



5 Advanced PostgreSQL Features

A reference guide to 5 advanced PostgreSQL-specific features.

FILTER Clause

Applying a WHERE clause to an aggregate function.

Before

```
SELECT  
COUNT(  
  CASE  
    WHEN status = 'cancelled' THEN 1 ELSE NULL  
  END  
) AS cancelled_orders  
FROM cust_order;
```

After

```
SELECT  
COUNT(*) FILTER (  
  WHERE status = 'cancelled'  
) AS cancelled_orders  
FROM cust_order;
```

DISTINCT ON

Return a single row per group.

```
SELECT DISTINCT ON (customer_id)  
  customer_id,  
  order_date,  
  total  
FROM cust_order  
ORDER BY customer_id, order_date DESC;
```

RETURNING Clause in DML

Return a value from the new or modified row.

```
INSERT INTO customers (name, email)  
VALUES ('Ben', 'ben@example.com')  
RETURNING id;
```

This returns the ID of the inserted row.

Richer Data Types

More data types for specific uses

UUID: unique identifiers
INET: good for IP addresses
MONEY: for financial values

Transactional DDL

Use DDL within transactions, to allow for COMMIT and ROLLBACK if needed.

```
BEGIN;  
CREATE TABLE test_table (id INT);  
ROLLBACK;
```