POSTGRESQL (2)

NỘI DUNG

1. Function/ Store procudure
2. Trigger
3. Q&A

1. FUNCTION

- Function – Hàm: sử dụng ngôn ngữ SQL để tạo ra các hàm, thực thi một nhiệm vụ cụ thể nào đó.

- Cú pháp Fucntion:

```
CREATE [ OR REPLACE ] FUNCTION function_name( [ [ argmode ] [ argname ] argtype [ { DEFAULT | = } default_expr ] [, ...] ] )
RETURNS [DATA TYPE]
{ LANGUAGE lang_name
   | COST execution_cost
   | ROWS result_rows
   | SET configuration_parameter { TO value | = value | FROM CURRENT }
   | AS 'definition'
   | AS 'obj_file', 'link_symbol‘   } ...
   [ WITH ( attribute [, ...] ) ]
```

```
CREATE FUNCTION add(integer, integer) RETURNS integer
    AS 'select $1 + $2;'
    LANGUAGE SQL
    IMMUTABLE
    RETURNS NULL ON NULL INPUT
```

```
CREATE OR REPLACE FUNCTION increment(i integer) RETURNS integer AS $$
    BEGIN
        RETURN i + 1;
    END;
$$ LANGUAGE plpgsql;
```

```
CREATE FUNCTION dup(in int, out f1 int, out f2 text)
    AS $$ SELECT $1, CAST($1 AS text) || ' is text' $$
    LANGUAGE SQL
```

2. TRIGGER

- Trigger: tương tự như function được viết bởi ngôn ngữ SQL.
- Được gọi đến khi có hành động nào đó được thao tác đối với 1 hay nhiều table (Được gọi đến khi insert hay update hay delete bản ghi đối với 1 hay nhiều table).

```
CREATE [ CONSTRAINT ] TRIGGER name { BEFORE | AFTER | INSTEAD OF } { event [ OR ... ] }
    ON table
    [ FROM referenced_table_name ]
    [ NOT DEFERRABLE | [ DEFERRABLE ] { INITIALLY IMMEDIATE | INITIALLY DEFERRED } ]
    [ FOR [ EACH ] { ROW | STATEMENT } ]
    [ WHEN ( condition ) ]
    EXECUTE PROCEDURE function_name ( arguments )
```

where event can be one of:

```
    INSERT | UPDATE [ OF column_name [, ... ] ] | DELETE | TRUNCATE
```

When	Event	Row-level	Statement-level
BEFORE	INSERT/UPDATE/DELETE	Tables	Tables and views
	TRUNCATE	—	Tables
AFTER	INSERT/UPDATE/DELETE	Tables	Tables and views
	TRUNCATE	—	Tables
INSTEAD OF	INSERT/UPDATE/DELETE	Views	—
	TRUNCATE	—	—

```
CREATE OR REPLACE FUNCTION update_timestam()
  RETURNS trigger AS
$BODY$
BEGIN
    NEW.ngaydathang = '2018-07-10';
    RETURN NEW;
END;
$BODY$
  LANGUAGE plpgsql
```

```
CREATE TRIGGER trigger_upd_timestamp_orderdetail
  BEFORE UPDATE
  ON orderdetail
  FOR EACH ROW
  EXECUTE PROCEDURE update_timestamp();
```

- THANKS FOR ATTENDING!