Music Store Data Analysis Project using SQL

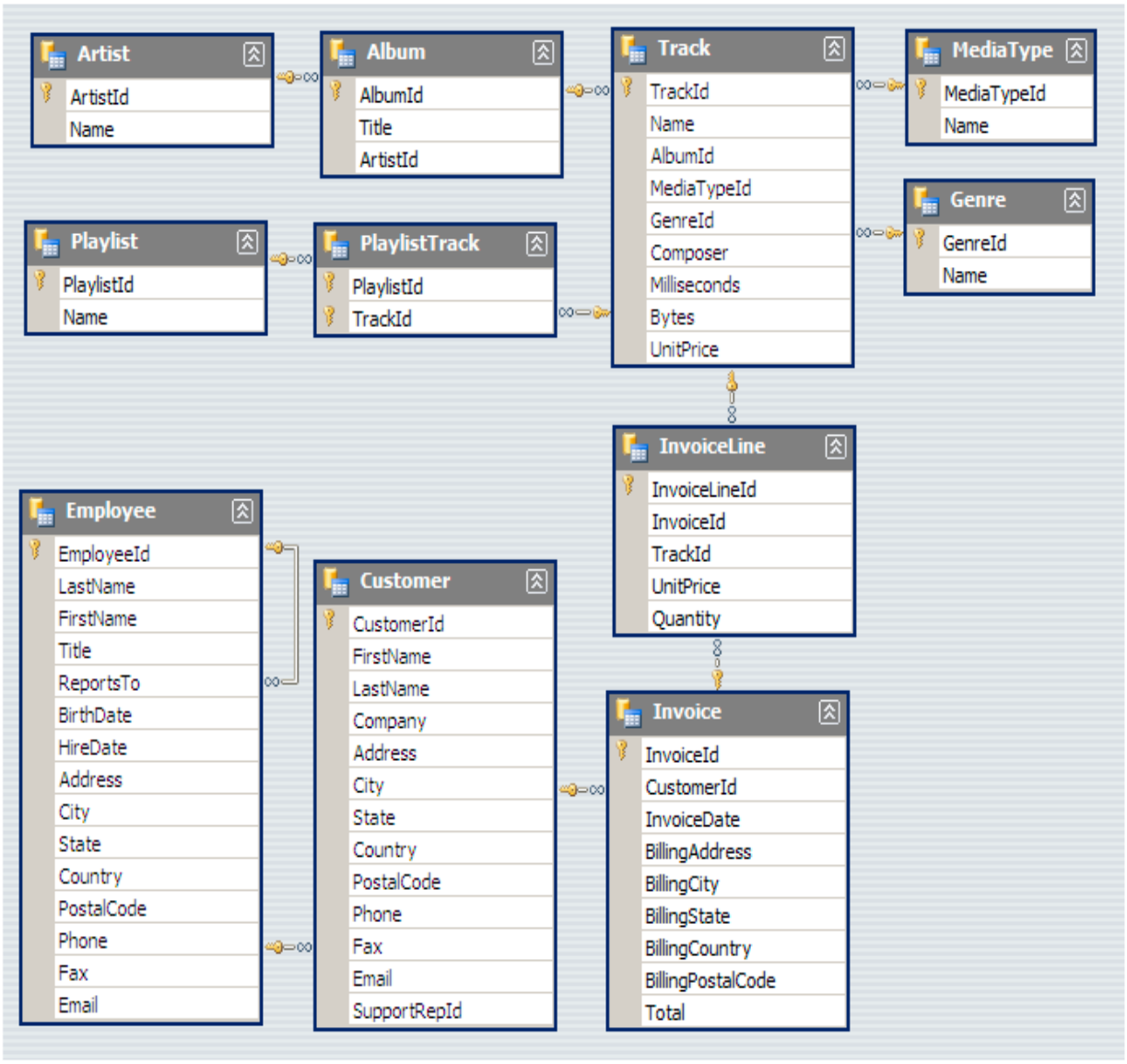
Objective

. Analyse data relate to sales, genres and artists.

. Analyse sales data to understand the distribution of sales across different countries.

. Determine the popularity of music genres and Artist in each country.

. Analyse which genres are most favoured by customers in different geographic locations, which can inform inventory selection and marketing efforts.


Artist
ArtistId
Name
Album
AlbumId
Title
ArtistId
Track
TrackId
Name
AlbumId
MediaTypeId
GenreId
Composer
Milliseconds
Bytes
UnitPrice
MediaType
MediaTypeId
Name
Genre
GenreId
Name
Playlist
PlaylistId
Name
PlaylistTrack
PlaylistId
TrackId
InvoiceLine
InvoiceLineId
InvoiceId
TrackId
UnitPrice
Quantity
Employee
EmployeeId
LastName
FirstName
Title
ReportsTo
BirthDate
HireDate
Address
City
State
Country
PostalCode
Phone
Fax
Email
Customer
CustomerId
FirstName
LastName
Company
Address
City
State
Country
PostalCode
Phone
Fax
Email
SupportRepId
Invoice
InvoiceId
CustomerId
InvoiceDate
BillingAddress
BillingCity
BillingState
BillingCountry
BillingPostalCode
Total

SQL Analysis with their Output

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

Query | Query History

```sql
/* Q1: Who is the senior most employee based on job title? */

SELECT title, first_name, last_name
FROM employee
ORDER BY levels DESC
LIMIT 1
```

Data Output | Messages | Notifications

	title character varying (50)	first_name character	last_name character
1	Senior General Manager	Mohan	Madan

SQL Analysis with their Output

Q2- Which countries have the most Invoices?

Query | Query History

```
1  /* Q2: Which countries have the most Invoices? */
2
3  SELECT COUNT(*) AS c, billing_country
4  FROM invoice
5  GROUP BY billing_country
6  ORDER BY c DESC
```

Data Output | Messages | Notifications

	c bigint	billing_country character varying (30)
1	131	USA
2	76	Canada
3	61	Brazil
4	50	France
5	41	Germany
6	30	Czech Republic
7	29	Portugal
8	28	United Kingdom

Total rows: 24 of 24 | Query complete 00:00:00.128

SQL Analysis with their Output

Q3-What are top 3 values of total invoice?

Query | Query History

```
1  /* Q3: What are top 3 values of total invoice? */
2
3  SELECT total FROM invoice
4  ORDER BY total DESC
5  LIMIT 3
```

Data Output | Messages | Notifications

	total double precision
1	23.759999999999998
2	19.8
3	19.8

Total rows: 3 of 3 | Query complete 00:00:00.075

SQL Analysis with their Output

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

Query | Query History

```
1  SELECT billing_city, sum(total) as Invoice_total FROM invoice
2  GROUP BY billing_city
3  ORDER BY Invoice_total DESC
4  LIMIT 1
```

Data Output | Messages | Notifications

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

Total rows: 1 of 1 | Query complete 00:00:00.085

SQL Analysis with their Output

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 | Query History

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

Data Output | Messages | Notifications

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

Total rows: 1 of 1 | Query complete 00:00:00.149

SQL Analysis with their Output

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

Query | Query History

```
1  SELECT DISTINCT first_name, last_name, email FROM customer
2  JOIN invoice ON customer.customer_id  = invoice.customer_id
3  JOIN invoice_line ON invoice.invoice_id = invoice_line.invoice_id
4  WHERE track_id IN(
5      SELECT track_id FROM track
6      JOIN genre ON track.genre_id = genre.genre_id
7      WHERE genre.name LIKE 'Rock'
8  )
9  ORDER BY email;
```

Data Output | Messages | Notifications

	first_name character	last_name character	email character varying (50)
1	Aaron	Mitchell	aaronmitchell@yahoo.ca
2	Alexandre	Rocha	alero@uol.com.br
3	Astrid	Gruber	astrid.gruber@apple.at
4	Bjørn	Hansen	bjorn.hansen@yahoo.no
5	Camille	Bernard	camille.bernard@yahoo.fr
6	Daan	Peeters	daan_peeters@apple.be
7	Diego	Gutiérrez	diego.gutierrez@yahoo.ar
8	Dan	Miller	dmiller@comcast.com

Total rows: 59 of 59 | Query complete 00:00:00.077

SQL Analysis with their Output

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 | Query History

```
1  SELECT artist.artist_id, artist.name, COUNT(artist.artist_id) AS number_of_songs
2  FROM track
3  JOIN album ON track.album_id = album.album_id
4  JOIN artist ON album.artist_id = artist.artist_id
5  JOIN genre ON track.genre_id = genre.genre_id
6  WHERE genre.name LIKE 'Rock'
7  GROUP BY artist.artist_id
8  ORDER BY number_of_songs DESC
9  LIMIT 10
```

Data Output | Messages | Notifications

	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
4	90	Iron Maiden	81
5	118	Pearl Jam	54
6	152	Van Halen	52
7	51	Queen	45
8	142	The Rolling Stones	41
9	76	Creedence Clearwater Revival	40

Total rows: 10 of 10 | Query complete 00:00:00.087

SQL Analysis with their Output

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 | Query History

```
1  SELECT name, milliseconds
2  FROM track
3  WHERE milliseconds > (
4      SELECT AVG(milliseconds) AS Avg_Track_Length
5      FROM track
6  )
7  ORDER BY milliseconds DESC;
```

Data Output | Messages | Notifications

	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

Total rows: 494 of 494 | Query complete 00:00:00.097

SQL Analysis with their Output

Q9-Find how much amount spent by each customer on artists? Write a query to return customer name, artist name and total spent

Query | Query History

```sql
WITH best_selling_price AS(
    SELECT artist.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, bsp.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_price bsp ON bsp.artist_id = alb.artist_id
GROUP BY 1,2,3,4
ORDER BY 5 DESC;
```

Query | Query History

Data Output | Messages | Notifications

	customer_id integer	first_name character	last_name character	artist_name character varying (120)	amount_spent double precision
1	46	Hugh	O'Reilly	Queen	27.719999999999985
2	38	Niklas	Schröder	Queen	18.81
3	3	François	Tremblay	Queen	17.82
4	34	João	Fernandes	Queen	16.830000000000002
5	53	Phil	Hughes	Queen	11.88
6	41	Marc	Dubois	Queen	11.88
7	47	Lucas	Mancini	Queen	10.89
8	33	Ellie	Sullivan	Queen	10.89
9	20	Dan	Miller	Queen	3.96
10	5	R	Madhav	Queen	3.96
11	23	John	Gordon	Queen	2.9699999999999998
12	54	Steve	Murray	Queen	2.9699999999999998
13	31	Martha	Silk	Queen	2.9699999999999998
14	16	Frank	Harris	Queen	1.98
15	17	Jack	Smith	Queen	1.98
16	24	Frank	Ralston	Queen	1.98
17	30	Edward	Francis	Queen	1.98

Total rows: 43 of 43 | Query complete 00:00:00.070

SQL Analysis with their Output

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

Query | Query History

```
WITH popular_genre AS
(
    SELECT COUNT(invoice_line.quantity) AS purchases, customer.country,
    genre.genre_id, genre.name,
    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
```

Data Output | Messages | Notifications

	purchases bigint	country character varying (50)	genre_id character varying (50)	name character varying (120)	rowno bigint
1	17	Argentina	4	Alternative & Punk	1
2	34	Australia	1	Rock	1
3	40	Austria	1	Rock	1
4	26	Belgium	1	Rock	1
5	205	Brazil	1	Rock	1

Total rows: 24 of 24 | Query complete 00:00:00.077

SQL Analysis with their Output

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

Query | Query History

```
WITH Customer_with_country AS(
    SELECT customer.customer_id, customer.first_name, customer.last_name,
    invoice.billing_country, SUM(invoice.total) AS total_spending,
    ROW_NUMBER() OVER(PARTITION BY invoice.billing_country ORDER BY SUM(invoice.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 Customer_with_country WHERE RowNo <=1
```

Data Output | Messages | Notifications

	customer_id integer	first_name character	last_name character	billing_country character varying (30)	total_spending double precision	rowno 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	60.38999999999999	1
5	1	Luís	Gonçalves	Brazil	108.89999999999998	1
6	3	François	Tremblay	Canada	99.99	1
7	57	Luis	Rojas	Chile	97.02000000000001	1
8	5	R	Madhav	Czech Republic	144.54000000000002	1

Total rows: 24 of 24 | Query complete 00:00:00.110