

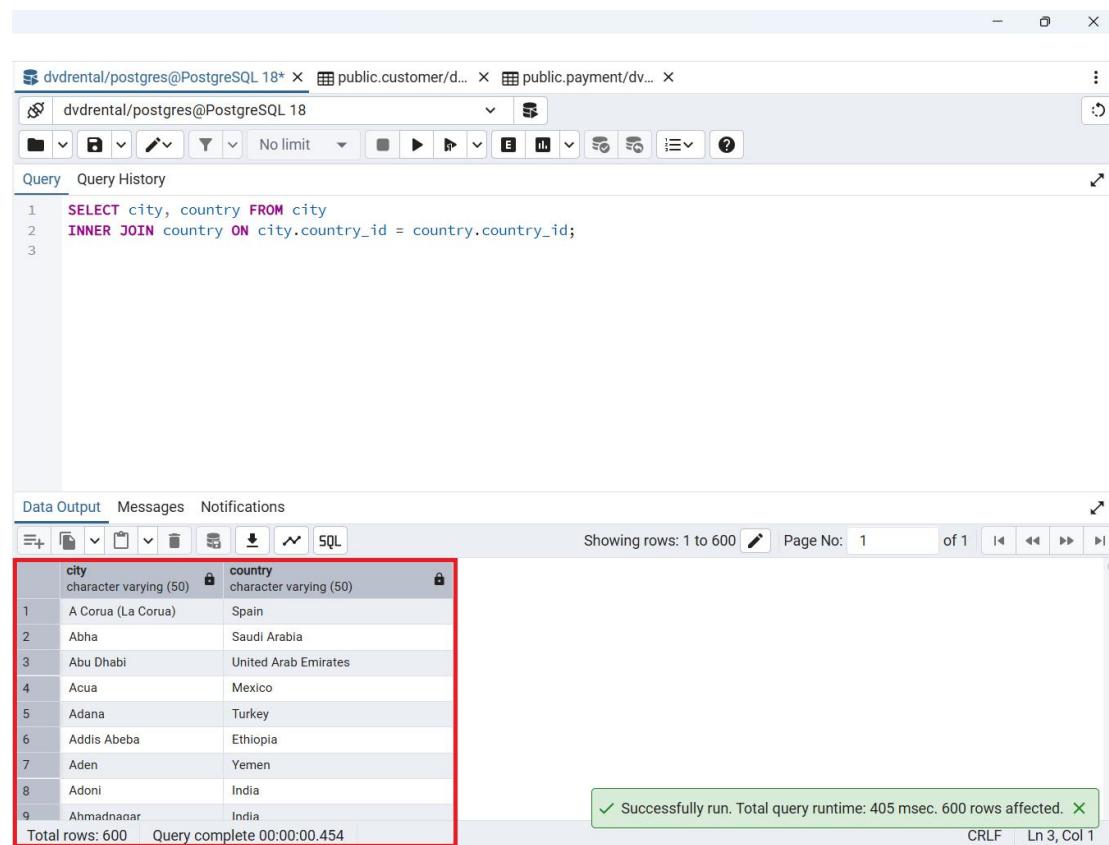
Assignment 9: DVDRental Queries

This assignment focuses on practicing **INNER JOIN operations** using the DVDRental database. The goal is to understand how related tables can be connected through key fields to retrieve combined information.

These exercises strengthen your understanding of **relational data, foreign keys**, and how JOINs allow you to merge data from multiple tables in a meaningful way.

Query 1

Description: Retrieve the city and country names together by joining the city and country tables. This query shows how location data is connected through the `country_id` field.



The screenshot shows a PostgreSQL client interface with a query editor and a results viewer. The query editor contains the following SQL code:

```
1 SELECT city, country FROM city
2 INNER JOIN country ON city.country_id = country.country_id;
3
```

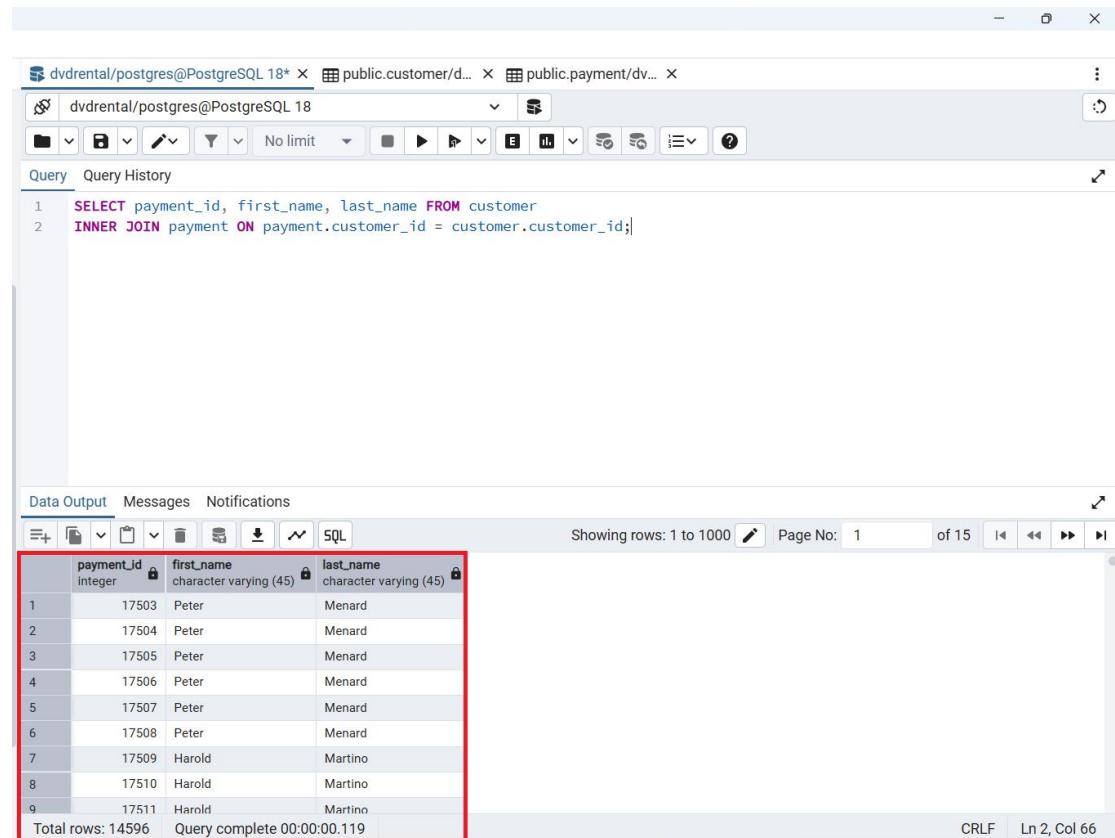
The results viewer displays a table with two columns: `city` and `country`. The table has 600 rows. A red box highlights the first few rows of the table. The table data is as follows:

	city	country
1	A Corua (La Corua)	Spain
2	Abha	Saudi Arabia
3	Abu Dhabi	United Arab Emirates
4	Acua	Mexico
5	Adana	Turkey
6	Addis Abeba	Ethiopia
7	Aden	Yemen
8	Adoni	India
9	Ahmadnagar	India
Total rows: 600		Query complete 00:00:00.454

A green message bar at the bottom right of the results viewer says: "Successfully run. Total query runtime: 405 msec. 600 rows affected."

Query 2

Description: Display each payment's ID along with the customer's first and last name. This query demonstrates how payment records are linked to the customers who made them.



The screenshot shows a PostgreSQL client interface with three tabs at the top: 'dvrental/postgres@PostgreSQL 18*' (active), 'public.customer/d...', and 'public.payment/dv...'. The main area is a query editor with the following SQL code:

```
1 SELECT payment_id, first_name, last_name FROM customer
2 INNER JOIN payment ON payment.customer_id = customer.customer_id;
```

Below the query editor is a data grid labeled 'Data Output' with three tabs: 'Data Output' (selected), 'Messages', and 'Notifications'. The data grid displays the results of the query:

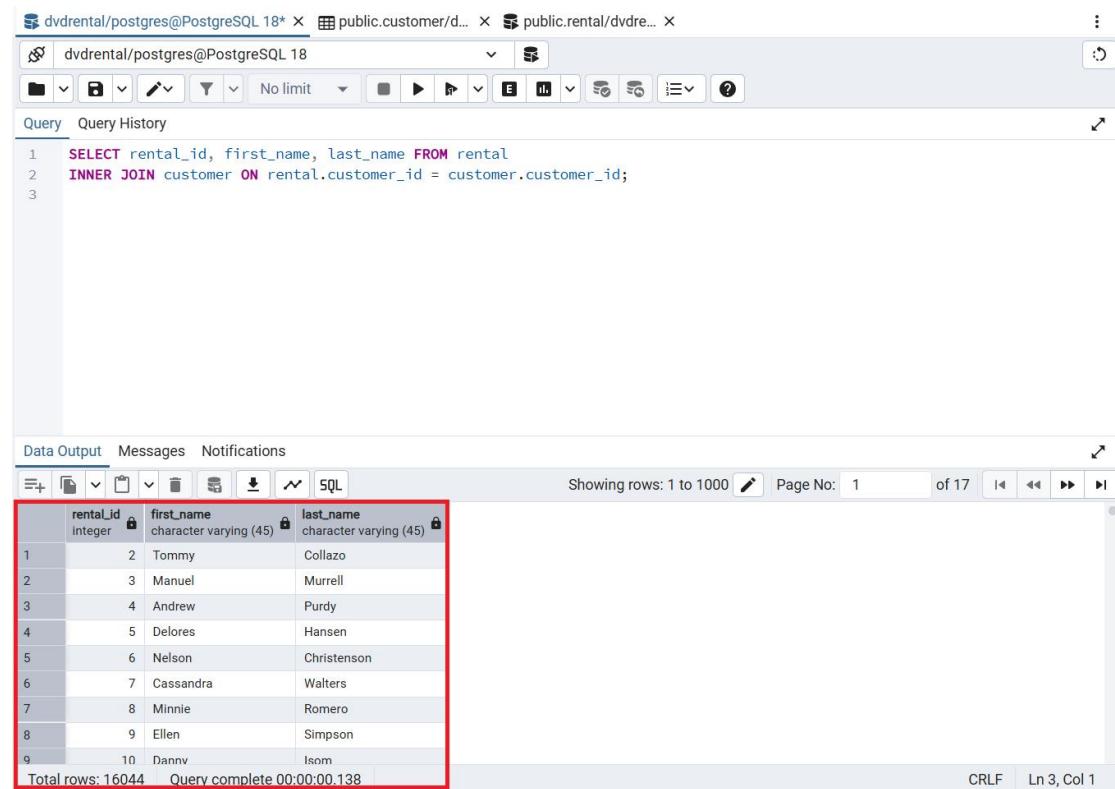
	payment_id	first_name	last_name
1	17503	Peter	Menard
2	17504	Peter	Menard
3	17505	Peter	Menard
4	17506	Peter	Menard
5	17507	Peter	Menard
6	17508	Peter	Menard
7	17509	Harold	Martino
8	17510	Harold	Martino
9	17511	Harold	Martino

Total rows: 14596 Query complete 00:00:00.119

CRLF Ln 2, Col 66

Query 3

Description: List each rental's ID together with the customer's first and last name.



The screenshot shows a PostgreSQL client interface with a query editor and a results viewer. The query editor contains the following SQL code:

```
1 SELECT rental_id, first_name, last_name FROM rental
2 INNER JOIN customer ON rental.customer_id = customer.customer_id;
3
```

The results viewer displays a table with three columns: rental_id, first_name, and last_name. The first 10 rows of the table are highlighted with a red border. The table data is as follows:

rental_id	first_name	last_name
1	2	Tommy Collazo
2	3	Manuel Murrell
3	4	Andrew Purdy
4	5	Delores Hansen
5	6	Nelson Christenson
6	7	Cassandra Walters
7	8	Minnie Romero
8	9	Ellen Simpson
9	10	Dannv Isom

Total rows: 16044 Query complete 00:00:00.138