

Tratarea excepțiilor în PL/SQL

Tratarea excepțiilor este un mecanism fundamental al limbajului PL/SQL, oferind capacitatea de a detecta, intercepta și gestiona situațiile anormale apărute în timpul execuției unui bloc PL/SQL. Excepțiile permit menținerea stabilității programelor și continuarea executării într-un mod controlat, chiar și atunci când apar erori.

Ce este o excepție?

O excepție reprezintă o condiție de eroare sau o situație neașteptată care apare în timpul execuției codului. Când apare o excepție:

- Execuția blocului normal (secțiunea EXECUTION) se întrerupe.
- Controlul este transferat secțiunii EXCEPTION (dacă aceasta există).
- Dacă nu se tratează excepția, ea este propagată către nivelul superior (procedura apelantă).

Excepțiile pot proveni din:

- funcții interne Oracle,
- erori de integritate,
- încălcarea constrângerilor,
- conversii incorecte de tip,
- codul PL/SQL (prin RAISE),
- baze de date (ORA-xxxx).

Structura unui bloc PL/SQL cu tratarea excepțiilor:

```
DECLARE
  -- declaratii
BEGIN
  -- cod executabil
EXCEPTION
  WHEN exceptie1 THEN
    -- tratare specifica
  WHEN exceptie2 THEN
```

```
-- alta tratare  
WHEN OTHERS THEN  
    -- tratarea oricărui altă excepție  
END;
```

Secțiunea EXCEPTION este optională, dar recomandată în majoritatea programelor.

Tipuri de erori în PL/SQL

Erorile se pot grupa astfel:

- Erori de sintaxă – detectate în faza de compilare.
- Erori logice – codul se execută corect, dar rezultatul este greșit.
- Erori de execuție – apar în timpul rulării. Acestea sunt tratate prin mecanismele de excepții.

Excepțiile se ocupă în special de erorile de execuție, dar pot gestiona și alte situații.

Clasificarea excepțiilor

PL/SQL oferă mai multe categorii de excepții:

1. Excepții predefinite (interne) - Sunt generate automat de Oracle și au nume predefinite în PL/SQL. Exemple:

Excepție	Cod	Descriere
NO_DATA_FOUND	ORA-01403	Nicio linie nu a fost returnată
TOO_MANY_ROWS	ORA-01422	O interogare a returnat prea multe linii
ZERO_DIVIDE	ORA-01476	Împărțire la zero
INVALID_NUMBER	ORA-01722	Conversie invalidă la număr
DUP_VAL_ON_INDEX	ORA-00001	Încălcare de constrângere unică

Aceste excepții pot fi tratate direct:

```
EXCEPTION  
  WHEN NO_DATA_FOUND THEN ...  
  WHEN ZERO_DIVIDE THEN ...
```

2. Excepții nepredefinite (interne) - Excepții generate tot de Oracle, dar care nu au nume în PL/SQL, având doar cod ORA-xxxx. Pentru a le utiliza, trebuie asociate cu un identificator folosind PRAGMA EXCEPTION_INIT.

Exemplu:

```
DECLARE  
  exceptie_vanzari EXCEPTION;  
  PRAGMA EXCEPTION_INIT(exceptie_vanzari, -2292); -- ORA-02292  
BEGIN  
  ...  
EXCEPTION  
  WHEN exceptie_vanzari THEN  
    DBMS_OUTPUT.put_line('Stergere imposibila: integritate referentiala.');
```

3. Excepții definite de utilizator (externe)

Programatorul poate declara și ridica propriile excepții.

Declarare: my_exception EXCEPTION;

Generare (raising): RAISE my_exception;

Exemple:

```
DECLARE  
  limita_depasita EXCEPTION;  
BEGIN  
  IF valoare > 100 THEN  
    RAISE limita_depasita;  
  END IF;  
EXCEPTION  
  WHEN limita_depasita THEN
```

```
DBMS_OUTPUT.put_line('Valoarea depăseste limita permisă.');
END;
```

Mecanisme de ridicare a excepțiilor

1. Ridicare automată - Oracle ridică excepția când apare o eroare internă.
2. Ridicare manuală cu RAISE

```
RAISE NO_DATA_FOUND;
RAISE exception_definita;
```

3. Ridicare cu RAISE_APPLICATION_ERROR - Permite generarea de coduri de eroare personalizate între -20000 și -20999.

```
RAISE_APPLICATION_ERROR(-20001, 'Mesaj personalizat.');
```

Funcții pentru identificarea excepțiilor

PL/SQL oferă funcții utile pentru inspectarea erorilor:

- SQLCODE – returnează codul numeric ORA al ultimei excepții.
 - SQLERRM – returnează mesajul corespunzător erorii.
- EXCEPTION
 - WHEN OTHERS THEN
 - DBMS_OUTPUT.put_line(SQLCODE);
 - DBMS_OUTPUT.put_line(SQLERRM);

Tratarea mai multor excepții în același bloc

Excepțiile se tratează într-o structură selectivă:

```
EXCEPTION
  WHEN NO_DATA_FOUND THEN ...
  WHEN TOO_MANY_ROWS THEN ...
  WHEN INVALID_NUMBER THEN ...
  WHEN OTHERS THEN ...
```

Important: Clauza WHEN OTHERS trebuie pusă ultima.

Cazuri speciale în tratarea excepțiilor

1. Excepții care nu declanșează implicit o eroare

Unele excepții, precum NO_DATA_FOUND într-un cursor explicit, nu apar automat — trebuie detectate de programator.

2. Excepții în triggeri

Triggerii nu pot folosi RETURN; singura modalitate de a opri operarea este ridicarea unei excepții.

3. Tratarea excepțiilor în blocuri imbricate

Fiecare bloc interior poate avea propria secțiune EXCEPTION.

4. Excepțiile relansate

Dacă dorim să tratăm parțial excepția, apoi să o lăsăm să continue:

```
WHEN NO_DATA_FOUND THEN
    log_error();
    RAISE; -- propagare mai departe
```

Propagarea excepțiilor

Dacă o excepție nu este tratată într-un bloc:

- Este transmisă blocului apelant.
- Continuu până la nivelul cel mai de sus.
- Dacă rămâne neîntreruptă, programul dă eroare și se oprește.

Excepțiile sunt locale unui bloc, dar pot fi transmise între proceduri și funcții.

Exemplu:

Funcția A apelează funcția B. B generează o excepție. Dacă B nu o tratează → se duce în A. Dacă A nu o tratează → se duce la SQL engine → ORA-xxxx afișat utilizatorului.

Excepții în proceduri și funcții

Procedurile pot trata local excepțiile, sau le pot propaga implicit. Funcțiile sunt mai sensibile: dacă apare o excepție necontrolată, valoarea returnată devine NULL sau funcția se oprește cu eroare.

Exemplu complet de tratare a excepțiilor

```
DECLARE
    ex_salariu_mic EXCEPTION;
BEGIN
    UPDATE angajati
    SET salariu = salariu - 500
    WHERE id = 10;

    IF SQL%ROWCOUNT = 0 THEN
        RAISE NO_DATA_FOUND;
    END IF;

    IF (SELECT salariu FROM angajati WHERE id = 10) < 0 THEN
        RAISE ex_salariu_mic;
    END IF;

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.put_line('Angajatul nu există.');
    WHEN ex_salariu_mic THEN
        DBMS_OUTPUT.put_line('Salariul rezultat nu poate fi negativ.');
    WHEN OTHERS THEN
        DBMS_OUTPUT.put_line('Eroare: ' || SQLERRM);
END;
```

Exercitii

- a) Să se creeze un bloc care insereaza in tabelul STUDENTS o intrare cu cheia primară dată de la tastatură. Celelalte coloane pot fi identice/ place holdere.
- b) Să se trateze eroarea apărută atunci când se dă de la tastatură o cheie primară care există deja. Inserția in tabel să fie făcută oricum.
- c)