

SQL PROJECT

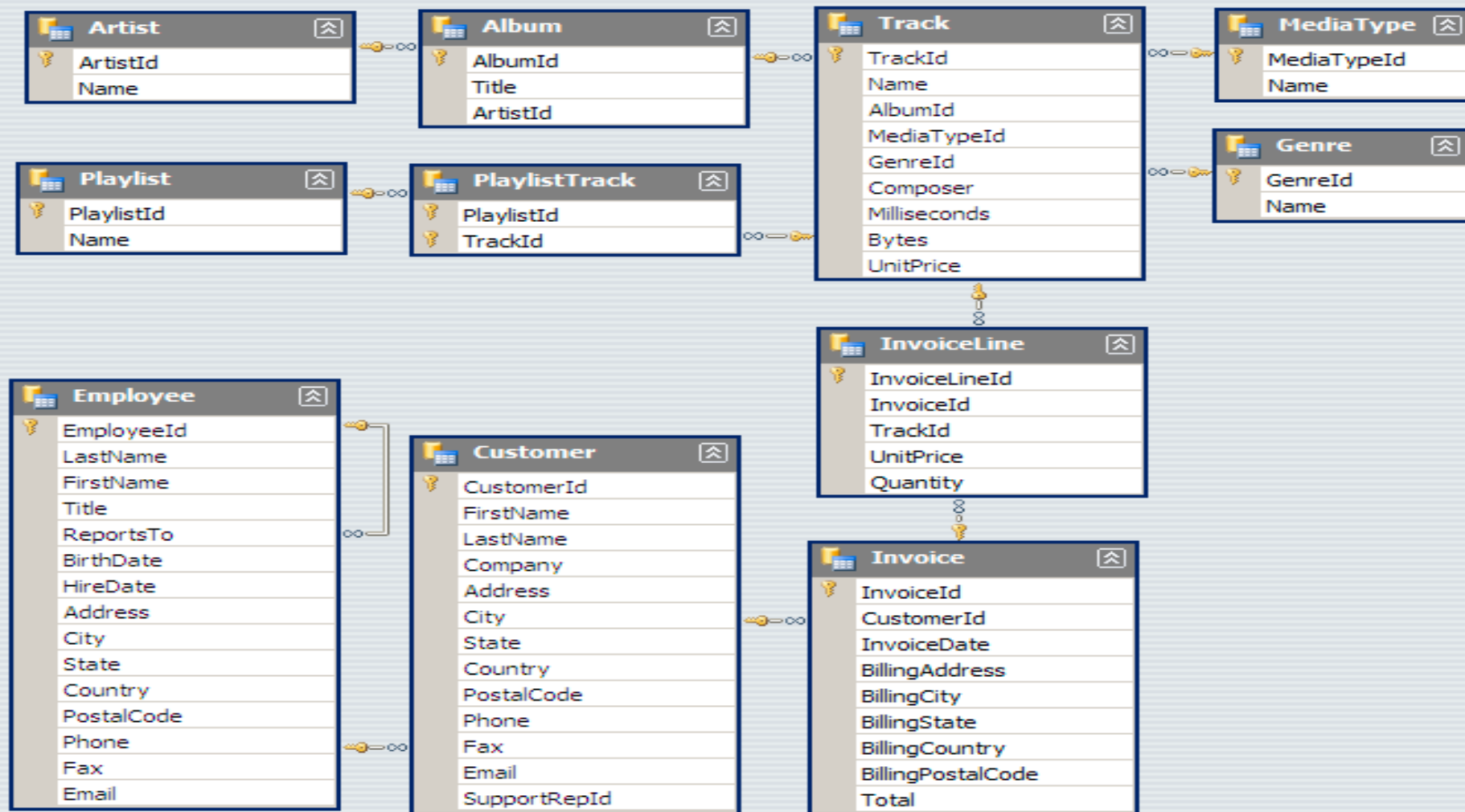
Music Store Analysis



Objective

1. Understand which music albums, artists, or genres are the most popular in each country.
2. Analyse data related to sales , genres, and artists
3. Identify genres, artists preferences across different regions
4. Analyse which genres are most favoured by customers in different geographic locations, which can inform inventory selection and marketing efforts.

Entity Relationship



SQL Analysis with their Output

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

Query

Query History

Scratch Pad

1

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

2

3

Select * from employee

4

order by levels desc

5

limit 1

Data Output

Messages

Notifications

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

SQL Analysis with their Output

Q2: Which countries have the most invoice?

Query

Query History

1 Q2: Which countries have the most invoice?

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

</

SQL Analysis with their Output

Q3: What are top 3 values of total invoice

```
select sum(total) as invoice_total, billing_city
from invoice
group by billing_city
order by invoice_total desc
limit 1
```

Output Messages Notifications











invoice_total		billing_city	
double precision	🔒	character varying (30)	🔒
273.240000000000007		Prague	

SQL Analysis with their Output

Q4: Which city has the best customers? we would like to throw a promotional music festivals 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 sum(total) as invoice_total, billing_city
from invoice
group by billing_city
order by invoice_total desc
limit 1
```

Output Messages Notifications









							
invoice_total					billing_city		
double precision					character varying (30)		
273.24000000000007					Prague		

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.

```
Select customer.customer_id, customer.first_name, customer.last_name, Sum(invoice.total) as total
from customer
join invoice on customer.customer_id=invoice.customer_id
group by customer.customer_id
order by total desc
limit 1|
```

SQL Output Messages Notifications

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

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

```
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;
```

Output Messages Notifications

email character varying (50)	first_name character	last_name character
aaronmitchell@yahoo.ca	Aaron	Mitchell
alero@uol.com.br	Alexandre	Rocha
astrid.gruber@apple.at	Astrid	Gruber
bjorn.hansen@yahoo.no	Bjørn	Hansen
camille.bernard@yahoo.fr	Camille	Bernard
daan_peeters@apple.be	Daan	Peeters

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

```
3
4 SELECT artist.artist_id, artist.name, COUNT(artist.artist_id) AS number_of_songs
5 FROM track
6 JOIN album ON album.album_id = track.album_id
7 JOIN artist ON artist.artist_id = album.artist_id
8 JOIN genre ON genre.genre_id = track.genre_id
9 WHERE genre.name LIKE 'Rock'
10 GROUP BY artist.artist_id
11 ORDER BY number_of_Songs desc
12 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
10	52	Kiss	35

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.

```
5 SELECT name,milliseconds
6 FROM track
7 WHERE milliseconds > (
8     SELECT AVG(milliseconds) AS avg_track_lengt
9     FROM track)
10 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
10	Fire In Space	2926593
11	The Long Patrol	2925008

SQL Analysis with their Output

Q9: Finds how much amount spent by each customers 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 alb ON alb.album_id = t.album_id  
JOIN best_selling_artist bsa ON bsa.artist_id = alb.artist_id  
GROUP BY 1,2,3,4  
ORDER BY 5 DESC;
```

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

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.

```
4
5 WITH most_popular_genre AS(
6 SELECT customer.country AS country, count(invoice_line.quantity) AS Purchases,
7 genre.genre_id As Genre_ID, genre.name AS Genre_name,
8 ROW_NUMBER() over(PARTITION BY customer.country ORDER BY count(invoice_line.quantity) DESC) as ROW_NO
9 FROM customer
10 JOIN invoice ON invoice.customer_id = customer.customer_id
11 JOIN invoice_line ON invoice_line.invoice_id=invoice.invoice_id
12 JOIN track ON track.track_id = invoice_line.track_id
13 JOIN genre ON genre.genre_id = track.genre_id
14 GROUP BY country, genre.genre_id, genre.name
15 ORDER BY country, purchases DESC
16 )
17 SELECT country, purchases, Genre_id, genre_name
18 from most_popular_genre where row_no <=1;
```

Data Output Messages Notifications



	country character varying (50) 🔒	purchases bigint 🔒	genre_id character varying (50) 🔒	genre_name character varying (120) 🔒
	Argentina	17	4	Alternative & Punk
2	Australia	34	1	Rock
8	Austria	40	1	Rock
1	Belgium	26	1	Rock

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.

WITH RECURSIVE

```
customer_with_country AS (  
  SELECT customer.customer_id, first_name, last_name, billing_country, sum(total) AS total_spending  
  FROM invoice  
  JOIN customer ON customer.customer_id = invoice.customer_id  
  GROUP BY 1,2,3,4  
  ORDER BY 2,3 DESC),
```

```
country_max_spending AS(  
  SELECT billing_country, MAX(total_spending) AS Max_spending  
  FROM customer_with_country  
  GROUP BY billing_country)
```

```
SELECT cc.billing_country, cc.total_spending, cc.first_name, cc.last_name  
FROM customer_with_country cc  
JOIN country_max_spending ms  
ON cc.billing_country = ms.billing_country  
WHERE cc.total_spending = ms.max_spending  
ORDER BY 1;
```

Data Output Messages Notifications

	billing_country character varying (30)	total_spending double precision	first_name character	last_name character
1	Argentina	39.6	Diego	Gutiérrez
2	Australia	81.18	Mark	Taylor
3	Austria	69.3	Astrid	Gruber
4	Belgium	60.389999999999999	Daan	Peeters
5	Brazil	108.89999999999998	Luis	Gonçalves
6	Canada	99.99	François	Tremblay
7	Chile	97.020000000000001	Luis	Rojas
8	Czech Republic	144.540000000000002	R	Madhav
9	Denmark	37.619999999999999	Kara	Nielsen
10	Finland	79.2	Terhi	Hämäläinen

THANK YOU