

Jazyk PL/SQL

Triggery

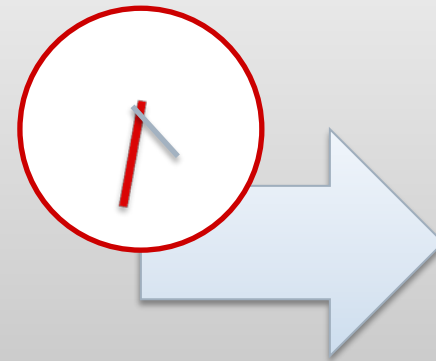
10

Uložené program. jednotky

Uložená programová jednotka (Podprogram) – pojmenované seskupení SQL a PL/SQL příkazů, pomocí nichž řešíme firemní a databázové úkoly. Jsou uloženy v rámci databázového systému ORACLE jako databázový objekt. Může být opakovaně spouštěn různými uživateli databáze.

□ Typy podprogramů:

- procedura (procedure)
- funkce (function)
- balík (package)
- **spouštěč (trigger)**



TRIGGERY

TRIGGER – uložený podprogram, který se spouští automaticky při modifikaci tabulky – nezávisle na uživateli a aplikaci.

☐ Odlišnosti triggerů od procedur, funkcí a balíků:

- jsou implicitně spouštěny při modifikaci tabulky
- definují se pouze pro databázové tabulky
- nepřijímají parametry
- Nelze použít příkazy DDL a řízení transakcí
- Spustí se pouze při DML příkazech:
 - ☐ Update
 - ☐ Insert
 - ☐ Delete

Triggery - použití

- ❑ nepovolí neplatné datové transakce
- ❑ zajišťují komplexní bezpečnost
- ❑ zajišťují referenční integritu přes všechny uzly v integrované databázi
- ❑ vytvářejí strategická a komplexní aplikační pravidla
- ❑ zajišťují audit (sledování)
- ❑ spravují synchronizaci tabulek
- ❑ zaznamenávají statistiku často modifikovaných tabulek

Trigger – referenční integrita

Triggery můžeme použít v případě, kdy situaci nelze zajistit deklarativní referenční integritou, tzn. není možno zajistit integritní omezení na úrovni tabulky pomocí:

- NOT NULL
- UNIQUE KEY
- PRIMARY KEY
- FOREIGN KEY
- CHECK
- UPDATE
- UPDATE | DELETE SET NULL
- UPDATE | DELETE SET DEFAULT



Trigger typy

□ Podle časování

■ **BEFORE** – před provedením manipulace

Slouží pro kontrolu; zabrání nevhodnému zpracování, rozhodne, zda se má daný příkaz dokončit, před dokončením tabulkové operace INSERT či UPDATE zjistí hodnotu sloupce.

■ **AFTER** – po dokončení manipulace

Slouží k reakci na požadovanou změnu dat.

Trigger typy

- Podle události (triggering event)
 - INSERT
 - UPDATE
 - DELETE
 - Kombinace INSERT, UPDATE, DELETE

Trigger - CREATE

❑ Vytvoření triggeru:

```
CREATE [ OR REPLACE ] TRIGGER <jméno triggeru>  
{BEFORE| AFTER} [INSTEAD OF]  
<triggering event> ON <jméno tabulky>  
[FOR EACH ROW]  
[WHEN ( <podmínka> )]
```

```
DECLARE
```

```
<deklarační část>
```

```
BEGIN
```

```
<tělo triggeru>
```

```
END;
```

Trigger - INSTEAD OF

Nová možnost od Oracle 8 - přehledný způsob modifikace pro pohledy, které není možno modifikovat přímo pomocí INSERT, UPDATE, DELETE, protože podřízené tabulky obsahují spojení.

Trigger provede SQL operace přímo na podřízených tabulkách

INSTEAD OF nemůže obsahovat v dotazech následující struktury:

- ☐ množinové operátory
- ☐ skupinové funkce
- ☐ spojení
- ☐ klauzuli DISTINCT

Trigger – FOR EACH ROW

FOR EACH ROW – fráze, která určuje, zda se jedná o řádkový či příkazový trigger.

☐ Řádkový

klauzule je uvedena - trigger se spouští jednou pro každý řádek.

V těle se můžeme odkazovat na hodnoty polí aktuálního řádku:

- ☐ **Stará hodnota** – odkaz **:OLD.jméno_sloupce**

- ☐ **Nová hodnota** – odkaz **:NEW.jméno sloupce**

☐ Příkazový

Není uvedena - trigger se spustí pouze jednou bez ohledu na množství modifikovaných řádek.

Nelze se odkazovat na hodnoty řádku.

Trigger - WHEN

Fráze WHEN určuje podmíněné spuštění triggeru – trigger se spustí jen v případě platnosti podmínky.

```
...  
WHEN (<podmínka>)  
DECLARE  
...  
BEGIN  
  
END;
```

