

Function predefined  
variables

- The PL/pgSQL functions have several special variables that are automatically created in the top-level block.
- For example, if the function returns a trigger, then several variables, such as `NEW`, `OLD`, and `TG_OP`, are created.

- In addition to the trigger special values, there is a Boolean variable called `FOUND`.
- This is often used in combination with DML and `PERFORM` statements, in order to conduct sanity checks.
- The value of the `FOUND` variable is affected by the `SELECT`, `INSERT`, `UPDATE`, `DELETE`, and `PERFORM` statements.
- These statements set `FOUND` to `true` if at least one row is selected, inserted, updated, or deleted.
- The `PERFORM` statement is similar to the `SELECT` statement, but it discards the result of the query.
- Finally, the `EXECUTE` statement does not change the value of the `FOUND` variable.

- The following examples show how the FOUND variable is affected by the INSERT and PERFORM statements:

```
car_portal=# DO $$
car_portal$# BEGIN
car_portal$#   CREATE TABLE t1(f1 int);
car_portal$#
car_portal$#   INSERT INTO t1 VALUES (1);
car_portal$#   RAISE NOTICE '%', FOUND;
car_portal$#
car_portal$#   PERFORM* FROM t1 WHERE f1 = 0;
car_portal$#   RAISE NOTICE '%', FOUND;
car_portal$#
car_portal$#   DROP TABLE t1;
car_portal$# END;
car_portal$# $$LANGUAGE plpgsql;
NOTICE:  t
NOTICE:  f
DO
car_portal=#
```

- In addition to the preceding query, you can get the last **object identifier (OID)** for an inserted row, as well as the affected number of rows, by using the `INSERT`, `UPDATE`, and `DELETE` statements, via the following commands:

```
GET DIAGNOSTICS variable = item;
```

- Assuming that there is a variable called `i`, of the `integer` type, you can get the affected number of rows, as follows:

```
GET DIAGNOSTICS i = ROW_COUNT;
```

- Assuming that there is a variable called `j`, of the `integer` type, you can get the OID of the last row inserted by the most recent `INSERT` command, as follows:

```
GET DIAGNOSTICS j = RESULT_OID;
```