# Задача №2

### Контекст

В PostgreSQL существует объекты называемые sequence. CREATE SEQUENCE создаёт генератор последовательности. Эта операция включает создание и инициализацию специальной таблицы *имя*, содержащей одну строку.

CREATE [ TEMPORARY | TEMP ] SEQUENCE [ IF NOT EXISTS ] имя [ INCREMENT [ BY ] шаг ]  
 [ MINVALUE мин\_значение | NO MINVALUE ] [ MAXVALUE макс\_значение | NO MAXVALUE ]  
 [ START [ WITH ] начало ] [ CACHE кеш ] [ [ NO ] CYCLE ]  
 [ OWNED BY { имя\_таблицы.имя\_столбца | NONE } ]

**Пример:** Создание возрастающей последовательности с именем serial, с начальным значением 101:

CREATE SEQUENCE serial START 101;

Получение следующего номера этой последовательности:

SELECT nextval('serial');  
  
 nextval  
---------  
 101

Получение следующего номера этой последовательности:

SELECT nextval('serial');  
  
 nextval  
---------  
 102

После создания последовательности работать с ней можно, вызывая функции nextval, currval и setval.

Использование сиквенсов для генерации таблички довольно непростая задача. Давайте в рамках этого задания напишем функцию, которая сгенерирует табличку из последовательных чисел.

### Постановка

Написать функцию serial\_generator(start\_val\_inc INTEGER, last\_val\_ex INTEGER), которая возвращает таблицу из последовательных чисел с шагом +1. Скрипт создания должен быть нечувствителен к уже объявленным функциям.

**Пример**

SELECT \* FROM serial\_generator(10, 12);  
  
 serial\_generator  
------------------  
 10  
 11

**Дополнительные условия** \* функция должна быть написана на plpgsql; \* функция генерирует значения начиная с start\_val\_inc до тех пор, пока оно строго меньше last\_val\_ex; \* **запрещено** использовать функцию generate\_series; \* самостоятельно изучите документацию как *генерировать* табличку из функции.

### Ожидаемый формат ответа

Выводить ничего не надо. Скрипт с решением должен содержать **только** объявление функции.