

Per l'accesso via software interno a delle basi di dati postgresql supporta un linguaggio di programmazione procedurale PL/pgSQL.

È più potente del normale SQL, permette cicli e strutture di controllo più avanzate non nativamente supportate da SQL.

I moduli software sono chiamati **function**, o **stored procedure**, ed una volta dichiarata è mantenuta nel database di riferimento, e può essere invocata da una **TRIGGER**.

Istruzione per creare una stored procedure o

```
CREATE FUNCTION (arg, argtype, ..., argn argtype)  
RETURNS {return-type | TABLE (cols coltype...)} AS  
$$
```

```
DECLARE
```

```
<deklarazioni variabili interne>
```

```
BEGIN
```

```
<istruzioni>
```

```
END
```

```
$$ language plpgsql;
```

se return type è void allora è una procedura

# TRIGGER in SQL

Un trigger è un meccanismo per specificare che una certa stored procedure debba essere automaticamente eseguita in coincidenza di un evento, in particolare l'inserimento, cancellazione o aggiornamento di una tupla da una relazione; Viene associato ad una tabella

## Sintassi

```
CREATE TRIGGER trigger_name [BEFORE | AFTER | INSTEAD OF]
[INSERT | DELETE | UPDATE] ON table_name
```

```
[FOR EACH ROW | FOR EACH STATEMENT] EXECUTE
PROCEDURE procedure_name;
```

perché la procedure possa essere utilizzata essa deve avere come return type un TRIGGER

## Semantica:

- **BEFORE | AFTER | INSTEAD OF**: un trigger può essere attivato prima o dopo che l'operazione di modifica avvenga, oppure può sostituire l'operazione con INSTEAD OF (per le viste)
- **INSERT | DELETE | UPDATE**: tipo di aggiornamento che attiva trigger
- **FOR EACH ROW | FOR EACH STATEMENT**:
  - se per ogni riga il trigger verrà attivato per ognuna delle tuple coinvolte
  - se per ogni statement, invece il trigger è attivato una volta sola per uno statement

## Alcune aggiunte

All'interno del blocco di istruzioni di una funzione definita per un trigger sono disponibili alcune variabili speciali:

- OLD: rappresenta la vecchia tupla per operazioni di UPDATE / DELETE, ed è NULL se è scelto FOR EACH STATEMENT o per gli INSERT
- NEW rappresenta la nuova tupla nelle operazioni di INSERT / UPDATE, ed è NULL per FOR EACH STATEMENT e per DELETE

**Obs:** Se il trigger è attivato da una transazione, le operazioni nell'ambito del trigger sono considerate appartenenti a tale transazione.

Se il trigger è definito BEFORE e FOR EACH ROW allora:

- se il valore  $v$  della procedura invocata dal trigger che opera sulla tupla  $t$  è NULL, allora l'operazione che ha scatenato il trigger non è eseguita per  $t$
- se invece non è NULL allora  $v$  è di tipo record e in quel caso l'operazione che ha scatenato il trigger è eseguita ed usa come tupla  $t$  proprio  $v$

Altrimenti, il valore di ritorno è ignorato