CREATE OR REPLACE VIEW to Update Views

by Sophia

WHAT'S COVERED

This tutorial explores using the REPLACE VIEW or ALTER VIEW to make changes to views in two parts:

1. Introduction
2. Example Change To View

1. Introduction

Some databases will not allow you to remove a column or add a column from a view. Rather, you would have to use the CREATE OR REPLACE VIEW command to make changes to a view. It is important to note that if a view already exists, the columns must be named the same, with the same data types and in the same order that it was created. With PostgreSQL, the database will allow you to add additional columns at the end of the list.

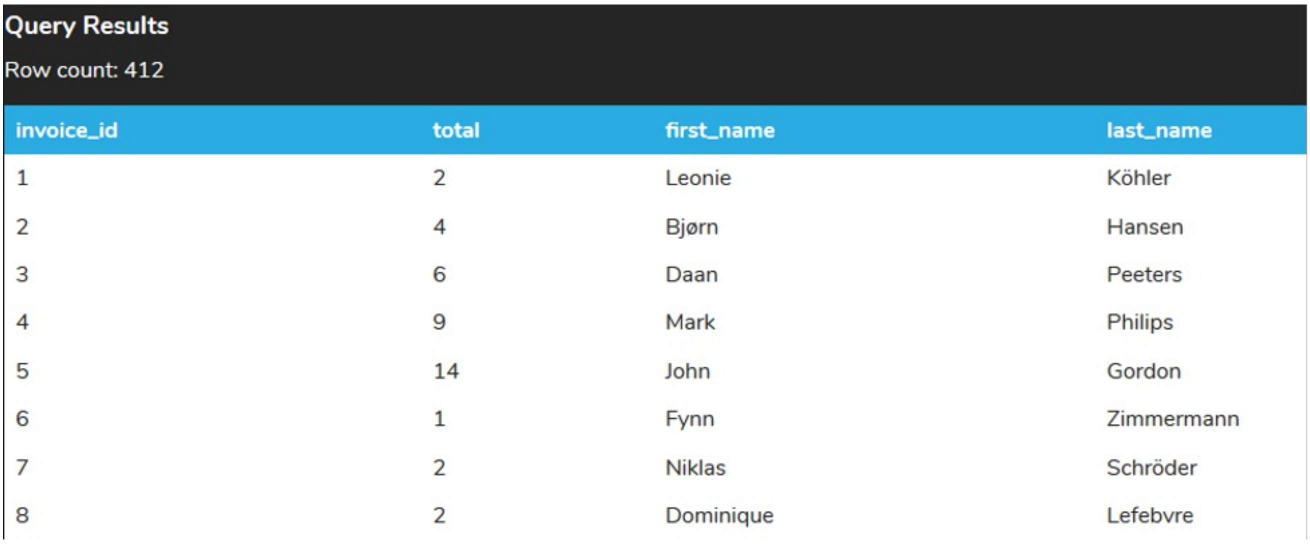
2. Example Change To View

For example, let's create a view that has the invoice_id, total, and first and last name of the customer:

```
CREATE VIEW invoice_information
AS
SELECT invoice_id, total, first_name, last_name
FROM invoice
INNER JOIN customer ON invoice.customer_id = customer.customer_id;
```

To query the view, we can run the following SELECT statement:

```
SELECT *
FROM invoice_information;
```



Query Results

Row count: 412

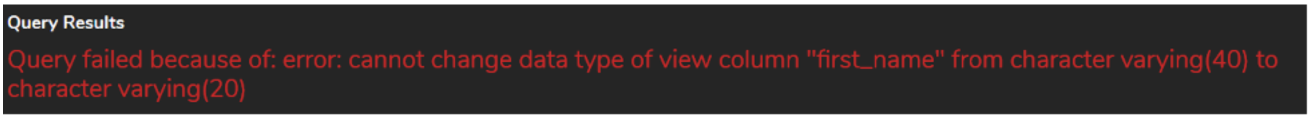
invoice_id	total	first_name	last_name
1	2	Leonie	Köhler
2	4	Bjørn	Hansen
3	6	Daan	Peeters
4	9	Mark	Philips
5	14	John	Gordon
6	1	Fynn	Zimmermann
7	2	Niklas	Schröder
8	2	Dominique	Lefebvre

If we wanted to have the employee's first and last name instead of the customer's first and last name, we could drop the view and then recreate it. However, if there were objects that depend on this view, we would have to drop those as well. Using the CREATE OR REPLACE VIEW allows us to simply recreate it without having to get rid of the dependent objects.

Let us make the change in the view:

```
CREATE OR REPLACE VIEW invoice_information
AS
SELECT invoice_id, total, employee.first_name, employee.last_name
FROM invoice
INNER JOIN customer ON invoice.customer_id = customer.customer_id
INNER JOIN employee ON customer.support_rep_id = employee.employee_id;
```

Even though the names of the columns are the same and are in the same order, we will get this error:



Query Results

Query failed because of: error: cannot change data type of view column "first_name" from character varying(40) to character varying(20)

This is because the first_name in the employee table is 20 characters, so the data size does not match. In this case, we could resize the employee table's first_name column to 40 characters, as well as the last_name:

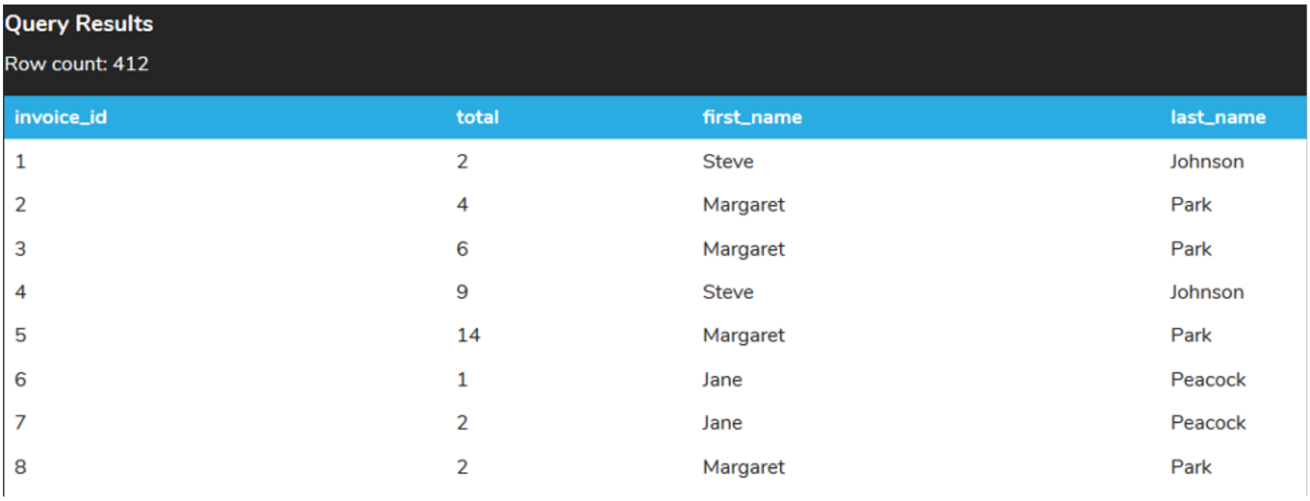
```
ALTER TABLE employee
ALTER COLUMN first_name TYPE VARCHAR (40);
ALTER TABLE employee
ALTER COLUMN last_name TYPE VARCHAR (40);
```

Now if we attempt to make the modification again:

```
CREATE OR REPLACE VIEW invoice_information
AS
SELECT invoice_id, total, employee.first_name, employee.last_name
FROM invoice
INNER JOIN customer ON invoice.customer_id = customer.customer_id
INNER JOIN employee ON customer.support_rep_id = employee.employee_id;
```

We are successful, as the data type, column names, and order all match:

```
SELECT *
FROM invoice_information;
```



Query Results

Row count: 412

invoice_id	total	first_name	last_name
1	2	Steve	Johnson
2	4	Margaret	Park
3	6	Margaret	Park
4	9	Steve	Johnson
5	14	Margaret	Park
6	1	Jane	Peacock
7	2	Jane	Peacock
8	2	Margaret	Park

Since the data has changed, we may want to also change the name of the view. We can use the ALTER VIEW statement to rename it:

```
ALTER VIEW invoice_information RENAME TO invoice_employee_info;
```

Video Transcription

Here, we have a view that's already created that has the invoice ID, total first name and last name between the invoice table and the customer table. It is important to note that once a view has been created, we can only replace it if the column names, as well as the data types and sizes, all match in exact order. So in this case here, if we want to make a modification to this particular view to get the employee information, rather than the information from the customer, we would have to modify this just slightly.

So you'll notice in this case here, we have a create or replace view with the same name for the view. And then in our query, we're going to have the employee first name and the employee last name. We've also joined the employee table. And then we're going to go ahead and try to create this. Once we've actually created this, the information that's going to be displayed is going to be based on the employee first name and last name instead.

TRY IT

Your turn! Open the SQL tool by clicking on the LAUNCH DATABASE button below. Then enter in one of the examples above and see how it works. Next, try your own choices for which columns you want the query to provide.

SUMMARY

The CREATE OR REPLACE VIEW allows you to replace an existing query as long as the column names, data types, data sizes and order of the columns matches.

Source: Authored by Vincent Tran