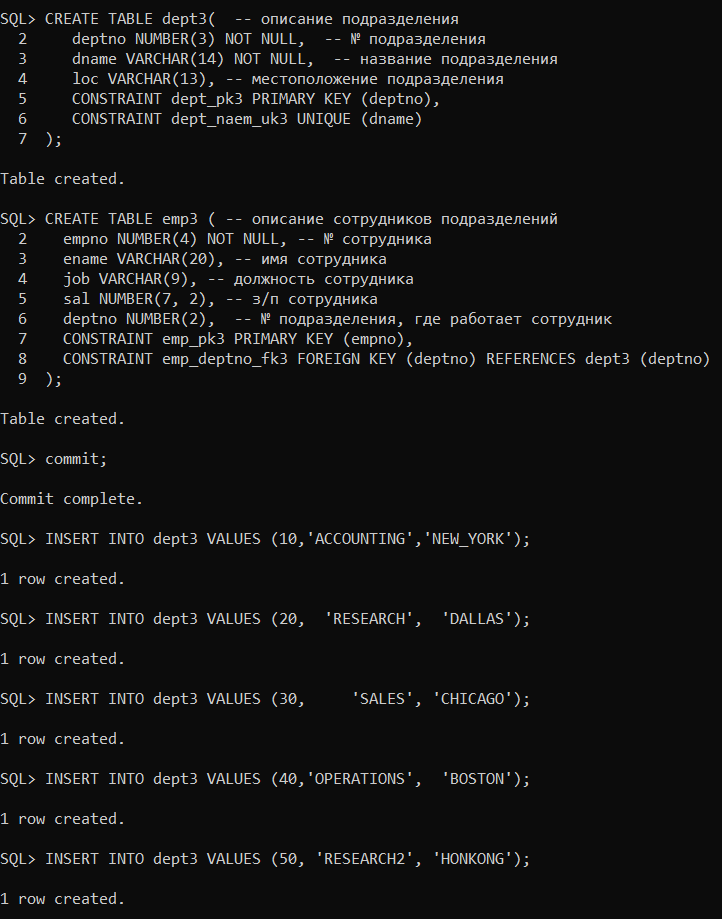
**deptno NUMBER(2), -- № подразделения, где работает сотрудник**

**CONSTRAINT emp\_pk3 PRIMARY KEY (empno),**

**CONSTRAINT emp\_deptno\_fk3 FOREIGN KEY (deptno) REFERENCES dept3 (deptno)**

**);**

**commit;**

****

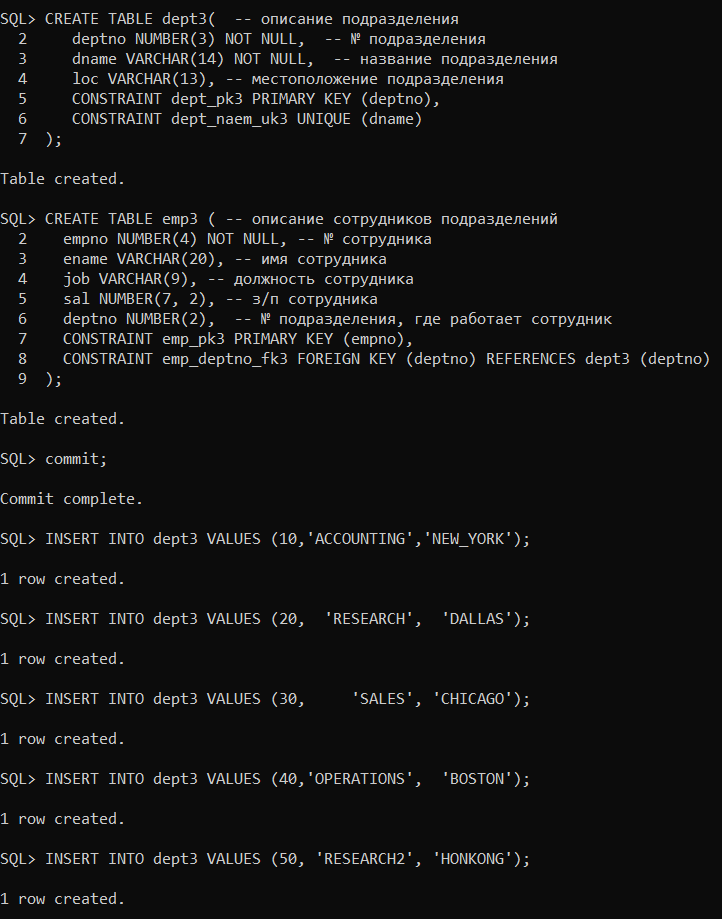
**INSERT INTO dept3 VALUES (10,'ACCOUNTING','NEW\_YORK');**

**INSERT INTO dept3 VALUES (20, 'RESEARCH', 'DALLAS');**

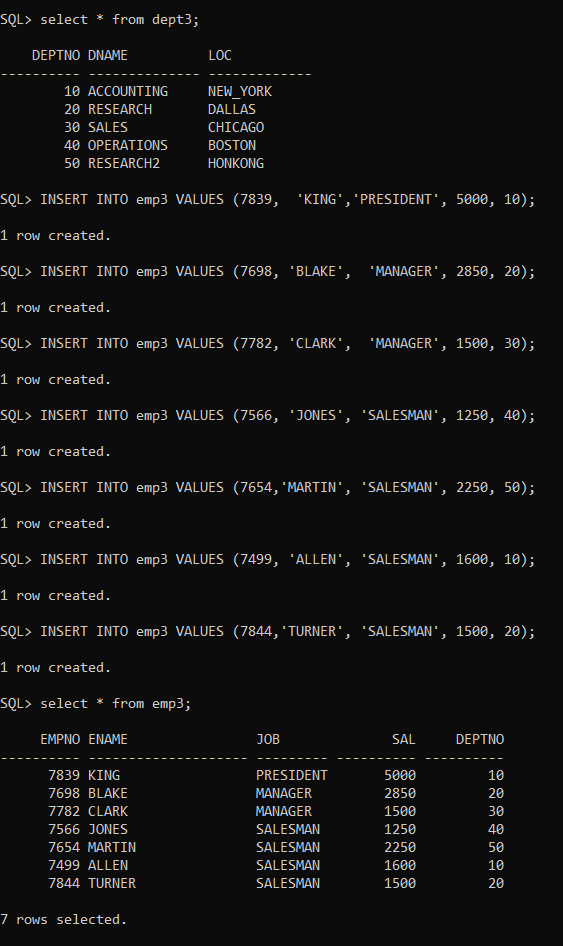
**INSERT INTO dept3 VALUES (30, 'SALES', 'CHICAGO');**

**INSERT INTO dept3 VALUES (40,'OPERATIONS', 'BOSTON');**

**INSERT INTO dept3 VALUES (50, 'RESEARCH2', 'HONKONG');**

****

**select \* from dept3;**

****

**INSERT INTO emp3 VALUES (7839, 'KING','PRESIDENT', 5000, 10);**

**INSERT INTO emp3 VALUES (7698, 'BLAKE', 'MANAGER', 2850, 20);**

**INSERT INTO emp3 VALUES (7782, 'CLARK', 'MANAGER', 1500, 30);**

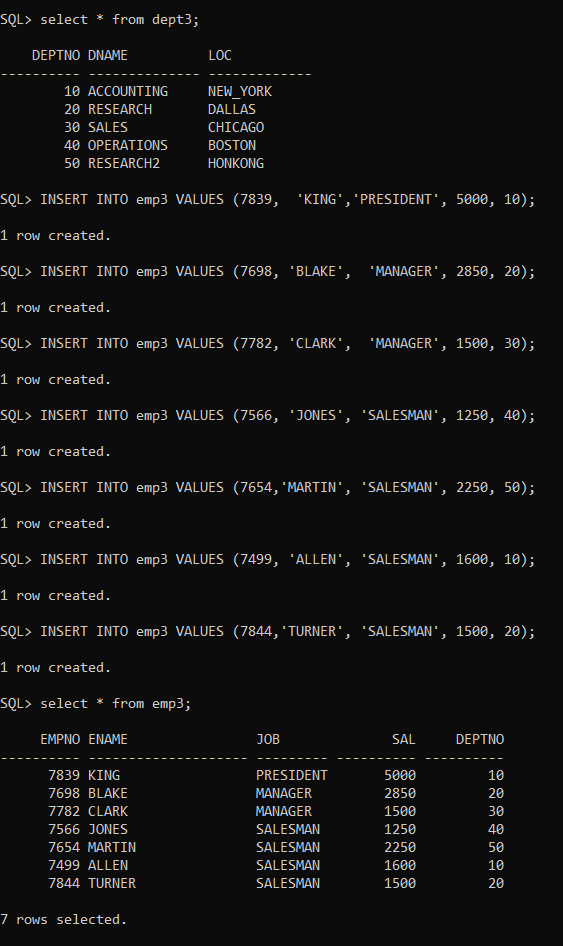
**INSERT INTO emp3 VALUES (7566, 'JONES', 'SALESMAN', 1250, 40);**

**INSERT INTO emp3 VALUES (7654,'MARTIN', 'SALESMAN', 2250, 50);**

**INSERT INTO emp3 VALUES (7499, 'ALLEN', 'SALESMAN', 1600, 10);**

**INSERT INTO emp3 VALUES (7844,'TURNER', 'SALESMAN', 1500, 20);**

**select \* from emp3;**

****

**commit;**

**create or replace trigger trig\_task3**

**after delete on dept3**

**referencing old as oldRow new as newRow**

**for each row**

**begin**

**delete from emp3**

**where emp3.deptno = :oldRow.deptno;**

**end;**

**/**

**delete from dept3**

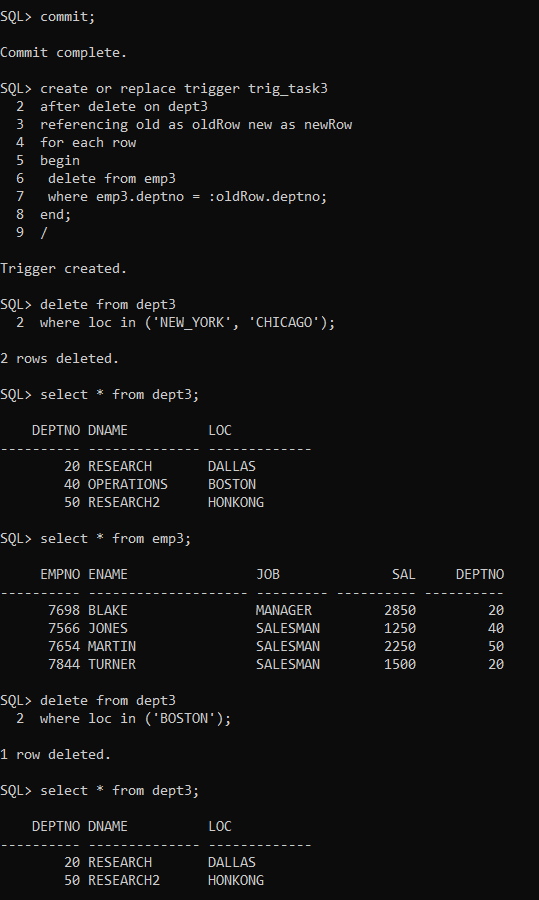
**where loc in ('NEW\_YORK', 'CHICAGO');**

**select \* from dept3;**

**select \* from emp3;**

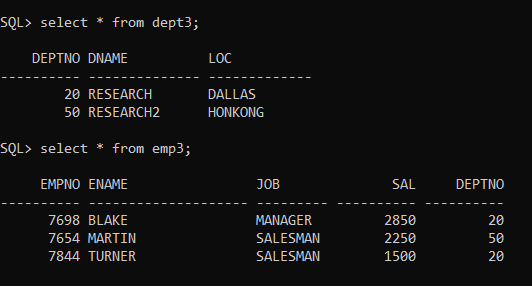
**delete from dept3**

**where loc in ('BOSTON');**

****

**select \* from dept3;**

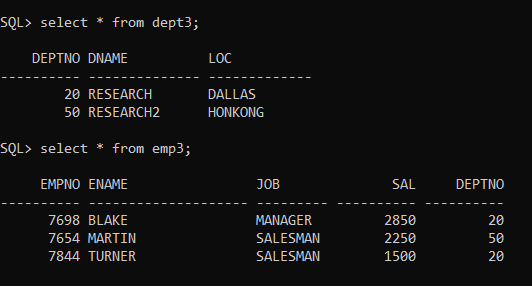
**select \* from emp3;**



4. \* Перевірте, що при цьому спрацьовує тригер із завдання 2.

**select \* from dept3;**

**select \* from emp3;**

****

**--drop table register2;**

**create table register2(ename char(10), old\_job char(10), new\_job char(10), time\_of\_change date);**

**create or replace trigger trig\_task2\_4**

**after insert or update or delete on emp3**

**referencing old as oldRow new as newRow**

**for each row**

**begin**

**if inserting then**

**insert into register2 values(:newRow.ename, 'HIRED', :newRow.job, sysdate);**

**elsif updating then**

**insert into register2 values(:oldRow.ename, :oldRow.job, :newRow.job, sysdate);**

**elsif deleting then**

**insert into register2 values(:oldRow.ename, :oldRow.job, 'DISMISSED', sysdate);**

**end if;**

**end;**

**/**

**delete from dept3**

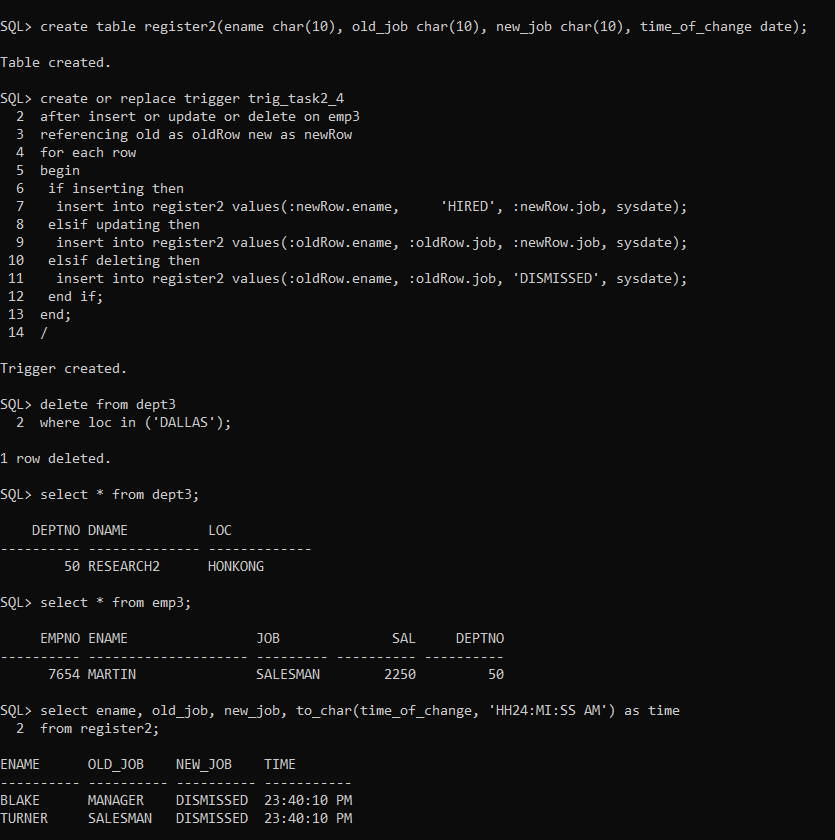
**where loc in ('DALLAS');**

**select \* from dept3;**

**select \* from emp3;**

**select ename, old\_job, new\_job, to\_char(time\_of\_change, 'HH24:MI:SS AM') as time**

**from register2;**



**drop table emp3;**

**drop table dept3;**

**drop table register2;**

