**1st Case Name :** SQL Query Creation

**Case Objective :** Write an SQL query to get the top 10 customers by net total spending in the last 6 months

**Task and Constraints:** The expected output should include:

* Customer\_id
* Customer\_name
* Country
* Total\_spent
* Unique\_products\_purchase
* Unique\_total\_orders
* Filtered only “completed” orders

**My approach to solve this case:**

1. Build PostgreSQL in a dockerized environment, through docker-compose.yaml

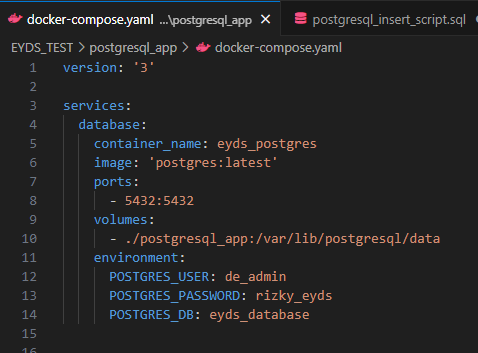


Figure - docker-compose.yaml syntax

1. Create database connections through DBeaver app

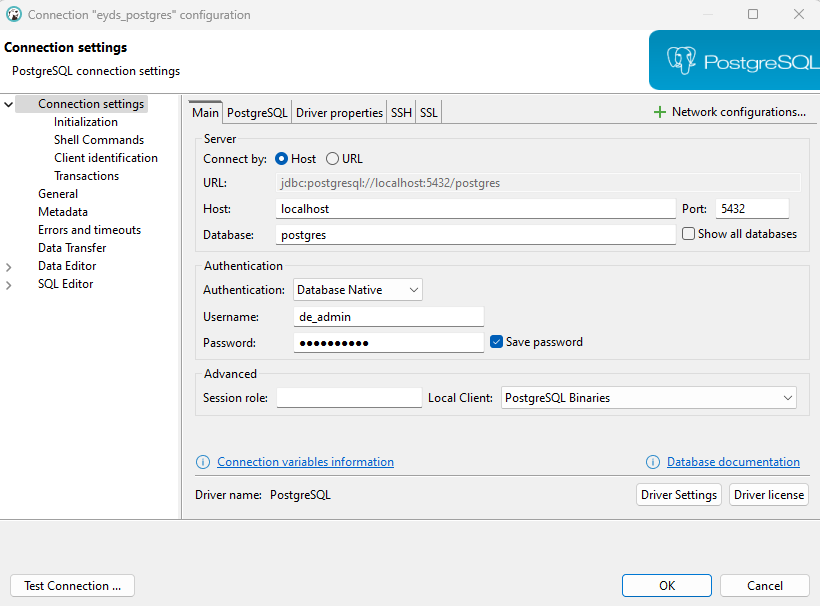


Figure Create connection in DBeaver

1. Create tables in PostgreSQL database using SQL DDL commands:
   * Create customers table:

**create** **table** customers (

customer\_id **INT** **primary** **key**,

name **VARCHAR**(50),

email **VARCHAR**(100),

country **VARCHAR**(50),

created\_at **TIMESTAMP**

);

* + Create products table:

**create** **table** products (

product\_id **INT** **primary** **key**,

name **varchar**(50),

category **varchar**(50),

base\_price **INT**

);

* + Create orders table:

**create** **table** orders (

order\_id **INT** **primary** **key**,

customer\_id **INT**,

order\_date **TIMESTAMP**,

status **VARCHAR**(255) **not** **null**,

**foreign** **key** (customer\_id) **references** customers(customer\_id)

);

* + Create order\_items table:

**create** **table** order\_items (

order\_item\_id **INT** **primary** **key**,

order\_id **INT**,

product\_id **INT**,

quantity **INT**,

price **INT**,

discount **INT**,

TAX **INT**,

**foreign** **key** (order\_id) **references** orders(order\_id),

**foreign** **key**(product\_id) **references** products(product\_id)

);

1. Generate sample data using AI
2. Convert generated data into DML commands and insert to table in PostgreSQL

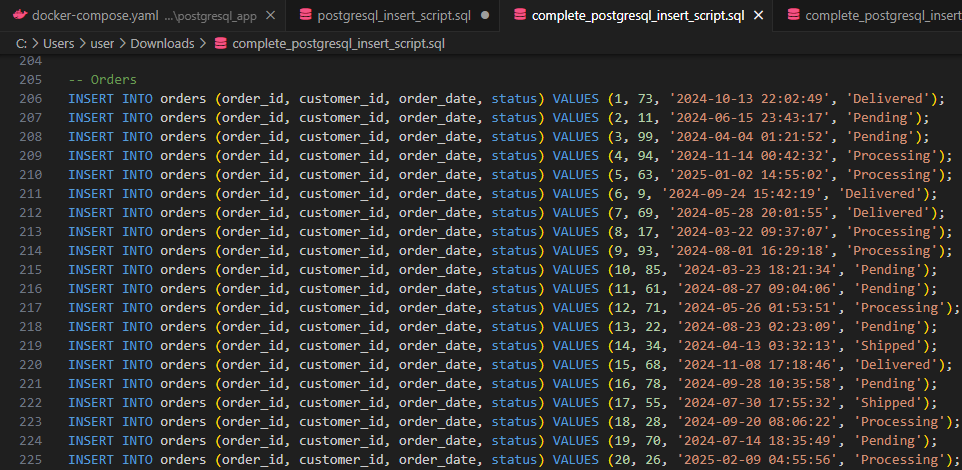
**

Figure - DML commands sample

1. Perform Query using Common Table Expression (CTE) in SQL

-- query top 10 customers by total\_spending in the last 6 months and delivered status only

**with** recent\_order **AS**(

**select**

*o*.order\_id,

*o*.customer\_id,

*o*.order\_date,

*o*.status,

*oi*.product\_id,

*oi*.quantity,

*oi*.price,

*oi*.discount,

*oi*.tax

**from** orders *o*

**left** **join** order\_items *oi*

**on** *o*.order\_id = *oi*.order\_id

**where** order\_date >= '2024-09-20' **and** status = 'Delivered'

**order** **by** order\_date **desc**

),

customer\_spending **AS**(

**select**

customer\_id,

**count**(**distinct** product\_id) **as** total\_unique\_products\_purchased,

**count**(**distinct** order\_id) **as** total\_unique\_orders,

**SUM** (quantity \* price) - **SUM**(discount) + **SUM**(tax) **as** total\_cust\_spending

**from** recent\_order

**group** **by** customer\_id

**order** **by** total\_unique\_orders **desc**

)

**select**

*c*.customer\_id,

*c*.name **as** *cust\_name*,

*c*.country,

*cs*.total\_unique\_orders,

*cs*.total\_unique\_products\_purchased,

*cs*.total\_cust\_spending

**from** customers *c*

**join** customer\_spending *cs*

**on** *c*.customer\_id = *cs*.customer\_id

**order** **by** total\_cust\_spending **desc**

**limit** 10

1. Query Results

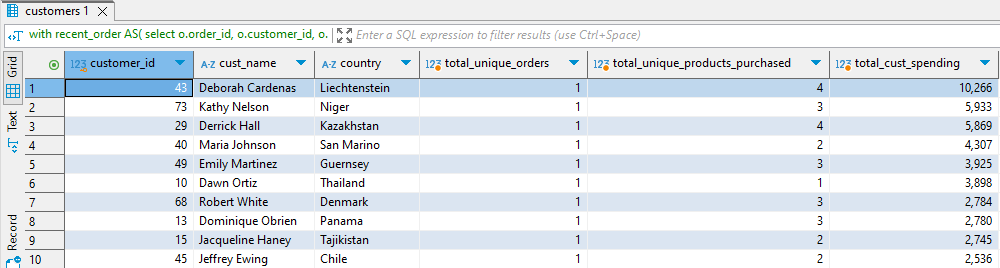


Figure - Query Results