



UNIPAR
UNIVERSIDADE PARANAENSE

Programando em SQL

Triggers, Stored Procedures e
funções

Objetivo

- Descrever as funcionalidades de triggers, store procedures e funções;
- Apresentar vantagens e benefícios;
- Criação e Gerenciamento de Store Program;

Triggers (gatilhos)

- Os triggers são gatilhos que disparam a execução de códigos armazenados no servidor, ou outros códigos SQL, sem a necessidade de uma chamada específica.
- Iniciados a partir de uma situação satisfeita no servidor.
 - Antes ou depois de inserção, alteração ou exclusão.
- Vantagem
 - Automatização de operações.

Triggers (gatilhos)

- Requisitos:
 - Especificar quando um trigger deve ser executado. Isso é desmembrado em um *evento* que faz com que o trigger seja verificado e uma *condição* que precisa ser satisfeita para que a execução do trigger prossiga.
 - Especificar as *ações* a serem tomadas quando o trigger for executado.
 - **Modelo evento-condição-ação**

Necessidade de trigger

- Mecanismos úteis para alertar usuários ou iniciar tarefas automaticamente, quando certas condições forem satisfeitas
- Exemplos:
 - Atualizar estoque em sistemas de venda;
 - Inserir empréstimos ao utilizar limite em sistemas bancários.
 - Incrementar contador.
 - Alertar estoque mínimos.
 - E outros...

Triggers

- Os triggers armazenados, quando criados e compilados, são inseridos em uma tabela chamada de TRIGGERS no BD INFORMATION_SCHEMA, que é um dos dicionários de dados do PostgreSQL;
 - `SELECT * FROM INFORMATION_SCHEMA.TRIGGERS;`



UNIPAR
UNIVERSIDADE PARANAENSE

Definir Trigger

- Pode-se definir mais de um trigger para uma determinada condição;
- O PostgreSQL utiliza ordem alfabética;
- Quando há o retorno de NULL, todos os demais triggers são abortados;

Tipos de Triggers

Modo	Descrição
BEFORE INSERT	O trigger é disparado antes de uma ação de inserção.
BEFORE UPDATE	O trigger é disparado antes de uma ação de alteração.
BEFORE DELETE	O trigger é disparado antes de uma ação de exclusão.
AFTER INSERT	O trigger é disparado depois de uma ação de inserção
AFTER UPDATE	O trigger é disparado depois de uma ação de alteração.
AFTER DELETE	O trigger é disparado depois de uma ação de exclusão.

Tipos de Triggers

- É possível combinar alguns dos modos, desde que tenham a operação de AFTER ou BEFORE em comum.

Modo	Descrição
BEFORE INSERT OR UPDATE OR DELETE	O trigger é disparado antes de uma ação de inserção ou de alteração ou de exclusão.



UNIPAR
UNIVERSIDADE PARANAENSE

Delimitador

- É utilizado o comando DELIMITER para alterar o delimitador da instrução.
- Isto permite que o delimitador ';' usado no corpo de procedure seja passado para o servidor em vez de ser interpretado.

Eventos e Variáveis de Contexto

Evento	Momento em que é disparado	Variáveis de contexto
Before insert	Antes de uma inserção	New
After insert	Depois de uma inserção	New*
Before update	Antes de uma atualização	New e Old
After update	Depois de uma atualização	New* e Old
Before delete	Antes de uma exclusão	Old
After delete	Após uma exclusão	Old

* Altera os valores das variáveis nesses eventos não causará qualquer efeito prático.

Trigger

- Sintaxe

```
CREATE TRIGGER name { BEFORE | AFTER } { event [ OR ... ] }  
ON table  
[ FOR [ EACH ] { ROW | STATEMENT } ]
```

```
EXECUTE PROCEDURE funcname ( arguments )
```

Exemplo

```
CREATE FUNCTION ftesteref() RETURNS trigger AS $$  
BEGIN  
    INSERT INTO test2 values (NEW.a1);  
    DELETE FROM test3 WHERE a3 = NEW.a1;  
    UPDATE test4 SET b4 = b4 + 1 WHERE a4 = NEW.a1;  
    RETURN NULL;  
END;  
$$ LANGUAGE plpgsql;
```

```
CREATE TRIGGER testref before INSERT ON test1  
FOR EACH ROW EXECUTE PROCEDURE ftesteref();
```



Para Praticar

- Baseado no DER, crie as tabelas necessárias, e crie uma função e um trigger que atualize a quantidade em estoque de cada produto vendido. Nesse momento, não é necessário validar se há produtos disponível para venda.



Referências Bibliográficas

- **Korth, Henry F. e Silbershcatz, Abraham; Sistemas de Banco de Dados; Elsevier; 5ª Ed., 2006.**
- **Elmasri, Ramez; Sistemas de Banco de Dados; Addison Wesley, 2005.**
- **MILANI, André; PostgreSQL Guia do Programador. Novatec, 2008**