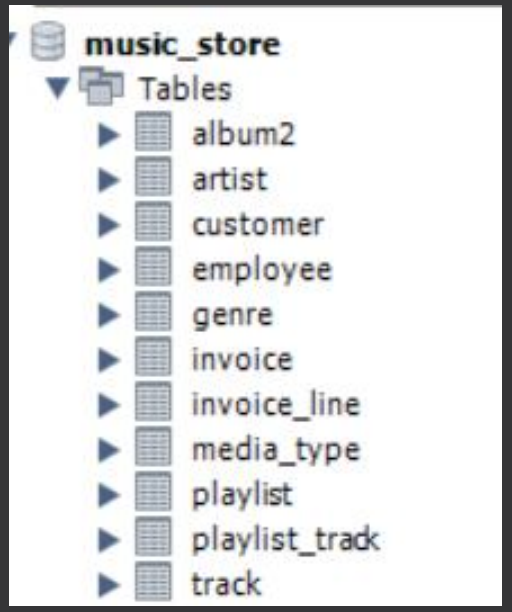
Music Store Data Analysis Using MySQL Workbench

Data Set Used :



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

Query :

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

Output :

employee_id	last_name	first_name	title	reports_to	levels	birthdate	hire_date
1	Adams	Andrew	General Manager	9	L6	18-02-1962 00:00	14-08-2016 00:00

Q2. Which country have the most invoices?

Query :

```
SELECT COUNT(*) AS invoice_count, billing_country
FROM invoice
GROUP BY billing_country
ORDER BY invoice_count DESC
LIMIT 1;
```

Output :

	invoice_count	billing_country
▶	131	USA

Q3. What are top 3 values of total invoice?

Query :

```
SELECT total FROM invoice
ORDER BY total DESC
LIMIT 3;
```

Output :

total
23.759999999999998
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 highest sum of invoice totals. Return both city name and sum of all invoice totals.

Query :

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

Output :

	invoice_total	billing_city
▶	273.24000000000007	Prague

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.

Query :

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

Output :

	customer_id	first_name	last_name	total_spent
▶	5	FrantiÅiek	WichterlovÃ¡	144.54000000000002

Q6. Write a 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.invoice_id
JOIN invoice_line ON invoice.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 LIKE 'Rock'
ORDER BY email;
```

Output :

email	first_name	last_name
aaronmitchell@yahoo.ca	Aaron	Mitchell
astrid.gruber@apple.at	Astrid	Gruber
dmiller@comcast.com	Dan	Miller
edfrancis@yachoo.ca	Edward	Francis
fharris@google.com	Frank	Harris
hholy@gmail.com	Helena	HolÃ½
jacksmith@microsoft.com	Jack	Smith
joakim.johansson@yahoo.se	Joakim	Johansson
johavanderberg@yahoo.nl	Johannes	Van der Berg
johngordon22@yahoo.com	John	Gordon
jubarnett@gmail.com	Julia	Barnett
kachase@hotmail.com	Kathy	Chase
lucas.mancini@yahoo.it	Lucas	Mancini
mark.taylor@yahoo.au	Mark	Taylor
masampaio@sapo.pt	Madalena	Sampaio
ricunningham@hotmail.com	Richard	Cunningham
roberto.almeida@riotur.gov...	Roberto	Almeida

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.

Query :

```
SELECT artist.artist_id, artist.name, COUNT(artist.artist_id) AS number_of_songs
FROM artist
JOIN album2 ON artist.artist_id = album2.artist_id
JOIN track ON album2.album_id = track.album_id
JOIN genre ON track.genre_id = genre.genre_id
WHERE genre.name LIKE 'Rock'
GROUP BY artist.artist_id, artist.name
ORDER BY number_of_songs DESC
LIMIT 10;
```

Output :

artist_id	name	number_of_songs
1	AC/DC	18
3	Aerosmith	15
8	Audioslave	14
22	Led Zeppelin	14
4	Alanis Morissette	13
5	Alice In Chains	12
23	Frank Zappa & Captain Beefheart	9
2	Accept	4

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.

Query :

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

Output :

name	milliseconds
How Many More Times	711836
Advance Romance	677694
Sleeping Village	644571
You Shook Me(2)	619467
Talkin' 'Bout Women Obviously	589531
Stratus	582086
No More Tears	555075
The Alchemist	509413
Wheels Of Confusion / The Straightener	494524
Book Of Thel	494393

Q9. 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 genre.name, genre.genre_id, customer.country,
    COUNT(invoice_line.quantity) AS purchases,
    ROW_NUMBER() OVER(PARTITION BY customer.country
    ORDER BY COUNT(invoice_line.quantity)DESC) AS RowNo
    FROM invoice_line
    JOIN invoice ON invoice_line.invoice_id = invoice.invoice_id
    JOIN customer ON invoice.customer_id = customer.customer_id
    JOIN track ON invoice_line.track_id = track.track_id
    JOIN genre ON track.genre_id = genre.genre_id
    GROUP BY genre.name, genre.genre_id, customer.country
    ORDER BY customer.country ASC, purchases DESC
)
SELECT * FROM popular_genre WHERE RowNo <= 1;
```

Output :

name	genre_id	country	purchases	RowNo
Rock	1	Argentina	1	1
Rock	1	Australia	18	1
Rock	1	Austria	6	1
Rock	1	Belgium	5	1
Rock	1	Brazil	26	1
Rock	1	Canada	57	1
Rock	1	Chile	7	1
Rock	1	Czech Republic	14	1
Rock	1	Denmark	6	1
Rock	1	Finland	6	1

Q10. 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 customer_with_country AS (
    SELECT customer.customer_id, customer.first_name, customer.last_name, invoice.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 customer.customer_id, customer.first_name, customer.last_name, invoice.billing_country
    ORDER BY billing_country ASC, total_spending DESC
)
SELECT * FROM customer_with_country WHERE RowNo = 1;
```

Output :

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	60.38999999999999	1
1	LuÃ-s	GonÃ§alves	Brazil	108.89999999999998	1
3	FranÃ§ois	Tremblay	Canada	99.99	1
57	Luis	Rojas	Chile	97.02000000000001	1
5	FrantiÅiek	WichterlovÃi	Czech Republic	144.54000000000002	1
9	Kara	Nielsen	Denmark	37.61999999999999	1
44	Terhi	HÃ¤mÃ¤lÃ¤inen	Finland	79.2	1