



PG ADMIN

SQL PROJECT

SQL DATA ANALYSIS ON MUSIC INDUSTRY DATASET USING PGADMIN

CONTENT

IN THIS PROJECT, I WORKED ON A MUSIC DATASET USING PGADMIN TO SOLVE A VARIETY OF SQL QUERIES. THE PROJECT IS DIVIDED INTO THREE SECTIONS BASED ON THE DIFFICULTY LEVEL: EASY, MODERATE, AND ADVANCED.

THIS PROJECT SHOWCASES MY ABILITY TO WRITE AND OPTIMIZE COMPLEX SQL QUERIES TO EXTRACT MEANINGFUL INSIGHTS FROM A MUSIC DATASET, WHICH CAN BE VALUABLE FOR BUSINESS DECISION-MAKING IN THE MUSIC INDUSTRY.

SQL CONCEPTS WHICH I HAVE APPLIED IN THIS PROJECT CONTAINS CTEs, PARTITION BY CLAUSE AND JOINS FOR JOINING TWO OR MORE THAN TWO TABLES TOGETHER.

OBJECTIVES

Question Set 1: Easy

1. **Senior Most Employee:** Identify the senior-most employee based on job title.
2. **Countries with Most Invoices:** Determine which countries generated the most invoices.
3. **Top 3 Invoice Values:** Extract the top three values of total invoices.
4. **Best Customer City:** Identify the city with the best customers by calculating the highest sum of invoice totals, ideal for hosting a promotional Music Festival.
5. **Best Customer:** Discover the customer who has spent the most money, earning the title of the best customer.

Question Set 3 : Advanced

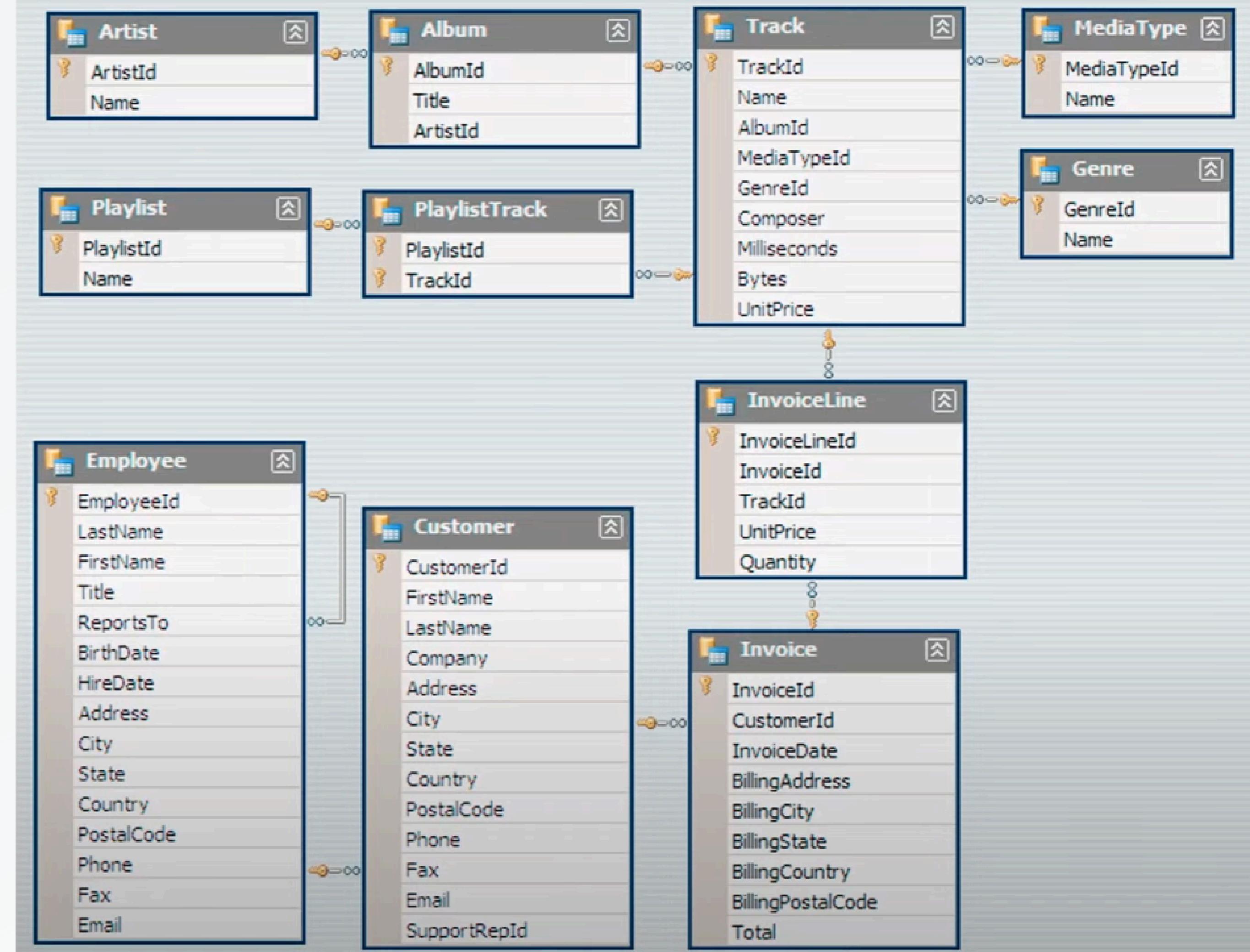
Question Set 2 : Moderate

1. **Rock Music Listeners:** Retrieve the email, first name, last name, and Genre of all Rock Music listeners, ordered alphabetically by email.
2. **Top Rock Bands:** List the top 10 rock bands by total track count to invite the artists who have written the most rock music.
3. **Longest Songs:** Find all tracks with a song length longer than the average, ordered by the longest duration first.

1. **Customer Spending by Artist:** Calculate the amount each customer has spent on different artists, returning the customer name, artist name, and total spent.
2. **Popular Genre by Country:** Determine the most popular music genre for each country based on the highest number of purchases, accounting for ties.
3. **Top Customer by Country:** Identify the customer who spent the most on music in each country, including ties.

DATABASE DIAGRAM TO UNDERSTAND THE DATABASE LAYOUT

All eleven tables from the database are shown in the diagram with the relationships between them.



EASY Q1: WHO IS THE SENIOR MOST EMPLOYEE BASED ON JOB TITLE?

Solution: For this question we will select employee table and order the output with respect to levels, the higher the level the senior the employee

QUERY

```
SELECT *
FROM EMPLOYEE
ORDER BY LEVELS DESC
LIMIT 1;
```

OUTPUT

	employee_id [PK] character varying (50)	last_name character (50)	first_name character (50)	title character varying (50)	reports_to character varying (30)	levels character varying (10)
1	9	Madan	Mohan	Senior General Manager	[null]	L7

- Based on our result we can see that Madan Mohan is the most senior employee with the highest level (L7)

EASY Q2: WHICH COUNTRIES HAVE THE MOST INVOICES?

Solution: For this question we will select invoice table , grouping the countries and count on invoice_id then sort the countries from highest invoices to lowest

QUERY

```
SELECT BILLING_COUNTRY, COUNT(INVOICE_ID)
FROM INVOICE
GROUP BY BILLING_COUNTRY
ORDER BY 2 DESC;
```

Invoice	
InvoiceId	
CustomerId	
InvoiceDate	
BillingAddress	
BillingCity	
BillingState	
BillingCountry	
BillingPostalCode	
Total	

- Based on our result we can see all the countries with their respective invoices in which USA has the most number of invoices(131 invoices) then Canada, brazil and so on.

OUTPUT

	billing_country	count
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
12	Spain	11
13	Finland	11
14	Australia	10
15	Netherlands	10
16	Sweden	10
17	Poland	10
18	Hungary	10
19	Denmark	10
20	Austria	9
21	Norway	9
22	Italy	9
23	Belgium	7
24	Argentina	5

EASY Q3: WHAT ARE TOP 3 VALUES OF TOTAL INVOICE?

Solution: For this question we will again select invoice table and sort the total values from highest to lowest

QUERY

```
SELECT INVOICE_ID, TOTAL AS TOTAL_VALUE  
FROM INVOICE  
ORDER BY 2 DESC  
LIMIT 3;
```

Invoice
InvoiceId
CustomerId
InvoiceDate
BillingAddress
BillingCity
BillingState
BillingCountry
BillingPostalCode
Total

OUTPUT

	invoice_id [PK] integer	total double precision
1	183	23.759999999999998
2	92	19.8
3	31	19.8

- We can see the top three invoices with the highest values in our output.

EASY Q4: 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

Solution: For the above question we will select Invoice table and group the values with respect to cities that will give us values according to each city then we will sum the total amount and sort the values from highest to lowest limiting the output to one that will give us the city with highest amount.

QUERY

```
SELECT BILLING_CITY, SUM(TOTAL) AS INVOICE_TOTAL  
FROM INVOICE  
GROUP BY BILLING_CITY  
ORDER BY 2 DESC  
LIMIT 1;
```

	Invoice
InvoiceId	
CustomerId	
InvoiceDate	
BillingAddress	
BillingCity	
BillingState	
BillingCountry	
BillingPostalCode	
Total	

OUTPUT

	billing_city	invoice_total
1	Prague	273.24000000000007

- From our result we can see that Prague is the city from which people have spent the most money.

EASY Q5: 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.

Solution: For the above question we will have to join customer table with invoice table as customer details are in customer table and amount spent details are in invoice table.

Then we can group the values by customer id that will generate the result according to each customer and sorting the total amount from highest to lowest.

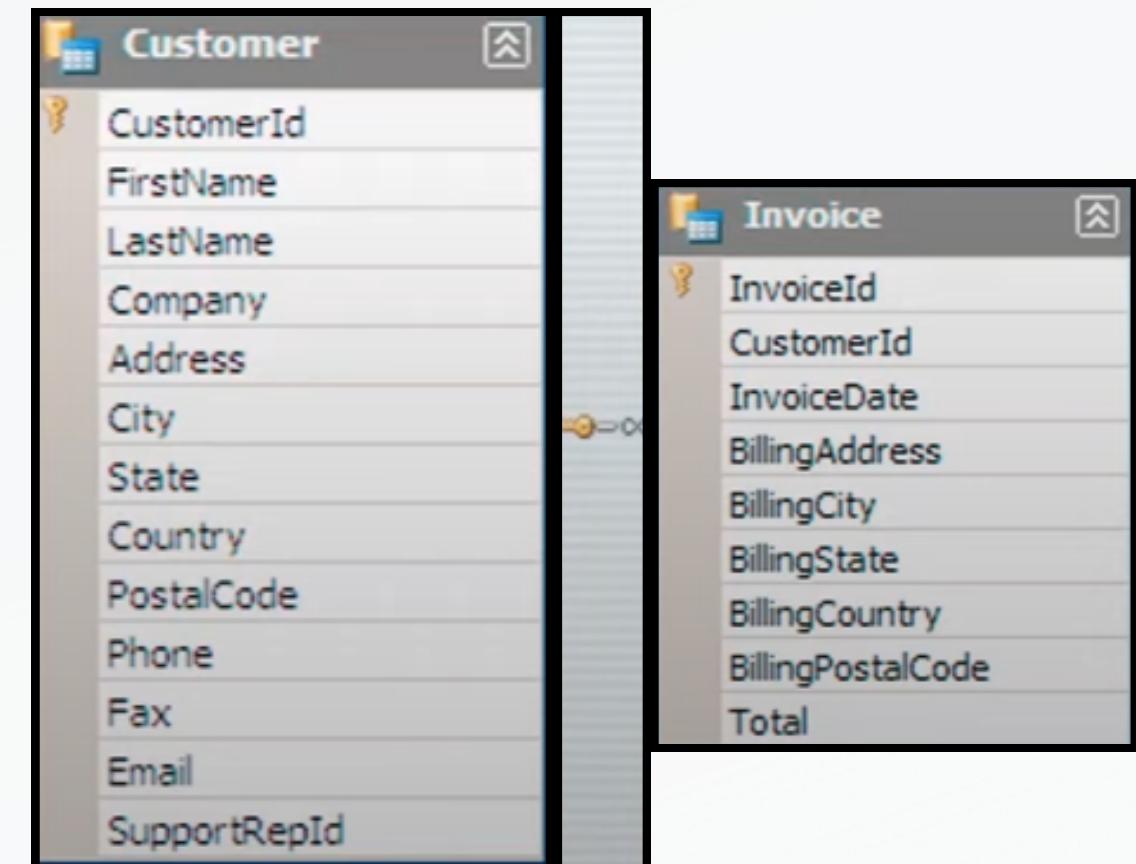
```
SELECT CUS.CUSTOMER_ID, CUS.FIRST_NAME, CUS.LAST_NAME, SUM(TOTAL) AS MONEY_SPENT  
FROM CUSTOMER CUS  
JOIN INVOICE INV  
ON CUS.CUSTOMER_ID = INV.CUSTOMER_ID  
GROUP BY CUS.CUSTOMER_ID  
ORDER BY MONEY_SPENT DESC  
LIMIT 1;
```

QUERY

OUTPUT

	customer_id [PK] integer	first_name character (50)	last_name character (50)	money_spent double precision
1	5	R	Madhav	144.54000000000002

- From our result we can see that R Madhav is the customer that has spent the most money.



MODERATE Q1: 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.

Solution: For the above question we will have to join three tables as track details are in Invoice Line table and customer details are in customer table.

For joining these two tables we will have to go through Invoice table because it contains elements of both these tables.

As we know that Rock Genre ID is 1 therefore we select our output based on tracks which have genre id = 1
because these are all rock music genre.

QUERY

```
SELECT CUS.EMAIL, CUS.FIRST_NAME, CUS.LAST_NAME
FROM CUSTOMER CUS
JOIN INVOICE INV
ON CUS.CUSTOMER_ID = INV.CUSTOMER_ID
JOIN INVOICE_LINE INV_L
ON INV_L.INVOICE_ID = INV.INVOICE_ID
WHERE TRACK_ID IN (
    SELECT TRACK_ID
    FROM TRACK
    WHERE CAST(GENRE_ID AS INT) = 1
)
GROUP BY CUS.CUSTOMER_ID
ORDER BY EMAIL;
```

OUTPUT

	email character varying (50)	first_name character (50)	last_name character (50)
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	dominiquelefrevre@gmail.c...	Dominique	Lefebvre
10	edfrancis@yachoo.ca	Edward	Francis

Total rows: 59 of 59 Query complete 00:00:00.141 Ln 21, Col 16

- From our output we can see all 59 customers who are rock music listeners.

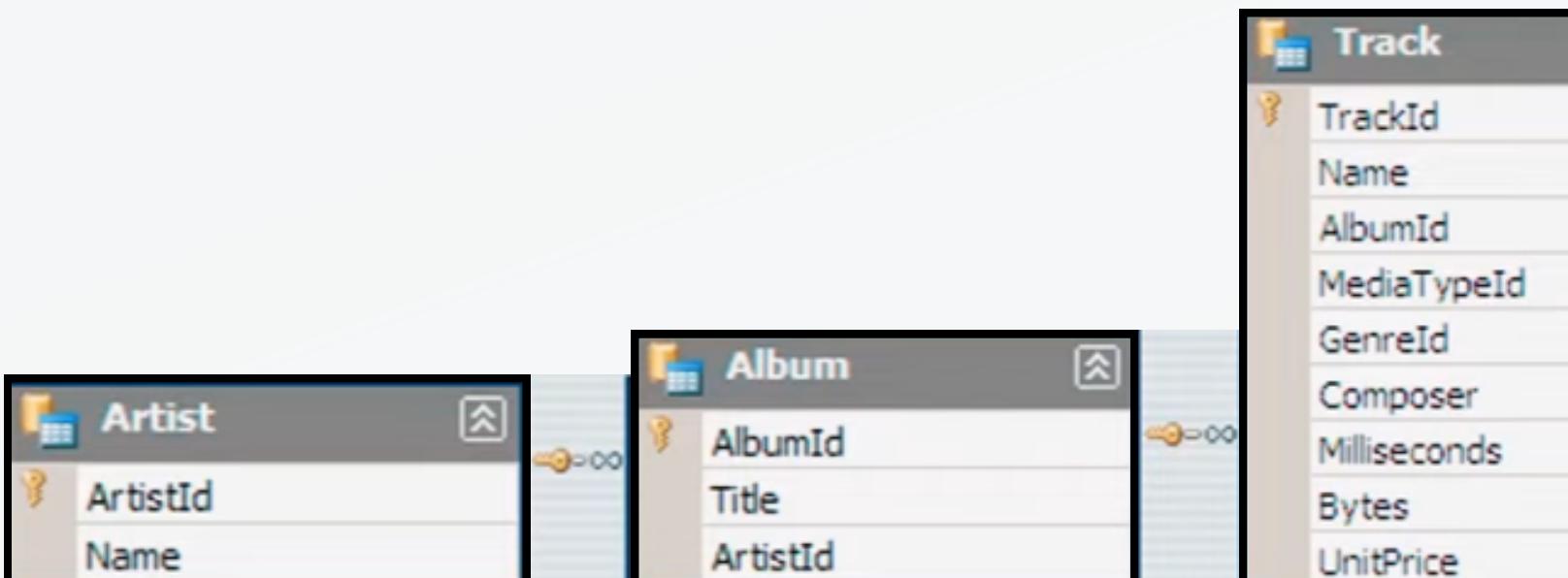
MODERATE Q2: LET'S INVITE THE ARTISTS WHO HAVE WRITTEN THE MOST ROCK MUSIC IN OUR DATASET. WRITE A QUERY THAT RETURNS THE ARTIST NAME AND TOTAL TRACK COUNT OF THE TOP 10 ROCK BANDS.

Solution: For the above question we will join three tables i-e Artist, Album and Track and filter the result where genre id = 1 as this is the rock genre.

Then we group by artist id for generating the output for each Artist then sorting the output and limiting the values to top 10.

QUERY

```
SELECT ART.NAME ARTIST_NAME, COUNT(TR.TRACK_ID) TOTAL_TRACK_COUNT
FROM ARTIST ART
JOIN ALBUM ALB
ON ART.ARTIST_ID = ALB.ARTIST_ID
JOIN TRACK TR
ON TR.ALBUM_ID = ALB.ALBUM_ID
WHERE CAST(TR.GENRE_ID AS INT) = 1
GROUP BY ART.ARTIST_ID
ORDER BY 2 DESC
LIMIT 10;
```



	artist_name	total_track_count
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

OUTPUT

- We can see the top 10 rock bands with their total number of songs

MODERATE Q3: 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.

Solution: For solving above question we will select track table and use a sub query where song length is greater than the average song length

QUERY

```
SELECT NAME AS TRACK_NAME, MILLISECONDS AS SONG_LENGTH
FROM TRACK
WHERE MILLISECONDS > (
    SELECT AVG(MILLISECONDS)
    FROM TRACK
)
ORDER BY 2 DESC;
```

Track	
TrackId	
Name	
AlbumId	
MediaTypeId	
GenreId	
Composer	
Milliseconds	
Bytes	
UnitPrice	

- In our result we can see all 494 songs which are longer than the average song length in our dataset.

OUTPUT

	track_name	song_length
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
12	The Magnificent Warriors	2924716
13	The Living Legend, Pt. 1	2924507
14	The Gun On Ice Planet Zero, Pt. 2	2924341

Total rows: 494 of 494

Query complete 00:00:00.12

ADVANCED Q1: FIND HOW MUCH AMOUNT SPENT BY EACH CUSTOMER ON ARTISTS? WRITE A QUERY TO RETURN CUSTOMER NAME, ARTIST NAME AND TOTAL SPENT

Solution: For the above question we will be joining six tables from Album table upto Customer table as we have to find all the customers spendings on each artist.

QUERY

```
SELECT CUS.CUSTOMER_ID, CUS.FIRST_NAME, CUS.LAST_NAME,  
SUM((INV_L.QUANTITY) * INV_L.UNIT_PRICE) AS TOTAL_SPENT,  
ART.NAME AS ARTIST_NAME  
FROM CUSTOMER CUS  
JOIN INVOICE INV  
ON CUS.CUSTOMER_ID = INV.CUSTOMER_ID  
JOIN INVOICE_LINE INV_L  
ON INV.INVOICE_ID = INV_L.INVOICE_ID  
JOIN TRACK TR  
ON INV_L.TRACK_ID = TR.TRACK_ID  
JOIN ALBUM ALB  
ON TR.ALBUM_ID = ALB.ALBUM_ID  
JOIN ARTIST ART  
ON ALB.ARTIST_ID = ART.ARTIST_ID  
GROUP BY CUS.CUSTOMER_ID, INV_L.UNIT_PRICE, ART.NAME  
ORDER BY 1, 4 DESC;
```

- We can see how much money each customer spent on each artist in our result.

OUTPUT

	customer_id integer	first_name character (50)	last_name character (50)	total_spent double precision	artist_name character varying (120)
1	1	Luís	Gonçalves	17.82	The Cult
2	1	Luís	Gonçalves	13.86000000	Rush
3	1	Luís	Gonçalves	12.87000000	JET
4	1	Luís	Gonçalves	7.920000000	AC/DC
5	1	Luís	Gonçalves	2.969999999	Aerosmith
6	1	Luís	Gonçalves	2.969999999	Miles Davis
7	1	Luís	Gonçalves	2.969999999	Guns N' Roses
8	1	Luís	Gonçalves	1.98	U2
9	1	Luís	Gonçalves	1.98	Iron Maiden
10	1	Luís	Gonçalves	1.98	Marisa Monte
11	1	Luís	Gonçalves	1.98	The Police
12	1	Luís	Gonçalves	1.98	The Who
13	1	Luís	Gonçalves	1.98	Amy Winehouse
14	1	Luís	Gonçalves	1.98	Queen

Total rows: 1000 of 2189 Query complete 00:00:00.180 Rows selected: 1 Ln 29, Col

ADVANCED Q2: 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.

Solution: For the above question we have joined three tables and used partition by clause for generating one row per each country where count is maximum.

QUERY

```
WITH POPULAR_MUSIC_CTE AS (
SELECT INV.BILLING_COUNTRY COUNTRY, GEN.NAME GENRE,
COUNT(INV_L.QUANTITY),
ROW_NUMBER() OVER(PARTITION BY INV.BILLING_COUNTRY
ORDER BY COUNT(INV_L.QUANTITY) DESC) AS ROW_NO
FROM INVOICE INV
JOIN INVOICE_LINE INV_L
ON INV.INVOICE_ID = INV_L.INVOICE_ID
JOIN TRACK TR
ON INV_L.TRACK_ID = TR.TRACK_ID
JOIN GENRE GEN
ON TR.GENRE_ID = GEN.GENRE_ID
GROUP BY INV.BILLING_COUNTRY, GEN.NAME
)
SELECT * FROM POPULAR_MUSIC_CTE WHERE ROW_NO = 1
;
```

- From the result we can see that Rock is the most popular genre in every country except Argentina whose most popular genre is Alternative & Punk.

OUTPUT

	country character varying (30)	genre character varying (12)	count bigint	row_no bigint
1	Argentina	Alternative & Punk	17	1
2	Australia	Rock	34	1
3	Austria	Rock	40	1
4	Belgium	Rock	26	1
5	Brazil	Rock	205	1
6	Canada	Rock	333	1
7	Chile	Rock	61	1
8	Czech Republic	Rock	143	1
9	Denmark	Rock	24	1
10	Finland	Rock	46	1
11	France	Rock	211	1
12	Germany	Rock	194	1
13	Hungary	Rock	44	1
14	India	Rock	102	1
15	Ireland	Rock	72	1
16	Italy	Rock	35	1
17	Netherlands	Rock	33	1
18	Norway	Rock	40	1
19	Poland	Rock	40	1
20	Portugal	Rock	108	1
21	Spain	Rock	46	1
22	Sweden	Rock	60	1
23	United Kingdom	Rock	166	1
24	USA	Rock	561	1

Total rows: 24 of 24

Query complete 00:00:00.091

Ln 52, Col 2

ADVANCED Q3: 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.

Solution: Similar to the last question we will use partition by clause as we have to see only one customer from each country who has spent the most money.

QUERY

```
WITH CTE AS (
SELECT INV.BILLING_COUNTRY COUNTRY, INV.CUSTOMER_ID,
CUS.FIRST_NAME, CUS.LAST_NAME, SUM(TOTAL) AMOUNT_SPENT,
ROW_NUMBER () OVER(PARTITION BY BILLING_COUNTRY
ORDER BY SUM(TOTAL) DESC) AS ROW_NO
FROM INVOICE INV
JOIN CUSTOMER CUS
ON INV.CUSTOMER_ID = CUS.CUSTOMER_ID
GROUP BY INV.CUSTOMER_ID, INV.BILLING_COUNTRY,
CUS.FIRST_NAME, CUS.LAST_NAME
)
SELECT * FROM CTE WHERE ROW_NO = 1
;
```

- From the result we can see the customer from each country who has spent the most money on music.

	country character varying (customer_id integer	first_name character (50) 	last_name character (50) 	amount_spent double precision
1	Argentina	56	Diego	Gutiérrez	39.6
2	Australia	55	Mark	Taylor	81.18
3	Austria	7	Astrid	Gruber	69.3
4	Belgium	8	Daan	Peeters	60.38999999
5	Brazil	1	Luís	Gonçalves	108.8999999
6	Canada	3	François	Tremblay	99.99
7	Chile	57	Luis	Rojas	97.02000000
8	Czech Republic	5	R	Madhav	144.5400000
9	Denmark	9	Kara	Nielsen	37.61999999
10	Finland	44	Terhi	Hämäläinen	79.2
11	France	42	Wyatt	Girard	99.99
12	Germany	37	Fynn	Zimmermann	94.05000000
13	Hungary	45	Ladislav	Kovács	78.21
14	India	58	Manoj	Pareek	111.8699999
15	Ireland	46	Hugh	O'Reilly	114.8399999
16	Italy	47	Lucas	Mancini	50.49
17	Netherlands	48	Johannes	Van der Berg	65.34
18	Norway	4	Bjørn	Hansen	72.27000000
19	Poland	49	Stanisław	Wójcik	76.22999999
20	Portugal	34	João	Fernandes	102.9600000
21	Spain	50	Enrique	Muñoz	98.01
22	Sweden	51	Joakim	Johansson	75.24
23	United Kingdom	53	Phil	Hughes	98.01
24	USA	17	Jack	Smith	98.01

Total rows: 24 of 24

Query complete 00:00:00.100

Ln 26, Col 2