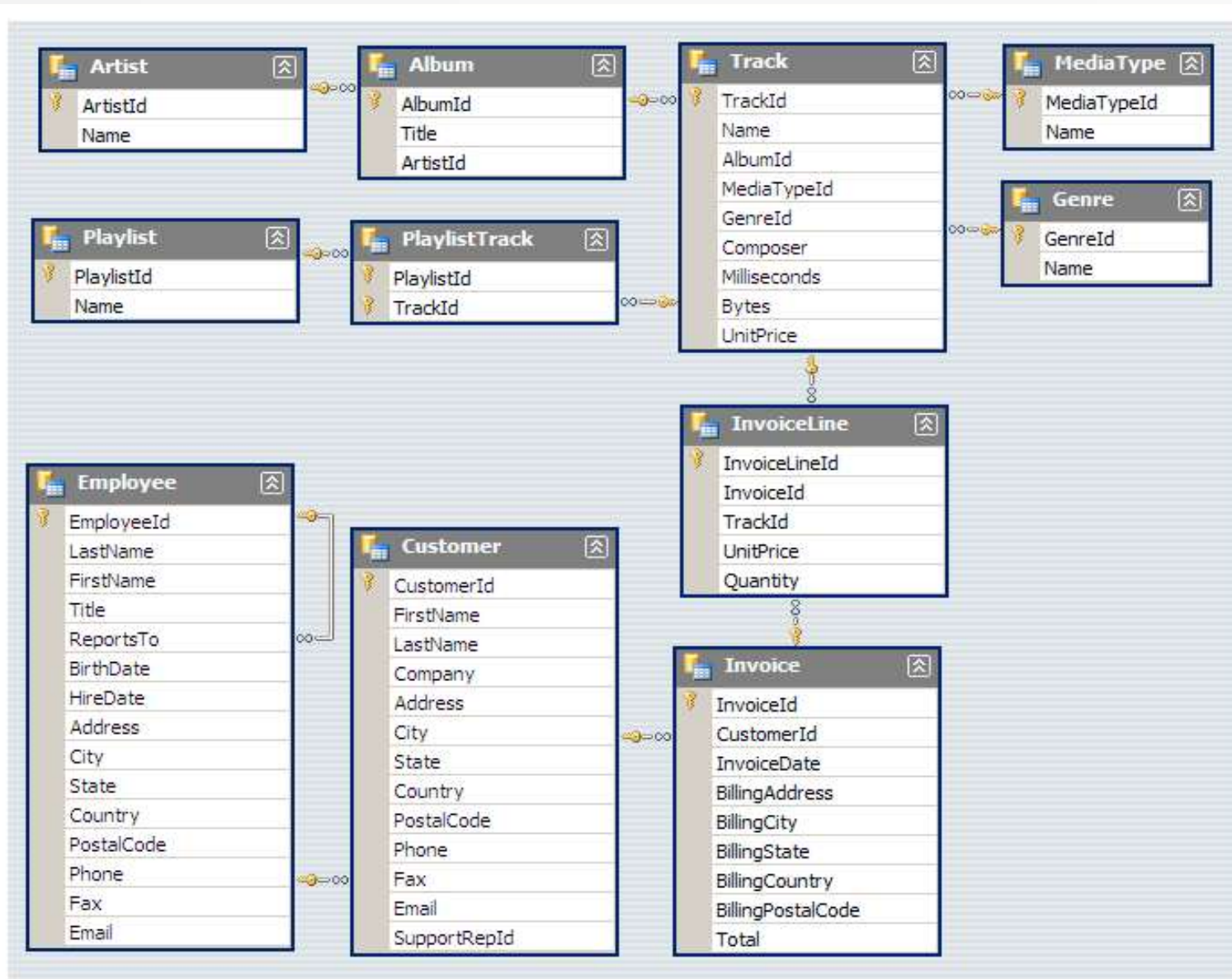


# **MUSIC STORE**

# **DATABASE ANALYSIS**

## SCHEMA :



### Question Set 1 : Easy

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

```
SELECT * FROM employee  
ORDER BY levels DESC  
LIMIT 1
```

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


## Q2. Which countries have the most invoices ?

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

total bigint		billing_country character varying (30)
131		USA
76		Canada
61		Brazil
50		France
41		Germany
30		Czech Republic
29		Portugal
28		United Kingdom
21		India
13		Chile
13		Ireland
11		Spain

Q3. What are top 3 values of total invoice ?

```
SELECT total AS total_invoice FROM invoice  
ORDER BY total DESC  
LIMIT 3
```

total_invoice	
double precision	
23.759999999999999998	
	19.8
	19.8

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.

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

	billing_city character varying (30) 🔒	total_invoice double precision 🔒
1	Prague	273.240000000000007

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.

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

	customer_id [PK] integer	first_name character	last_name character	total double precision
1	5 R	...	Madhav	144.540000000000002



## Question Set 2 : 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.

```
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 LIKE 'Rock')
ORDER BY email ASC
```

	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@yahoo.ca	Edward	Francis
11	eduardo@woodstock.com.br	Eduardo	Martins
12	ellie.sullivan@shaw.ca	Ellie	Sullivan
13	emma.jones@hotmail.com	Emma	Jones
Total rows: 59 of 59    Query complete 00:00:00.132			



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.artist_id,artist.name,COUNT(artist.artist_id) AS number_of_songs
FROM track
JOIN album ON track.album_id = album.album_id
JOIN artist ON album.artist_id = artist.artist_id
JOIN genre ON track.genre_id = genre.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
22	Led Zeppelin	114
150	U2	112
58	Deep Purple	92
90	Iron Maiden	81
118	Pearl Jam	54
152	Van Halen	52
51	Queen	45
142	The Rolling Stones	41
76	Creedence Clearwater Revival	40
52	Kiss	35

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 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
character varying (150)	integer
Occupation / Precipice	5286953
Through a Looking Glass	5088838
Greetings from Earth, Pt. 1	2960293
The Man With Nine Lives	2956998
Battlestar Galactica, Pt. 2	2956081
Battlestar Galactica, Pt. 1	2952702
Murder On the Rising Star	2935894
Battlestar Galactica, Pt. 3	2927802
rows: 494 of 494	Query complete 00:00:00.162

### Question Set 3 : Advance

Q1. Find how much amount spent by each customer on artists? Write a query to return customer name, artist name and total spent.

```
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 1
    ORDER BY 3 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 a ON a.album_id = t.album_id
JOIN best_selling_artist bsa ON bsa.artist_id = a.artist_id
GROUP BY 1,2,3,4
ORDER BY 5 DESC
```

	customer_id integer	first_name character	last_name character	artist_name character varying (120)	amount_spent double precision
	46	Hugh	O'Reilly	Queen	27.719999999999985
	38	Niklas	Schröder	... Queen	18.81
	3	François	... Tremblay	... Queen	17.82
	34	João	Fernandes	... Queen	16.830000000000002
	53	Phil	Hughes	... Queen	11.88
	41	Marc	Dubois	Queen	11.88
	47	Lucas	Mancini	... Queen	10.89
	33	Ellie	Sullivan	Queen	10.89
	20	Dan	Miller	Queen	3.96
	5	R	Madhav	... Queen	3.96
total rows: 43 of 43					Query complete 00:00:00.095

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. 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 Row_no
    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 Row_no <=1
```

purchases bigint	country character varying (50)	name character varying (120)	genre_id character varying (50)	row_no bigint
17	Argentina	Alternative & Punk	4	1
34	Australia	Rock	1	1
40	Austria	Rock	1	1
26	Belgium	Rock	1	1
205	Brazil	Rock	1	1
333	Canada	Rock	1	1
61	Chile	Rock	1	1
143	Czech Republic	Rock	1	1
24	Denmark	Rock	1	1
46	Finland	Rock	1	1
211	France	Rock	1	1

al rows: 24 of 24    Query complete 00:00:00.087



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. For countries where the top amount spent is shared, provide all customers who spent this amount.

```
WITH customer_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 Row_no
    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 customer_with_country WHERE Row_no <= 1
```

	customer_id integer	first_name character	last_name character	billing_country character varying (30)	total_spending double precision	row_no bigint
1	56	Diego	... Gutiérrez	... Argentina	39.6	1
2	55	Mark	... Taylor	Australia	81.18	1
3	7	Astrid	Gruber	... Austria	69.3	1
4	8	Daan	... Peeters	... Belgium	60.389999999999999	1
5	1	Luis	Gonçalves	... Brazil	108.899999999999998	1
6	3	François	... Tremblay	... Canada	99.99	1
Total rows: 24 of 24		Query complete 00:00:00.124				