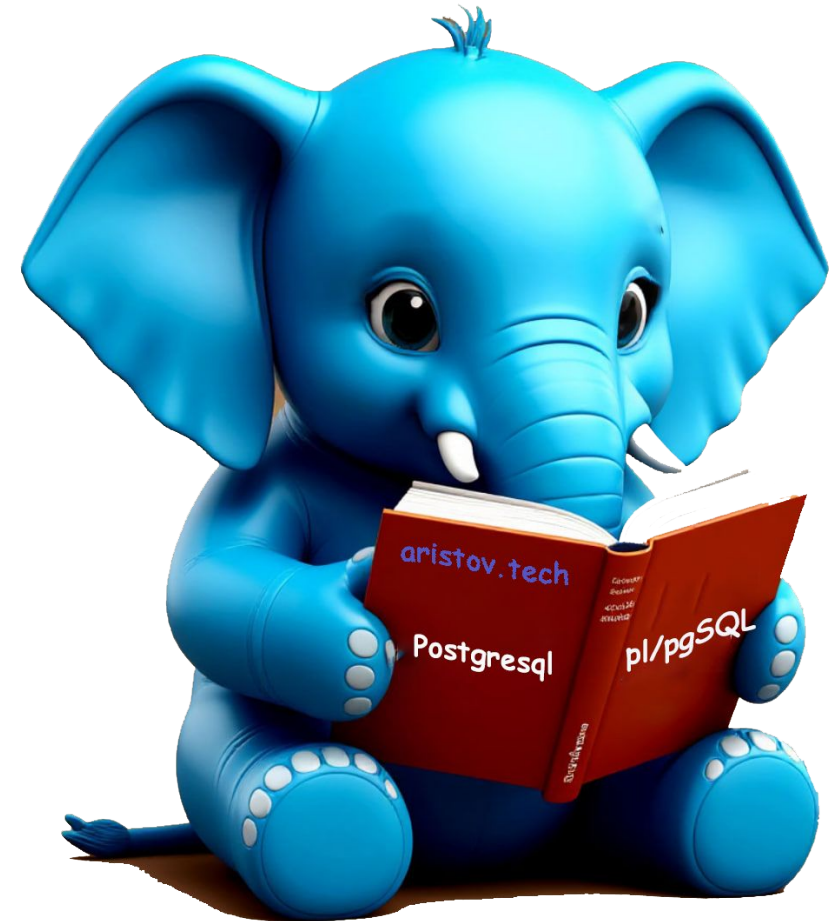


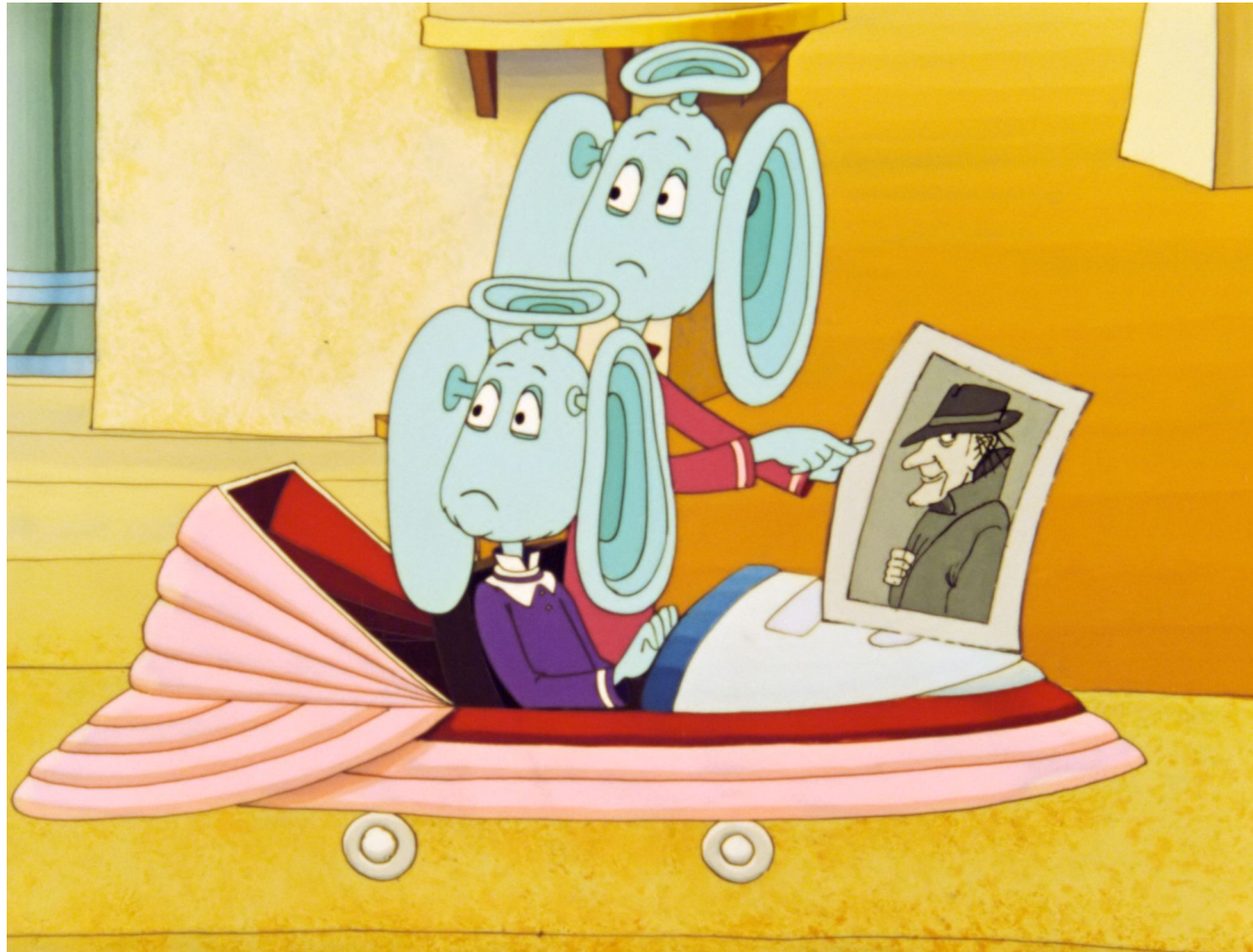
Аристов Евгений

PL/pgSQL в PostgreSQL

за 31 занятие

Полиморфизм функций





**Аристов
Евгений
Николаевич**



<https://aristov.tech>

Founder & CEO 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

Полиморфизм функций

Ограничения

Лучшие практики

Полиморфизм функций

Полиморфизм функций

Полиморфизм функций в PostgreSQL позволяет создавать функции, которые могут работать с разными типами данных. Это более гибкий подход по сравнению с перегрузкой, поскольку одна функция может обрабатывать множественные типы данных.

Полиморфные типы данных

- **anyelement** - любой тип данных
- **anyarray** - любой массив
- **anynonarray** - любой не-массивный тип
- **anyenum** - любой enum тип
- **anyrange** - любой диапазонный тип

Полиморфизм функций

Например:

```
CREATE FUNCTION maximum(a anyelement, b anyelement) RETURNS anyelement AS $$  
    SELECT CASE WHEN a > b THEN a ELSE b END;  
$$ LANGUAGE SQL;
```

```
SELECT maximum(1,2);
```

Главное, чтобы над таким типов данных при вызове работал оператор (<>= и т.д.)

Полиморфизм функций. Ограничения

Ограничение: нельзя использовать `anyelement` в RETURNS TABLE:

sql

НЕ РАБОТАЕТ:

```
CREATE OR REPLACE FUNCTION get_polymorphic_table()  
RETURNS TABLE(id anyelement, name text) AS $$ ... $$;
```

РЕШЕНИЕ: использовать OUT параметры - но помним разницу

```
CREATE OR REPLACE FUNCTION get_polymorphic_data(in i anyelement, OUT id anyelement, OUT data  
jsonb)  
RETURNS SETOF record AS $$ – возврат анонимного множества
```

обязательно должен быть задан 1 IN элемент параметр!

Полиморфизм функций. Итоги

Мощный инструмент для:

- ❖ Создания универсальных утилит
- ❖ Уменьшения дублирования кода
- ❖ Повышения гибкости API

Основные преимущества:

- ❖ Одна функция для многих типов
- ❖ Автоматическое определение типов
- ❖ Сокращение количества функций

Ограничения:

- ❖ Сложность отладки
- ❖ Ограниченная поддержка в некоторых контекстах
- ❖ Возможные проблемы с производительностью

Практика

Итоги

Итоги

Остались ли вопросы?

Увидимся на следующем занятии

Спасибо за внимание!

Когда дальше и куда?

Аристов Евгений