Structured Query Language project by Navjoth Singh on Music Store Data

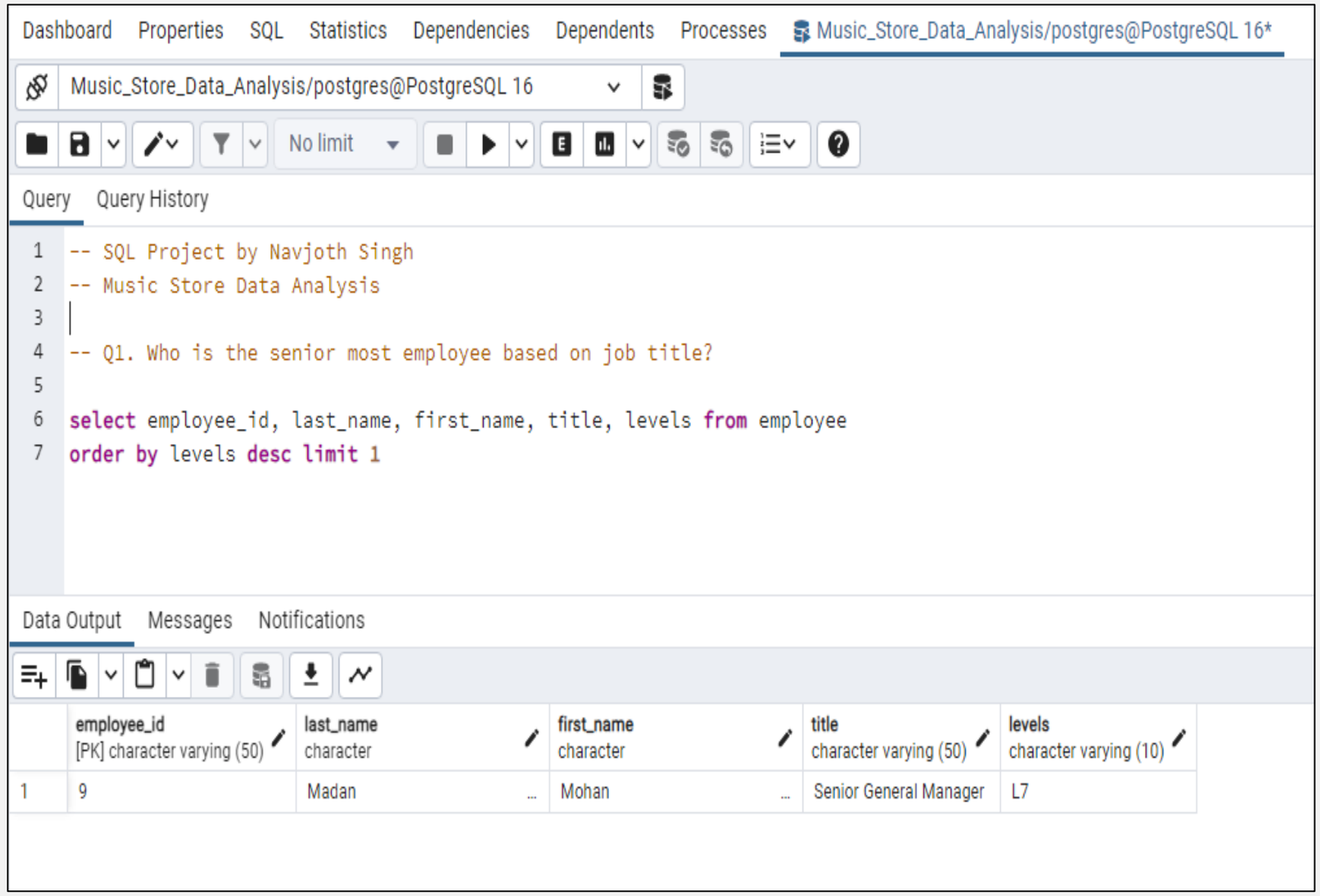
Question Set 1 – Easy (5 Questions)

Q1: Who is the senior most employee based on job title?

SQL Query

- select employee_id, last_name, first_name, title, levels from employee
- order by levels desc limit 1

Data Output



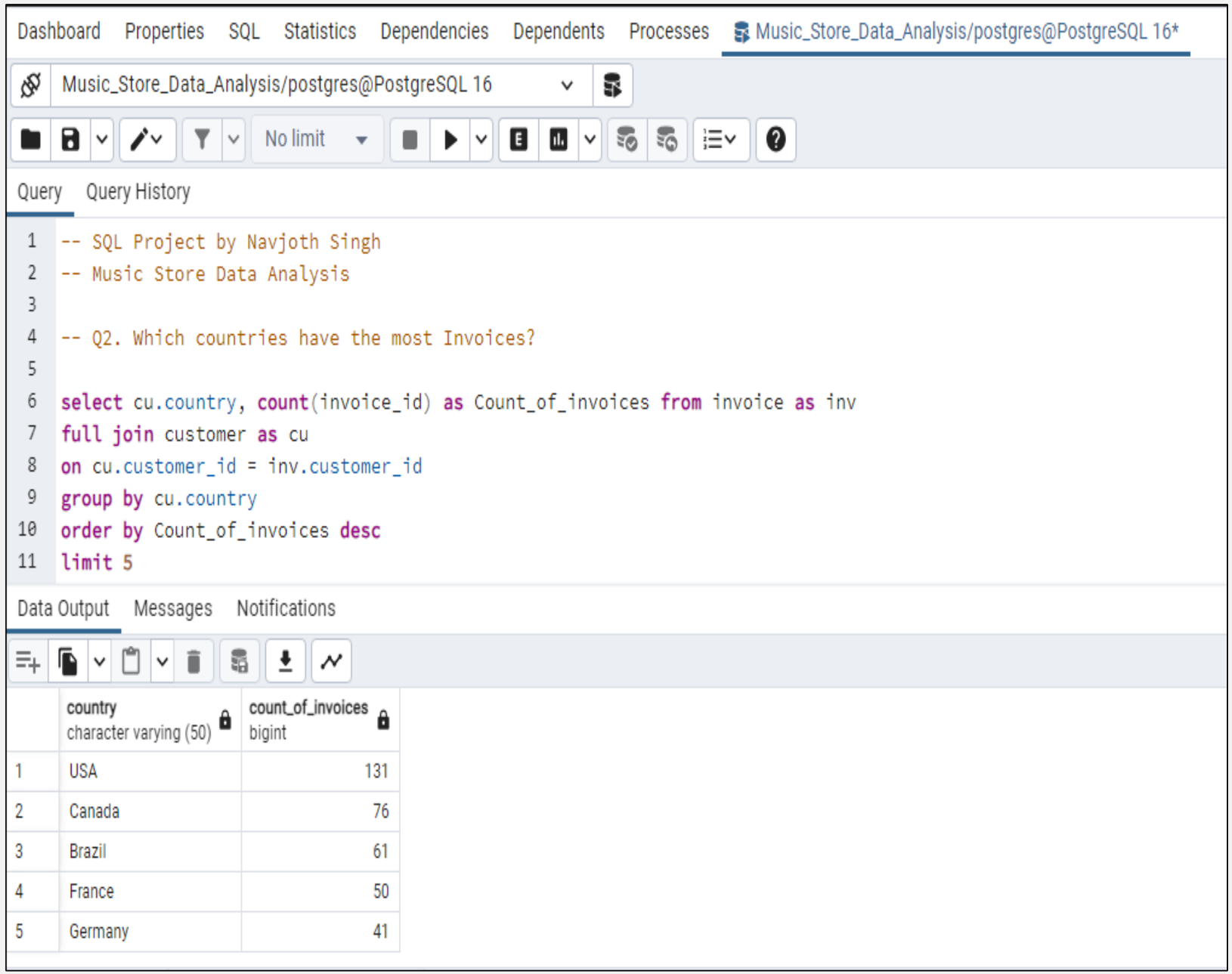
SQL Project by Navjoth Singh – 09-Oct.-2023 – PostgreSQL

Q2: Which countries have the most Invoices?

SQL Query

- select cu.country, count(invoice_id) as Count_of_invoices from invoice as inv
- full join customer as cu
- on cu.customer_id = inv.customer_id
- group by cu.country
- order by Count_of_invoices desc
- limit 5

Data Output

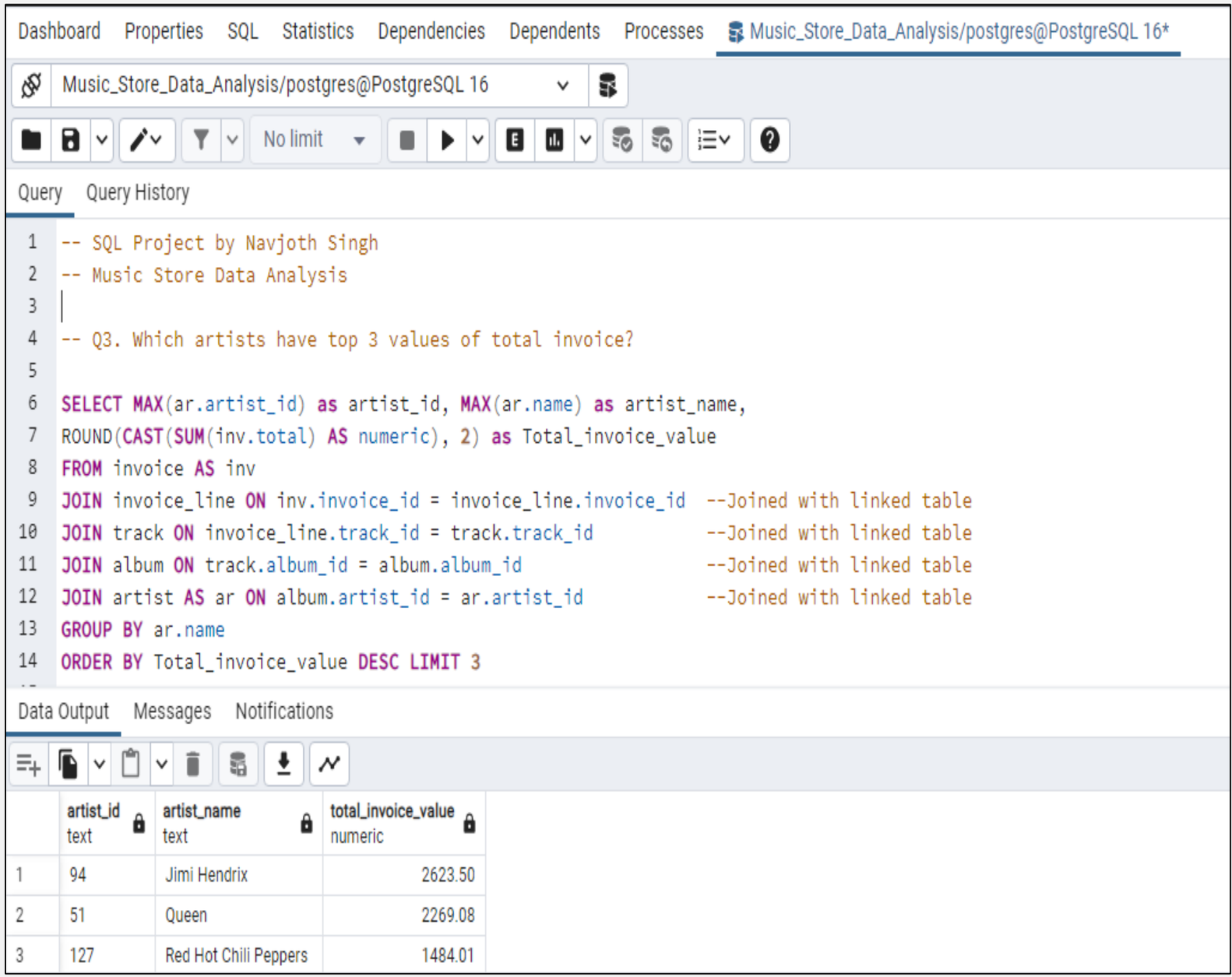


Q3: Which artists have top 3 values of total invoice?

SQL Query

- SELECT MAX(ar.artist_id) as artist_id,
  MAX(ar.name) as artist_name,
- ROUND(CAST(SUM(inv.total) AS numeric), 2) as
  Total_invoice_value
- FROM invoice AS inv
- JOIN invoice_line ON inv.invoice_id =
  invoice_line.invoice_id
- JOIN track ON invoice_line.track_id = track.track_id
- JOIN album ON track.album_id = album.album_id
- JOIN artist AS ar ON album.artist_id = ar.artist_id
- GROUP BY ar.name
- ORDER BY Total_invoice_value DESC LIMIT 3

Data Output

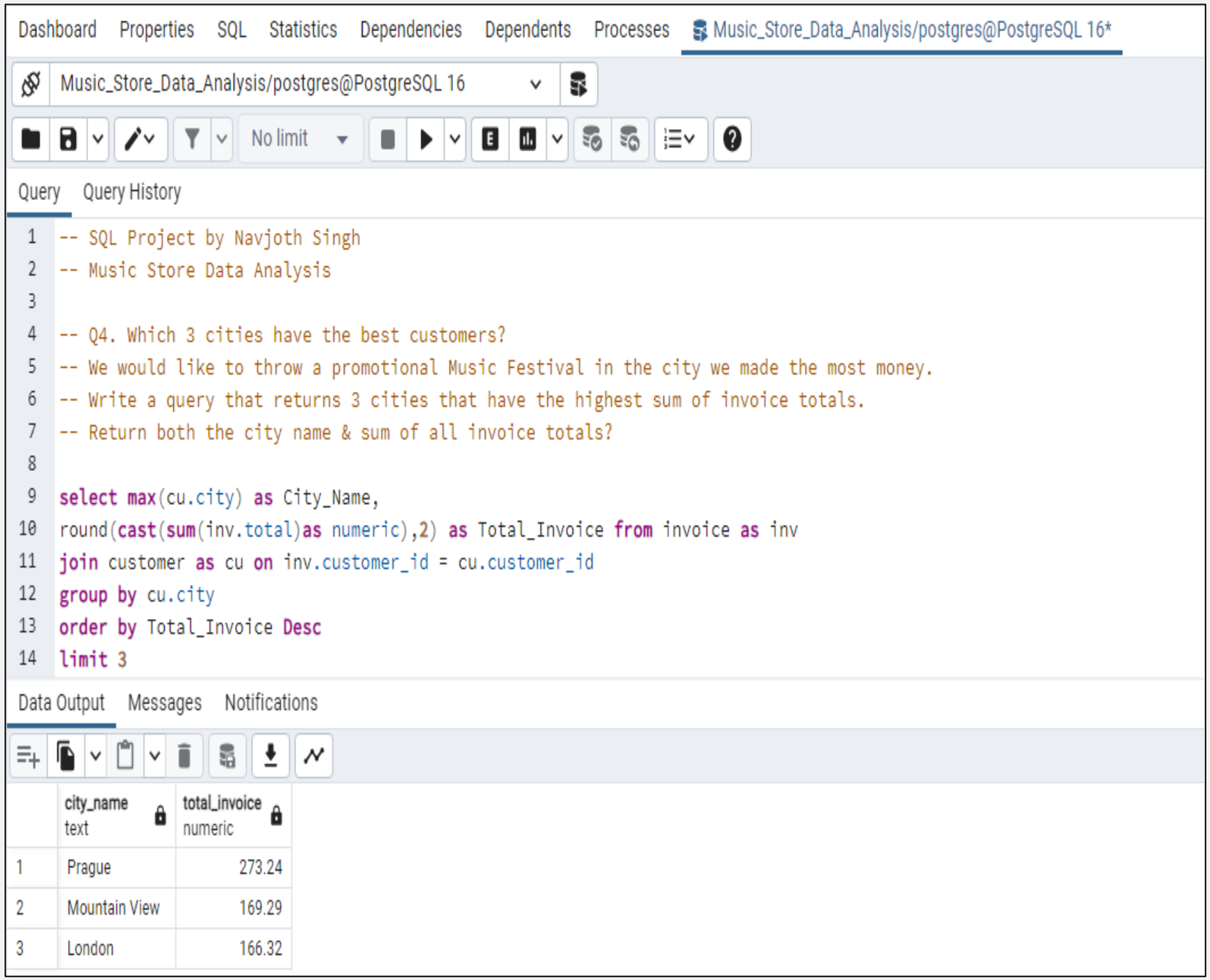


Q4: Which 3 cities have the best customers? We would like to throw a promotional Music Festival in the city we made the most money. Write a query that returns 3 cities that have the highest sum of invoice totals. Return both the city name & sum of all invoice totals?

SQL Query

- select max(cu.city) as City_Name,
- round(cast(sum(inv.total)as numeric),2) as Total_Invoice from invoice as inv
- join customer as cu on inv.customer_id = cu.customer_id
- group by cu.city
- order by Total_Invoice Desc
- limit 3

Data Output

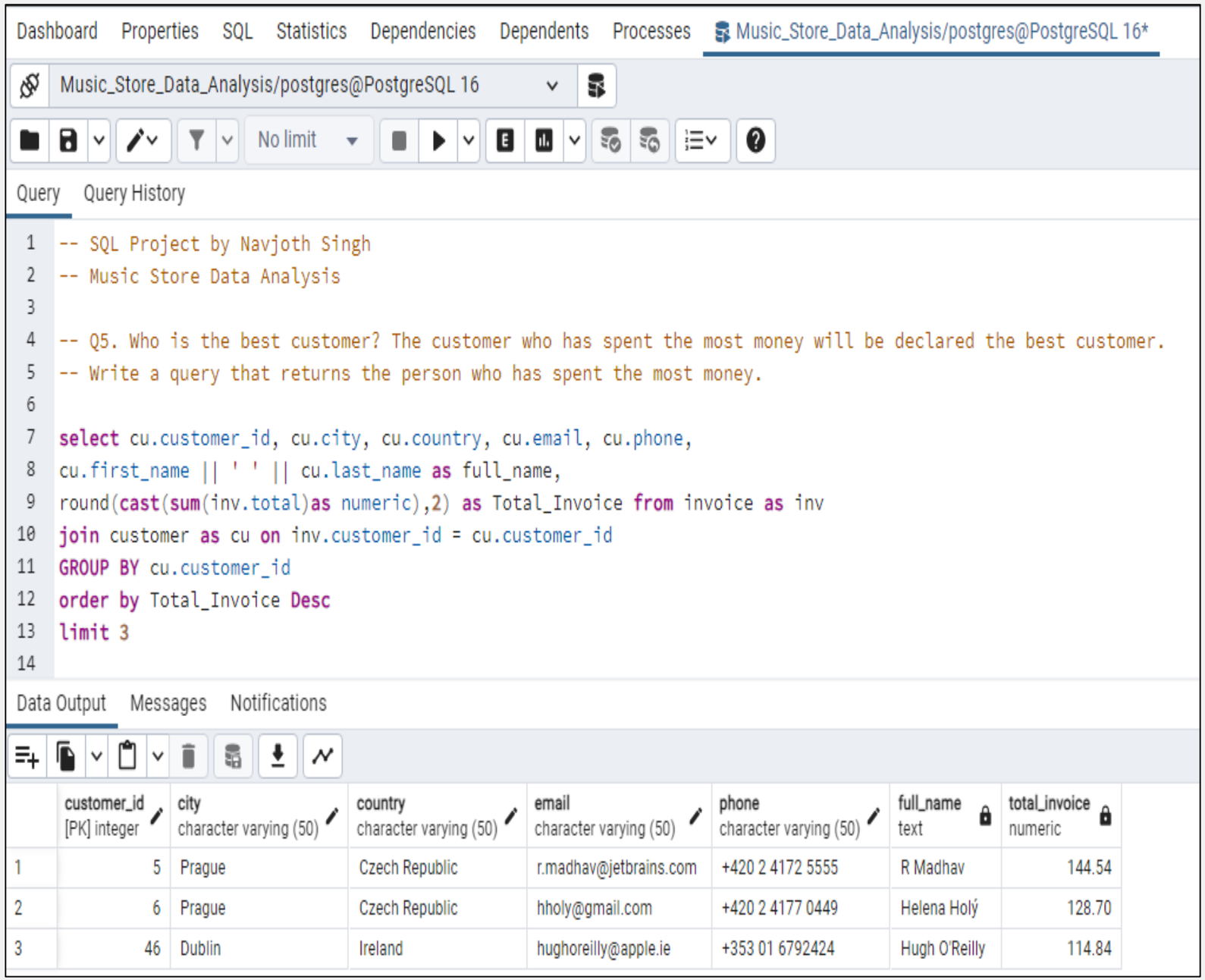


Question 5: Who is the best customer? The customer who has spent the most money will be declared the best customer. Write a query that returns the person who has spent the most money.

SQL Query

- select cu.customer_id, cu.city, cu.country, cu.email, cu.phone,
- cu.first_name || ' ' || cu.last_name as full_name,
- round(cast(sum(inv.total)as numeric),2) as Total_Invoice from invoice as inv
- join customer as cu on inv.customer_id = cu.customer_id
- GROUP BY cu.customer_id
- order by Total_Invoice Desc
- limit 3

Data Output



	customer_id [PK] integer	city character varying (50)	country character varying (50)	email character varying (50)	phone character varying (50)	full_name text	total_invoice numeric
1	5	Prague	Czech Republic	r.madhav@jetbrains.com	+420 2 4172 5555	R Madhav	144.54
2	6	Prague	Czech Republic	hholy@gmail.com	+420 2 4177 0449	Helena Holý	128.70
3	46	Dublin	Ireland	hughoreilly@apple.ie	+353 01 6792424	Hugh O'Reilly	114.84

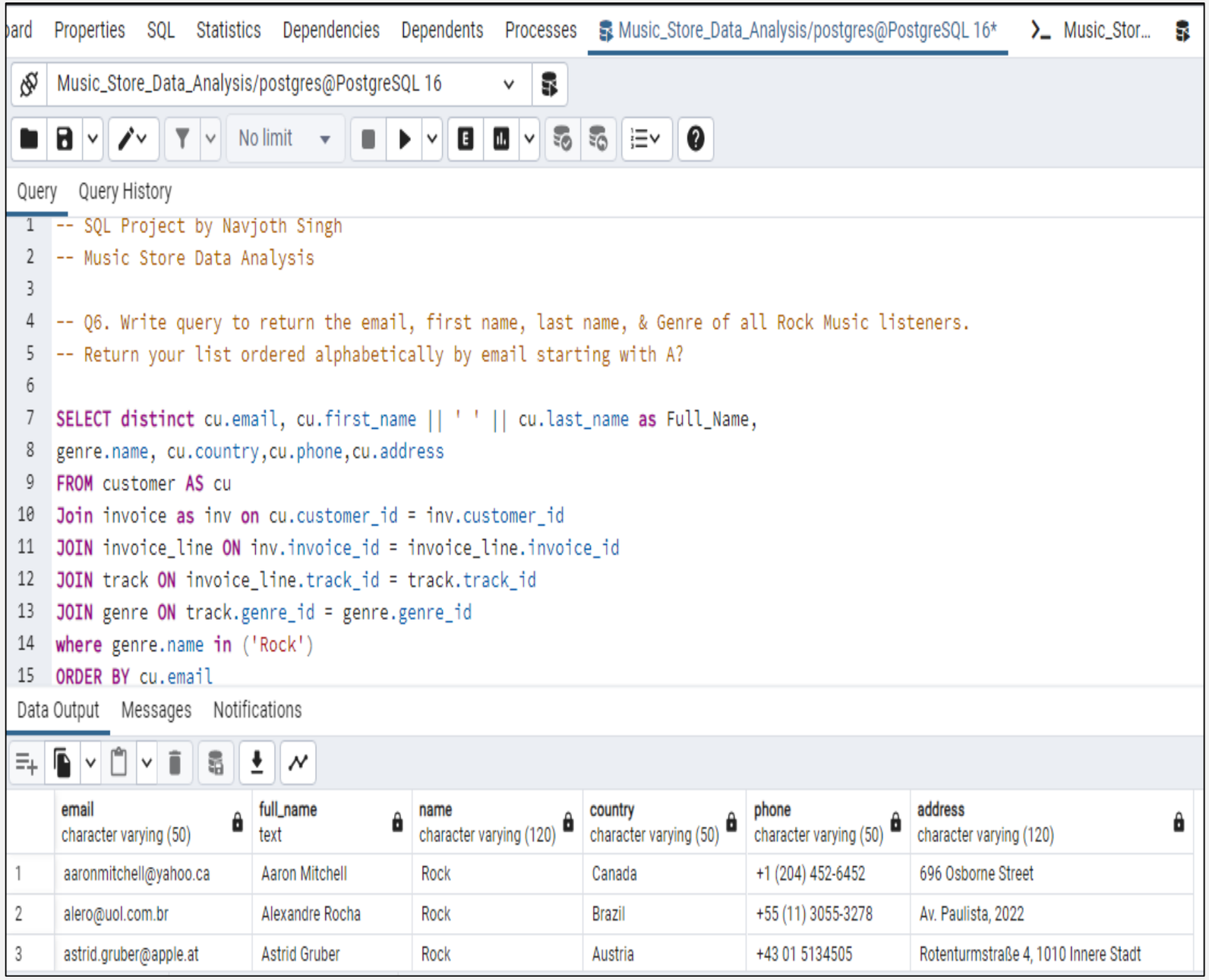
Question Set 2 – Moderate (3 Questions)

Q6: Write query to return the email, first name, last name, & Genre of all Rock Music listeners. Return your list ordered alphabetically by email starting with A?

SQL Query

- SELECT distinct cu.email, cu.first_name || ' ' || cu.last_name as Full_Name,
- genre.name, cu.country,cu.phone,cu.address
- FROM customer AS cu
- Join invoice as inv on cu.customer_id = inv.customer_id
- JOIN invoice_line ON inv.invoice_id = invoice_line.invoice_id
- JOIN track ON invoice_line.track_id = track.track_id
- JOIN genre ON track.genre_id = genre.genre_id
- where genre.name in ('Rock')
- ORDER BY cu.email

Data Output

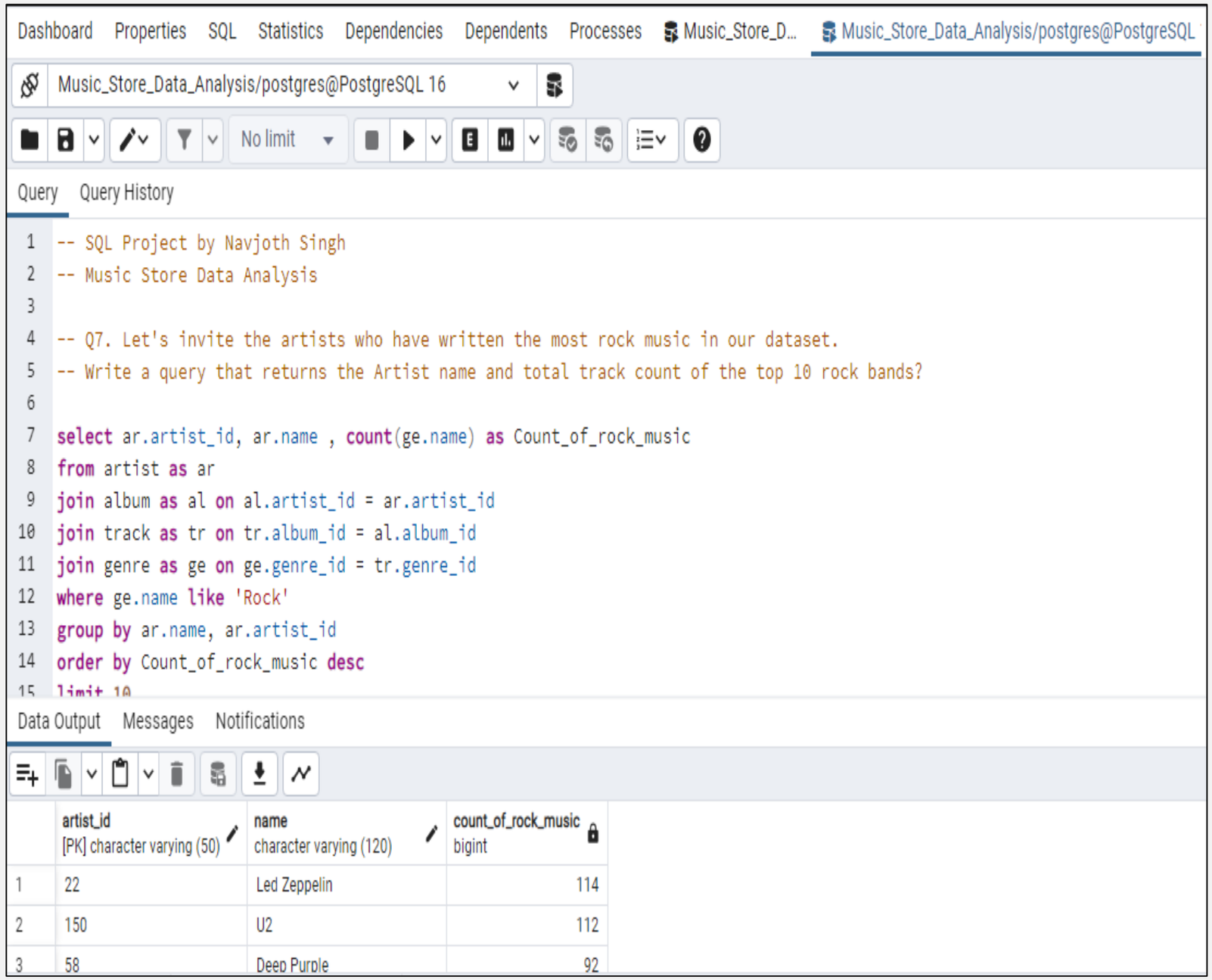


Q7: Let's invite the artists who have written the most rock music in our dataset. Write a query that returns the Artist name and total track count of the top 10 rock bands?

SQL Query

- select ar.artist_id, ar.name , count(ge.name) as Count_of_rock_music
- from artist as ar
- join album as al on al.artist_id = ar.artist_id
- join track as tr on tr.album_id = al.album_id
- join genre as ge on ge.genre_id = tr.genre_id
- where ge.name like 'Rock'
- group by ar.name, ar.artist_id
- order by Count_of_rock_music desc
- limit 10

Data Output



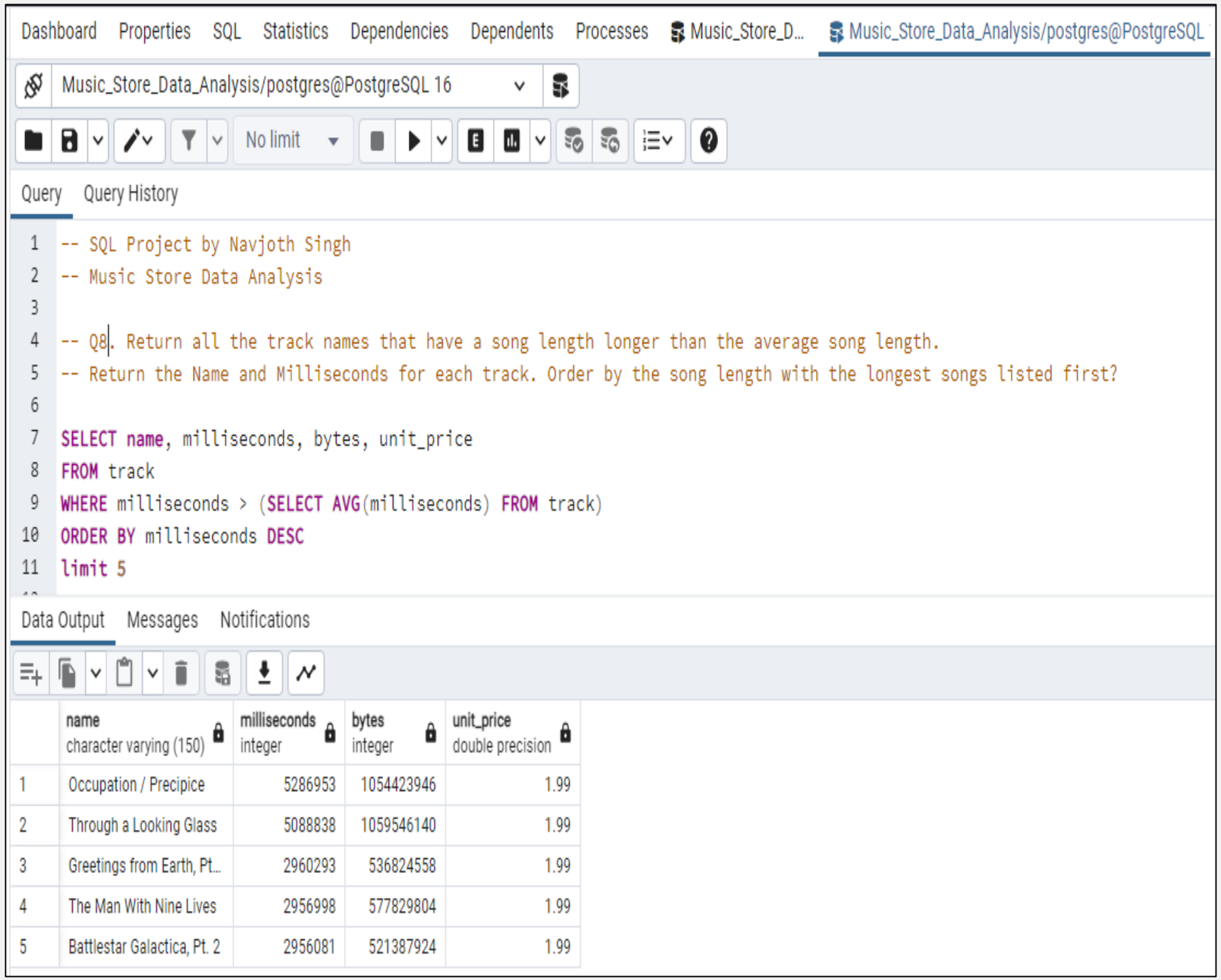
SQL Project by Navjoth Singh – 09-Oct.-2023 – PostgreSQL

Q8: Return all the track names that have a song length longer than the average song length. Return the Name and Milliseconds for each track. Order by the song length with the longest songs listed first?

SQL Query

- SELECT name, milliseconds, bytes, unit_price
- FROM track
- WHERE milliseconds > (SELECT AVG(milliseconds) FROM track)
- ORDER BY milliseconds DESC
- limit 5

Data Output



	name character varying (150)	milliseconds integer	bytes integer	unit_price double precision
1	Occupation / Precipice	5286953	1054423946	1.99
2	Through a Looking Glass	5088838	1059546140	1.99
3	Greetings from Earth, Pt...	2960293	536824558	1.99
4	The Man With Nine Lives	2956998	577829804	1.99
5	Battlestar Galactica, Pt. 2	2956081	521387924	1.99

Question Set 3 - Advance (3 Questions)

Q9: Find how much amount spent by each customer on best-selling artists? Write a query to return customer name, artist name and total spent?

SQL Query

- with best_selling_artist as(
  - select art.artist_id, art.name, round(cast(sum(invl.unit_price*invl.quantity) as numeric),2)
  - as total_invoice from invoice as inv
  - join invoice_line as invl on invl.invoice_id = inv.invoice_id
  - join track as trk on trk.track_id = invl.track_id
  - join album as alb on alb.album_id = trk.album_id
  - join artist as art on art.artist_id = alb.artist_id
  - group by 1,2
  - order by 3 Desc
  - limit 1)
- Select cu.customer_id, cu.first_name || ' ' || cu.last_name as Full_Name_of_Customer,
- max(art.name) as Artist_Name, max(art.artist_id) as artist_id, round(cast(sum(invl.quantity*invl.unit_price) as numeric),2)
- as total_incoice from customer as cu
- join Invoice as inv on inv.customer_id = cu.customer_id
- join invoice_line as invl on invl.invoice_id = inv.invoice_id
- join track as trk on trk.track_id = invl.track_id
- join album as alb on alb.album_id = trk.album_id
- join best_selling_artist as art on art.artist_id = alb.artist_id
- group by 1,2
- order by 5 Desc

Data Output

Dashboard | Properties | SQL | Statistics | Dependencies | Dependents | Processes | Music_Store_Data_Analysis/postgres@PostgreSQL 16*

Music_Store_Data_Analysis/postgres@PostgreSQL 16

No limit

Query | Query History

```sql
-- SQL Project by Navjoth Singh
-- Music Store Data Analysis

-- Q9. Find how much amount spent by each customer on best-selling artists?
-- Write a query to return customer name, artist name and total spent?

with best_selling_artist as(
    select art.artist_id, art.name, round(cast(sum(invl.unit_price*invl.quantity) as numeric),2)
    as total_invoice from invoice as inv
    join invoice_line as invl on invl.invoice_id = inv.invoice_id
    join track as trk on trk.track_id = invl.track_id
    join album as alb on alb.album_id = trk.album_id
    join artist as art on art.artist_id = alb.artist_id
```

Data Output | Messages | Notifications

	customer_id [PK] integer	full_name_of_customer text	artist_name text	artist_id text	total_incoice numeric
1	46	Hugh O'Reilly	Queen	51	27.72
2	38	Niklas Schröder	Queen	51	18.81
3	3	François Tremblay	Queen	51	17.82
4	34	João Fernandes	Queen	51	16.83

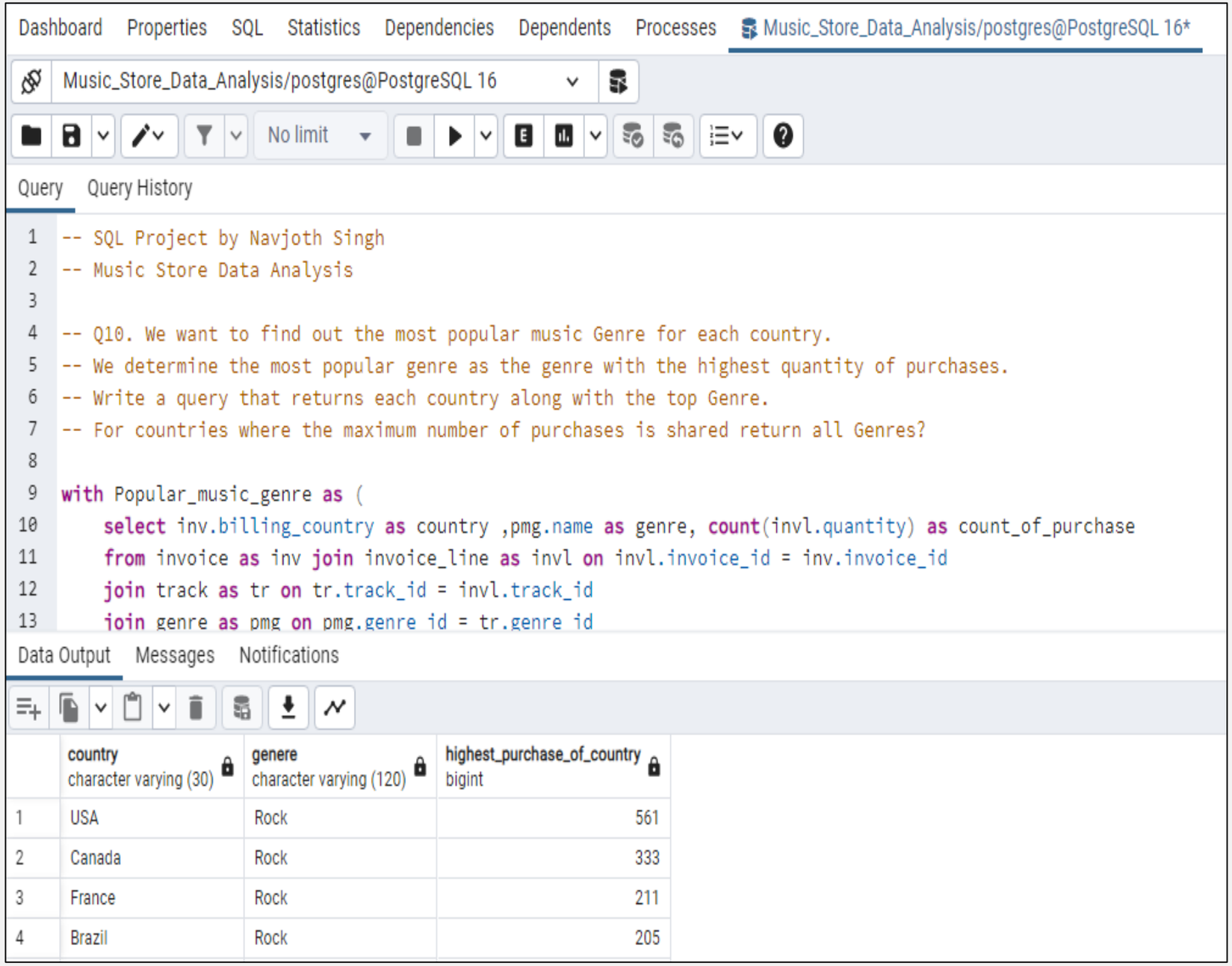
SQL Project by Navjoth Singh – 09-Oct.-2023 – PostgreSQL

Q10: 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 quantity 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?

SQL Query (This can be solve with the help of RowNumber() & RECURSIVE Query)

- with Popular_music_genre as (
  - select inv.billing_country as country ,pmg.name as genre, count(invl.quantity) as count_of_purchase
  - from invoice as inv join invoice_line as invl on invl.invoice_id = inv.invoice_id
  - join track as tr on tr.track_id = invl.track_id
  - join genre as pmg on pmg.genre_id = tr.genre_id
  - group by 1,2
  - order by 3 desc)
- select pmg.country as Country, pmg.genre as Genere,
- max(pmg.count_of_purchase) as Highest_purchase_of_country
- from Popular_music_genre as pmg
- join (select pmg.country as gof_name, max(pmg.count_of_purchase) as purchase
  - from Popular_music_genre as pmg group by 1)
- as gof on gof.gof_name = pmg.country
- where pmg.count_of_purchase = gof.purchase
- group by 1,2
- order by 3 Desc

Data Output



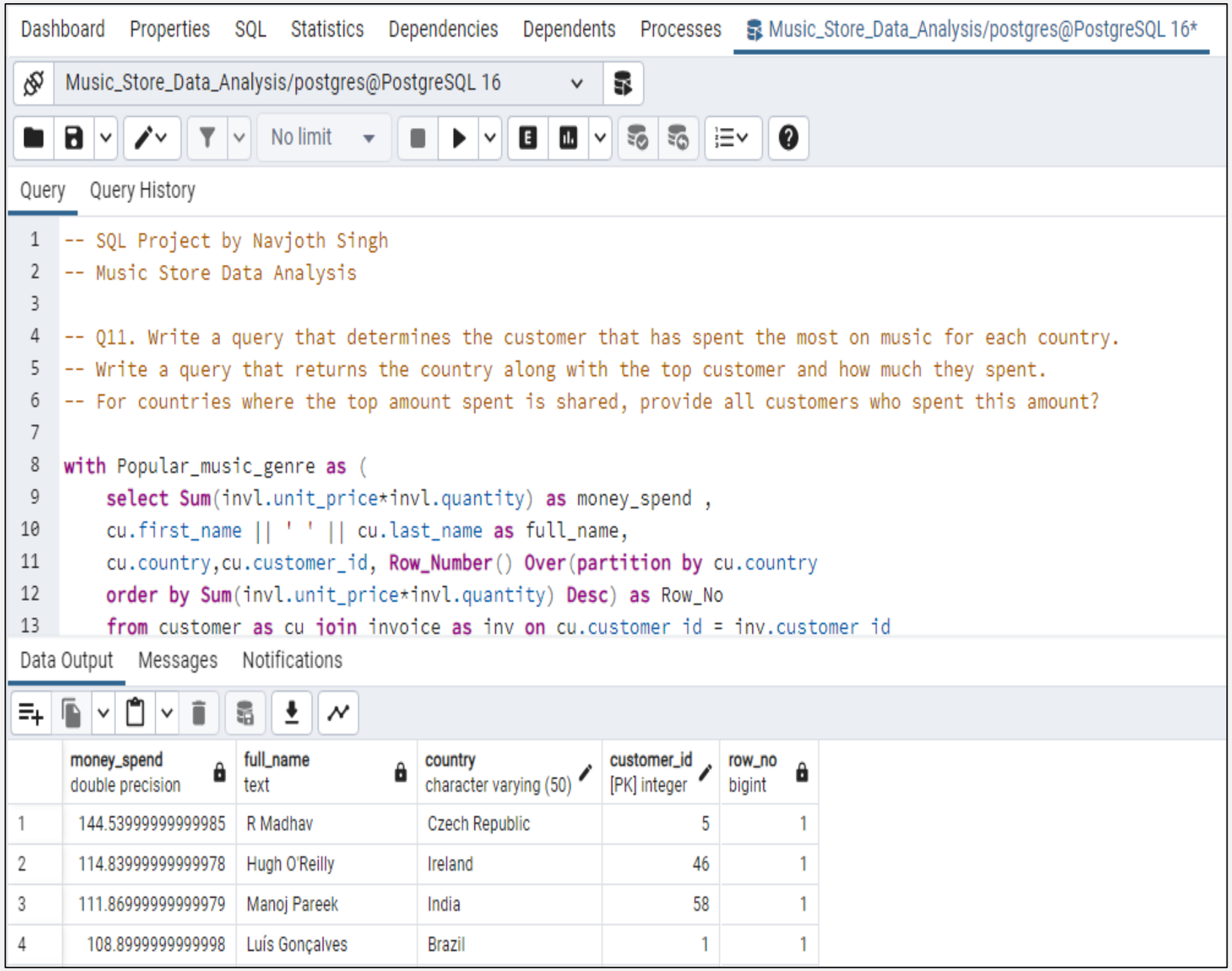
SQL Project by Navjoth Singh – 09-Oct.-2023 – PostgreSQL

Q11: Write a query that determines the customer that has spent the most on music for each country. Write a query that returns the country along with the top customer and how much they spent. For countries where the top amount spent is shared, provide all customers who spent this amount?

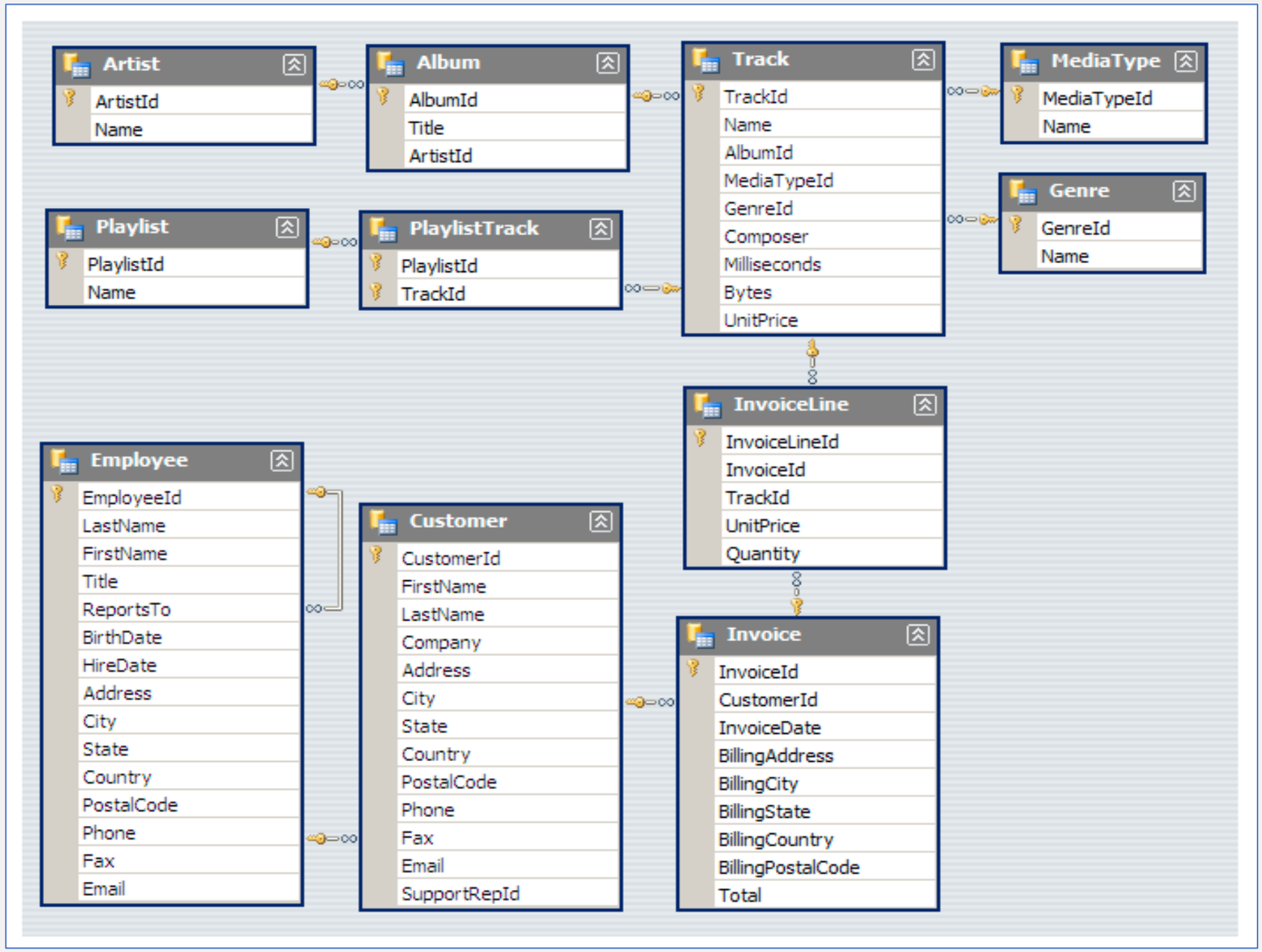
SQL Query

- with Popular_music_genre as (
  - select Sum(invl.unit_price*invl.quantity) as money_spend ,
  - cu.first_name || ' ' || cu.last_name as full_name,
  - cu.country,cu.customer_id, Row_Number() Over(partition by cu.country
  - order by Sum(invl.unit_price*invl.quantity) Desc) as Row_No
  - from customer as cu join invoice as inv on cu.customer_id = inv.customer_id
  - join invoice_line as invl on invl.invoice_id = inv.invoice_id
  - group by 2,3,4
  - order by 2 Asc, 1 Desc)
- select * from Popular_music_genre as pmg
- where Row_No = 1
- order by pmg.money_spend Desc

Data Output



Schema Relationship Diagram Between 11 Tables



Question Set 1 - Easy

1. Who is the senior most employee based on job title?

2. Which countries have the most Invoices?

3. Which artists have top 3 values of total invoice?

4. Which city has the best customers? We would like to throw a promotional Music Festival in the city we made the most money. Write a query that returns one city that has the highest sum of invoice totals. Return both the city name & sum of all invoice totals?

5. Who is the best customer? The customer who has spent the most money will be declared the best customer. Write a query that returns the person who has spent the most money?

Question Set 2 – Moderate

6. Write query to return the email, first name, last name, & Genre of all Rock Music listeners. Return your list ordered alphabetically by email starting with A?

7. 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?

8. Return all the track names that have a song length longer than the average song length. Return the Name and Milliseconds for each track. Order by the song length with the longest songs listed first?

Question Set 3 – Advance

9. Find how much amount spent by each customer on artists? Write a query to return customer name, artist name and total spent?

10. 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 Quantity 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?

11. Write a query that determines the customer that has spent the most on music for each country. Write a query that returns the country along with the top customer and how much they spent. For countries where the top amount spent is shared, provide all customers who spent this amount?