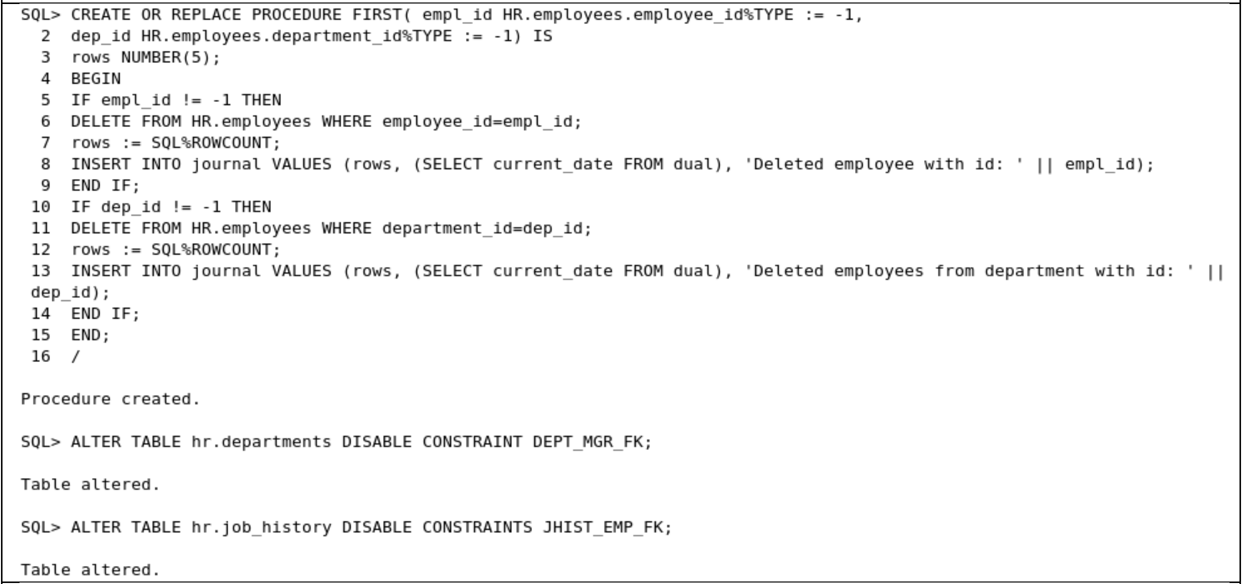
DATABASE ADMINISTRATION

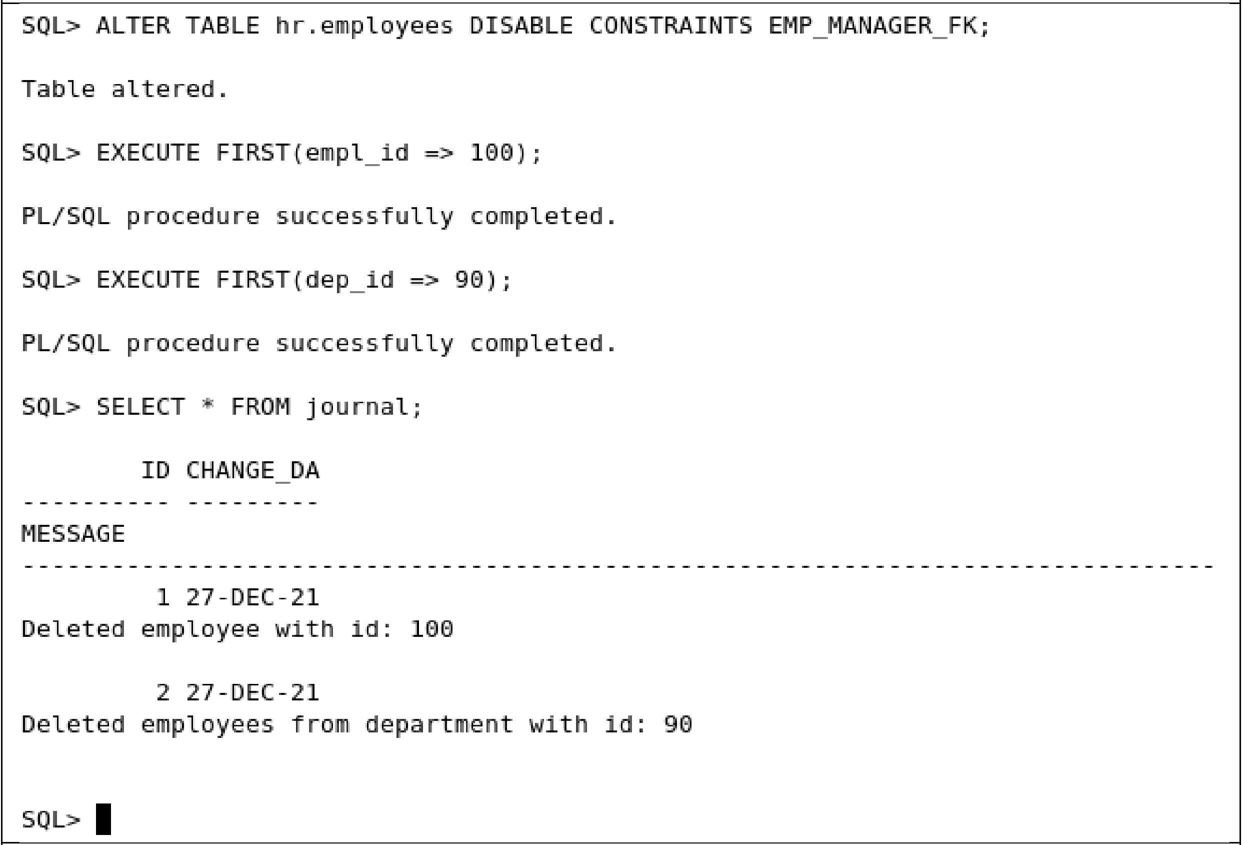
Lab 10 – PL / SQL exception handling, triggers

229840 – Wiktor Bechciński
229850 – Kamil Budzyn

1. Write a procedure to remove from the table an employee with the given number or all employees from the given department (given as parameters). Then count how many rows have been changed and insert the appropriate comment into the journal table (definition in the instruction from class 10.). Catch any errors and insert the appropriate comment into the log table.

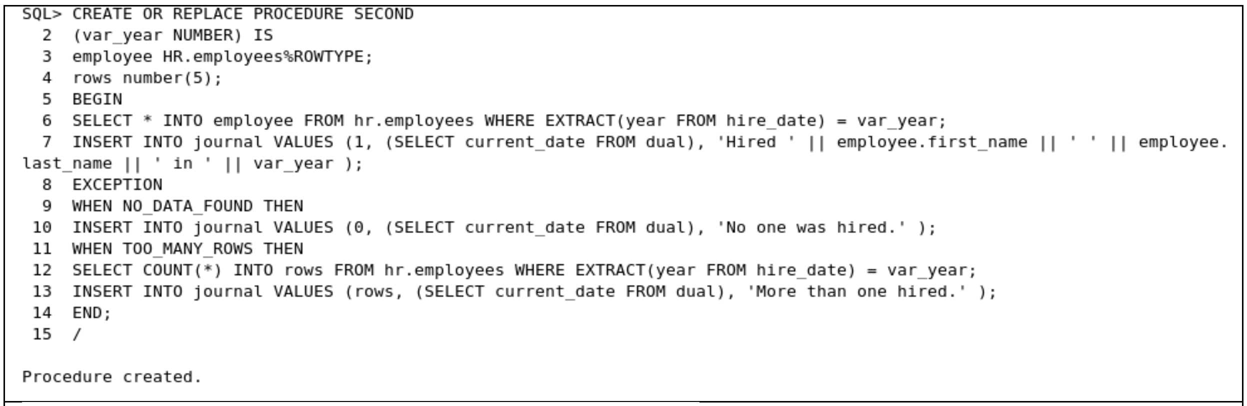


```
SQL> CREATE OR REPLACE PROCEDURE FIRST( empl_id HR.employees.employee_id%TYPE := -1,
dep_id HR.employees.department_id%TYPE := -1) IS
rows NUMBER(5);
BEGIN
IF empl_id != -1 THEN
DELETE FROM HR.employees WHERE employee_id=empl_id;
rows := SQL%ROWCOUNT;
INSERT INTO journal VALUES (rows, (SELECT current_date FROM dual), 'Deleted employee with id: ' || empl_id);
END IF;
IF dep_id != -1 THEN
DELETE FROM HR.employees WHERE department_id=dep_id;
rows := SQL%ROWCOUNT;
INSERT INTO journal VALUES (rows, (SELECT current_date FROM dual), 'Deleted employees from department with id: ' || dep_id);
END IF;
END;
/

Procedure created.

SQL> ALTER TABLE hr.departments DISABLE CONSTRAINT DEPT_MGR_FK;

Table altered.

SQL> ALTER TABLE hr.job_history DISABLE CONSTRAINTS JHIST_EMP_FK;

Table altered.
```

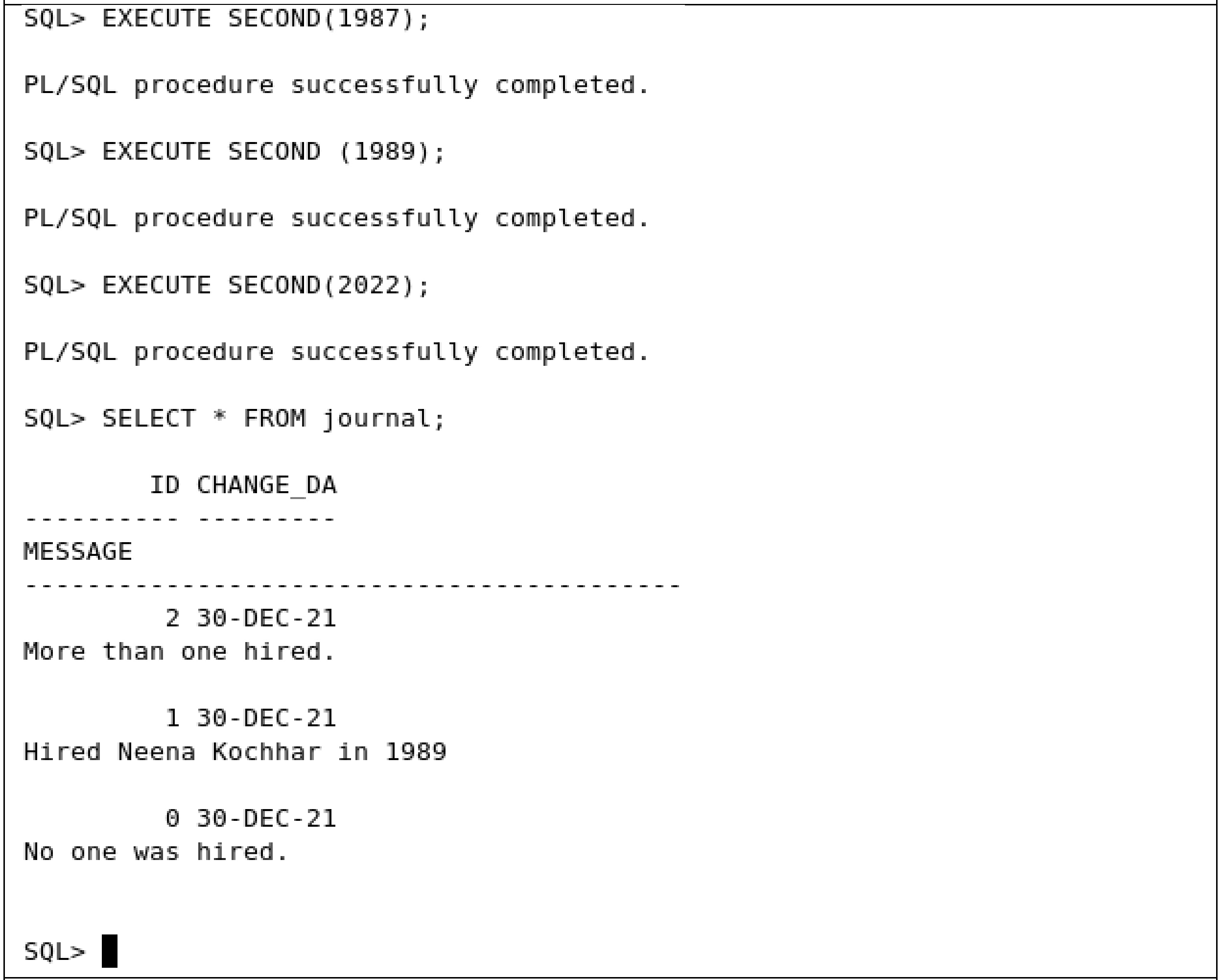


```
SQL> ALTER TABLE hr.employees DISABLE CONSTRAINTS EMP_MANAGER_FK;

Table altered.

SQL> EXECUTE FIRST(empl_id => 100);

PL/SQL procedure successfully completed.

SQL> EXECUTE FIRST(dep_id => 90);

PL/SQL procedure successfully completed.

SQL> SELECT * FROM journal;

        ID CHANGE_DA
---------- ---------
MESSAGE
--------------------------------------------------------------------------------
         1 27-DEC-21
Deleted employee with id: 100

         2 27-DEC-21
Deleted employees from department with id: 90


SQL> █
```

2. Write a procedure that inserts into the diary a comment about the number of employees employed in the indicated year on the basis of error handling (Hired ... / No one was hired. / More than one hired. / Error No. ...).

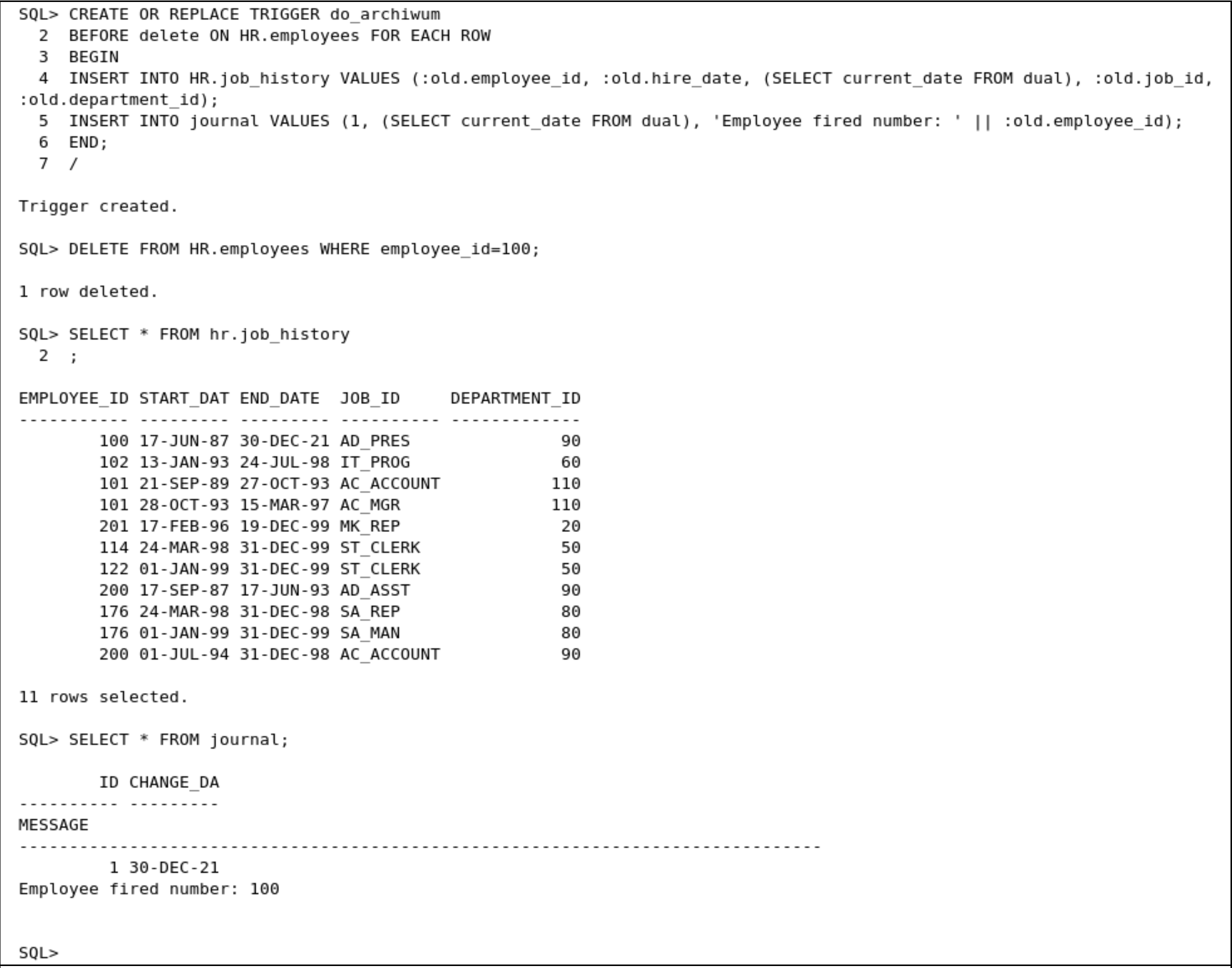


```
SQL> CREATE OR REPLACE PROCEDURE SECOND
     (var_year NUMBER) IS
     employee HR.employees%ROWTYPE;
     rows number(5);
     BEGIN
     SELECT * INTO employee FROM hr.employees WHERE EXTRACT(year FROM hire_date) = var_year;
     INSERT INTO journal VALUES (1, (SELECT current_date FROM dual), 'Hired ' || employee.first_name || ' ' || employee.
last_name || ' in ' || var_year );
     EXCEPTION
     WHEN NO_DATA_FOUND THEN
     INSERT INTO journal VALUES (0, (SELECT current_date FROM dual), 'No one was hired.' );
     WHEN TOO_MANY_ROWS THEN
     SELECT COUNT(*) INTO rows FROM hr.employees WHERE EXTRACT(year FROM hire_date) = var_year;
     INSERT INTO journal VALUES (rows, (SELECT current_date FROM dual), 'More than one hired.' );
     END;
     /

Procedure created.
```



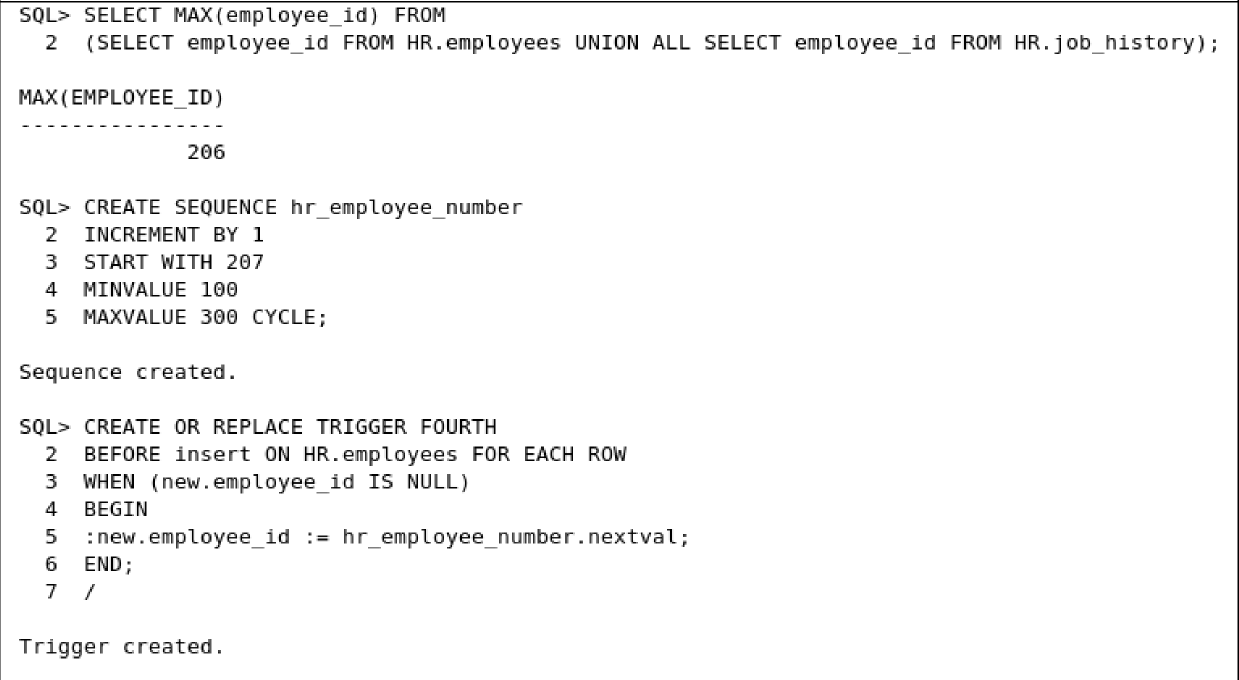
```
SQL> EXECUTE SECOND(1987);

PL/SQL procedure successfully completed.

SQL> EXECUTE SECOND (1989);

PL/SQL procedure successfully completed.

SQL> EXECUTE SECOND(2022);

PL/SQL procedure successfully completed.

SQL> SELECT * FROM journal;

        ID CHANGE_DA
---------- ---------
MESSAGE
-------------------------------------------
         2 30-DEC-21
More than one hired.

         1 30-DEC-21
Hired Neena Kochhar in 1989

         0 30-DEC-21
No one was hired.


SQL> 
```

3. Create a do_archiwum trigger that moves the employee's data to the archive table if the employee is dismissed (employees are removed from the table). Add a comment to the board log: Employee fired number: .....



```
SQL> CREATE OR REPLACE TRIGGER do_archiwum
  2  BEFORE delete ON HR.employees FOR EACH ROW
  3  BEGIN
  4  INSERT INTO HR.job_history VALUES (:old.employee_id, :old.hire_date, (SELECT current_date FROM dual), :old.job_id,
:old.department_id);
  5  INSERT INTO journal VALUES (1, (SELECT current_date FROM dual), 'Employee fired number: ' || :old.employee_id);
  6  END;
  7  /

Trigger created.

SQL> DELETE FROM HR.employees WHERE employee_id=100;

1 row deleted.

SQL> SELECT * FROM hr.job_history
  2  ;

EMPLOYEE_ID START_DAT END_DATE  JOB_ID     DEPARTMENT_ID
----------- --------- --------- ---------- -------------
        100 17-JUN-87 30-DEC-21 AD_PRES               90
        102 13-JAN-93 24-JUL-98 IT_PROG               60
        101 21-SEP-89 27-OCT-93 AC_ACCOUNT           110
        101 28-OCT-93 15-MAR-97 AC_MGR               110
        201 17-FEB-96 19-DEC-99 MK_REP                20
        114 24-MAR-98 31-DEC-99 ST_CLERK              50
        122 01-JAN-99 31-DEC-99 ST_CLERK              50
        200 17-SEP-87 17-JUN-93 AD_ASST               90
        176 24-MAR-98 31-DEC-98 SA_REP                80
        176 01-JAN-99 31-DEC-99 SA_MAN                80
        200 01-JUL-94 31-DEC-98 AC_ACCOUNT            90

11 rows selected.

SQL> SELECT * FROM journal;

        ID CHANGE_DA
---------- ---------
MESSAGE
--------------------------------------------------------------------------------
         1 30-DEC-21
Employee fired number: 100


SQL>
```

4. Create a trigger that, if you insert data into the array, employees without providing a number, will insert this number using the appropriate sequence.



```
SQL> SELECT MAX(employee_id) FROM
  2  (SELECT employee_id FROM HR.employees UNION ALL SELECT employee_id FROM HR.job_history);

MAX(EMPLOYEE_ID)
----------------
             206

SQL> CREATE SEQUENCE hr_employee_number
  2  INCREMENT BY 1
  3  START WITH 207
  4  MINVALUE 100
  5  MAXVALUE 300 CYCLE;

Sequence created.

SQL> CREATE OR REPLACE TRIGGER FOURTH
  2  BEFORE insert ON HR.employees FOR EACH ROW
  3  WHEN (new.employee_id IS NULL)
  4  BEGIN
  5  :new.employee_id := hr_employee_number.nextval;
  6  END;
  7  /

Trigger created.
```

```
SQL> INSERT INTO HR.employees VALUES (NULL, 'name', 'surname', 'email', 'phone', (SELECT current_date FROM dual), 'AD_PR
ES', 24100, NULL, NULL, 90, 'NO');

1 row created.

SQL> SELECT * FROM HR.employees WHERE employee_id=207;

EMPLOYEE_ID FIRST_NAME           LAST_NAME
----------- -------------------- -------------------------
EMAIL                     PHONE_NUMBER         HIRE_DATE JOB_ID         SALARY
------------------------- -------------------- --------- ---------- ----------
COMMISSION_PCT MANAGER_ID DEPARTMENT_ID EXC
-------------- ---------- ------------- ---
        207 name                 surname
email                     phone                30-DEC-21 AD_PRES         24100
                                     90 NO


SQL>
```