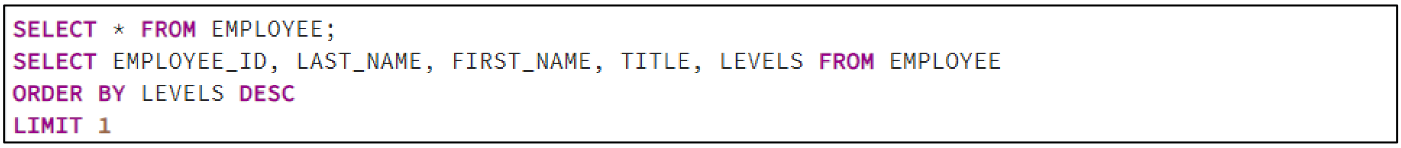
SQL PROJECT- MUSIC STORE DATA ANALYSIS

Question Set 1 - Easy

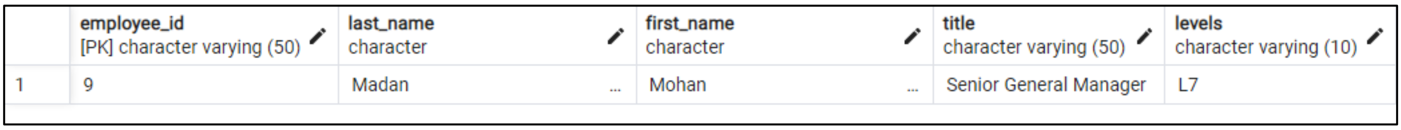
1. Who is the senior most employee based on job title?

QUERY:



```sql
SELECT * FROM EMPLOYEE;
SELECT EMPLOYEE_ID, LAST_NAME, FIRST_NAME, TITLE, LEVELS FROM EMPLOYEE
ORDER BY LEVELS DESC
LIMIT 1
```

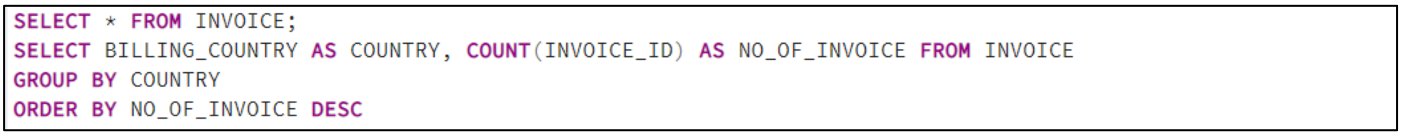
OUTPUT:



	employee_id [PK] character varying (50)	last_name character	first_name character	title character varying (50)	levels character varying (10)
1	9	Madan	Mohan	Senior General Manager	L7

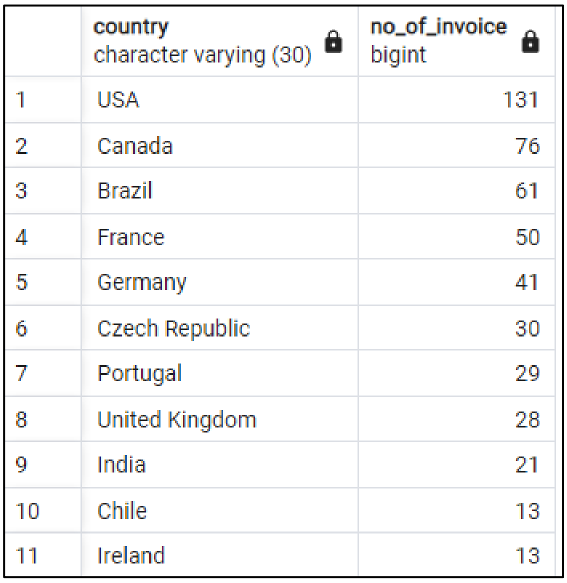
2. Which countries have the most Invoices?

QUERY:



```sql
SELECT * FROM INVOICE;
SELECT BILLING_COUNTRY AS COUNTRY, COUNT(INVOICE_ID) AS NO_OF_INVOICE FROM INVOICE
GROUP BY COUNTRY
ORDER BY NO_OF_INVOICE DESC
```

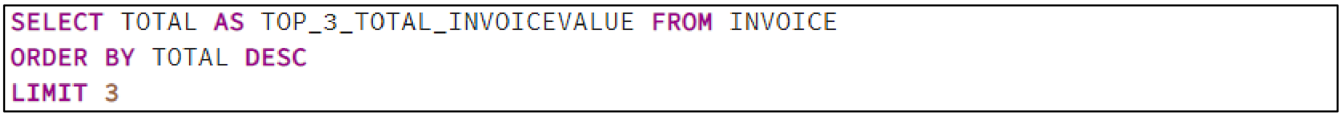
OUTPUT:



	country character varying (30)	no_of_invoice bigint
1	USA	131
2	Canada	76
3	Brazil	61
4	France	50
5	Germany	41
6	Czech Republic	30
7	Portugal	29
8	United Kingdom	28
9	India	21
10	Chile	13
11	Ireland	13

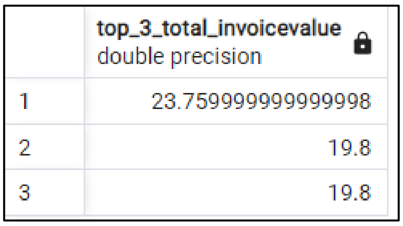
3. What are top 3 values of total invoice?

QUERY:



```
SELECT TOTAL AS TOP_3_TOTAL_INVOICEVALUE FROM INVOICE
ORDER BY TOTAL DESC
LIMIT 3
```

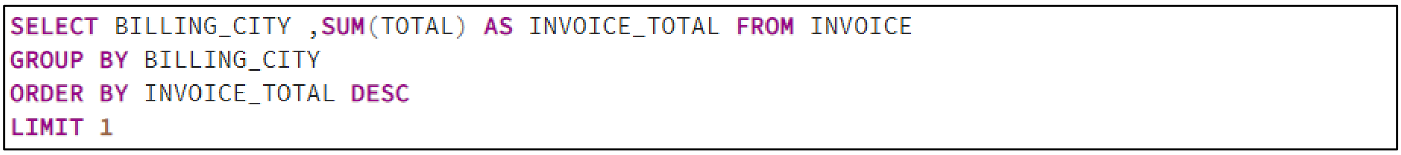
OUTPUT:



	top_3_total_invoicevalue double precision
1	23.759999999999998
2	19.8
3	19.8

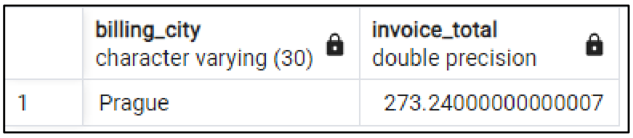
4. Which city has the best customers? We would like to throw a promotional Music Festival in the city we made the most money. Write a query that returns one city that has the highest sum of invoice totals. Return both the city name & sum of all invoice totals

QUERY:



```
SELECT BILLING_CITY ,SUM(TOTAL) AS INVOICE_TOTAL FROM INVOICE
GROUP BY BILLING_CITY
ORDER BY INVOICE_TOTAL DESC
LIMIT 1
```

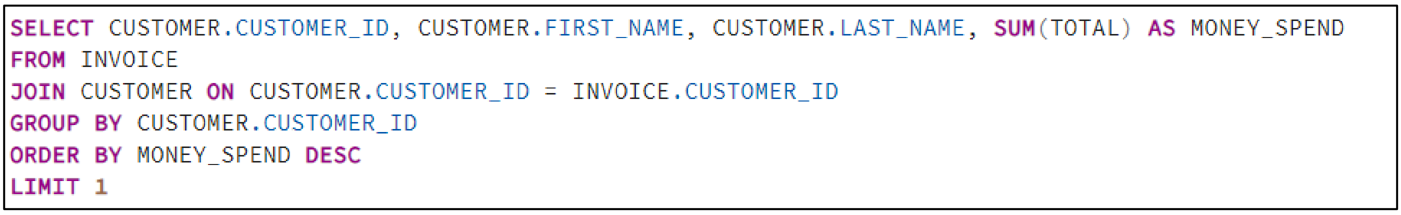
OUTPUT:



	billing_city character varying (30)	invoice_total double precision
1	Prague	273.24000000000007

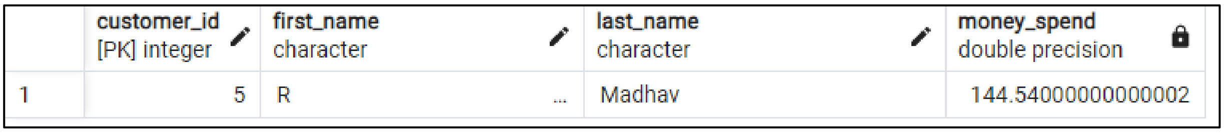
5. Who is the best customer? The customer who has spent the most money will be declared the best customer. Write a query that returns the person who has spent the most money

QUERY:



```
SELECT CUSTOMER.CUSTOMER_ID, CUSTOMER.FIRST_NAME, CUSTOMER.LAST_NAME, SUM(TOTAL) AS MONEY_SPEND
FROM INVOICE
JOIN CUSTOMER ON CUSTOMER.CUSTOMER_ID = INVOICE.CUSTOMER_ID
GROUP BY CUSTOMER.CUSTOMER_ID
ORDER BY MONEY_SPEND DESC
LIMIT 1
```

OUTPUT:

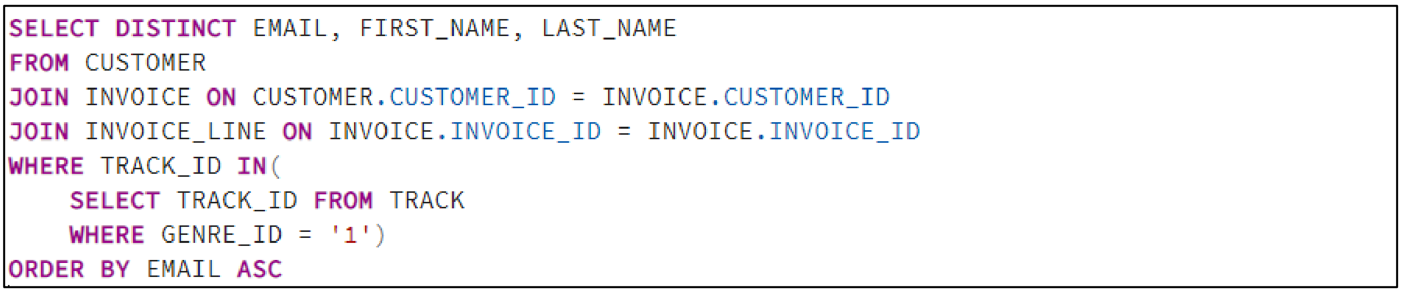


	customer_id [PK] integer	first_name character	last_name character	money_spend double precision
1	5	R ...	Madhav	144.54000000000002

Question Set 2 – Moderate

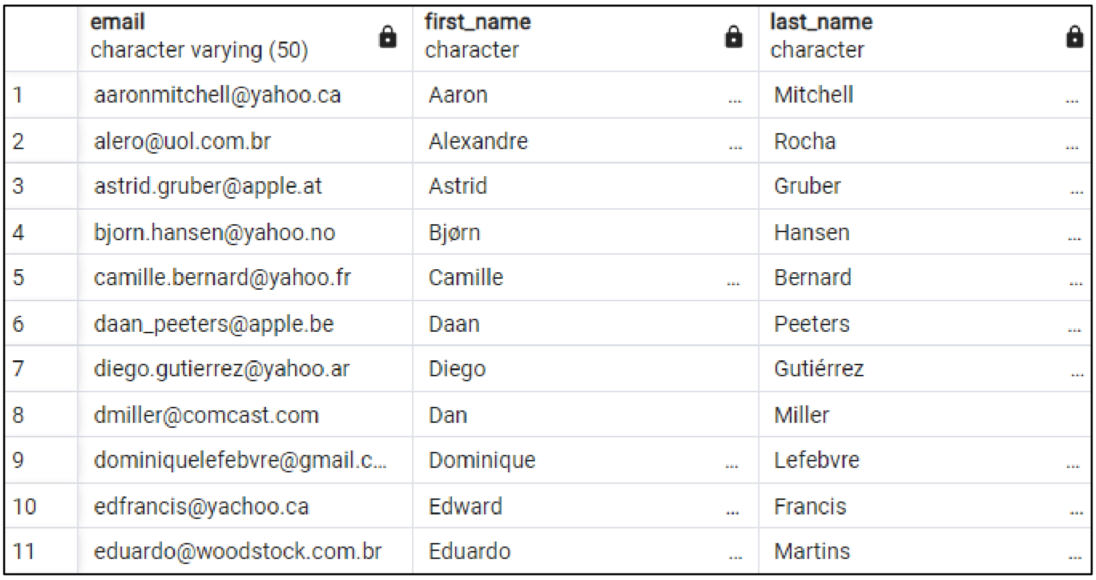
1. Write query to return the email, first name, last name, & Genre of all Rock Music listeners. Return your list ordered alphabetically by email starting with A

QUERY:



```
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.INVOICE_ID
WHERE TRACK_ID IN(
    SELECT TRACK_ID FROM TRACK
    WHERE GENRE_ID = '1')
ORDER BY EMAIL ASC
```

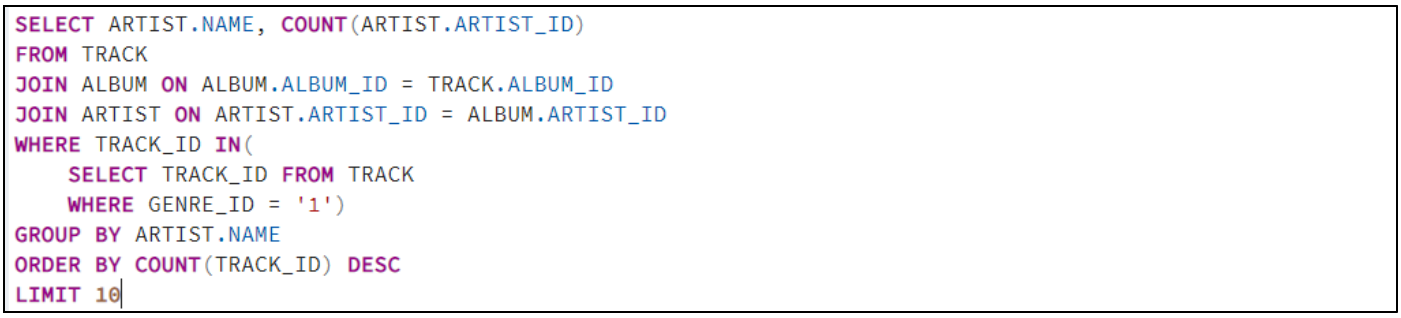
OUTPUT:



	email character varying (50)	first_name character	last_name character
1	aaronmitchell@yahoo.ca	Aaron	Mitchell
2	alero@uol.com.br	Alexandre	Rocha
3	astrid.gruber@apple.at	Astrid	Gruber
4	bjorn.hansen@yahoo.no	Bjørn	Hansen
5	camille.bernard@yahoo.fr	Camille	Bernard
6	daan_peeters@apple.be	Daan	Peeters
7	diego.gutierrez@yahoo.ar	Diego	Gutiérrez
8	dmiller@comcast.com	Dan	Miller
9	dominiquelefebvre@gmail.c...	Dominique	Lefebvre
10	edfrancis@yachoo.ca	Edward	Francis
11	eduardo@woodstock.com.br	Eduardo	Martins

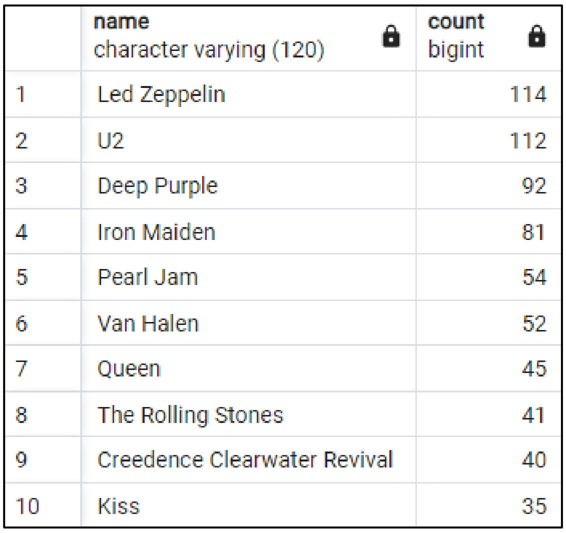
2. Let's invite the artists who have written the most rock music in our dataset. Write a query that returns the Artist name and total track count of the top 10 rock bands

QUERY:



```
SELECT ARTIST.NAME, COUNT(ARTIST.ARTIST_ID)
FROM TRACK
JOIN ALBUM ON ALBUM.ALBUM_ID = TRACK.ALBUM_ID
JOIN ARTIST ON ARTIST.ARTIST_ID = ALBUM.ARTIST_ID
WHERE TRACK_ID IN(
    SELECT TRACK_ID FROM TRACK
    WHERE GENRE_ID = '1')
GROUP BY ARTIST.NAME
ORDER BY COUNT(TRACK_ID) DESC
LIMIT 10
```

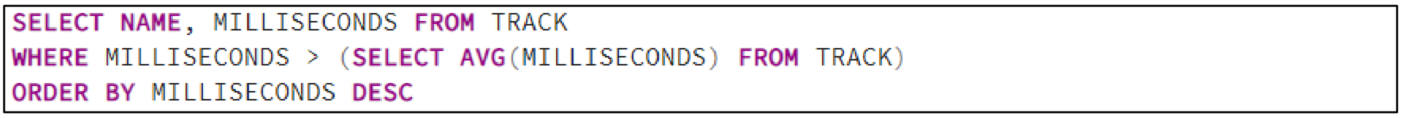
OUTPUT:



	name character varying (120)	count bigint
1	Led Zeppelin	114
2	U2	112
3	Deep Purple	92
4	Iron Maiden	81
5	Pearl Jam	54
6	Van Halen	52
7	Queen	45
8	The Rolling Stones	41
9	Creedence Clearwater Revival	40
10	Kiss	35

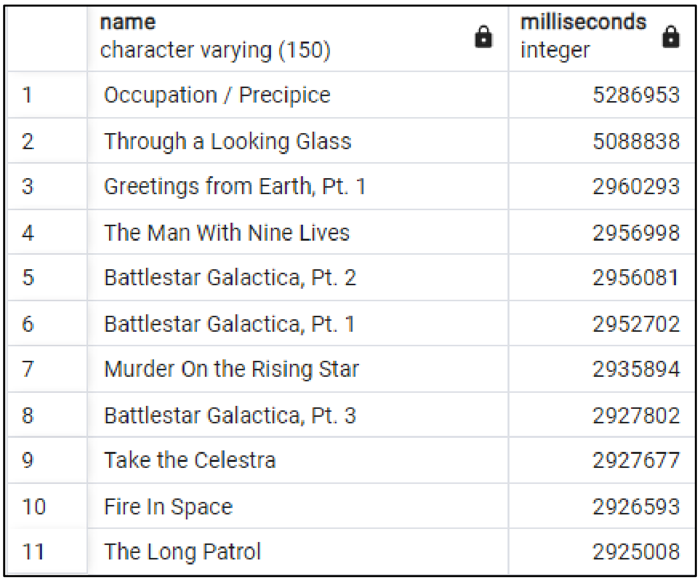
3. Return all the track names that have a song length longer than the average song length. Return the Name and Milliseconds for each track. Order by the song length with the longest songs listed first

QUERY:



```
SELECT NAME, MILLISECONDS FROM TRACK
WHERE MILLISECONDS > (SELECT AVG(MILLISECONDS) FROM TRACK)
ORDER BY MILLISECONDS DESC
```

OUTPUT:

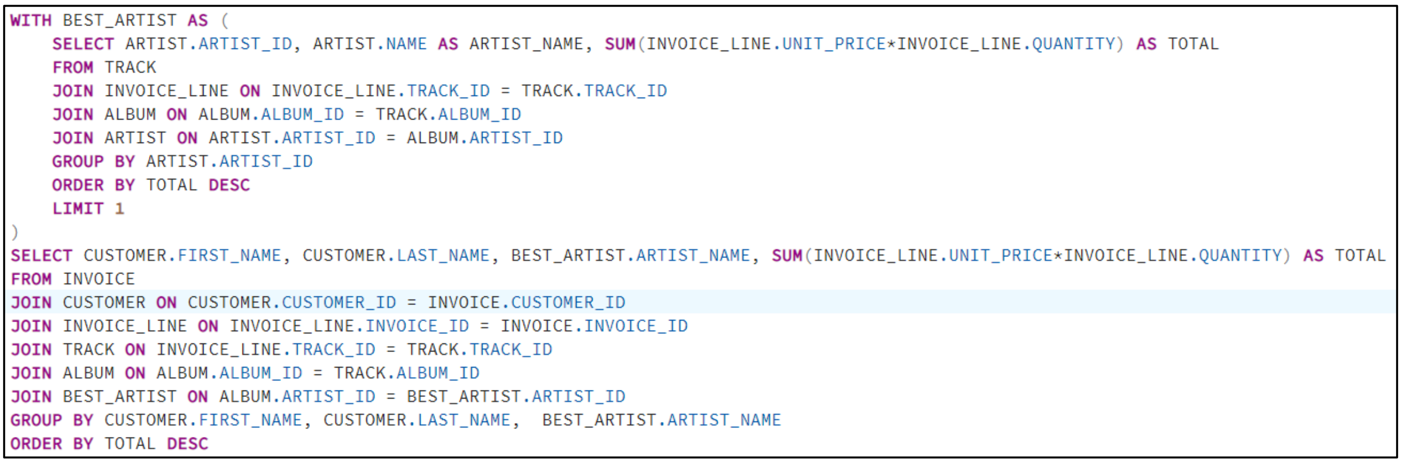


	name character varying (150)	milliseconds integer
1	Occupation / Precipice	5286953
2	Through a Looking Glass	5088838
3	Greetings from Earth, Pt. 1	2960293
4	The Man With Nine Lives	2956998
5	Battlestar Galactica, Pt. 2	2956081
6	Battlestar Galactica, Pt. 1	2952702
7	Murder On the Rising Star	2935894
8	Battlestar Galactica, Pt. 3	2927802
9	Take the Celestra	2927677
10	Fire In Space	2926593
11	The Long Patrol	2925008

Question Set 3 – Advance

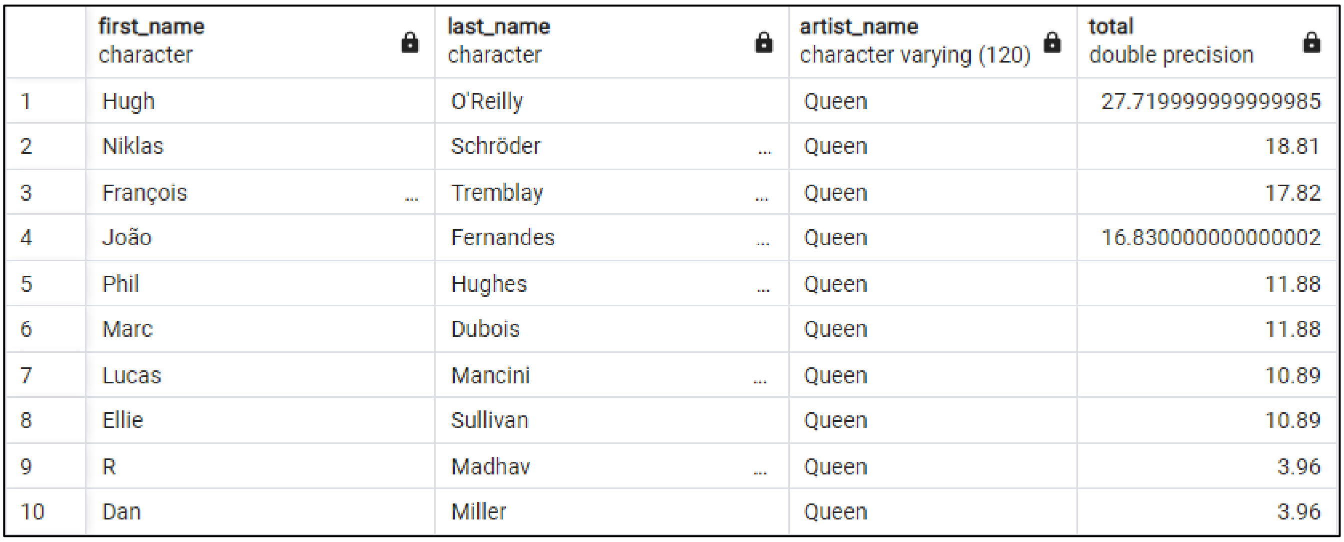
1. Find how much amount spent by each customer on artists? Write a query to return customer name, artist name and total spent

QUERY:



```
WITH BEST_ARTIST AS (
    SELECT ARTIST.ARTIST_ID, ARTIST.NAME AS ARTIST_NAME, SUM(INVOICE_LINE.UNIT_PRICE*INVOICE_LINE.QUANTITY) AS TOTAL
    FROM TRACK
    JOIN INVOICE_LINE ON INVOICE_LINE.TRACK_ID = TRACK.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
    ORDER BY TOTAL DESC
    LIMIT 1
)
SELECT CUSTOMER.FIRST_NAME, CUSTOMER.LAST_NAME, BEST_ARTIST.ARTIST_NAME, SUM(INVOICE_LINE.UNIT_PRICE*INVOICE_LINE.QUANTITY) AS TOTAL
FROM INVOICE
JOIN CUSTOMER ON CUSTOMER.CUSTOMER_ID = INVOICE.CUSTOMER_ID
JOIN INVOICE_LINE ON INVOICE_LINE.INVOICE_ID = INVOICE.INVOICE_ID
JOIN TRACK ON INVOICE_LINE.TRACK_ID = TRACK.TRACK_ID
JOIN ALBUM ON ALBUM.ALBUM_ID = TRACK.ALBUM_ID
JOIN BEST_ARTIST ON ALBUM.ARTIST_ID = BEST_ARTIST.ARTIST_ID
GROUP BY CUSTOMER.FIRST_NAME, CUSTOMER.LAST_NAME,  BEST_ARTIST.ARTIST_NAME
ORDER BY TOTAL DESC
```

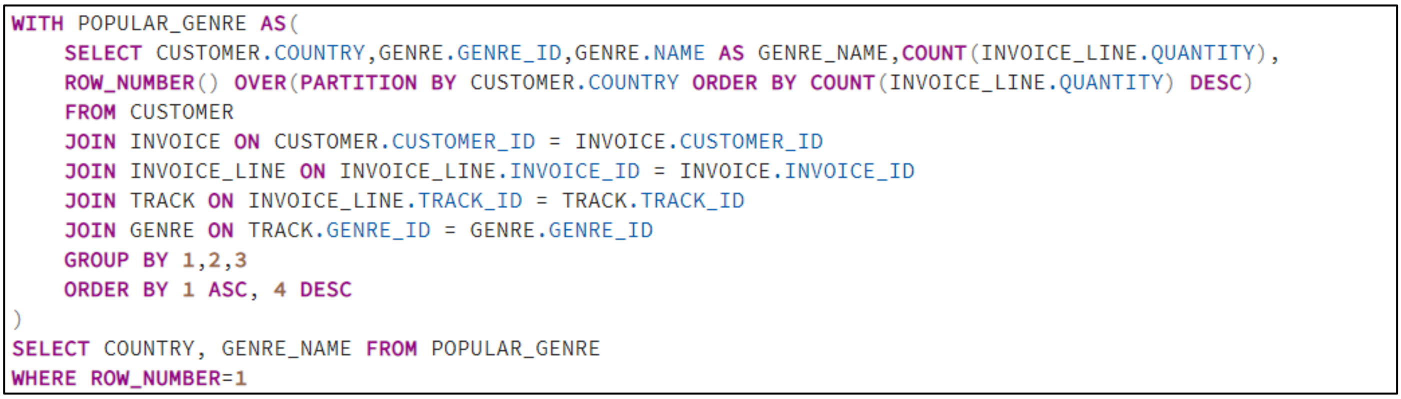
OUTPUT:



	first_name character	last_name character	artist_name character varying (120)	total double precision
1	Hugh	O'Reilly	Queen	27.719999999999985
2	Niklas	Schröder	Queen	18.81
3	François	Tremblay	Queen	17.82
4	João	Fernandes	Queen	16.830000000000002
5	Phil	Hughes	Queen	11.88
6	Marc	Dubois	Queen	11.88
7	Lucas	Mancini	Queen	10.89
8	Ellie	Sullivan	Queen	10.89
9	R	Madhav	Queen	3.96
10	Dan	Miller	Queen	3.96

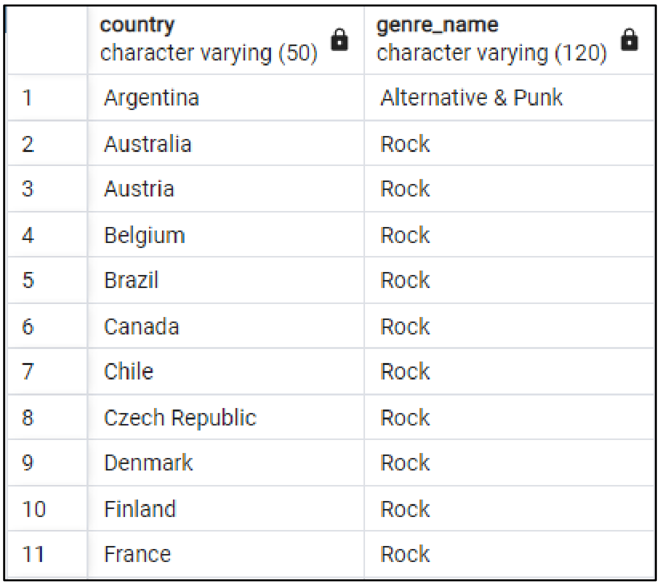
2. We want to find out the most popular music Genre for each country. We determine the most popular genre as the genre with the highest amount of purchases. Write a query that returns each country along with the top Genre. For countries where the maximum number of purchases is shared return all Genres

QUERY:



```
WITH POPULAR_GENRE AS(
    SELECT CUSTOMER.COUNTRY,GENRE.GENRE_ID,GENRE.NAME AS GENRE_NAME,COUNT(INVOICE_LINE.QUANTITY),
    ROW_NUMBER() OVER(PARTITION BY CUSTOMER.COUNTRY ORDER BY COUNT(INVOICE_LINE.QUANTITY) DESC)
    FROM CUSTOMER
    JOIN INVOICE ON CUSTOMER.CUSTOMER_ID = INVOICE.CUSTOMER_ID
    JOIN INVOICE_LINE ON INVOICE_LINE.INVOICE_ID = INVOICE.INVOICE_ID
    JOIN TRACK ON INVOICE_LINE.TRACK_ID = TRACK.TRACK_ID
    JOIN GENRE ON TRACK.GENRE_ID = GENRE.GENRE_ID
    GROUP BY 1,2,3
    ORDER BY 1 ASC, 4 DESC
)
SELECT COUNTRY, GENRE_NAME FROM POPULAR_GENRE
WHERE ROW_NUMBER=1
```

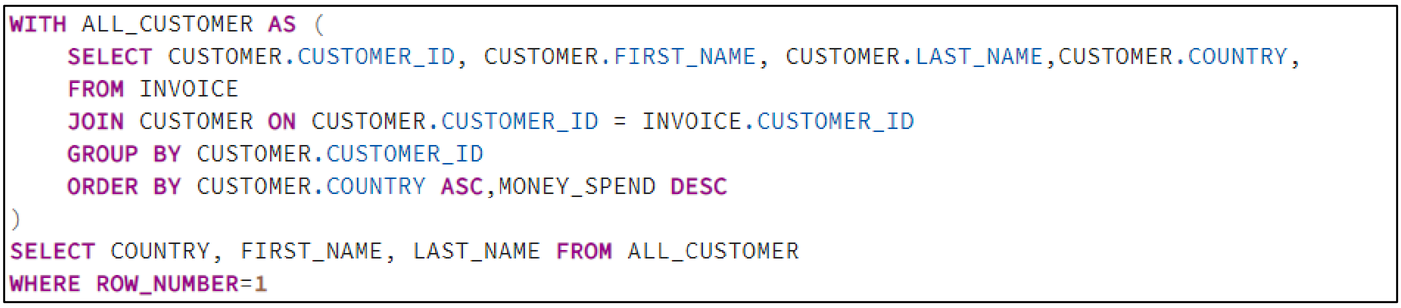
OUTPUT:



	country character varying (50)	genre_name character varying (120)
1	Argentina	Alternative & Punk
2	Australia	Rock
3	Austria	Rock
4	Belgium	Rock
5	Brazil	Rock
6	Canada	Rock
7	Chile	Rock
8	Czech Republic	Rock
9	Denmark	Rock
10	Finland	Rock
11	France	Rock

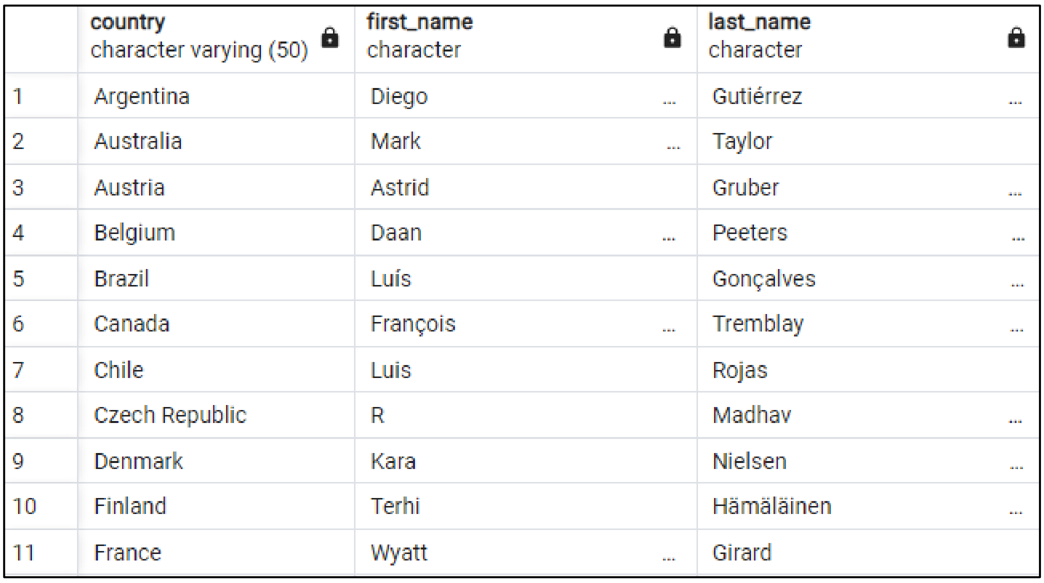
3. Write a query that determines the customer that has spent the most on music for each country. Write a query that returns the country along with the top customer and how much they spent. For countries where the top amount spent is shared, provide all customers who spent this amount

QUERY:



```
WITH ALL_CUSTOMER AS (
    SELECT CUSTOMER.CUSTOMER_ID, CUSTOMER.FIRST_NAME, CUSTOMER.LAST_NAME,CUSTOMER.COUNTRY,
    FROM INVOICE
    JOIN CUSTOMER ON CUSTOMER.CUSTOMER_ID = INVOICE.CUSTOMER_ID
    GROUP BY CUSTOMER.CUSTOMER_ID
    ORDER BY CUSTOMER.COUNTRY ASC,MONEY_SPEND DESC
)
SELECT COUNTRY, FIRST_NAME, LAST_NAME FROM ALL_CUSTOMER
WHERE ROW_NUMBER=1
```

OUTPUT:



	country character varying (50)	first_name character	last_name character
1	Argentina	Diego ...	Gutiérrez ...
2	Australia	Mark ...	Taylor
3	Austria	Astrid	Gruber ...
4	Belgium	Daan ...	Peeters ...
5	Brazil	Luís	Gonçalves ...
6	Canada	François ...	Tremblay ...
7	Chile	Luis	Rojas
8	Czech Republic	R	Madhav ...
9	Denmark	Kara	Nielsen ...
10	Finland	Terhi	Hämäläinen ...
11	France	Wyatt ...	Girard