**SQL Music Database**

-- Q1: Write a query to return email, first name, last name and genre of all rock music listeners.

-- write your list ordered alphabetically by email starting with A.

SELECT DISTINCT email, first\_name, last\_name

FROM customer

JOIN invoice

ON customer.customer\_id = invoice.customer\_id

JOIN invoice\_line

ON invoice.invoice\_id = invoice\_line.invoice\_id

WHERE track\_id IN(

SELECT track\_id FROM track

JOIN genre ON track.genre\_id = genre.genre\_id

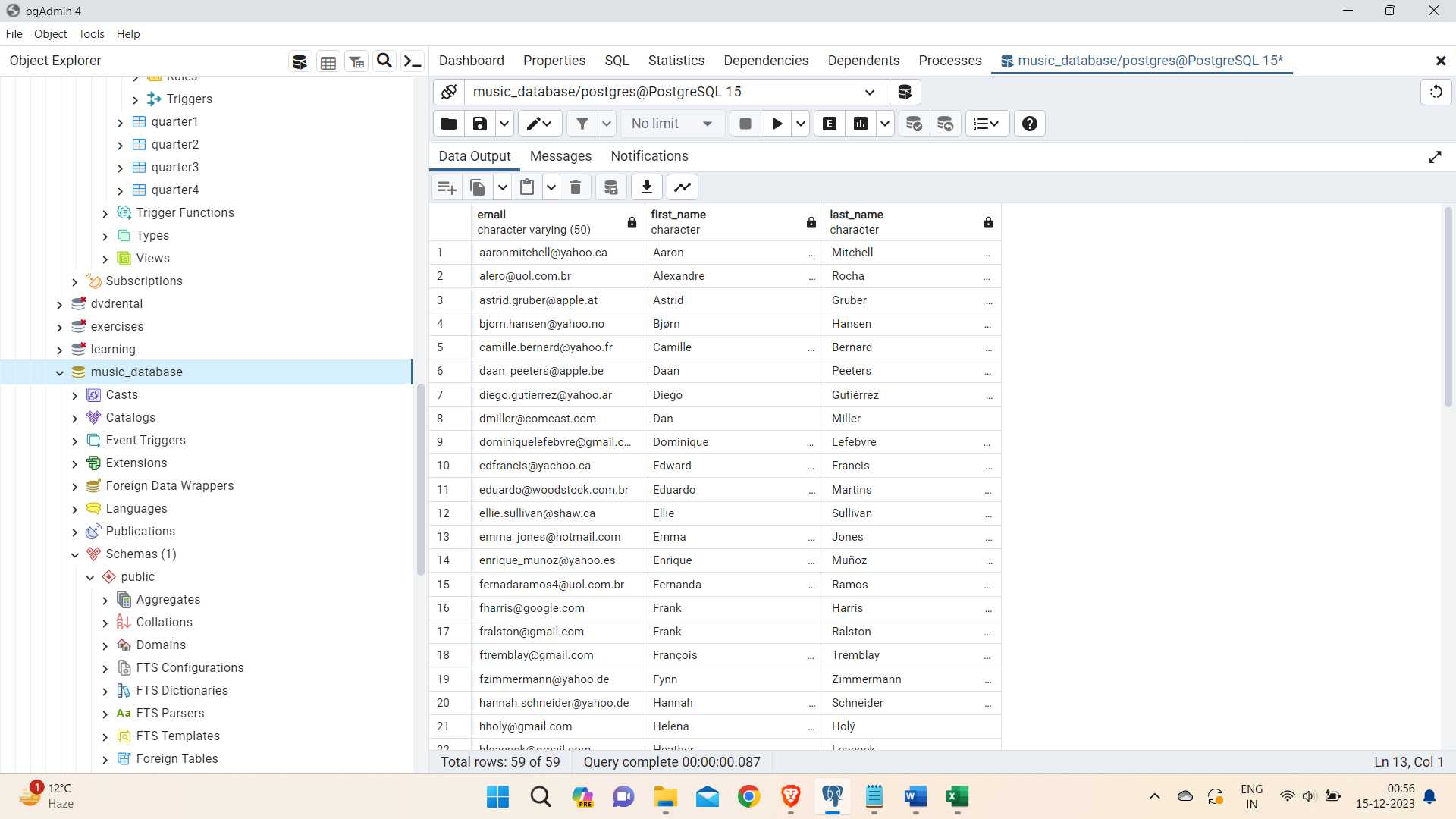
WHERE genre.name = 'Rock'

)

ORDER BY email;

OUTPUT:

Following tables shows few rows of total 59 rows.



--Q2: Let's invite the artists who have written the most rock music in our dataset.

--Write a query that returns the artist name and total track count of the top 10 rock bands.

SELECT artist.name as artist\_name, COUNT(\*) AS track\_count

FROM artist

JOIN album ON artist.artist\_id = album.artist\_id

JOIN track ON album.album\_id = track.album\_id

JOIN genre ON track.genre\_id = genre.genre\_id

WHERE genre.name = 'Rock'

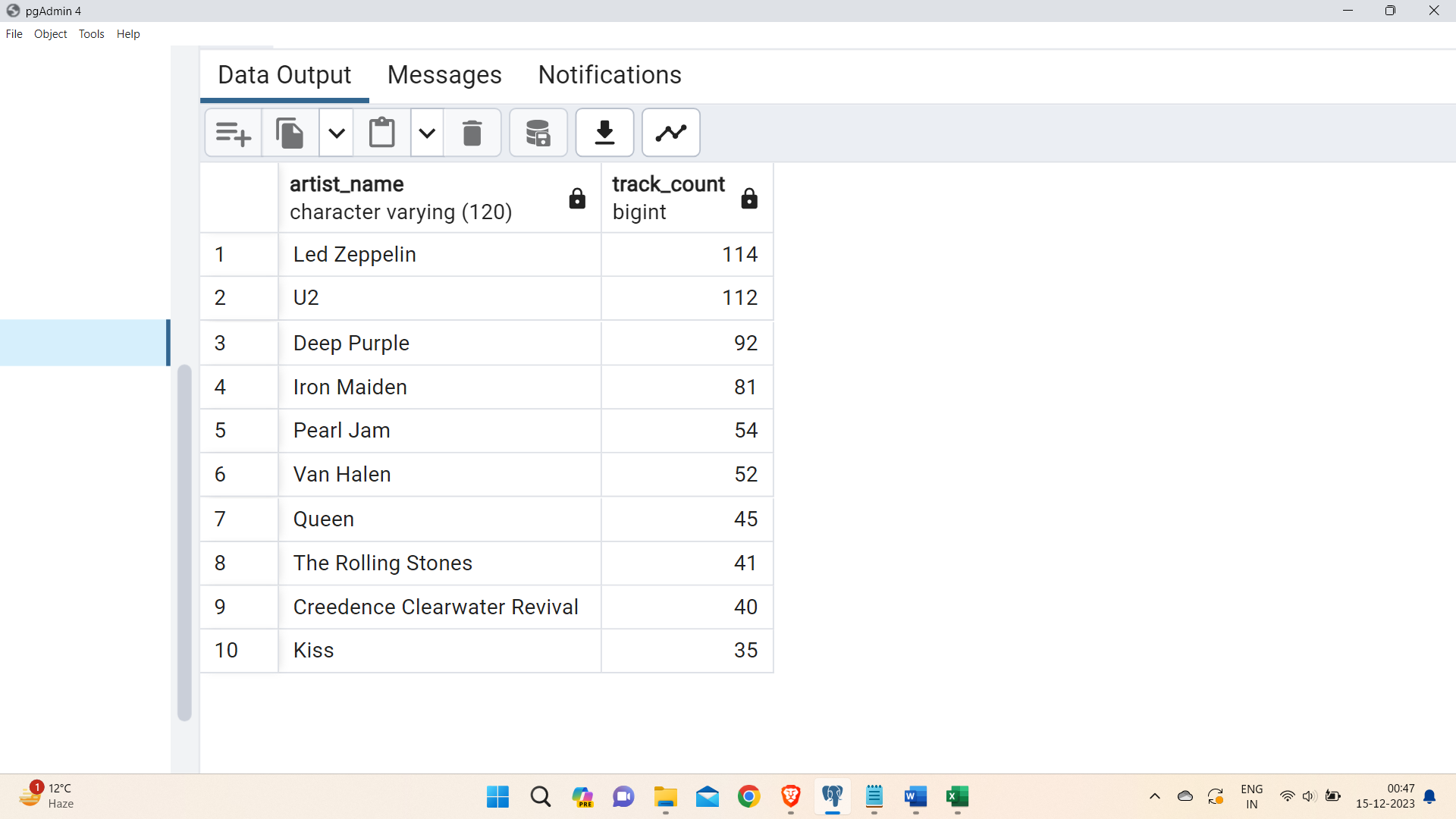
GROUP BY artist.name

ORDER BY track\_count DESC

LIMIT 10;

OUTPUT:

Total of 10 rows were resulted.



--Q3: Return all the track names that have a song length longer than the average song length.

--Return the names and milliseconds for each track.

--Order by the song length with the longest song listed first.

SELECT name, milliseconds

FROM track

WHERE milliseconds > (

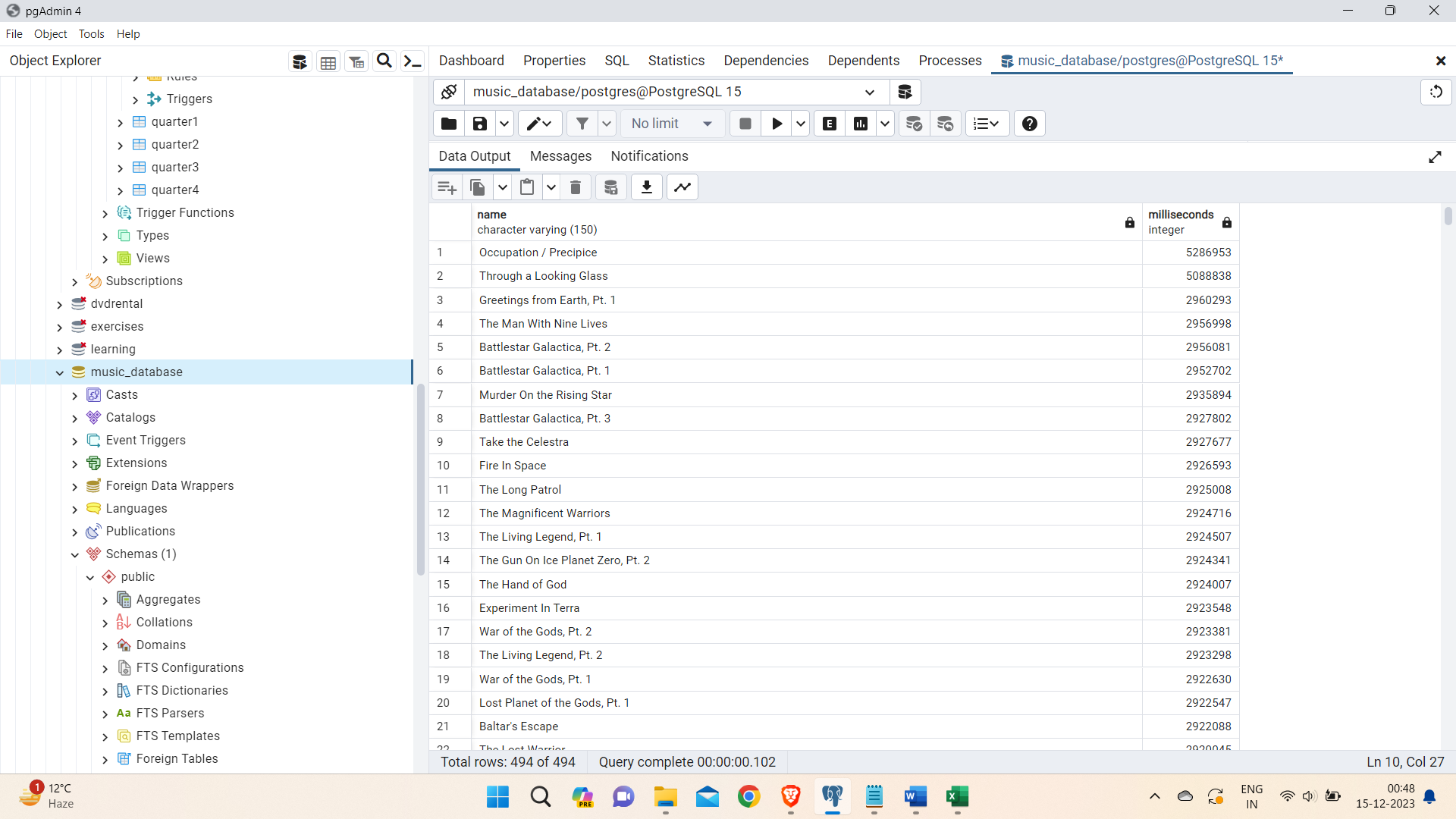
SELECT AVG(milliseconds) AS average

FROM track)

ORDER BY milliseconds DESC;

OUTPUT:

Following table shows first few rows out of 494 total rows.



--Q4: Write how much amount spent by each customer on artists?

--Write a query to return customer name and total spent.

SELECT c.customer\_id, c.first\_name, c.last\_name, SUM(il.unit\_price\*il.quantity) AS amount\_spent

FROM invoice i

JOIN customer c ON c.customer\_id = i.customer\_id

JOIN invoice\_line il ON il.invoice\_id = i.invoice\_id

JOIN track t ON t.track\_id = il.track\_id

JOIN album alb ON alb.album\_id = t.album\_id

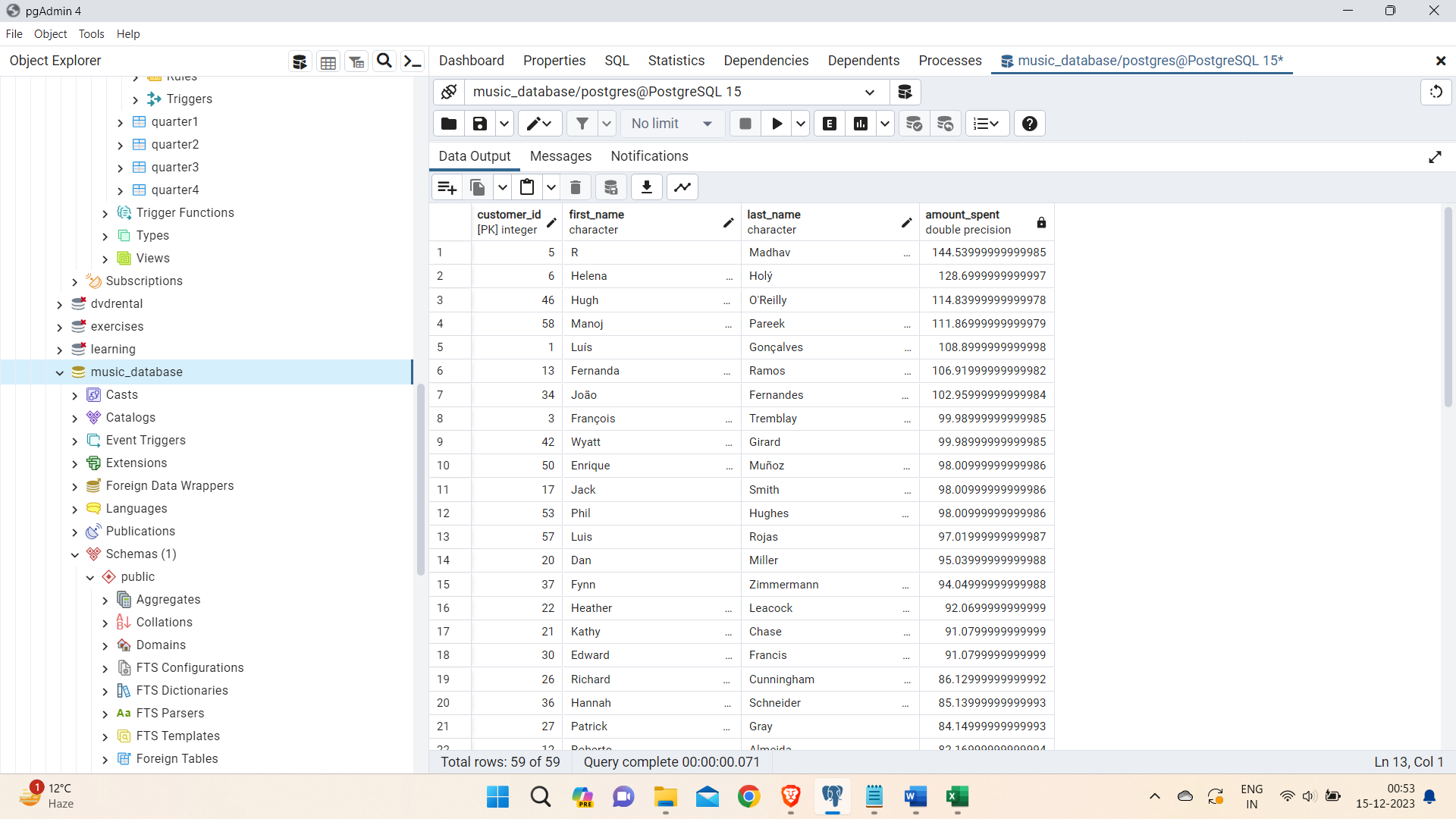
join artist on artist.artist\_id = alb.artist\_id

GROUP BY 1,2,3

order by amount\_spent desc;

OUTPUT:

Following table shows first few rows out of 59 total rows.



--Q5 We want to find out the most popular music genre for each country.

--We determine the most popular genre as the genre with the highest amount of purchases.

--Write a query that returns each country along with the top genre.

--For countries where the maximum number of purchases is shared return all genres.

WITH popular\_genre AS

(

SELECT COUNT(invoice\_line.quantity) AS purchases, customer.country, genre.name, genre.genre\_id,

ROW\_NUMBER() OVER(PARTITION BY customer.country ORDER BY COUNT(invoice\_line.quantity) DESC) AS RowNo

FROM invoice\_line

JOIN invoice ON invoice.invoice\_id = invoice\_line.invoice\_id

JOIN customer ON customer.customer\_id = invoice.customer\_id

JOIN track ON track.track\_id = invoice\_line.track\_id

JOIN genre ON genre.genre\_id = track.genre\_id

GROUP BY 2,3,4

ORDER BY 1 DESC

)

SELECT \* FROM popular\_genre WHERE RowNo <= 1

OUTPUT:

Following table shows first few rows out of 24 total rows.

