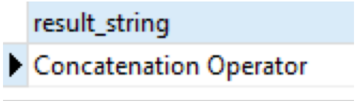
PostgreSQL CONCAT Function

Summary: in this tutorial, we will show you how to use the PostgreSQL CONCAT and CONCAT_WS functions to concatenate two or more strings into one.

Introduction to PostgreSQL CONCAT function

To concatenate two or more strings into one, you use the string concatenation operator || as the following example:

```
SELECT
   'Concatenation' || ' ' || 'Operator' AS result_string;
```



result_string
Concatenation Operator

The following statement concatenates a string with a `NULL` value:

```
SELECT
   'Concat with ' || NULL AS result_string;
```

It returns a `NULL` value.

Since version 9.1, PostgreSQL introduced a built-in string function named `CONCAT` to concatenate two or more strings into one.

The following illustrates the syntax of the `CONCAT` function:

```
CONCAT(str_1, str_2, ...)
```

The `CONCAT` function accepts a list of arguments. The argument needs to be convertible to a string. A string in this context means any of the following data types (https://www.postgresqltutorial.com/postgresql-tutorial/postgresql-data-types/) : char, varchar, or text.

The `CONCAT` function is variadic. It means that the `CONCAT` function accepts an array (https://www.postgresqltutorial.com/postgresql-tutorial/postgresql-array/) as the argument. In this case, you need to mark the array with the `VARIADIC` keyword. The `CONCAT` function treats each array element as an argument.

Unlike the concatenation operator ||, the `CONCAT` function ignores `NULL` arguments. We will see it in detail in the following section.

PostgreSQL CONCAT function examples

The following SELECT statement (https://www.postgresqltutorial.com/postgresql-tutorial/postgresql-select/) uses the `CONCAT` function to concatenate two strings into one:

```
SELECT
	CONCAT ('CONCAT',' ', 'function');
```

The following statement concatenates values in the `first_name` and `last_name` columns of the `customer` table in the sample database (https://www.postgresqltutorial.com/download/dvd-rental-sample-database/) .

```
SELECT
	CONCAT  (first_name, ' ', last_name) AS "Full name"
FROM
	customer;
```

In the following example, we use `CONCAT` function to concatenate a string with a `NULL` value.

```
SELECT
        CONCAT('Concat with ',NULL) AS result_string;
```

As you see, unlike the string concatenation operator ||, the `CONCAT` function ignores the `NULL` arguments.

The following statement concatenates strings with a number returned from the LENGTH function (https://www.postgresqltutorial.com/postgresql-length-function/) .

```
SELECT
        first_name,
        concat (
                'Your first name has ',
                LENGTH (first_name),
                ' characters'
        )
FROM
        customer;
```

PostgreSQL CONCAT_WS function

Besides the `CONCAT` function, PostgreSQL also provides you with the `CONCAT_WS` function that concatenates strings into one separated by a particular separator. By the way, WS stands for **w**ith **s**eparator.

Like the `CONCAT` function, the `CONCAT_WS` function is also variadic and ignored `NULL` values.

The following illustrates the syntax of the `CONCAT_WS` function.

```
CONCAT_WS(separator,str_1,str_2,...);
```

The `separator` is a string that separates all arguments in the result string.

The `str_1`, `str_2`, etc., are strings or any arguments that can be converted into strings.

The `CONCAT_WS` function returns a combined string that is the combination of `str_1`, `str_2`, etc., separated by the `separator`.

PostgreSQL CONCAT_WS function example

The following statement concatenates the last name and first name and separates them by a comma and a space:

```
SELECT
        concat_ws (', ', last_name, first_name) AS full_name
FROM
        customer
```

```
ORDER BY
        last_name;
```

In this tutorial, you have learned how to use the PostgreSQL `CONCAT()` and `CONCAT_WS` function to concatenates two or more strings into one.