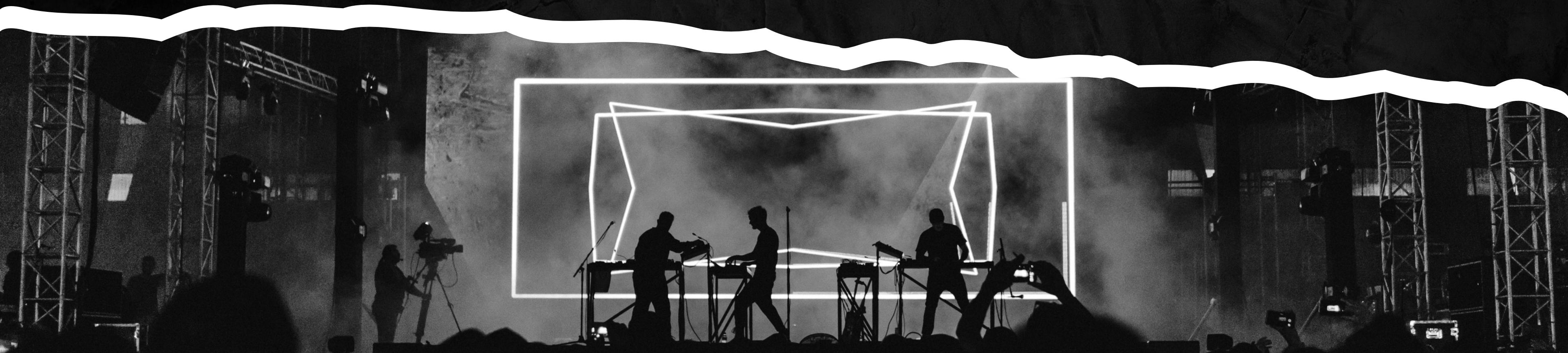


SQL ANALYSIS
MUSIC STORE

PROD. GAURAV THAKUR FT. SQL



Query Query History

```
1 Q1: Who is the senior most employee based on job title?  
2  
3 select first_name, last_name, levels from employee  
4 order by levels desc  
5 limit 1;  
6  
7  
8  
9
```

Data Output Messages Notifications



	first_name character	last_name character	levels character varying (10)
1	Mohan	...	Madan

Query Query History

```
12  
13  
14 Q2: Which countries have the most Invoices?  
15 |  
16 select billing_country, count(*) as count_of_invoice from invoice  
17 group by billing_country  
18 order by count(*) desc  
19 limit 1;  
20  
21
```

Data Output Messages Notifications



	billing_country character varying (30)	count_of_invoice bigint
1	USA	131

Query Query History

```
23  
24  
25 Q3: What are top 3 values of total invoice  
26  
27     select total from invoice  
28     order by total desc  
29     limit 3;  
30  
31  
32
```

Data Output Messages Notifications



	total double precision	🔒
1	23.759999999999998	
2		19.8
3		19.8

Query Query History

```
32  
33  
34  
35 Q4: Which city has the best customers? We would like to throw a  
36 promotional Music Festival in the city we made the most money. Write a  
37 query that returns one city that has the highest sum of invoice totals.  
38 Return both the city name & sum of all invoice totals  
39  
40 select billing_city, sum(total) as invoice_total from invoice  
41 group by billing_city  
42 order by sum(total) desc  
43 limit 1;  
44  
45  
46  
47
```

Data Output Messages Notifications



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

Query Query History

```
51
52 Q5: Who is the best customer? The customer who has spent the most
53 money will be declared the best customer. Write a query that return
54 the person who has spent the most money.
55
56 select c.customer_id, c.first_name, c.last_name, sum(i.total) as total from customer c
57 join invoice i on c.customer_id = i.customer_id
58 group by c.customer_id
59 order by total desc
60 limit 1;
61
62
```

Data Output Messages Notifications



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

Query Query History

```
70
71 Q6 : Write query to return the email, first name, last name, & Genre
72 of all Rock Music listeners. Return your list ordered alphabetically
73 by email starting with A
74
75 SELECT distinct c.email, c.first_name, c.last_name
76 FROM customer c
77 join invoice i on i.customer_id = c.customer_id
78 join invoice_line li on li.invoice_id = i.invoice_id
79 where li.track_id in (
80     select t.track_id from track t
81     join genre g on g.genre_id = t.genre_id
82     where g.name = 'Rock'
83 )
84 order by c.email
85
86
```

Data Output Messages Notifications



	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

Total rows: 59 of 59

Query complete 00:00:00.080

Query Query History

```
91 Q7: Lets invite the artists who have written the most rock music in
92 our dataset. Write a query that returns the Artist name and total
93 track count of the top 10 rock bands
94
95 SELECT a.artist_id, a.name, COUNT(*) AS number_of_songs
96 FROM track t
97 JOIN album al ON al.album_id = t.album_id
98 JOIN artist a ON a.artist_id = al.artist_id
99 JOIN genre g ON g.genre_id = t.genre_id
100 WHERE g.name = 'Rock'
101 GROUP BY a.artist_id, a.name
102 ORDER BY number_of_songs DESC
103 LIMIT 10;
104
105
```

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

Total rows: 10 of 10

Query complete 00:00:00.076

Query Query History

```
110  
111  
112 Q8: Return all the track names that have a song length longer than  
113 the average song length. Return the Name and Milliseconds for  
114 each track. Order by the song length with the longest songs listed  
115 first.  
116  
117 select name, milliseconds from track  
118 where milliseconds > (select avg(milliseconds) from track)  
119 order by milliseconds desc  
120  
121  
122  
123
```

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

Total rows: 494 of 494

Query complete 00:00:00.096

Query Query History

```
142
143 Q9: Find how much amount spent by each customer on best selling artists? Write a
144 query to return customer name, artist name and total spent
145
146 WITH cte AS (
147     SELECT a.artist_id AS artist_id, a.name AS artist_name,
148     SUM(il.unit_price * il.quantity) AS total_sales
149     FROM invoice_line il
150     JOIN track t ON t.track_id = il.track_id
151     JOIN album al ON al.album_id = t.album_id
152     JOIN artist a ON a.artist_id = al.artist_id
153     GROUP BY a.artist_id
154     ORDER BY total_sales DESC
155     LIMIT 1
156 )
157 SELECT c.customer_id, c.first_name, c.last_name, cte.artist_name,
158 SUM(il.unit_price * il.quantity) AS amount_spent
159 FROM invoice i
160 JOIN customer c ON c.customer_id = i.customer_id
161 JOIN invoice_line il ON il.invoice_id = i.invoice_id
162 JOIN track t ON t.track_id = il.track_id
163 JOIN album al ON al.album_id = t.album_id
164 JOIN cte ON cte.artist_id = al.artist_id
165 GROUP BY c.customer_id, c.first_name, c.last_name, cte.artist_name
166 ORDER BY amount_spent DESC;
167
```

Data Output Messages Notifications

	customer_id	first_name	last_name	artist_name	amount_spent
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

Total rows: 43 of 43

Query complete 00:00:00.061

Query Query History

```
172 Q10: We want to find out the most popular music Genre for each country.  
173 ( We determine the most popular genre as the genre with the highest  
174 amount of purchases. Write a query that returns each country along with  
175 the top Genre. For countries where the maximum number of purchases  
176 is shared return all Genres.  
177  
178  
179 WITH cte AS  
180 (  
181     SELECT c.country, g.name, COUNT(*) AS purchases,  
182     ROW_NUMBER() OVER(PARTITION BY c.country ORDER BY COUNT(*) DESC) AS RN  
183     FROM invoice_line il  
184     JOIN invoice i ON i.invoice_id = il.invoice_id  
185     JOIN customer c ON c.customer_id = i.customer_id  
186     JOIN track t ON t.track_id = il.track_id  
187     JOIN genre g ON g.genre_id = t.genre_id  
188     GROUP BY c.country, g.name  
189     ORDER BY c.country, COUNT(*) DESC  
190 )  
191 SELECT * FROM cte WHERE RN = 1  
192
```

Data Output Messages Notifications



	country character varying (50)	name character varying (120)	purchases bigint	rn 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

Total rows: 24 of 24

Query complete 00:00:00.073

Query Query History

```
196  
197  
198 Q11: Write a query that determines the customer that has spent the most  
199 on music for each country. Write a query that returns the country along  
200 with the top customer and how much they spent. For countries where  
201 the top amount spent is shared, provide all customers who spent this  
202 amount  
203  
204 WITH cte AS (  
205     SELECT c.customer_id, c.first_name,c.last_name,i.billing_country,SUM(i.total) AS total_spending,  
206     ROW_NUMBER() OVER(PARTITION BY billing_country ORDER BY SUM(total) DESC) AS RN  
207     FROM invoice i  
208     JOIN customer c ON c.customer_id = i.customer_id  
209     GROUP BY c.customer_id, c.first_name,c.last_name,i.billing_country  
210     ORDER BY total_spending DESC )  
211  
212 SELECT * FROM cte WHERE RN = 1  
213  
214  
215
```

Data Output Messages Notifications



	customer_id integer	first_name character	last_name character	billing_country character varying (30)	total_spending double precision	rn bigint
1	5	R	Madhav	Czech Republic	144.54000000000002	1
2	46	Hugh	O'Reilly	Ireland	114.83999999999997	1
3	58	Manoj	Pareek	India	111.86999999999999	1
4	1	Luís	Gonçalves	Brazil	108.89999999999998	1
5	34	João	Fernandes	Portugal	102.96000000000001	1
6	42	Wyatt	Girard	France	99.99	1
7	3	François	Tremblay	Canada	99.99	1
8	50	Enrique	Muñoz	Spain	98.01	1
9	53	Phil	Hughes	United Kingdom	98.01	1
10	17	Jack	Smith	USA	98.01	1

Total rows: 24 of 24

Query complete 00:00:00.075