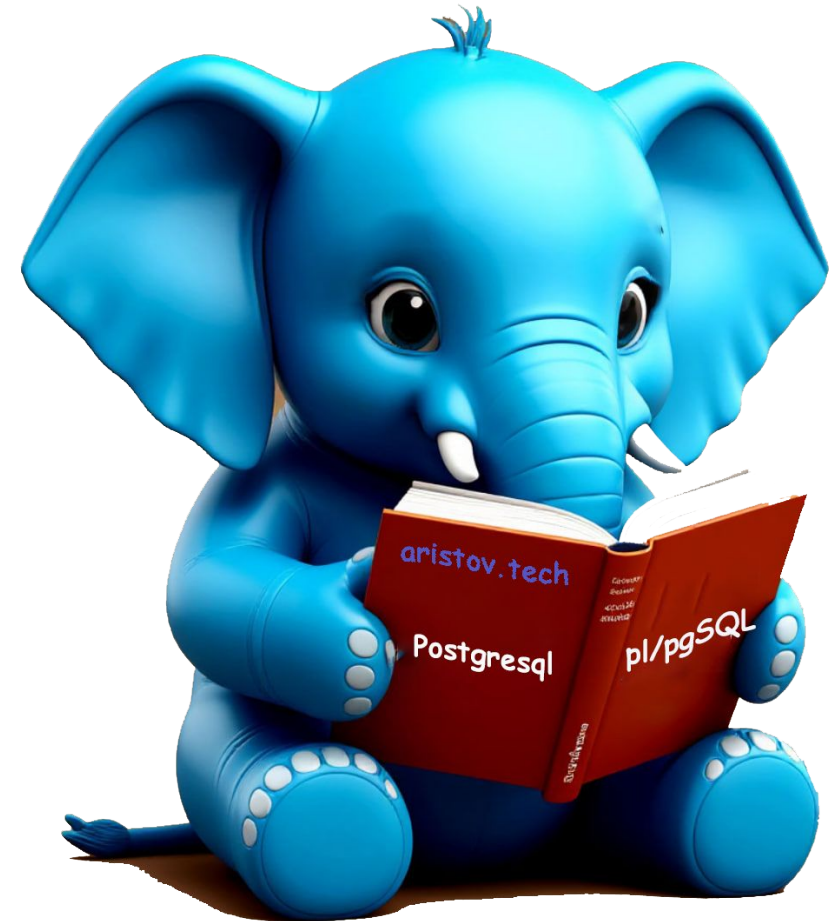


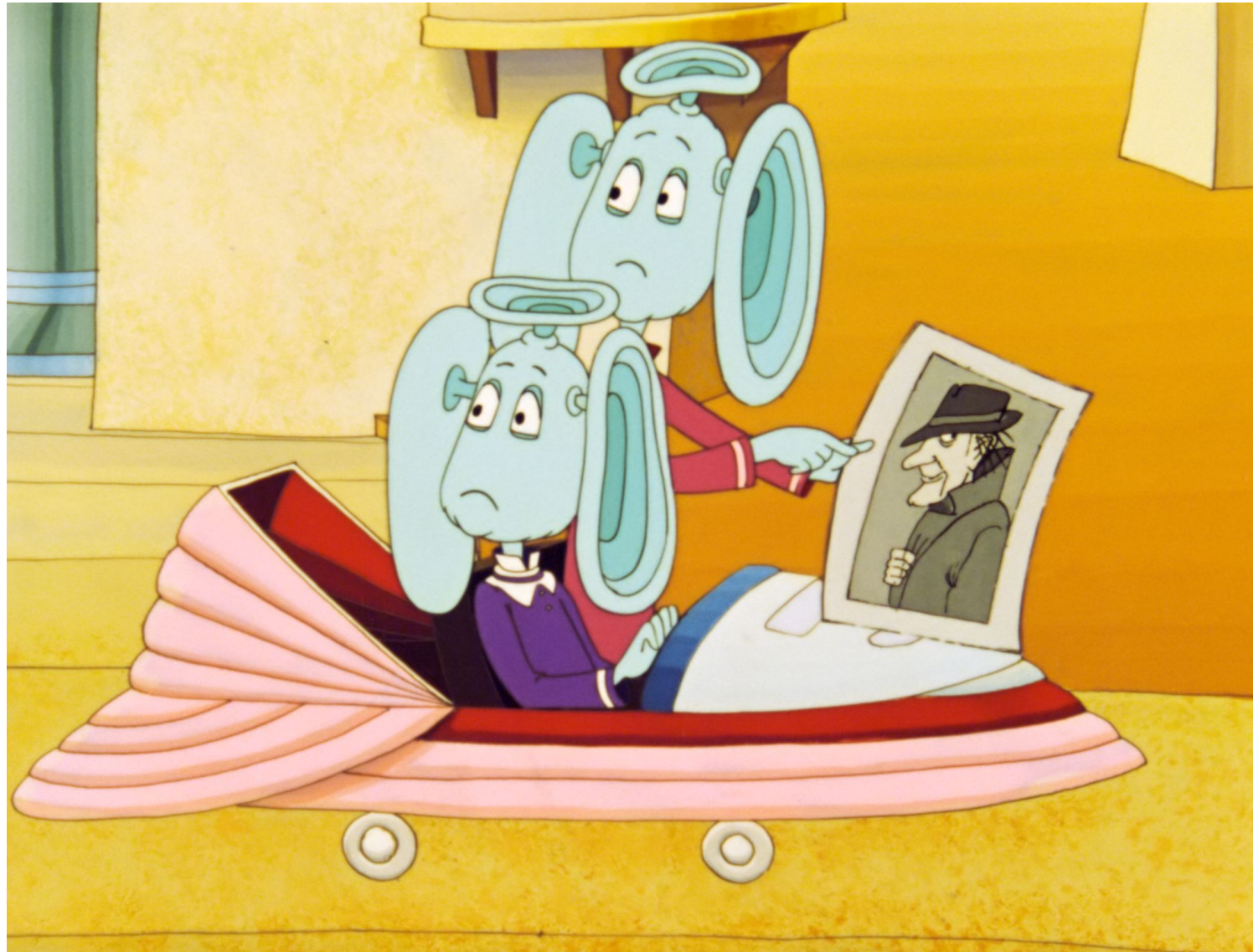
**Аристов Евгений**

# **PL/pgSQL в PostgreSQL**

**за 31 занятие**

**Команды возврата из функции**





**Аристов  
Евгений  
Николаевич**



<https://aristov.tech>

Founder & CEO [aristov.tech](https://aristov.tech)

25 лет занимаюсь разработкой БД и ПО

Архитектор высоконагруженных баз данных и инфраструктуры

Спроектировал и разработал более ста проектов для финансового сектора, сетевых магазинов, фитнес-центров, отелей.

Сейчас решаю актуальные для бизнеса задачи: аудит и оптимизация БД и инфраструктуры, миграция на PostgreSQL, обучение сотрудников.

Автор более 10 практических курсов по PostgreSQL, MySQL, Mongo и др..

Автор книг по PostgreSQL. Новинка [PostgreSQL 16: лучшие практики оптимизации](#)

<https://aristov.tech>

# Правила вебинара

Задаем вопрос в чат

Вопросы вижу, отвечу в момент логической паузы

Если есть вопрос голосом - поставьте знак ? в чат

Если остались вопросы, можно их задать на следующем занятии или в комментариях к записи

# Маршрут вебинара

Возврат значений из функции

Варианты и их отличия

Особенности применения

# Входные выходные параметры

**FUNCTION** (параметр 1, параметр 2 ...)

**RETURN** тип\_результата

## Варианты возвращаемых данных:

- ❖ **тип\_результата** - может быть **базовый**, **составной**, либо ссылка на **тип столбца таблицы (13 лекция)**. Если функция не должна возвращать значение, в качестве типа результата указывается **void** (**ничего не вернуть**)
- ❖ Указание SETOF показывает, что функция возвращает множество, а не единственный элемент
- ❖ Можно вернуть таблицу
- ❖ *Последние 2 варианта более подробно рассмотрим на 10 лекции*



## Возвращение значения

❖ **RETURN** выражение;

RETURN с последующим выражением прекращает выполнение функции и возвращает значение выражения в вызывающую программу. Эта форма используется для функций PL/pgSQL, которые не возвращают набор строк.

В функции, возвращающей скалярный тип, результирующее выражение автоматически приводится к типу возвращаемого значения. Однако, чтобы вернуть составной тип (строку), возвращаемое выражение должно в точности содержать требуемый набор столбцов. При этом может потребоваться явное приведение типов.

Для функции с выходными параметрами просто используйте RETURN без выражения. Будут возвращены текущие значения выходных параметров.

Для функции, возвращающей **void**, RETURN можно использовать в любом месте, но **без** выражения после RETURN.

## Возвращение значения

Возвращаемое значение функции не может остаться не определённым. Если достигнут конец блока верхнего уровня, а оператор RETURN так и не встретился, происходит ошибка времени исполнения.

Это не касается функций с выходными параметрами и функций, возвращающих void. Для них оператор RETURN выполняется автоматически по окончании блока верхнего уровня.

Несколько примеров:

-- Функции, возвращающие скалярный тип данных

**RETURN 1 + 2;**

**RETURN scalar\_var;**

-- Функции, возвращающие составной тип данных

**RETURN composite\_type;**

**RETURN (1, 2, 'three'::text);** -- требуется приведение типов



# Возвращение значения

## ❖ PERFORM

Если результат выполнения функции не важен, то можно использовать PERFORM

**!!!Но только в рамках другой процедуры или функции!!!**

# Основные операторы

Присвоение значения переменной PL/pgSQL записывается в виде:

***переменная { := | = } выражение;***

Примеры:

```
tax := total * 0.04;  
user_id = 40;
```

<https://www.postgresql.org/docs/current/plpgsql-statements.html>

# Практика

# Итоги

# Итоги

Остались ли вопросы?

Увидимся на следующем занятии

# Спасибо за внимание!

Когда дальше и куда?

Аристов Евгений