EX 1: Ce exceptii pot sa apara la exemplul 4.14?

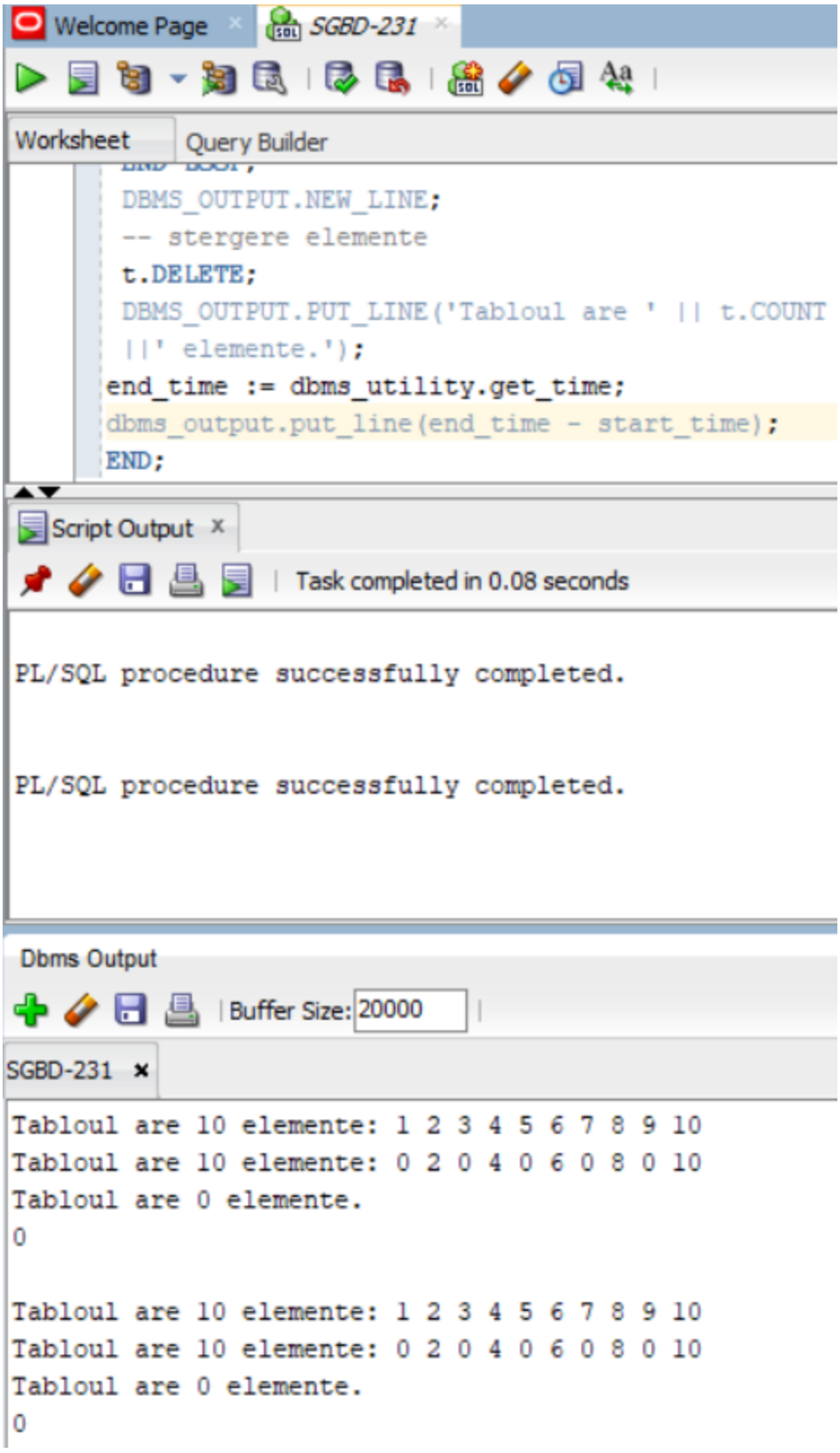
RASPUNS: NO_DATA_FOUND (se foloseste un “id_categorie” care nu exista), VALUE_ERROR (comentam clauza “select into” si facem direct afisarea => colectia nu e initializata corect), COLLECTION_IS_NULL (daca exista un rand care are coloane “null” si datele sunt preluate in “select into”).

Ar mai putea sa apara TOO_MANY_ROWS daca nu era unique “id_categorie” (totusi, este unique).

EX 2: In cat timp ruleaza exemplul cu vectori de la 4.2.6? Folosim dbms_utility.get_time la inceput si la final pentru start_time si end_time si calculam diferenta.

```
DECLARE
 TYPE tab_vec IS VARRAY(10) OF NUMBER;
 t tab_vec := tab_vec();
 start_time pls_integer;
 end_time pls_integer;
BEGIN
start_time := dbms_utility.get_time;
 -- atribuire valori
 FOR i IN 1..10 LOOP
 t.EXTEND;
 t(i):=i;
 END LOOP;
 --parcurgere
 DBMS_OUTPUT.PUT('Tabloul are ' || t.COUNT ||' elemente: ');
 FOR i IN t.FIRST..t.LAST LOOP
 DBMS_OUTPUT.PUT(t(i) || ' ');
 END LOOP;
 DBMS_OUTPUT.NEW_LINE;
 -- numar elemente
 FOR i IN 1..10 LOOP
```

```
 IF i mod 2 = 1 THEN t(i):=null;
 END IF;
 END LOOP;
 DBMS_OUTPUT.PUT('Tabloul are ' || t.COUNT ||' elemente: ');
 FOR i IN t.FIRST..t.LAST LOOP
 DBMS_OUTPUT.PUT(nvl(t(i), 0) || ' ');
 END LOOP;
 DBMS_OUTPUT.NEW_LINE;
 -- stergere elemente
 t.DELETE;
 DBMS_OUTPUT.PUT_LINE('Tabloul are ' || t.COUNT
 ||' elemente.');
end_time := dbms_utility.get_time;
dbms_output.put_line(end_time - start_time);
END;
```


Welcome Page
SGBD-231
Worksheet
Query Builder
DBMS_OUTPUT.NEW_LINE;
-- stergere elemente
t.DELETE;
DBMS_OUTPUT.PUT_LINE('Tabloul are ' || t.COUNT
||' elemente.');
end_time := dbms_utility.get_time;
dbms_output.put_line(end_time - start_time);
END;
Script Output
Task completed in 0.08 seconds
PL/SQL procedure successfully completed.
PL/SQL procedure successfully completed.
Dbms Output
Buffer Size: 20000
SGBD-231
Tabloul are 10 elemente: 1 2 3 4 5 6 7 8 9 10
Tabloul are 10 elemente: 0 2 0 4 0 6 0 8 0 10
Tabloul are 0 elemente.
0
Tabloul are 10 elemente: 1 2 3 4 5 6 7 8 9 10
Tabloul are 10 elemente: 0 2 0 4 0 6 0 8 0 10
Tabloul are 0 elemente.
0

4. Testare UNION/EXCEPT/INTERSECT cu ALL/DISTINCT.

```
declare
   type tablou_imbricat is table of number;
   t1 tablou_imbricat     := tablou_imbricat(1, 1, 2, 2, 3, 4, 5, 5, 5);
   t2 tablou_imbricat     := tablou_imbricat(2, 2, 4, 4, 4, 5, 5, 6, 6, 7, 7);
   result tablou_imbricat := tablou_imbricat();
begin
   result := t1 multiset except all t2;      -- (1, 1, 3, 5)
--    result := t1 multiset except distinct t2; -- (1, 3)
--    result := t2 multiset except all t1;      -- (4, 4, 6, 6, 7, 7)
--    result := t2 multiset except distinct t1; -- (6, 7)

--    result := t1 multiset union all t2;      -- (1, 1, 2, 2, 3, 4, 5, 5, 5, 2, 2, 4, 4, 4, 5, 5, 6, 6, 7, 7)
--    result := t1 multiset union distinct t2; -- (1, 2, 3, 4, 5, 6, 7)

--    result := t1 multiset intersect all t2;      -- (2, 2, 4, 5, 5)
--    result := t1 multiset intersect distinct t2; -- (2, 4, 5)

   for i in 1..result.last loop
      dbms_output.put_line(result(i));
   end loop;

   exception
      when value_error then
         dbms_output.put_line('VALUE_ERROR');
end;
```