

[CASA](#)[PROCEDIMIENTOS ALMACENADOS](#)[ÍNDICES](#)[PUNTOS DE VISTA](#)[GATILLOS](#)[INTERFACES](#) ▾[FUNCIONES](#) ▾

Resumen : en este tutorial, aprenderá cómo usar la `ARRAY_AGG()` función agregada de PostgreSQL para devolver una matriz de un conjunto de valores de entrada.

Introducción a la `ARRAY_AGG()` función PostgreSQL .

La `ARRAY_AGG()` función PostgreSQL es una [función agregada](#) que acepta un conjunto de valores y devuelve una [matriz](#) en la que cada valor en el conjunto de entrada se asigna a un elemento de la matriz.

A continuación se muestra la sintaxis de la `ARRAY_AGG()` función:

```
1 ARRAY_AGG(expression [ORDER BY [sort_expression {ASC | DESC}], [...])
```

El `ARRAY_AGG()` acepta una expresión que devuelve un valor de cualquier tipo que sea válido para un elemento de matriz.

La `ORDER BY` cláusula es una cláusula opcional. Especifica el orden de las filas que se procesan en la agregación, lo que determina el orden de los elementos en la matriz de resultados.

Similar a otras funciones de agregación tales como `AVG()`, `COUNT()`, `MAX()`, `MIN()`, y `SUM()`, el `ARRAY_AGG()` se utiliza a menudo con la `GROUP BY` cláusula.

`ARRAY_AGG()` Ejemplos de funciones de PostgreSQL

Utilizaremos las `film`, `film_actor` y `actor` las tablas de la [base de datos ejemplo](#) para la demostración.

Usando la `ARRAY_AGG()` función PostgreSQL sin el `ORDER BY` ejemplo de la cláusula

The following example uses the `ARRAY_AGG()` function to return the list of film title and a list of actors for each film:

```
1 SELECT
2     title,
3     ARRAY_AGG (first_name || ' ' || last_name) actors
4 FROM
5     film
```

CASA

PROCEDIMIENTOS ALMACENADOS

ÍNDICES

PUNTOS DE VISTA

GATILLOS

INTERFACES ▾

FUNCIONES ▾

Ace Goldfinger	{ "Minnie Zellweger", "Chris Depp", "Bob Fawcett", "Sean Guinness" }
Adaptation Holes	{ "Cameron Streep", "Bob Fawcett", "Nick Wahlberg", "Ray Johansson", "Julianne Dench" }
Affair Prejudice	{ "Jodie Degeneres", "Kenneth Pesci", "Fay Winslet", "Oprah Kilmer", "Scarlett Damon" }
African Egg	{ "Dustin Tautou", "Matthew Leigh", "Gary Phoenix", "Matthew Carrey", "Thora Temple" }
Agent Truman	{ "Warren Nolte", "Sandra Kilmer", "Jayne Neeson", "Morgan Williams", "Kirsten Paltrow", "Kenneth Hoffman", "Reese West" }
Airplane Sierra	{ "Mena Hopper", "Jim Mostel", "Michael Bolger", "Oprah Kilmer", "Richard Penn" }
Airport Pollock	{ "Lucille Dee", "Susan Davis", "Fay Kilmer", "Gene Willis" }

As you can see, the actors in each film are arbitrarily ordered. To sort the actors by last name or first name, you can use the `ORDER BY` clause in the `ARRAY_AGG()` function.

Using PostgreSQL `ARRAY_AGG()` function with the `ORDER BY` clause example

This example uses the `ARRAY_AGG()` function to return a list of films and a list of actors for each film sorted by the actor's first name:

```

1 SELECT
2     title,
3     ARRAY_AGG (
4         first_name || ' ' || last_name
5         ORDER BY
6             first_name
7     ) actors
8 FROM
9     film
10 INNER JOIN film_actor USING (film_id)
11 INNER JOIN actor USING (actor_id)
12 GROUP BY
13     title
14 ORDER BY
15     title;
```

The following shows the partial output:

title	actors
Academy Dinosaur	{ "Christian Gable", "Johnny Cage", "Lucille Tracy", "Mary Keitel", "Mena Temple", "Oprah Kilmer", "Penelope Guinness", "Rock Dukakis", "Sandra Peck", "Warren Nolte" }
Ace Goldfinger	{ "Bob Fawcett", "Chris Depp", "Minnie Zellweger", "Sean Guinness" }
Adaptation Holes	{ "Bob Fawcett", "Cameron Streep", "Julianne Dench", "Nick Wahlberg", "Ray Johansson" }
Affair Prejudice	{ "Fay Winslet", "Jodie Degeneres", "Kenneth Pesci", "Oprah Kilmer", "Scarlett Damon" }
African Egg	{ "Dustin Tautou", "Gary Phoenix", "Matthew Leigh", "Matthew Carrey", "Thora Temple" }
Agent Truman	{ "Jayne Neeson", "Kenneth Hoffman", "Kirsten Paltrow", "Morgan Williams", "Reese West", "Sandra Kilmer", "Warren Nolte" }
Airplane Sierra	{ "Jim Mostel", "Mena Hopper", "Michael Bolger", "Oprah Kilmer", "Richard Penn" }
Airport Pollock	{ "Fay Kilmer", "Gene Willis", "Lucille Dee", "Susan Davis" }

[CASA](#)[PROCEDIMIENTOS ALMACENADOS](#)[ÍNDICES](#)[PUNTOS DE VISTA](#)[GATILLOS](#)[INTERFACES](#) ▾[FUNCIONES](#) ▾

```
7      last_name DESC
8    ) actors
9 FROM
10    film
11 INNER JOIN film_actor USING (film_id)
12 INNER JOIN actor USING (actor_id)
13 GROUP BY
14    title
15 ORDER BY
16    title;
```

This picture shows the partial output of the query:

title	actors
Academy Dinosaur	{'Christian Gable', 'Johnny Cage', 'Lucille Tracy', 'Mary Keitel', 'Mena Temple', 'Oprah Kilmer', 'Penelope Guinness', 'Rock Dukakis', 'Sandra Peck', 'Warren Nolte'}
Ace Goldfinger	{'Bob Fawcett', 'Chris Depp', 'Minnie Zellweger', 'Sean Guinness'}
Adaptation Holes	{'Bob Fawcett', 'Cameron Streep', 'Julianne Dench', 'Nick Wahlberg', 'Ray Johansson'}
Affair Prejudice	{'Fay Winslet', 'Jodie Degeneres', 'Kenneth Pesci', 'Oprah Kilmer', 'Scarlett Damon'}
African Egg	{'Dustin Tautou', 'Gary Phoenix', 'Matthew Leigh', 'Matthew Carrey', 'Thora Temple'}
Agent Truman	{'Jayne Neeson', 'Kenneth Hoffman', 'Kirsten Paltrow', 'Morgan Williams', 'Reese West', 'Sandra Kilmer', 'Warren Nolte'}
Airplane Sierra	{'Jim Mostel', 'Mena Hopper', 'Michael Bolger', 'Oprah Kilmer', 'Richard Penn'}
Airport Pollock	{'Fay Kilmer', 'Gene Willis', 'Lucille Dee', 'Susan Davis'}

En este tutorial, aprendió a usar la `ARRAY_AGG()` función PostgreSQL para devolver una matriz de un conjunto de valores de entrada.

¿Fue útil este tutorial?



Si



no



Tutorial Anterior:

[Función PostgreSQL SUM](#)

Siguiente tutorial:

[Función PostgreSQL STRING_AGG](#)



[CASA](#)[PROCEDIMIENTOS ALMACENADOS](#)[ÍNDICES](#)[PUNTOS DE VISTA](#)[GATILLOS](#)[INTERFACES ▾](#)[FUNCIONES ▾](#)

[Instalar PostgreSQL](#)[Conectarse a la base de datos](#)[Descargar PostgreSQL Sample Database](#)[Cargar base de datos de muestra](#)[Explorar objetos de servidor y base de datos](#)[FUNDAMENTOS DE POSTGRESQL](#)[SELECCIONAR](#)[ORDEN POR](#)[SELECCIONAR DISTINTO](#)[DÓNDE](#)[LÍMITE](#)[HA PODIDO RECUPERAR](#)[EN](#)[ENTRE](#)[ME GUSTA](#)[ES NULO](#)[ALIAS](#)[Se une](#)[UNIR INTERNAMENTE](#)[UNIRSE IZQUIERDO](#)[Auto-unirse](#)[ÚNETE COMPLETO](#)[Cruzar](#)

CASA	PROCEDIMIENTOS ALMACENADOS	ÍNDICES	PUNTOS DE VISTA	GATILLOS
INTERFACES ▾	FUNCIONES ▾			
				EXCEPCIONES
				Conjuntos de agrupación
				CUBO
				ENROLLAR
				Subconsulta
				ALGUNA
				TODOS
				EXISTE
				INSERTAR
				ACTUALIZAR
				ACTUALIZACIÓN Únete
				Upsert
				BORRAR
				GESTIÓN DE LA ESTRUCTURA DE LA MESA
				Tipos de datos PostgreSQL
				CREAR MESA
				SELECCIONAR EN
				CREAR TABLA COMO
				Auto-Increment Column with SERIAL
				Identity Column
				ALTER TABLE
				DROP TABLE
				TRUNCATE TABLE
				CHECK Constraint

[CASA](#)[PROCEDIMIENTOS ALMACENADOS](#)[ÍNDICES](#)[PUNTOS DE VISTA](#)[GATILLOS](#)[INTERFACES ▾](#)[FUNCIONES ▾](#)

[Managing PostgreSQL Views](#)[Creating Updatable Views](#)[PostgreSQL Materialized Views](#)[The WITH CHECK OPTION Views](#)[PostgreSQL Recursive View](#)[POSTGRESQL TRIGGERS](#)[Introduction to Trigger](#)[Creating A Trigger](#)[Managing PostgreSQL Triggers](#)

ABOUT POSTGRESQL TUTORIAL

PostgreSQLTutorial.com is a website dedicated to developers and database administrators who are working on PostgreSQL database management system.

We constantly publish useful PostgreSQL tutorials to keep you up-to-date with the latest PostgreSQL features and technologies. All PostgreSQL tutorials are simple, easy-to-follow and practical.

RECENT POSTGRESQL TUTORIALS

[PostgreSQL STRING_AGG Function](#)[PostgreSQL ARRAY_AGG Function](#)[PostgreSQL CREAT PROCEDIMIENTO](#)[Transacción PostgreSQL](#)[PostgreSQL ISNULL](#)[PostgreSQL CTE](#)[PostgreSQL es nulo](#)[Índices de columnas múltiples de PostgreSQL](#)[Tipos de índices de PostgreSQL](#)[PostgreSQL Listar índices](#)

[CASA](#)[PROCEDIMIENTOS ALMACENADOS](#)[ÍNDICES](#)[PUNTOS DE VISTA](#)[GATILLOS](#)[INTERFACES ▾](#)[FUNCIONES ▾](#)

[PostgreSQL JDBC](#)[Recursos PostgreSQL](#)[INFORMACIÓN DEL SITIO](#)[Casa](#)[Sobre nosotros](#)[Contáctenos](#)[Política de privacidad](#)

Copyright © 2019 by [PostgreSQL Tutorial del](#) sitio web. Todos los derechos reservados.