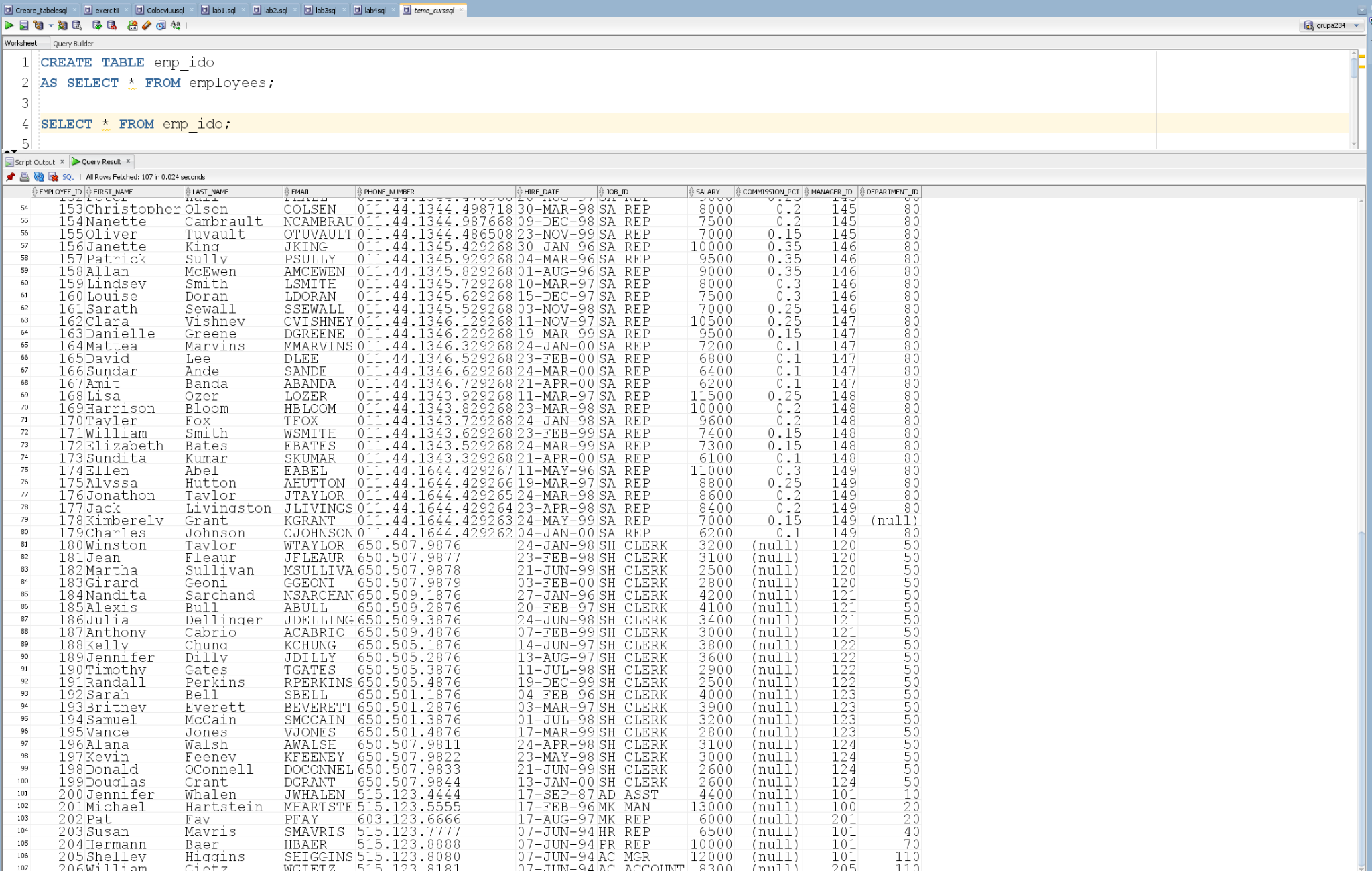
**Testare exceptii pe clauzele DELETE si UPDATE avand clauza RETURNING**

Voi lua ca referinta tabela emp\_ido care a fost creata dupa tabela employees din cadrul laboratorului:

CREATE TABLE emp\_ido

AS SELECT \* FROM employees;

SELECT \* FROM emp\_ido;



Pentru cerintele care urmeaza, trebuie notat faptul ca employee\_id ia valori de la 100 la 206 inclusiv. De aceea, pentru NO DATA FOUND voi lua ca employee\_id = 1000 ceea ce bineinteles nu se regaseste in tabela mea, pentru TOO MANY ROWS voi lua employee\_id > 100, iar pentru o singura linie voi lua doar employee\_id = 100. Intotdeauna clauzele vor fi executate pe liniile aflate din tabela emp\_ido nemodificata.

1. Pentru **DELETE**:

**A1) Exceptia TOO MANY ROWS:**

--DELETE pentru too many rows

DECLARE

v\_emp\_id NUMBER(6);

v\_first\_name VARCHAR(20);

v\_last\_name VARCHAR(20);

BEGIN

DELETE FROM emp\_ido WHERE employee\_id > 100

RETURNING employee\_id, first\_name, last\_name

INTO v\_emp\_id, v\_first\_name, v\_last\_name;

COMMIT;

END;



In acest caz, vom primi eroare.

**A2) O singura linie din tabela:**

DECLARE

v\_emp\_id NUMBER(6);

v\_first\_name VARCHAR2(20);

v\_last\_name VARCHAR2(25);

BEGIN

DELETE FROM emp\_ido WHERE employee\_id = 100

RETURNING employee\_id, first\_name, last\_name

INTO v\_emp\_id, v\_first\_name, v\_last\_name;

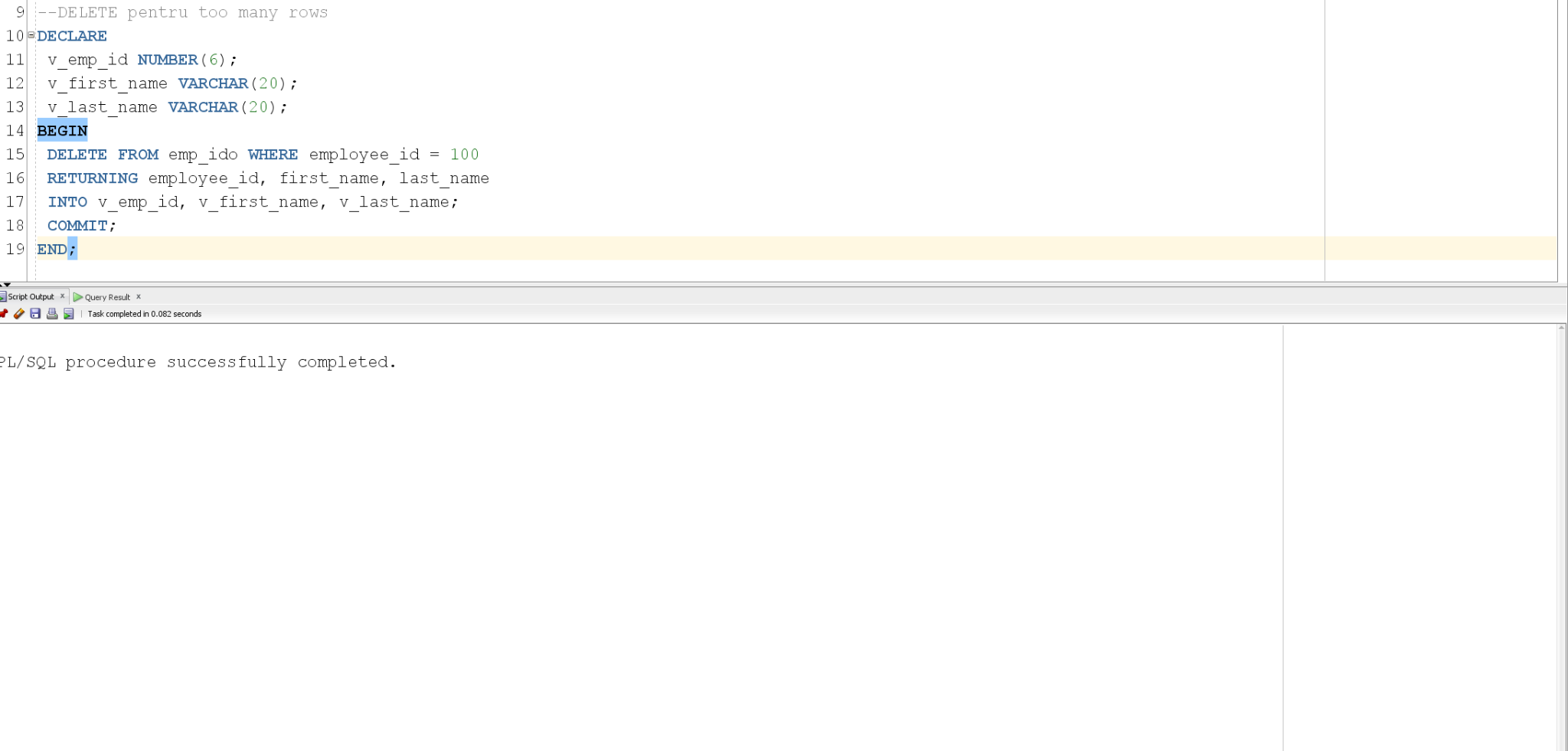
DBMS\_OUTPUT.PUT\_LINE(v\_emp\_id);

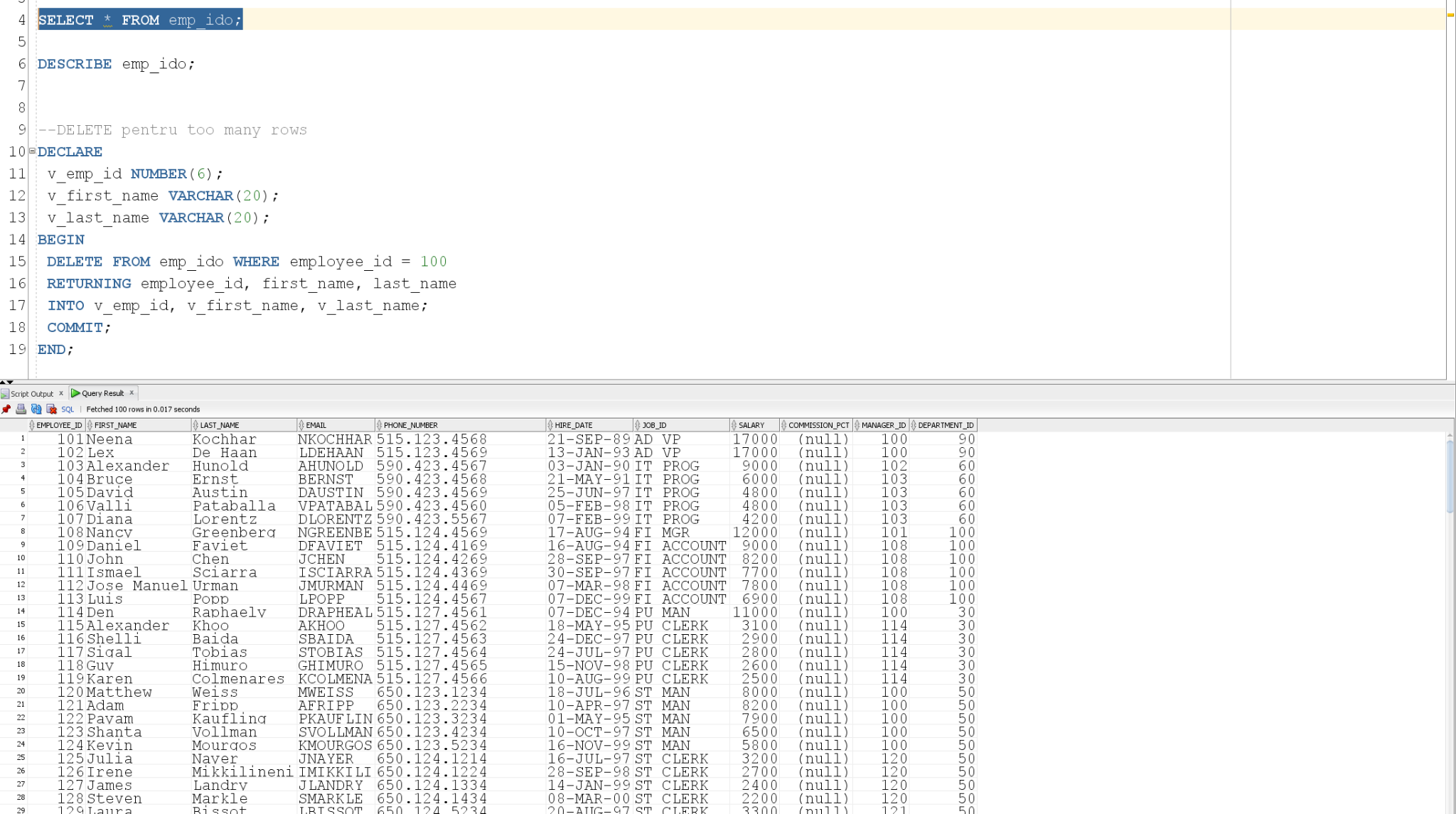
DBMS\_OUTPUT.PUT\_LINE(v\_first\_name);

DBMS\_OUTPUT.PUT\_LINE(v\_last\_name);

COMMIT;

END;





Dupa cum se poate observa, totul a mers cum ne asteptam: A fost eliminata linia din tabela in care employee\_id = 100 fara niciun fel de exceptii.

**A3) Exceptia NO DATA FOUND:**

--DELETE pentru NO DATA FOUND

DECLARE

v\_emp\_id NUMBER(6);

v\_first\_name VARCHAR2(20);

v\_last\_name VARCHAR2(25);

BEGIN

DELETE FROM emp\_ido WHERE employee\_id = 1000

RETURNING employee\_id, first\_name, last\_name

INTO v\_emp\_id, v\_first\_name, v\_last\_name;

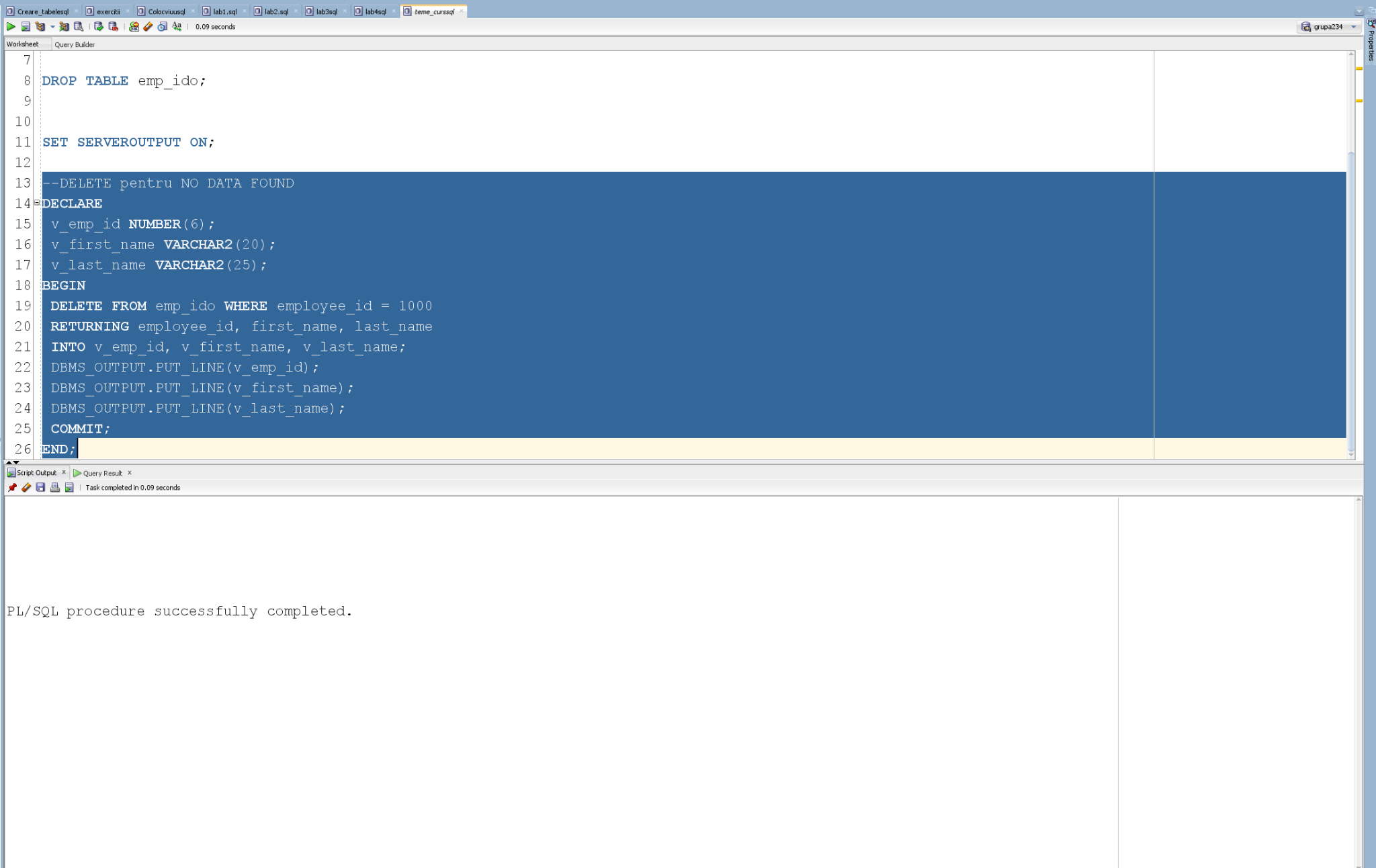
DBMS\_OUTPUT.PUT\_LINE(v\_emp\_id);

DBMS\_OUTPUT.PUT\_LINE(v\_first\_name);

DBMS\_OUTPUT.PUT\_LINE(v\_last\_name);

COMMIT;

END;





Dupa cum se poate vedea, nu vom primi eroare, iar valorile din cauza RETURNING vor ramane la fel.

1. Pentru **UPDATE**

**B1) Exceptia TOO MANY ROWS:**

DECLARE

v\_emp\_id NUMBER(6) := 0;

v\_first\_name VARCHAR2(20) := 'Hello';

v\_last\_name VARCHAR2(25) := 'World!';

BEGIN

UPDATE emp\_ido

SET employee\_id = 1000

WHERE employee\_id > 100

RETURNING employee\_id, first\_name, last\_name

INTO v\_emp\_id, v\_first\_name, v\_last\_name;

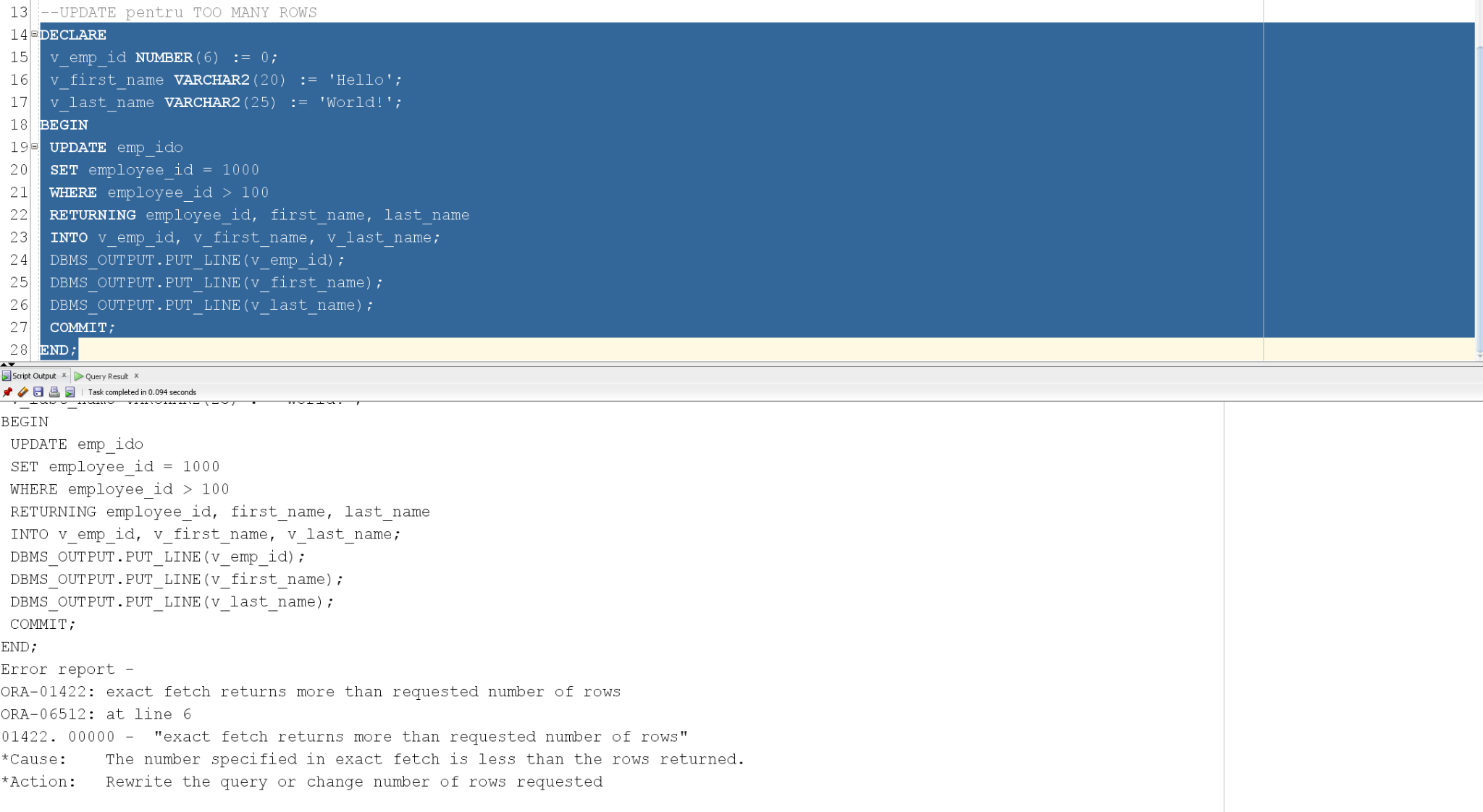
DBMS\_OUTPUT.PUT\_LINE(v\_emp\_id);

DBMS\_OUTPUT.PUT\_LINE(v\_first\_name);

DBMS\_OUTPUT.PUT\_LINE(v\_last\_name);

COMMIT;

END;



Iarasi vom primi aceeasi eroare ce ne spune ca avem prea multe linii pe care incercam sa le updatam.

**A2) O singura linie din tabela:**

--UPDATE pentru o linie

DECLARE

v\_emp\_id NUMBER(6) := 0;

v\_first\_name VARCHAR2(20) := 'Hello';

v\_last\_name VARCHAR2(25) := 'World!';

BEGIN

UPDATE emp\_ido

SET employee\_id = 1000

WHERE employee\_id = 100

RETURNING employee\_id, first\_name, last\_name

INTO v\_emp\_id, v\_first\_name, v\_last\_name;

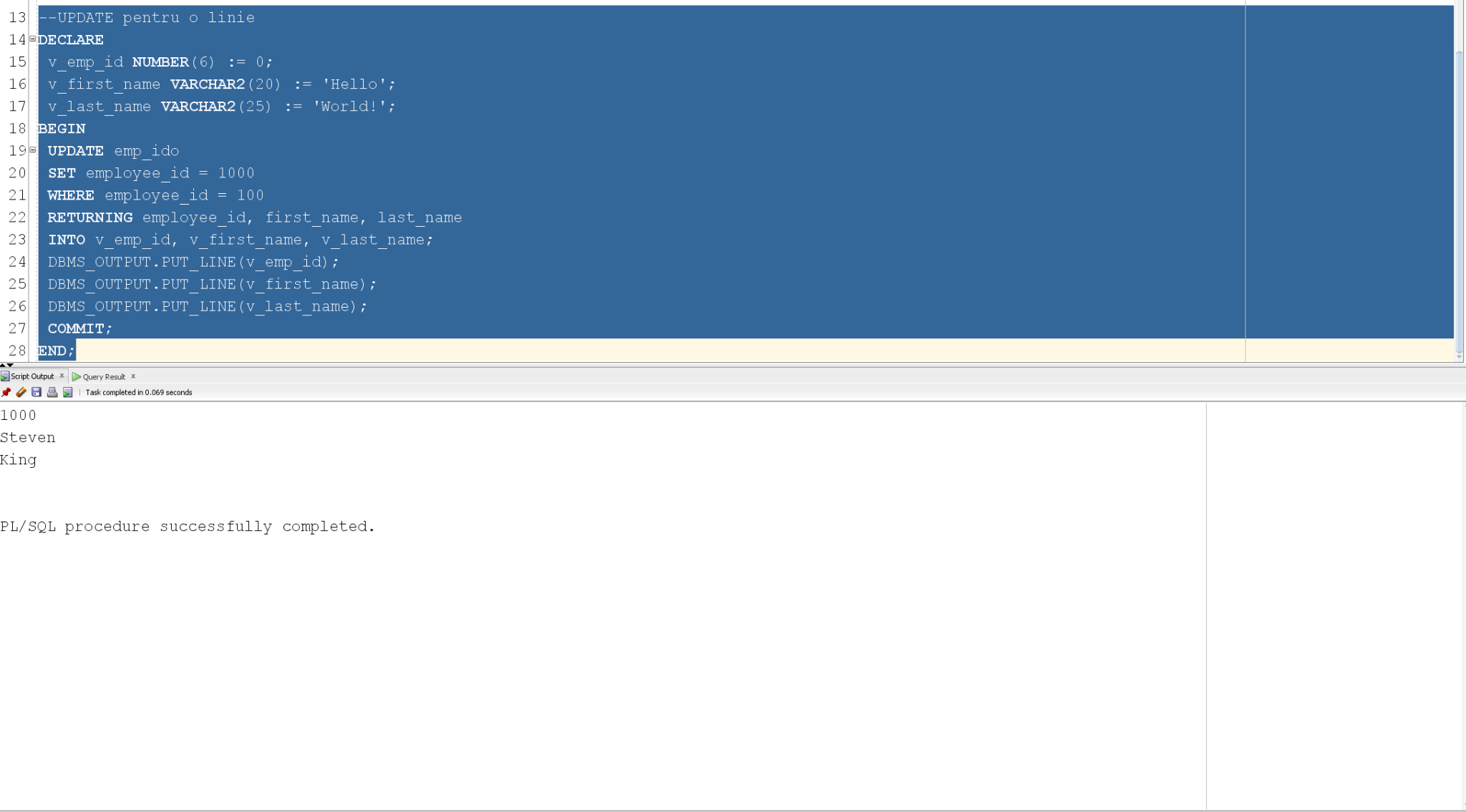
DBMS\_OUTPUT.PUT\_LINE(v\_emp\_id);

DBMS\_OUTPUT.PUT\_LINE(v\_first\_name);

DBMS\_OUTPUT.PUT\_LINE(v\_last\_name);

COMMIT;

END;

****

In acest caz s-a facut update-ul cum ne asteptam, si id\_ul lui Steven King s-a modificat la 1000 asa cum ne asteptam din clauza RETURNING.

**B3) Exceptia NO DATA FOUND:**

--UPDATE pentru NO DATA FOUND

DECLARE

v\_emp\_id NUMBER(6) := 0;

v\_first\_name VARCHAR2(20) := 'Hello';

v\_last\_name VARCHAR2(25) := 'World!';

BEGIN

UPDATE emp\_ido

SET employee\_id = 1000

WHERE employee\_id = 1000

RETURNING employee\_id, first\_name, last\_name

INTO v\_emp\_id, v\_first\_name, v\_last\_name;

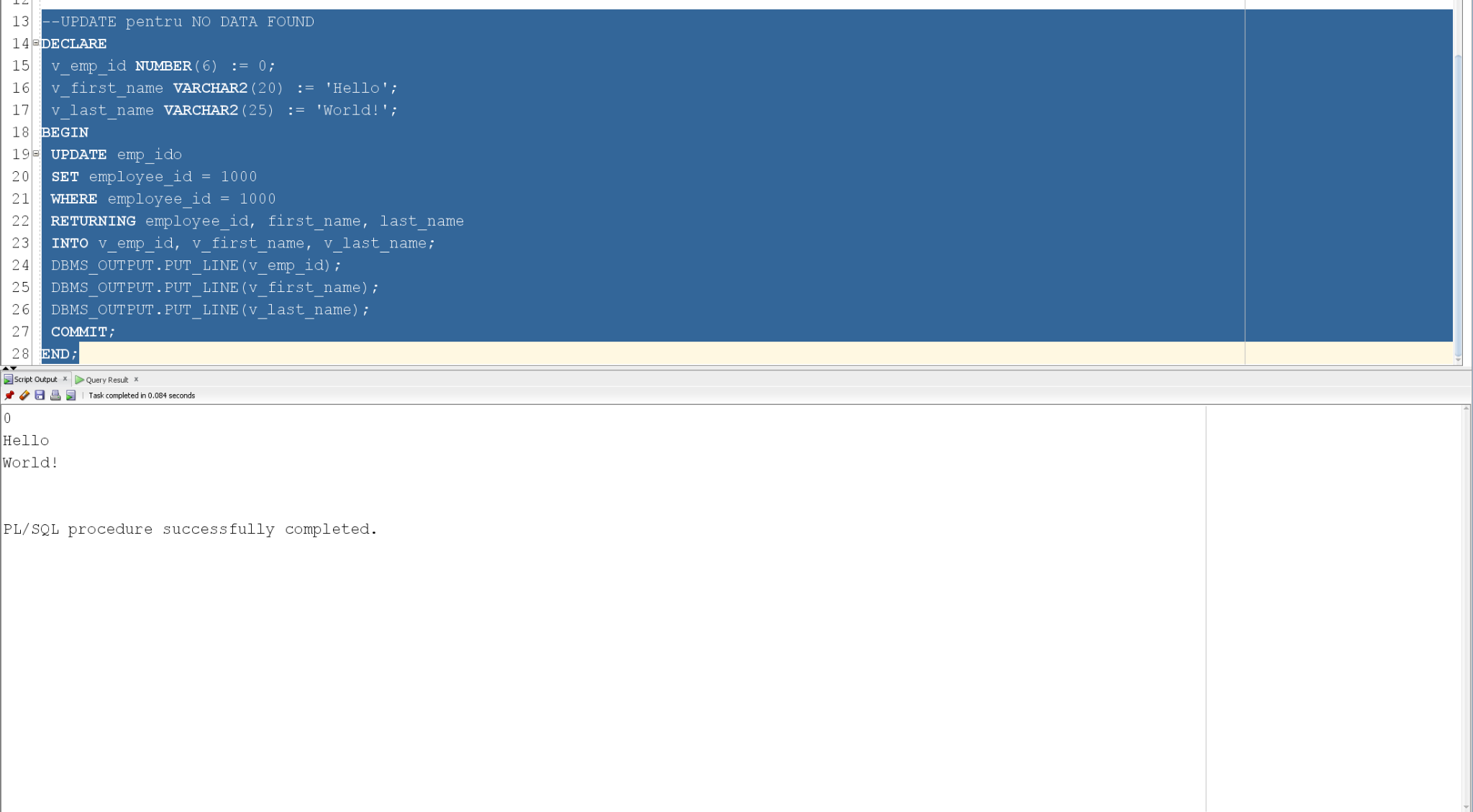
DBMS\_OUTPUT.PUT\_LINE(v\_emp\_id);

DBMS\_OUTPUT.PUT\_LINE(v\_first\_name);

DBMS\_OUTPUT.PUT\_LINE(v\_last\_name);

COMMIT;

END;



Din nou, se poate observa ca pentru NO DATA FOUND, nu primim eroare, nu se updateaza nimic, si valorile variabilelor raman aceleasi.

**CONCLUZIE:** Asadar, atat pentru DELETE cat si pentru UPDATE, cand incercam sa accesam mai multe linii din tabela si avem exceptia TOO MANY ROWS, clauza nu se va executa cu succes avand un mesaj de eroare.

Cand incercam sa accesam o linie din tabela, totul va rula cu succes si vom avea in variabilele noastre informatii din linia pe care trebuie sa o stergem / updatam.

Nu in ultimul rand, cand incercam sa accesam din tabela o linie care

nu exista, ne-am fi asteptat sa primim eroare, dar de fapt clauza se executa cu succes, iar variabilele declarate de noi cat si tabela initiala raman neschimbate.