**Personalizati o tema, folosind pachete predefinite, precum DBMS\_JOB, DBMS\_DDL, DBMS\_SQL**

Am creat mai intai o tabela pe care vom testa aceste pachete:  
  
CREATE TABLE example\_table (

id NUMBER,

name VARCHAR2(100)

);

Am Programat un “Job” sa inseram date (**DBMS\_JOB**) cu delay de un minut:

DECLARE

job\_number NUMBER;

BEGIN

DBMS\_JOB.submit(

job => job\_number,

what => 'BEGIN

INSERT INTO example\_table VALUES (1, ''John'');

INSERT INTO example\_table VALUES (2, ''Jane'');

COMMIT;

END;',

next\_date => SYSTIMESTAMP + INTERVAL '1' MINUTE

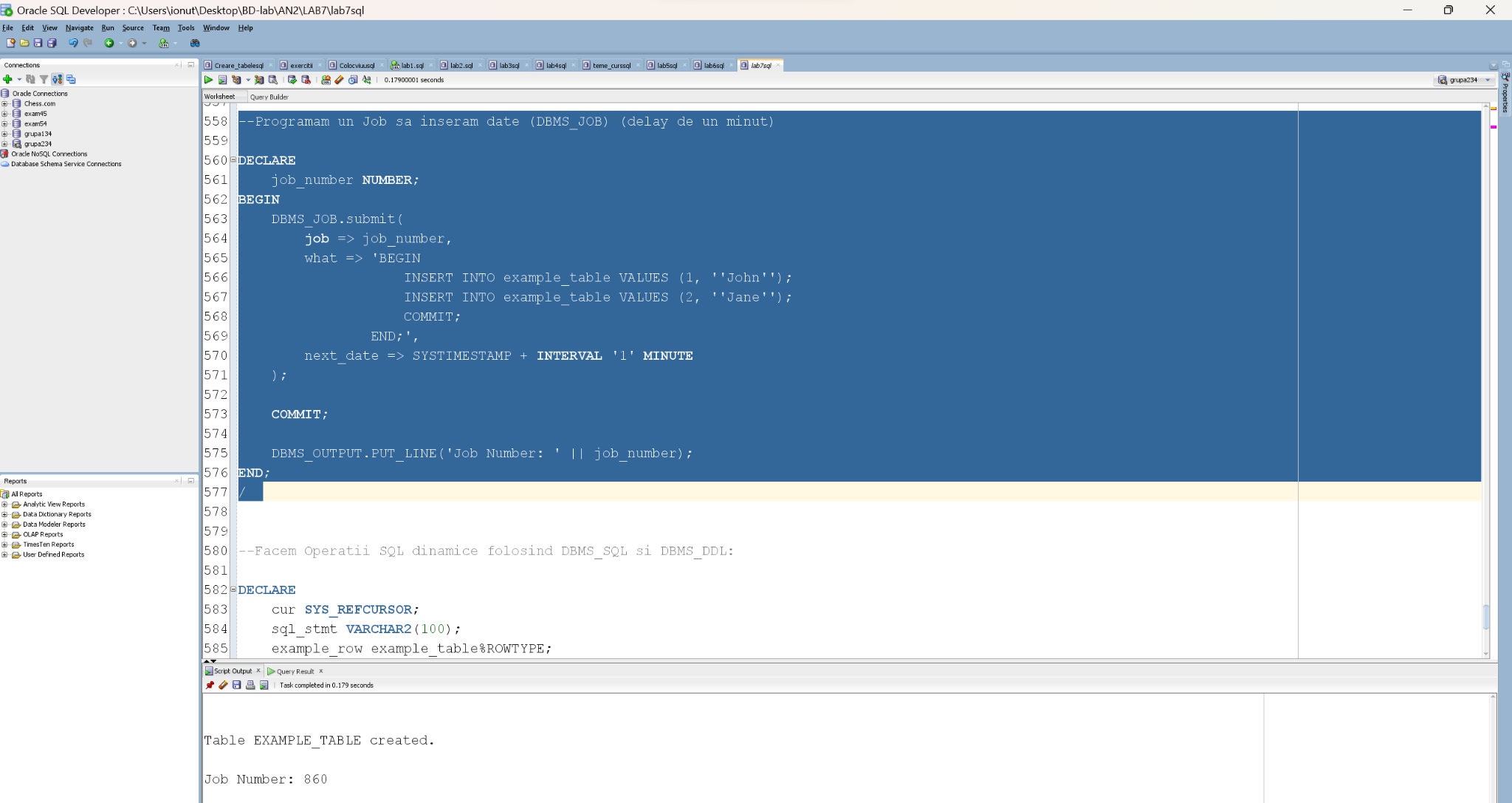
);

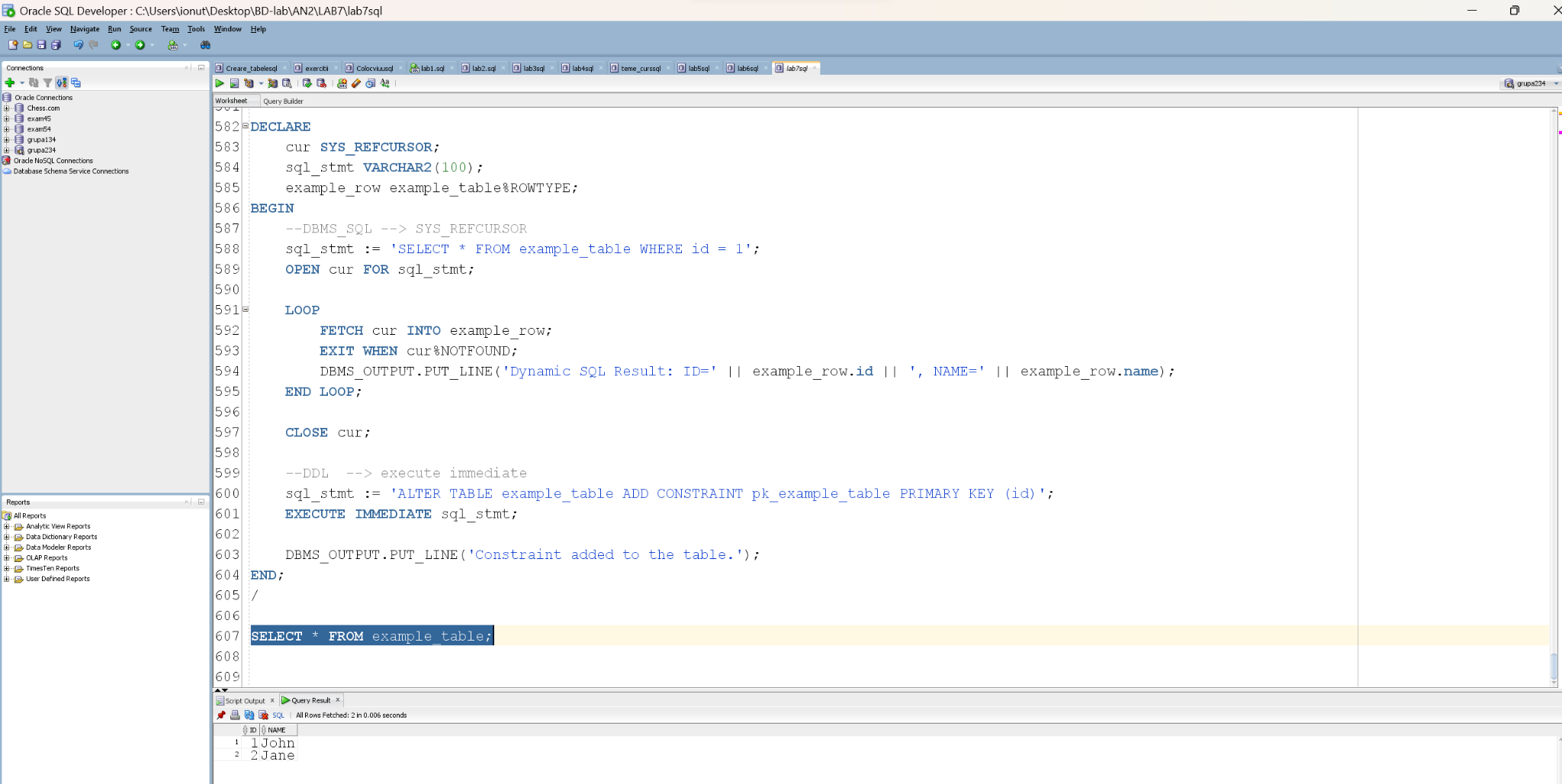
COMMIT;

DBMS\_OUTPUT.PUT\_LINE('Job Number: ' || job\_number);

END;

/



Dupa un minut:  
  


Facem Operatii SQL dinamice folosind **DBMS\_SQL** si **DBMS\_DDL**:

Folosim DBMS\_SQL pentru a selecta date din tabela bazata pe o conditie. De asemenea, folosim DBMS\_DDL pentru a adauga o cheie primara in mod dinamic tabelei.

DECLARE

cur SYS\_REFCURSOR;

sql\_stmt VARCHAR2(100);

example\_row example\_table%ROWTYPE;

BEGIN

--DBMS\_SQL --> SYS\_REFCURSOR

sql\_stmt := 'SELECT \* FROM example\_table WHERE id = 1';

OPEN cur FOR sql\_stmt;

LOOP

FETCH cur INTO example\_row;

EXIT WHEN cur%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE('Dynamic SQL Result: ID=' || example\_row.id || ', NAME=' || example\_row.name);

END LOOP;

CLOSE cur;

--DDL --> execute immediate

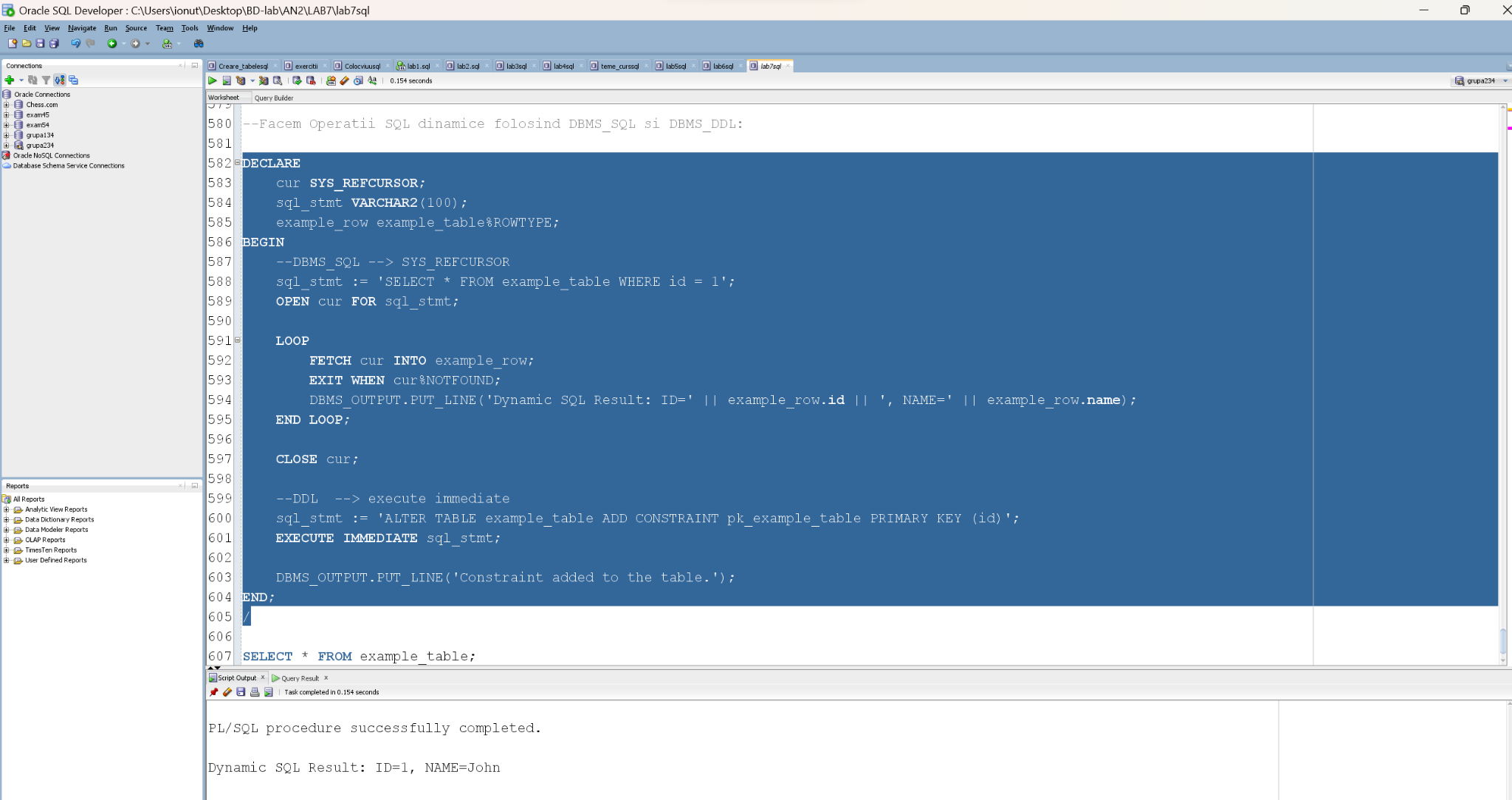
sql\_stmt := 'ALTER TABLE example\_table ADD CONSTRAINT pk\_example\_table PRIMARY KEY (id)';

EXECUTE IMMEDIATE sql\_stmt;

DBMS\_OUTPUT.PUT\_LINE('Constraint added to the table.');

END;

/



**Se poate defini o variabila globala a carei valoare sa poata fi citita sau modificata din sesiuni diferite?**

Raspunsul este ca da, se poate.

Intr-o sesiune cream urmatorul pachet, cu variabila my\_variable globala:

CREATE OR REPLACE PACKAGE my\_package AS

my\_variable NUMBER;

END my\_package;

CREATE OR REPLACE PACKAGE BODY my\_package AS

BEGIN

my\_variable := 42;

END my\_package;

In cealalta sesiune afisam valoarea din pachet:  
  
SET SERVEROUTPUT ON;

DECLARE

v\_value NUMBER;

BEGIN

v\_value := my\_package.my\_variable;

DBMS\_OUTPUT.PUT\_LINE('Value: ' || v\_value);

END;

