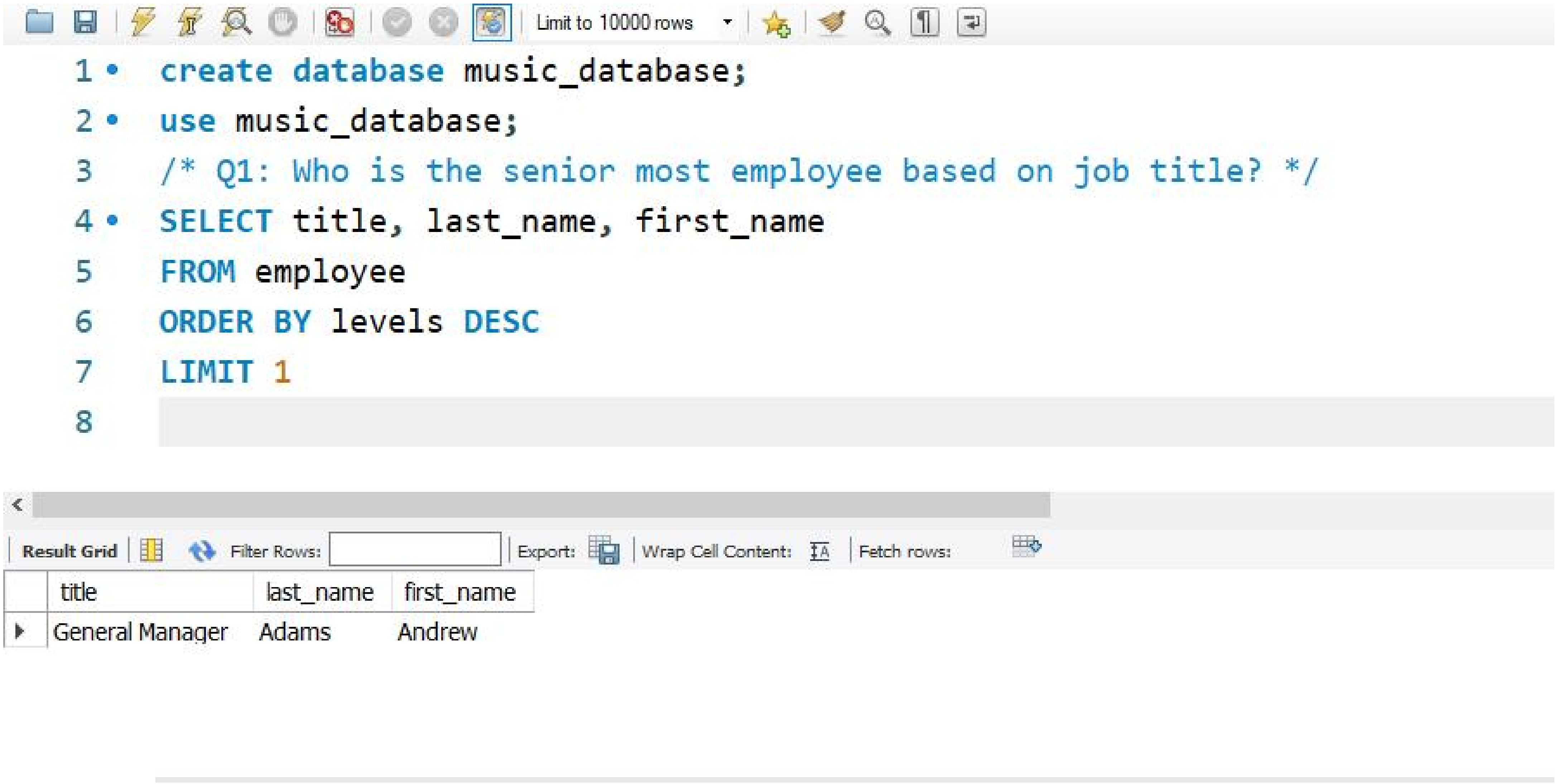
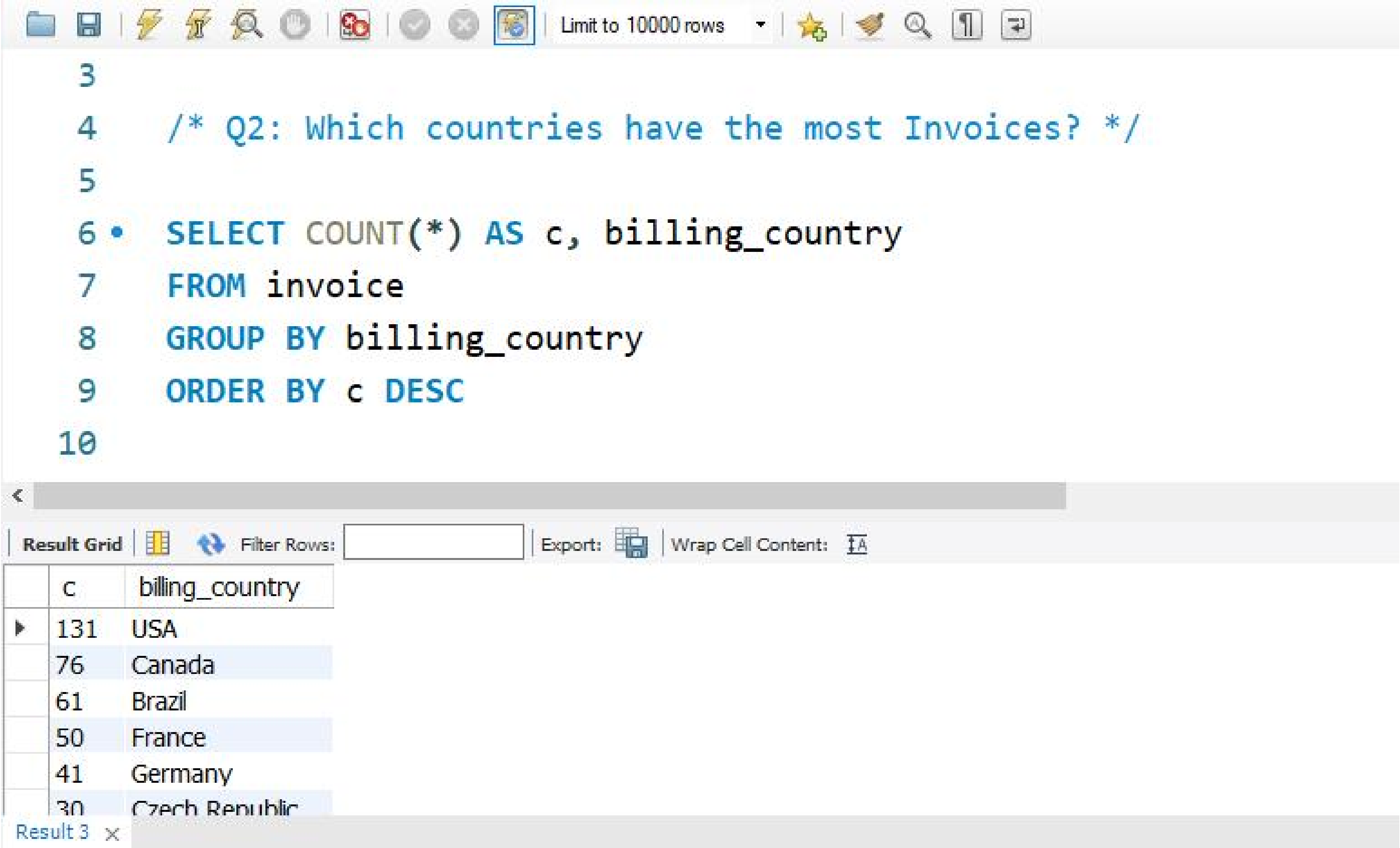


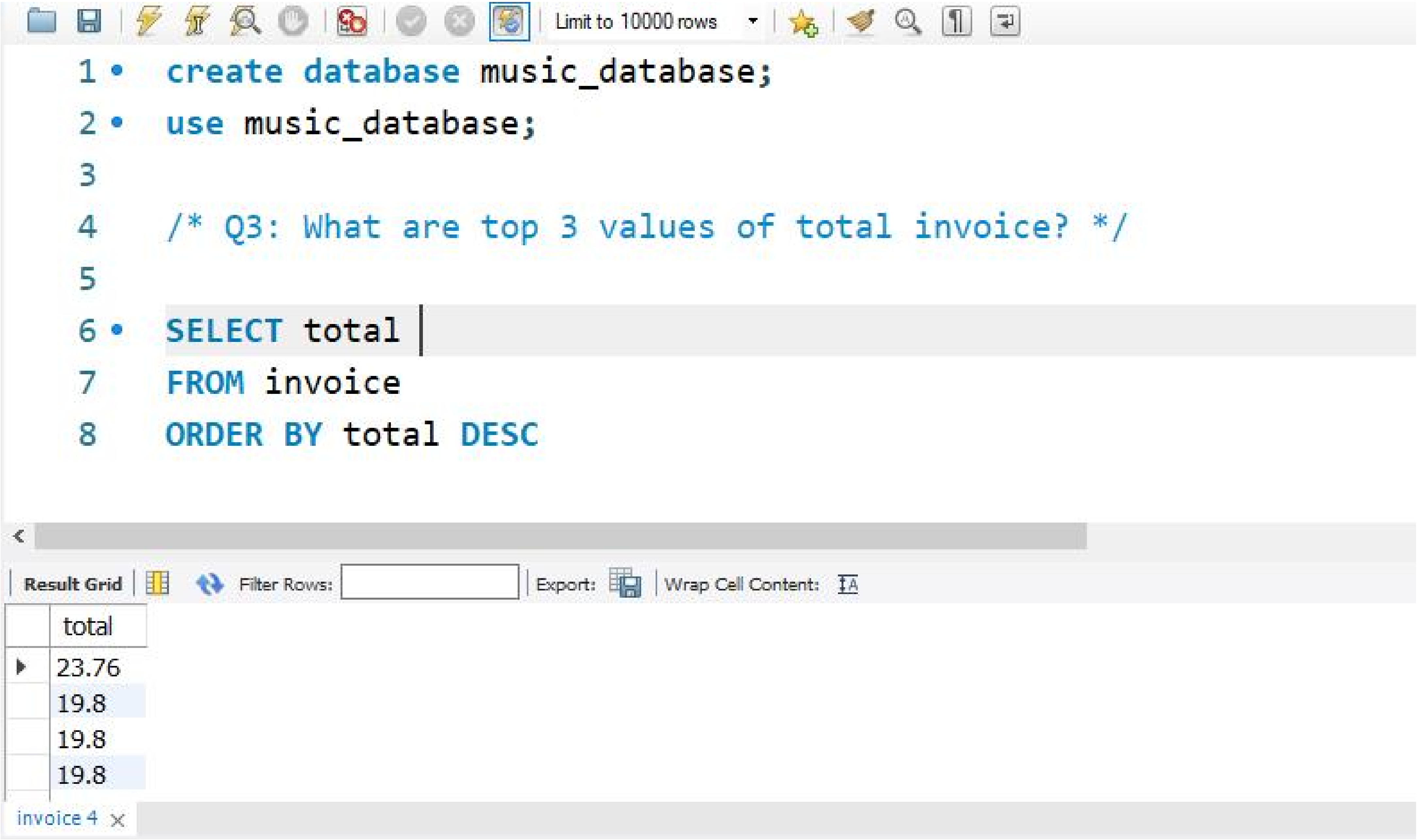
```sql
create database music_database;
use music_database;
/* Q1: Who is the senior most employee based on job title? */
SELECT title, last_name, first_name
FROM employee
ORDER BY levels DESC
LIMIT 1

```

title	last_name	first_name
General Manager	Adams	Andrew



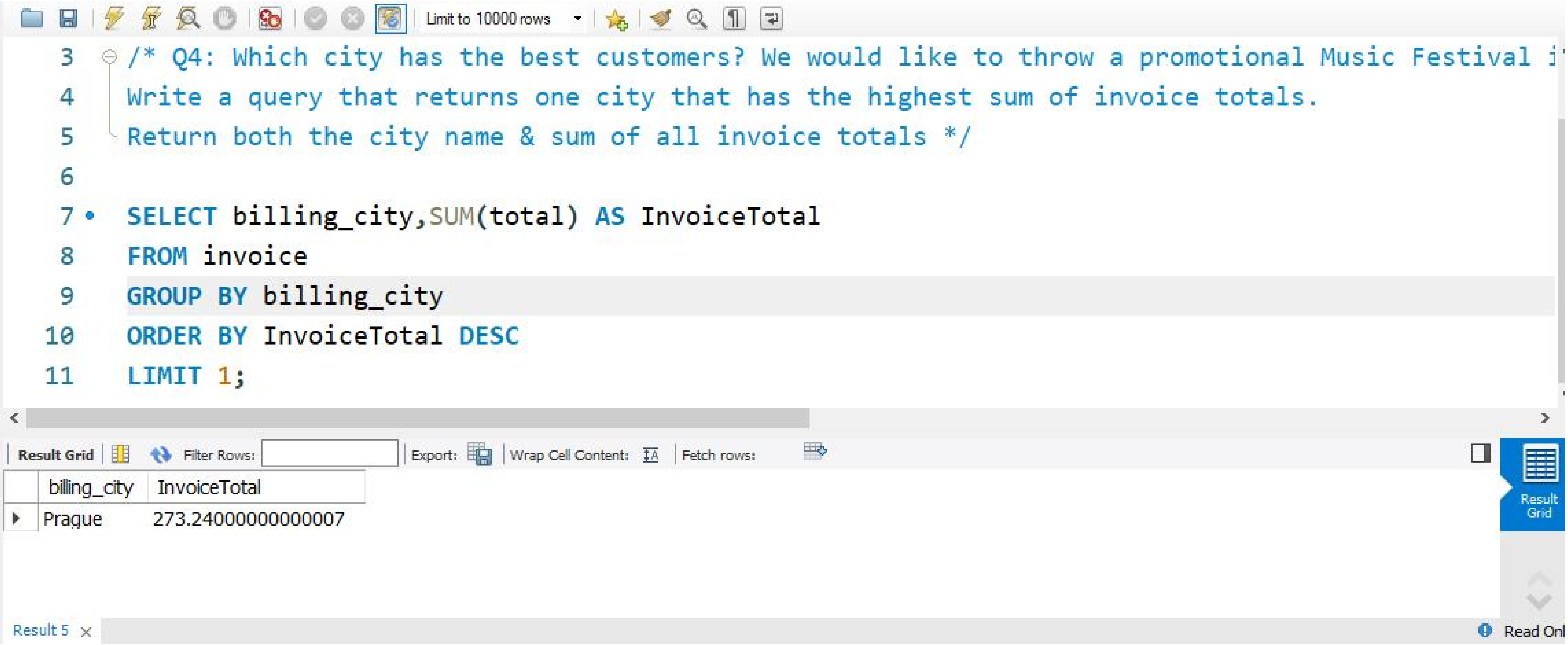
```sql
/* Q2: Which countries have the most Invoices? */

SELECT COUNT(*) AS c, billing_country
FROM invoice
GROUP BY billing_country
ORDER BY c DESC
```

c	billing_country
131	USA
76	Canada
61	Brazil
50	France
41	Germany
30	Czech Republic



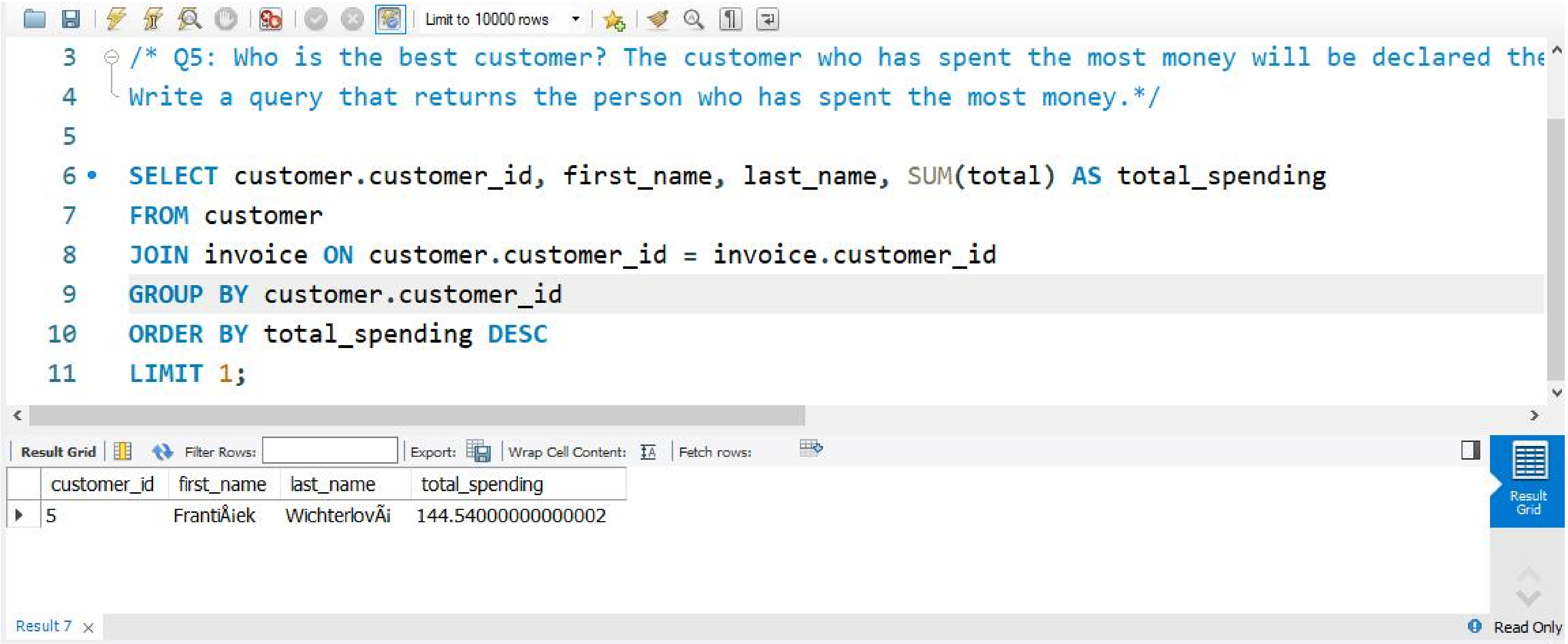
```sql
create database music_database;
use music_database;

/* Q3: What are top 3 values of total invoice? */

SELECT total
FROM invoice
ORDER BY total DESC
```

total
23.76
19.8
19.8
19.8



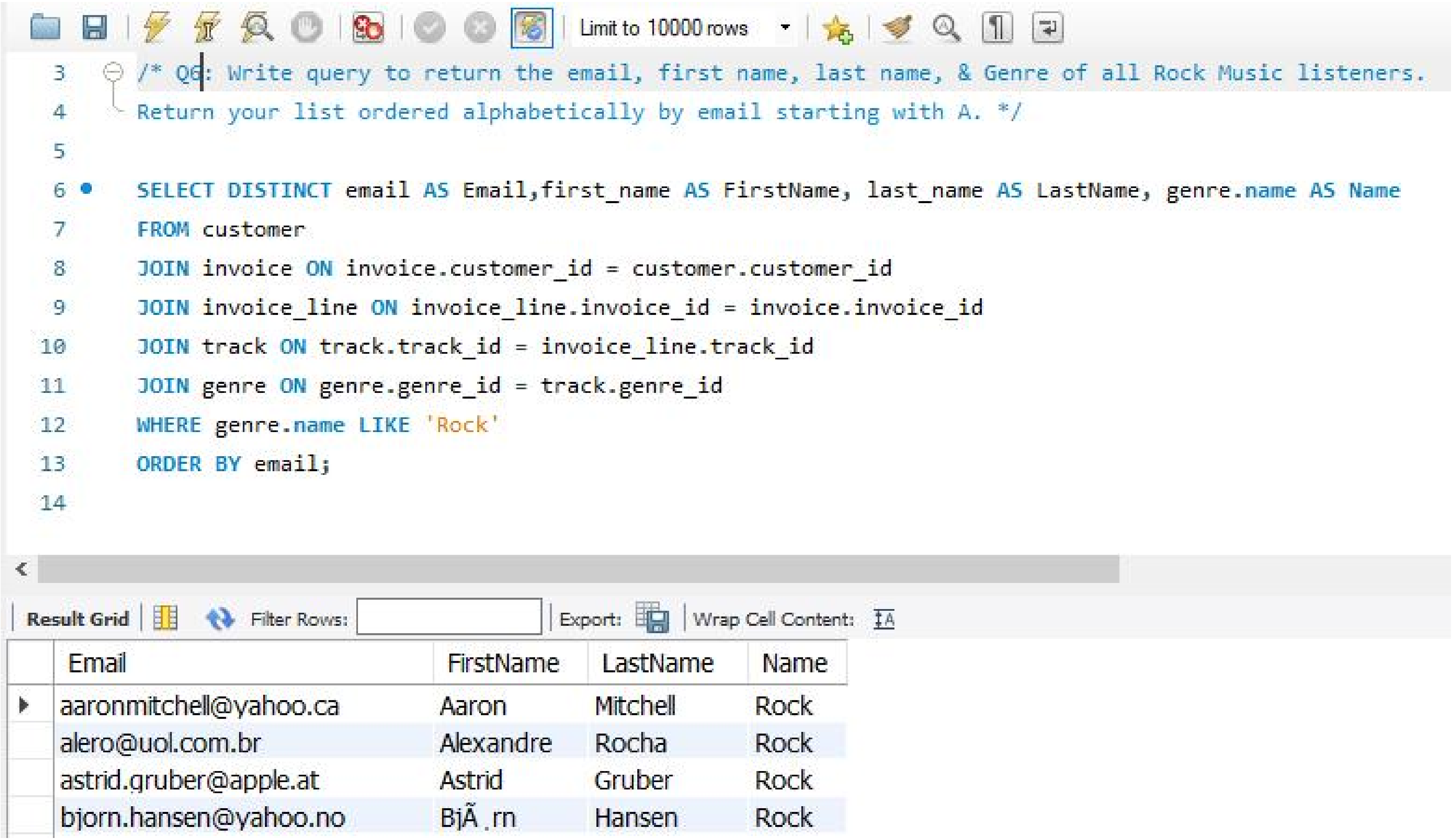
```
/* Q4: Which city has the best customers? We would like to throw a promotional Music Festival i
Write a query that returns one city that has the highest sum of invoice totals.
Return both the city name & sum of all invoice totals */

SELECT billing_city,SUM(total) AS InvoiceTotal
FROM invoice
GROUP BY billing_city
ORDER BY InvoiceTotal DESC
LIMIT 1;
```

billing_city	InvoiceTotal
Prague	273.24000000000007



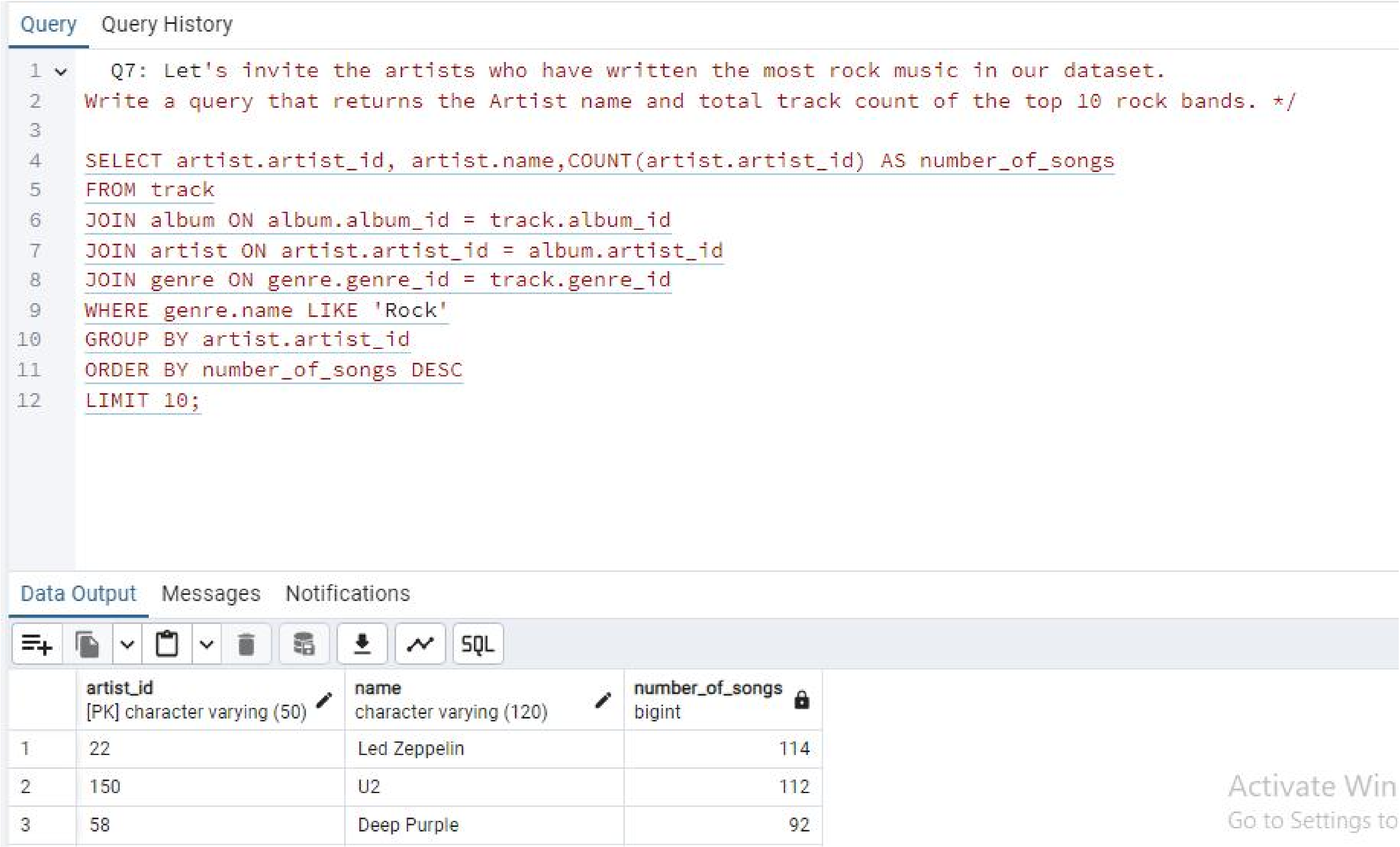
```
/* Q5: Who is the best customer? The customer who has spent the most money will be declared the
Write a query that returns the person who has spent the most money.*/

SELECT customer.customer_id, first_name, last_name, SUM(total) AS total_spending
FROM customer
JOIN invoice ON customer.customer_id = invoice.customer_id
GROUP BY customer.customer_id
ORDER BY total_spending DESC
LIMIT 1;
```

customer_id	first_name	last_name	total_spending
5	FrantiÅiek	WichterlovÃi	144.54000000000002



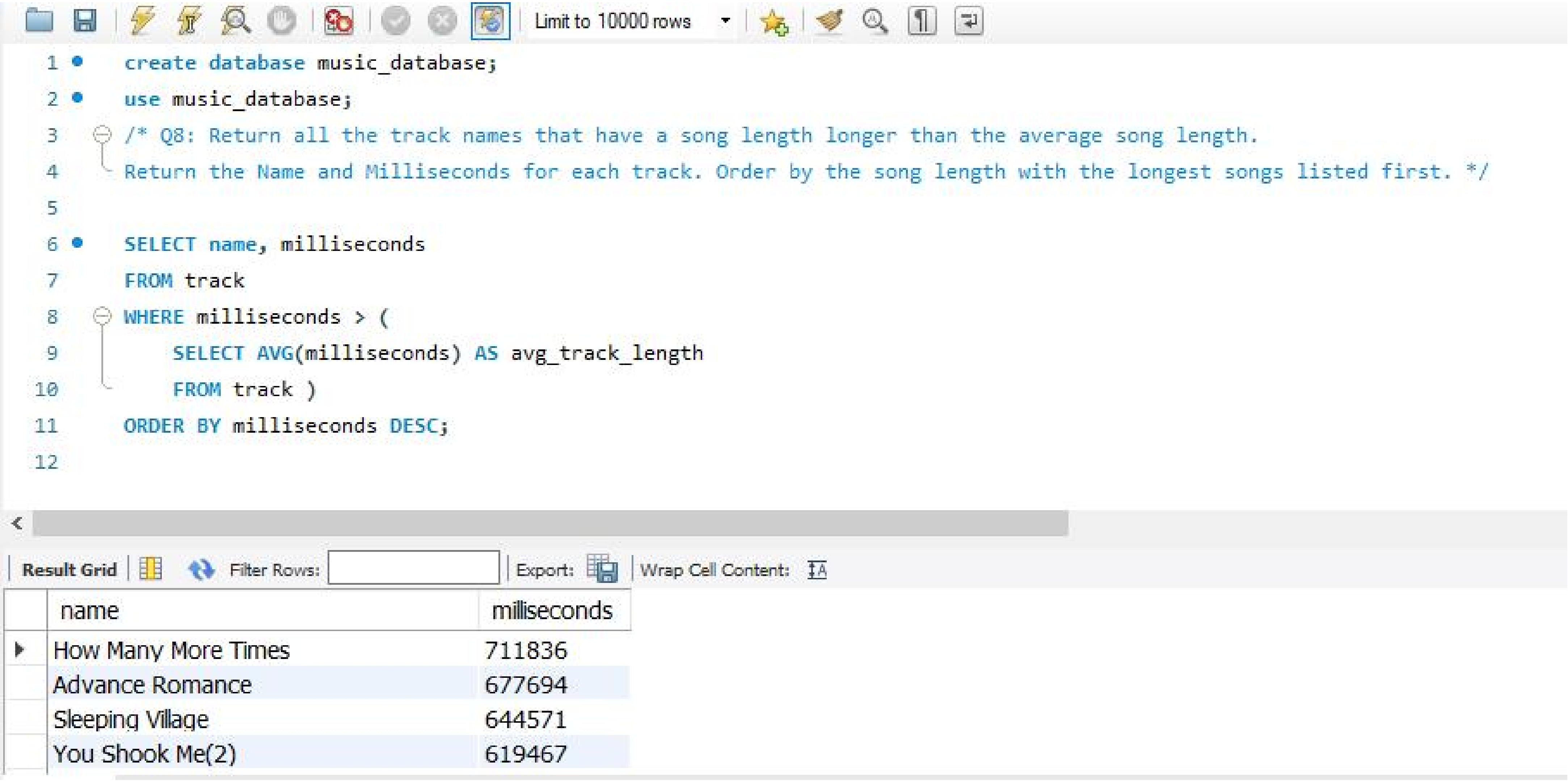
```sql
/* Q6: Write query to return the email, first name, last name, & Genre of all Rock Music listeners.
Return your list ordered alphabetically by email starting with A. */

SELECT DISTINCT email AS Email,first_name AS FirstName, last_name AS LastName, genre.name AS Name
FROM customer
JOIN invoice ON invoice.customer_id = customer.customer_id
JOIN invoice_line ON invoice_line.invoice_id = invoice.invoice_id
JOIN track ON track.track_id = invoice_line.track_id
JOIN genre ON genre.genre_id = track.genre_id
WHERE genre.name LIKE 'Rock'
ORDER BY email;
```

Email	FirstName	LastName	Name
aaronmitchell@yahoo.ca	Aaron	Mitchell	Rock
alero@uol.com.br	Alexandre	Rocha	Rock
astrid.gruber@apple.at	Astrid	Gruber	Rock
bjorn.hansen@yahoo.no	BjÃ¸rn	Hansen	Rock



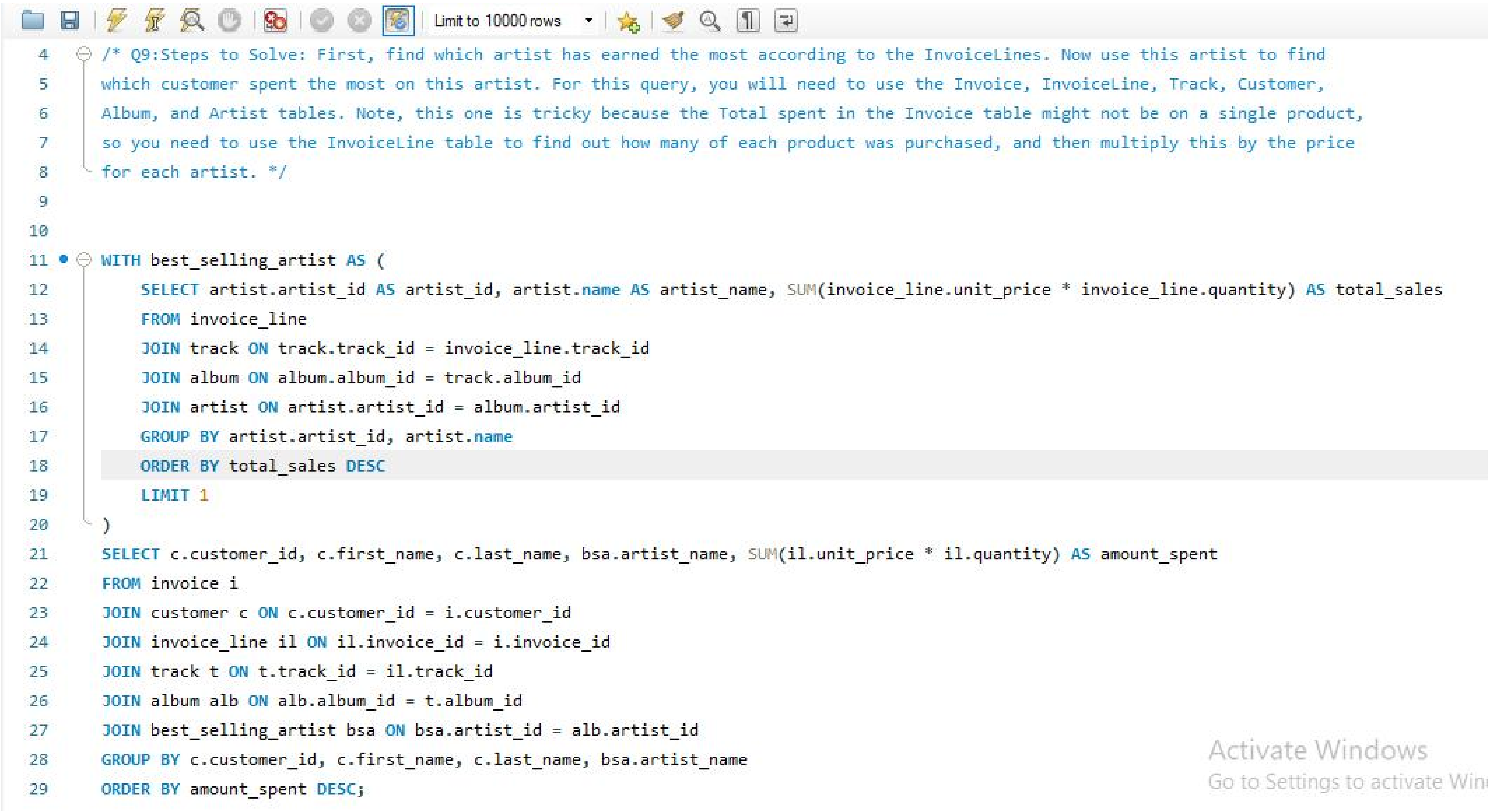
```
  Q7: Let's invite the artists who have written the most rock music in our dataset.
Write a query that returns the Artist name and total track count of the top 10 rock bands. */

SELECT artist.artist_id, artist.name,COUNT(artist.artist_id) AS number_of_songs
FROM track
JOIN album ON album.album_id = track.album_id
JOIN artist ON artist.artist_id = album.artist_id
JOIN genre ON genre.genre_id = track.genre_id
WHERE genre.name LIKE 'Rock'
GROUP BY artist.artist_id
ORDER BY number_of_songs DESC
LIMIT 10;
```

	artist_id [PK] character varying (50)	name character varying (120)	number_of_songs bigint
1	22	Led Zeppelin	114
2	150	U2	112
3	58	Deep Purple	92

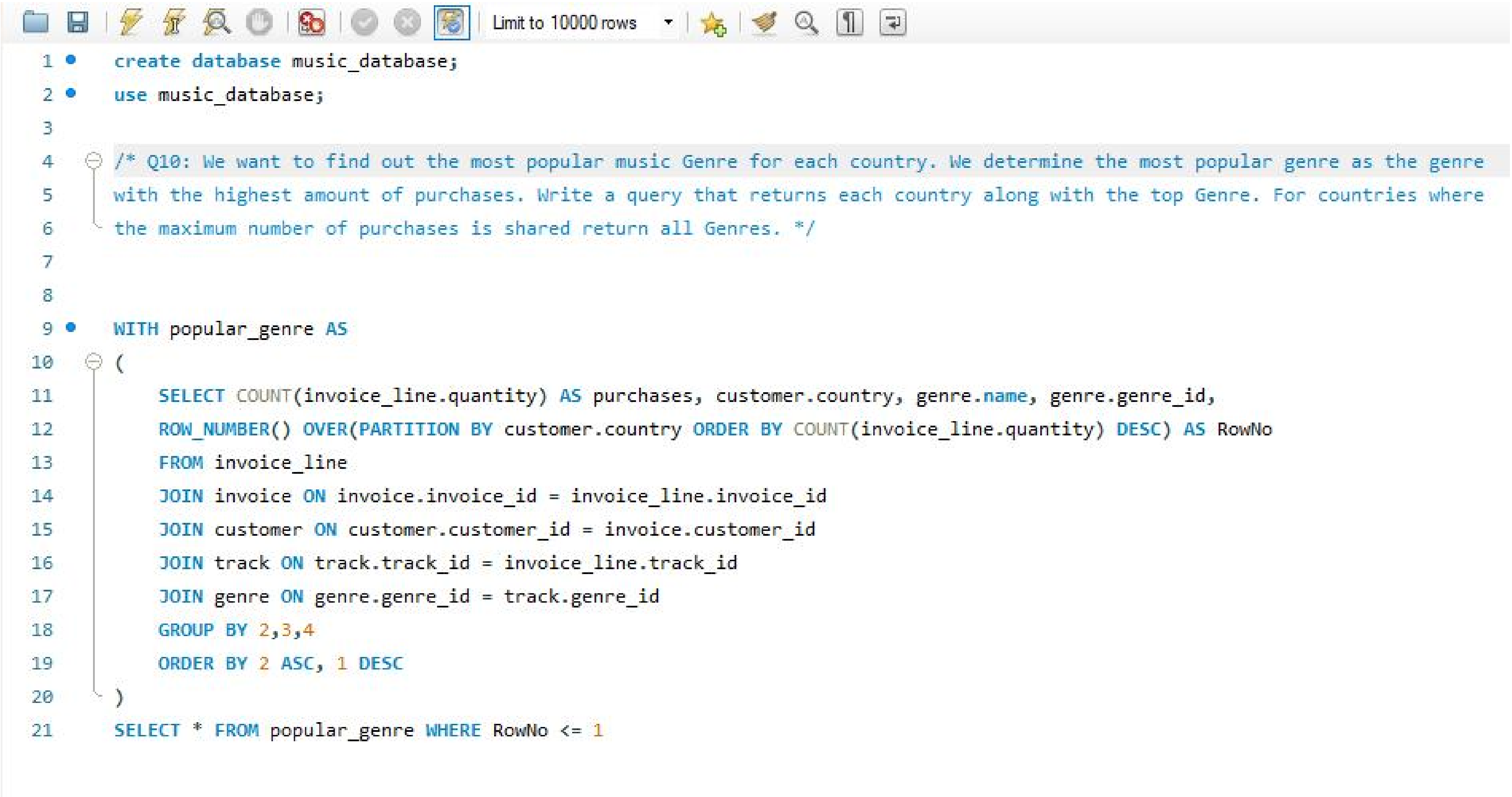


```
create database music_database;
use music_database;
/* Q8: Return all the track names that have a song length longer than the average song length.
Return the Name and Milliseconds for each track. Order by the song length with the longest songs listed first. */

SELECT name, milliseconds
FROM track
WHERE milliseconds > (
    SELECT AVG(milliseconds) AS avg_track_length
    FROM track )
ORDER BY milliseconds DESC;
```

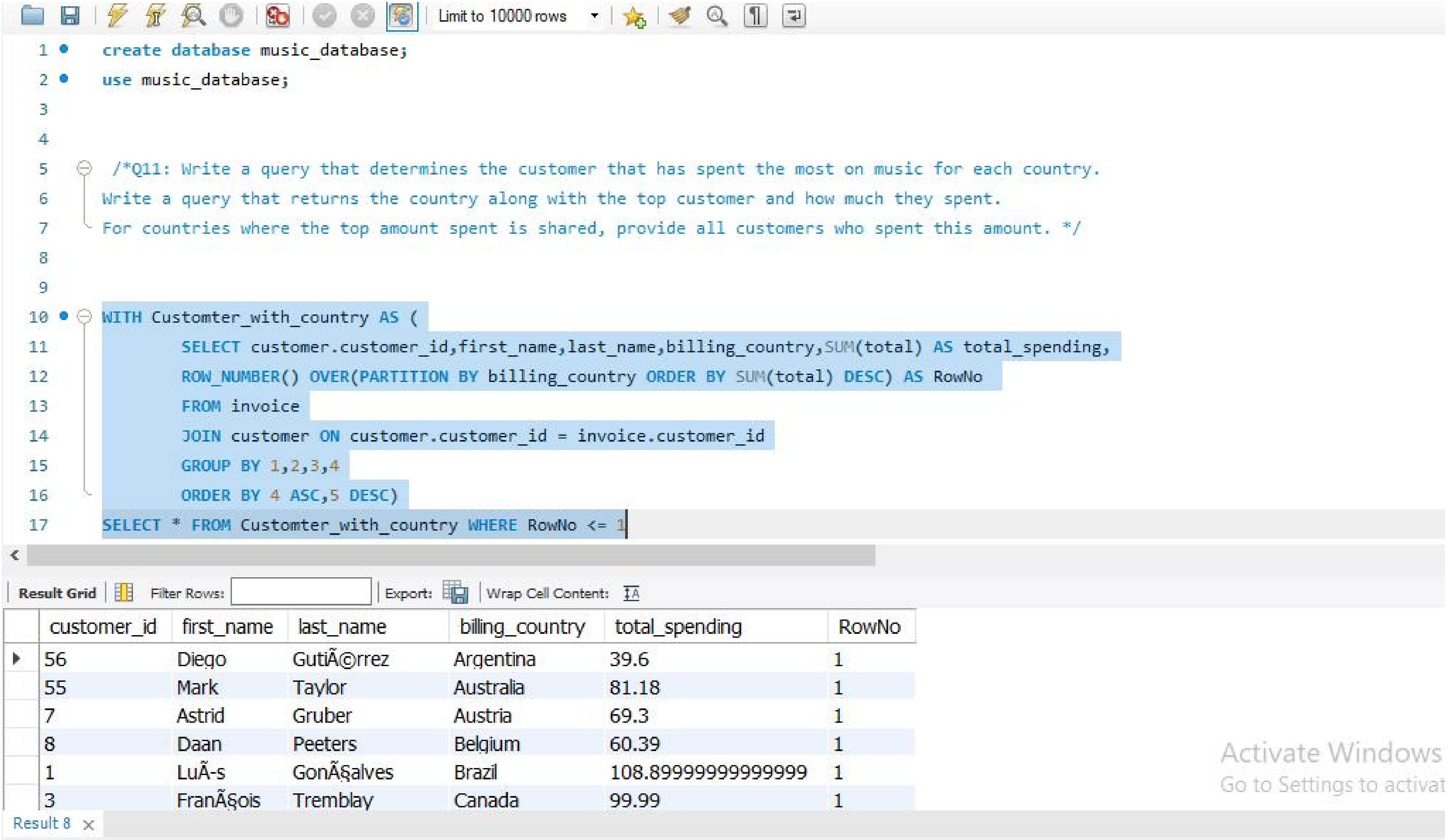
name	milliseconds
How Many More Times	711836
Advance Romance	677694
Sleeping Village	644571
You Shook Me(2)	619467



```sql
/* Q9:Steps to Solve: First, find which artist has earned the most according to the InvoiceLines. Now use this artist to find
which customer spent the most on this artist. For this query, you will need to use the Invoice, InvoiceLine, Track, Customer,
Album, and Artist tables. Note, this one is tricky because the Total spent in the Invoice table might not be on a single product,
so you need to use the InvoiceLine table to find out how many of each product was purchased, and then multiply this by the price
for each artist. */


WITH best_selling_artist AS (
    SELECT artist.artist_id AS artist_id, artist.name AS artist_name, SUM(invoice_line.unit_price * invoice_line.quantity) AS total_sales
    FROM invoice_line
    JOIN track ON track.track_id = invoice_line.track_id
    JOIN album ON album.album_id = track.album_id
    JOIN artist ON artist.artist_id = album.artist_id
    GROUP BY artist.artist_id, artist.name
    ORDER BY total_sales DESC
    LIMIT 1
)
SELECT c.customer_id, c.first_name, c.last_name, bsa.artist_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 best_selling_artist bsa ON bsa.artist_id = alb.artist_id
GROUP BY c.customer_id, c.first_name, c.last_name, bsa.artist_name
ORDER BY amount_spent DESC;
```



```
create database music_database;
use music_database;

/* 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 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 2 ASC, 1 DESC
)
SELECT * FROM popular_genre WHERE RowNo <= 1
```



```
create database music_database;
use music_database;


 /*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. */


WITH Customter_with_country AS (
        SELECT customer.customer_id,first_name,last_name,billing_country,SUM(total) AS total_spending,
        ROW_NUMBER() OVER(PARTITION BY billing_country ORDER BY SUM(total) DESC) AS RowNo
        FROM invoice
        JOIN customer ON customer.customer_id = invoice.customer_id
        GROUP BY 1,2,3,4
        ORDER BY 4 ASC,5 DESC)
SELECT * FROM Customter_with_country WHERE RowNo <= 1
```

customer_id	first_name	last_name	billing_country	total_spending	RowNo
56	Diego	GutiÃ©rrez	Argentina	39.6	1
55	Mark	Taylor	Australia	81.18	1
7	Astrid	Gruber	Austria	69.3	1
8	Daan	Peeters	Belgium	60.39	1
1	LuÃ-s	GonÃ§alves	Brazil	108.89999999999999	1
3	FranÃ§ois	Tremblay	Canada	99.99	1