

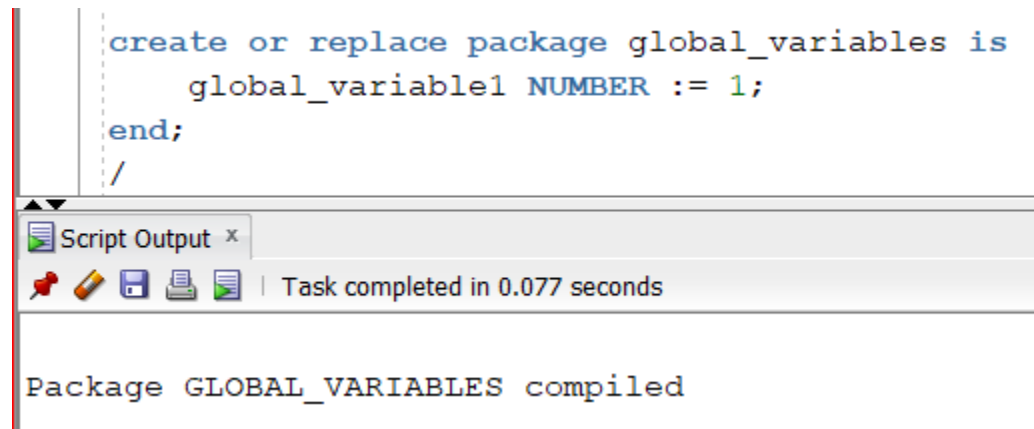
## *Variabile globale*

În PL/SQL, *variabilele globale* se definesc cu ajutorul pachetelor.

Astfel, vom vedea dacă:

1. *valoarea variabilei globale poate fi citită/modificată de programe din aceeași sesiune*
2. *valoarea variabilei globale poate fi citită/modificată de programe din sesiuni diferite*

Începem prin a crea un pachet denumit GLOBAL\_VARIABLES, care va conține o variabilă numită GLOBAL\_VARIABLE1, inițializată cu 1.



```
create or replace package global_variables is
    global_variable1 NUMBER := 1;
end;
/
```

Script Output x

Task completed in 0.077 seconds

Package GLOBAL\_VARIABLES compiled

### 1. Valoarea variabilei globale poate fi citită/modificată de programe din aceeași sesiune?

Pentru a vedea acest lucru, ne creăm un bloc anonim, în care încercăm să modificăm variabila noastră globală astfel:

```
BEGIN
    --inainte de modificare
    dbms_output.put_line('INAINTE DE MODIFICARE: global_variable1 = ' ||
                          global_variables.global_variable1);

    --modificare
    global_variables.global_variable1 := 10;

    --dupa modificare
    dbms_output.put_line('DUPA MODIFICARE: global_variable1 = ' ||
                          global_variables.global_variable1);

END;
/
```

Script Output x

Task completed in 0.077 seconds

PL/SQL procedure successfully completed.

Rezultatul este cel așteptat:

Dbms Output

Buffer Size: 20000

SGBD 231 x

```
INAINTE DE MODIFICARE: global_variable1 = 1
DUPA MODIFICARE: global_variable1 = 10
```

## 2. Valoarea variabilei globale poate fi citită/modificată de programe din sesiuni diferite?

Recompilăm pachetul și procedăm asemănător exercițiului anterior.

**SESIUNEA 1**

```
create or replace package global_variables is
  global_variable1 NUMBER := 1;
end;
/

BEGIN
  --inainte de modificare
  dbms_output.put_line('INAINTE DE MODIFICARE: global_variable1 = ' ||
    global_variables.global_variable1);

  --modificare
  global_variables.global_variable1 := 10;

  --dupa modificare
  dbms_output.put_line('DUPA MODIFICARE: global_variable1 = ' ||
    global_variables.global_variable1);
END;
/
```

Script Output: Task completed in 0.03 seconds

Package GLOBAL\_VARIABLES compiled

PL/SQL procedure successfully completed.

Dbms Output: Buffer Size:20000

SGRD 231

INAINTE DE MODIFICARE: global\_variable1 = 1  
DUPA MODIFICARE: global\_variable1 = 10

**SESIUNEA 2**

```
BEGIN
  --inainte de modificare
  dbms_output.put_line('INAINTE DE MODIFICARE: global_variable1 = ' ||
    global_variables.global_variable1);

  --modificare
  global_variables.global_variable1 := 384;

  --dupa modificare
  dbms_output.put_line('DUPA MODIFICARE: global_variable1 = ' ||
    global_variables.global_variable1);
END;
/
```

Script Output: Task completed in 0.177 seconds

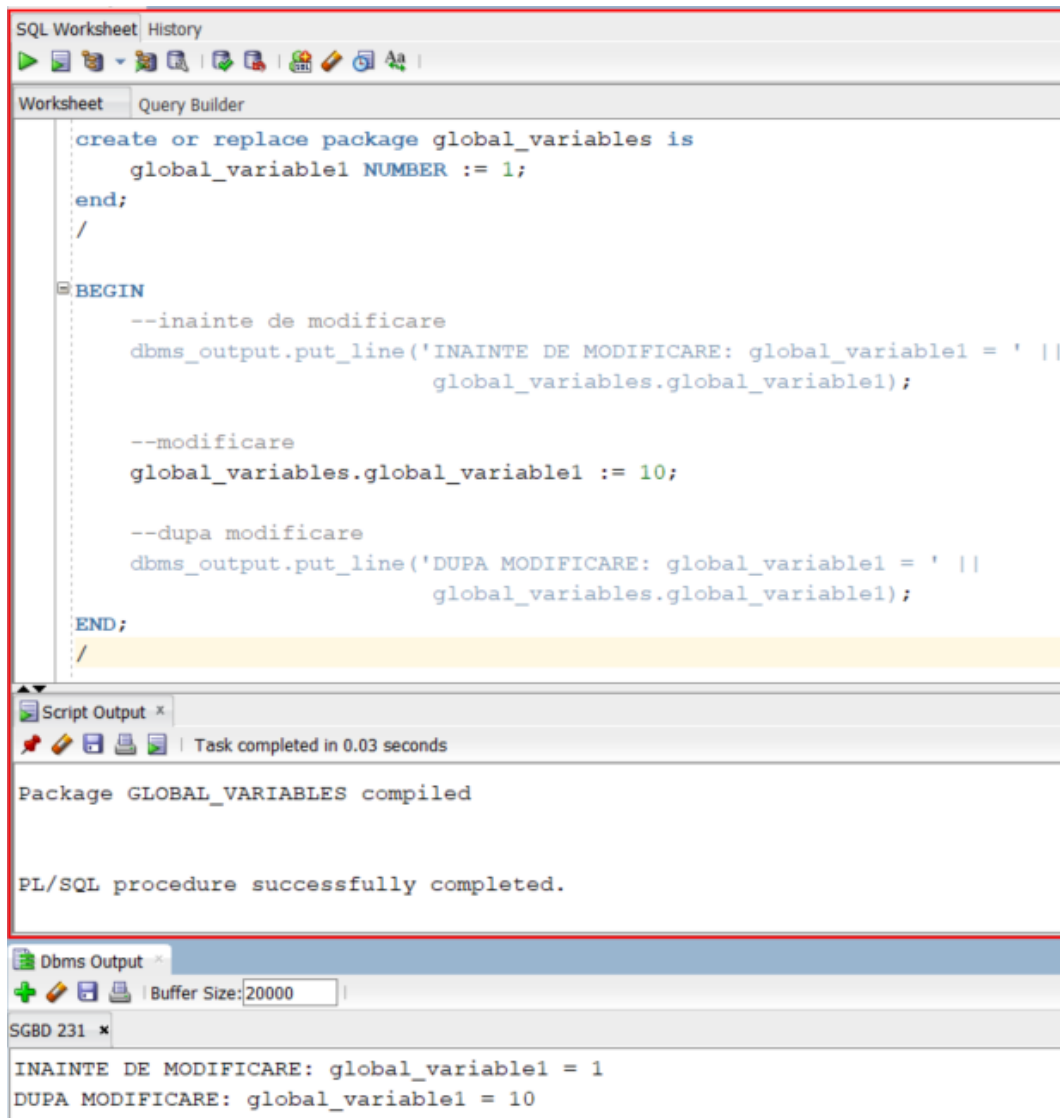
PL/SQL procedure successfully completed.

Dbms Output: Buffer Size:20000

SGRD 231

INAINTE DE MODIFICARE: global\_variable1 = 1  
DUPA MODIFICARE: global\_variable1 = 384

## Sesiunea 1:



The screenshot displays an SQL Worksheet interface with a red border. The main window is titled 'SQL Worksheet' and contains a script for creating and replacing a package named 'global\_variables'. The script includes comments in Romanian and uses 'dbms\_output.put\_line' to display the current value of 'global\_variable1' before and after a modification. Below the script, the 'Script Output' pane shows the execution results, including the message 'Package GLOBAL\_VARIABLES compiled' and 'PL/SQL procedure successfully completed.'. The 'Dbms Output' pane at the bottom shows the output of the 'put\_line' statements, displaying the value of 'global\_variable1' as 1 and then 10.

```
create or replace package global_variables is
    global_variable1 NUMBER := 1;
end;
/

BEGIN
    --inainte de modificare
    dbms_output.put_line('INAINTE DE MODIFICARE: global_variable1 = ' ||
        global_variables.global_variable1);

    --modificare
    global_variables.global_variable1 := 10;

    --dupa modificare
    dbms_output.put_line('DUPA MODIFICARE: global_variable1 = ' ||
        global_variables.global_variable1);
END;
/
```

Script Output x

Task completed in 0.03 seconds

Package GLOBAL\_VARIABLES compiled

PL/SQL procedure successfully completed.

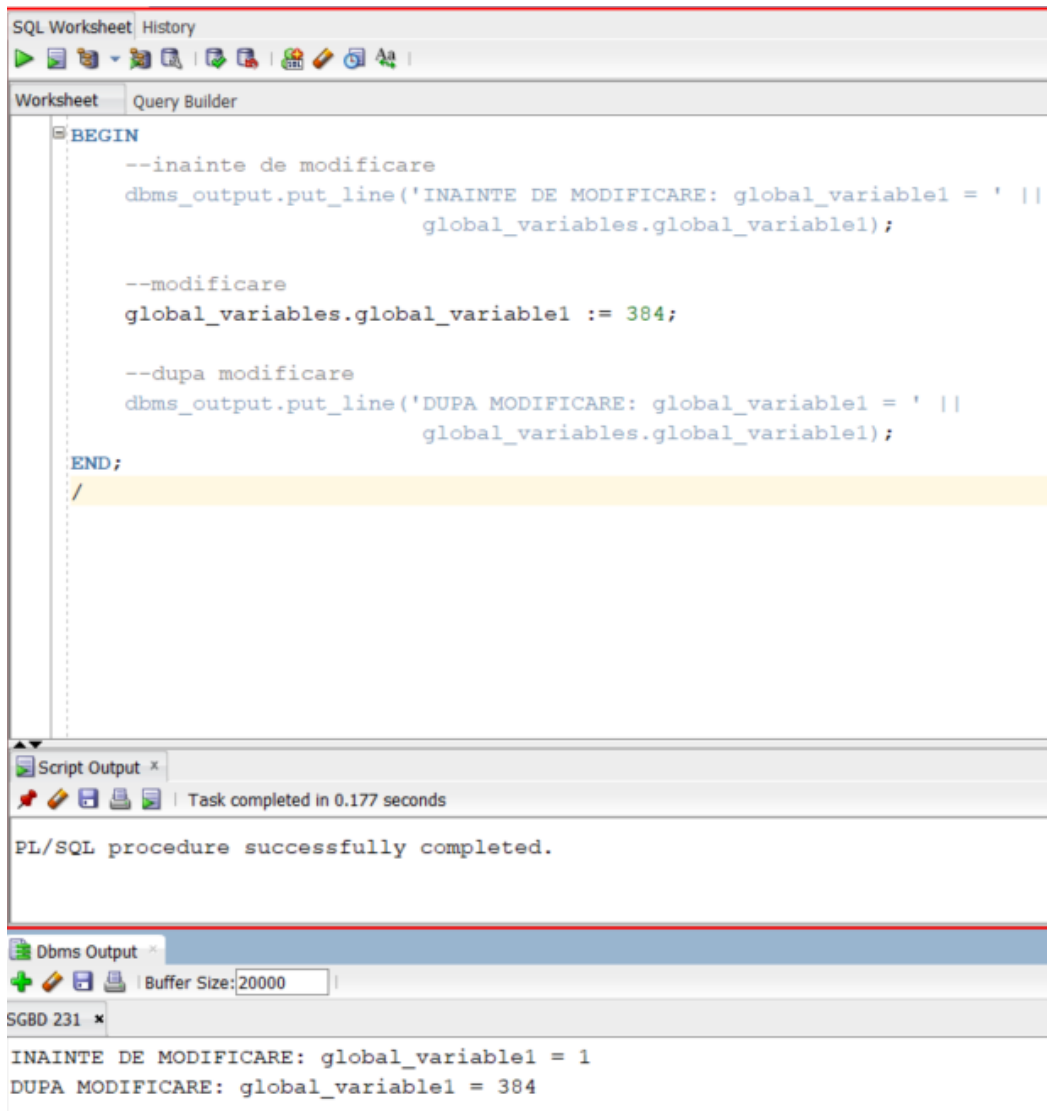
Dbms Output x

Buffer Size: 20000

SGBD 231 x

INAINTE DE MODIFICARE: global\_variable1 = 1  
DUPA MODIFICARE: global\_variable1 = 10

## Sesiunea 2:



The screenshot displays an SQL Worksheet interface with three main panes. The top pane, titled 'Worksheet', contains a PL/SQL procedure named 'BEGIN' with the following code:

```
BEGIN
--inainte de modificare
dbms_output.put_line('INAINTE DE MODIFICARE: global_variable1 = ' ||
                      global_variables.global_variable1);

--modificare
global_variables.global_variable1 := 384;

--dupa modificare
dbms_output.put_line('DUPA MODIFICARE: global_variable1 = ' ||
                      global_variables.global_variable1);
END;
```

The middle pane, titled 'Script Output', shows the execution status: 'Task completed in 0.177 seconds' and 'PL/SQL procedure successfully completed.'.

The bottom pane, titled 'Dbms Output', shows the output of the procedure:

```
INAINTE DE MODIFICARE: global_variable1 = 1
DUPA MODIFICARE: global_variable1 = 384
```

Observăm că modificările au fost făcute, mai exact, și sesiunea 2 a recunoscut pachetul. În schimb, putem vedea faptul că modificarea variabilei dintr-o sesiune nu a fost vizibilă și în cealaltă sesiune.

Afișam, din nou, valoarea variabilei globale în cele două sesiuni:

### Sesiunea 1:

```
SET SERVEROUTPUT ON
BEGIN
    dbms_output.put_line('SESIUNEA 1: global_variable1 = ' ||
                          global_variables.global_variable1);
END;
/
```

Script Output x

Task completed in 0.044 seconds

SESIUNEA 1: global\_variable1 = 1

PL/SQL procedure successfully completed.

### Sesiunea 2:

```
SET SERVEROUTPUT ON
BEGIN
    dbms_output.put_line('SESIUNEA 2: global_variable1 = ' ||
                          global_variables.global_variable1);
END;
/
```

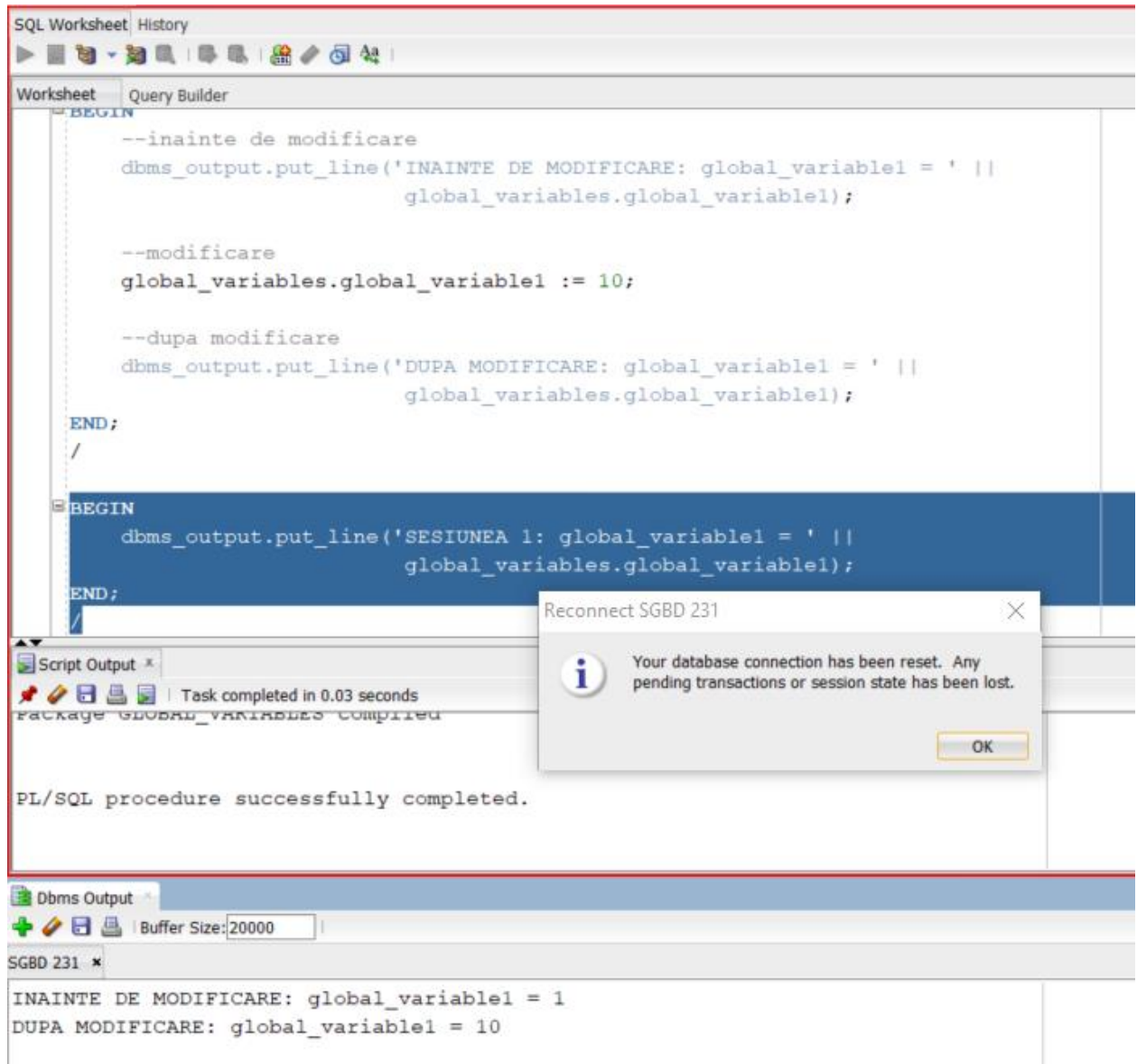
Script Output x

Task completed in 0.067 seconds

SESIUNEA 2: global\_variable1 = 384

PL/SQL procedure successfully completed.

La încercarea de a face acest lucru, am întâmpinat o problemă în sesiunea 1 (care a fost rulată înainte de sesiunea 2):



Acest lucru ne-a determinat să afișăm în Script Output (cu ajutorul SET SERVEROUTPUT ON) variabila.

După cum putem vedea mai sus, citirea/modificarea unei variabile globale în mai multe sesiuni nu are un final neapărat fericit.