

Практика 4

Функции и триггеры

Цель практики:

1. Научиться писать функции на языке PL/pgSQL
2. Научиться писать триггеры

Задание:

1. Написать функцию, которая выводит первое слово из строки
2. Написать функцию для вывода количества строк в заранее заданной таблице, где содержится введённое слово
3. Написать функцию для проставления статуса "ready" для всех задач в работе "in_work" со временем начала больше 4 часов назад
4. Написать функцию и добавить триггер в таблицу, в результате чего вместо удаления строки будет проставляться флаг, что строка удалена
Усложнение (необязательно): для записей старше недели - удалять, а меньше - ставить флаг
5. Написать функцию и добавить триггер для хранения текущего количества строк в таблице
6. *(Дополнительное необязательное задание)* Написать функцию и триггер для логирования изменений в таблице

Подсказки к заданию 1

Для выполнения задания нужно использовать строковые функции

- <https://postgrespro.ru/docs/postgrespro/12/functions-string>
- <https://sqlbackupandftp.com/blog/postgresql-string-functions-cheat-sheet>

Пример входных и выходных данных:

- only_word → only_word
- first_word second_word → first_word
- NULL → NULL (не слово NULL, а значение)

Пример вызова функции:

```
SELECT first_word('first_word second_word');
```

Подсказки и подготовка к заданию 2

Предварительно нужно создать таблицу, в которой будет текстовое поле, по которому будет осуществляться поиск. Пример таблицы, поиск будет осуществляться по полю text:

```
CREATE TABLE test (id serial, text text);
```

Нужно написать функцию, которая будет искать по полю text в таблице test (только в ней).

Пример вызова:

```
SELECT search_word('text');
```

Для сравнения строк нужно использовать LIKE

Подсказки и подготовка к заданию 3

Для заданий 3-5 можно использовать таблицу:

```
CREATE TABLE test_table(id serial, last_update timestamp, task_status text, is_deleted bool);
```

Для выполнения задания нужно использовать функции работы с датой-временем:

<https://postgrespro.ru/docs/postgresql/12/functions-datetime>

Пример: `select now() + interval '1 week';`

Функция текущего времени: `NOW()`

Подсказки и подготовка к заданию 5

Предварительно нужно создать таблицу статистики, где будет храниться текущее количество строк для каждой таблицы, для которой происходит отслеживание

Вид таблицы статистики: название_таблицы, количество строк

Например:

```
CREATE TABLE table_count(table_name text, count bigint);
```

Перед добавлением триггера (но после создания таблицы статистики) в таблицу статистики предварительно нужно добавить строчку с текущим количеством строк для отслеживаемой таблицы. Это делается для того, чтобы не было необходимости создавать начальную строку в таблице статистики в функции, если ещё нет записей в отслеживаемой таблице.

После написания функции проверить работу при добавлении, изменении и удалении данных.

Добавить триггер на две таблицы и проверить, что в таблице статистики две актуальные записи.

Подсказки и подготовка к заданию 6

Предварительно нужно создать таблицу истории, куда будут писаться все данные.

Что должно храниться в таблице истории: время изменения, с какой таблицей произошли изменения, тип операции, данные строки (новое значение для изменения и добавления, старые — для удаляемых данных)

Для записи добавленных/удалённых данных используется тип JSON.

Возможный вид таблицы:

```
CREATE TABLE table_history(id serial, mod_time timestamp, table_name text, op_type text, data json);
```

Как записать строчку в поле json?

- `row_to_json(OLD)` - для удаляемых строк
- `row_to_json(NEW)` - для новых и обновляемых строк

Приведение типов:

- `TG_TABLE_NAME::text` - название таблицы привести к типу текст