

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«Санкт-Петербургский национальный исследовательский университет  
информационных технологий, механики и оптики»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И  
КОМПЬЮТЕРНОЙ ТЕХНИКИ

**ЛАБОРАТОРНАЯ РАБОТА №1**  
**по дисциплине**  
**«Распределённые системы хранения данных»**

Вариант №367086

***Выполнил:***  
Студент группы Р3334  
Баянов Равиль  
Динарович  
***Преподаватель:***  
Николаев Владимир  
Вячеславович

# Оглавление

<b>Задание</b> .....	3
Код .....	4
Сложности .....	6
<b>Вывод</b> .....	7

# Задание

Используя сведения из системных каталогов, сформировать запрос, отключающий все ограничения целостности типа NOT NULL в пределах заданной схемы. Запрос должен выводить информацию о количестве отключённых ограничений целостности:

Схема: s100000

Ограничений целостности типа NOT NULL отключено: 8

Программу оформить в виде анонимного блока.

# Код

```
--Проверка пришло ли имя схемы
if [ -z "$1" ]; then
    echo "Ошибка: Укажите имя схемы при запуске скрипта."
    exit 1
fi

SCHEMA_NAME="$1"
BD_NAME="$2"

psql -d $BD_NAME -c "DO \$\$
DECLARE
    i RECORD;
    c_count INTEGER := 0;
BEGIN
    -- Удаление обычных ограничений NOT NULL
    FOR i IN
        EXECUTE format(
            'SELECT s.nspname AS schema_name, c.relname AS table_name, a.attname AS
column_name
            FROM pg_catalog.pg_attribute AS a
            JOIN pg_catalog.pg_class AS c ON a.attrelid = c.oid
            JOIN pg_catalog.pg_namespace AS s ON c.relnamespace = s.oid
            LEFT JOIN pg_catalog.pg_index AS ix ON ix.indrelid = c.oid AND a.attnum =
ANY(ix.indkey)
            WHERE a.attnotnull = true AND s.nspname = %L
            AND a.attnum > 0 - Исключение из списка системных столбцов при удалении
ограничений
            AND ix.indexrelid IS NULL - Исключение столбцов, использующихся в индексах
            AND c.relkind != ''S'',
            '$SCHEMA_NAME')
        LOOP
            EXECUTE format('ALTER TABLE %I.%I ALTER COLUMN %I DROP NOT NULL;', i.schema_name,
i.table_name, i.column_name);
            c_count := c_count + 1;
        END LOOP;

    --Удаление ограничений NOT NULL через ADD CONSTRAINTS

    FOR i IN
        EXECUTE format(
            'SELECT s.nspname AS schema_name, c.relname AS table_name, constr.conname AS
constraint_name
            FROM pg_catalog.pg_constraint AS constr
            JOIN pg_catalog.pg_class AS c ON constr.conrelid = c.oid
            JOIN pg_catalog.pg_namespace AS s ON s.oid = c.relnamespace
            WHERE constr.contype = ''c'' -Нахождение ограничений типа CHECK
            AND s.nspname = %L
            AND pg_get_constraintdef(constr.oid) LIKE ''%%CHECK (%%IS NOT NULL%%)%%''
            AND pg_get_constraintdef(constr.oid) NOT LIKE ''%%AND%%'';', -- Условие для
выборки ограничений только NOT NULL
            '$SCHEMA_NAME')
        LOOP
            EXECUTE format('ALTER TABLE %I.%I DROP CONSTRAINT %I;', i.schema_name,
i.table_name, i.constraint_name);
            c_count := c_count + 1;
        END LOOP;

    -- Вывод ответа
```

```
RAISE NOTICE E'\rCхема: %', '$SCHEMA_NAME';  
RAISE NOTICE E'\rОграничений целостности типа NOT NULL отключено: %', c_count;  
END \$$ LANGUAGE plpgsql;" | grep -v "^DO$" - Удаление ненужного вывода в консоль
```

# Сложности

В ходе данной лабораторной работы мне дали интересное задание усовершенствовать мой скрипт для удаления ограничений NOT NULL, а именно следовало удалять не только обычные ограничения, но и ограничения, добавленные на колонки с помощью конструкции ADD CONSTRAINTS. У меня получилось это реализовать с помощью второго цикла по таблицам в схеме, таблицы pg\_constraint и системной функции pg\_get\_constraintdef(). Также не сразу было понятно, как можно реализовать правильную и логичную передачу аргумента bash утилиты в sql-скрипт. Но и это у меня получилось. Для выполнения данной работы пришлось тщательно изучить функции RAISE и EXECUTE format, а также узнать много нового о системных каталогах PostgreSQL.

# Вывод

В данной лабораторной работе я научился работать с системными каталогами в postgresql. Научился писать анонимные блоки для отключения ограничений целостности любых таблиц в определённых схемах.