



P O R T F O L I O P R O J E C T

ANALYZING ONLINE MUSIC STORE DATA WITH SQL



S A R A H R O D R I G U E Z



PROJECT OVERVIEW

In this project, you will explore and analyze the dataset of an online music store. The dataset contains comprehensive information spread across 11 tables:

- Album: Details about the music albums available in the store.
- Artist: Information about the artists who have created the albums.
- Customer: Data on the customers who have purchased music from the store.
- Employee: Information on the employees managing the store.
- Genre: Details about the different genres of music offered.
- Invoice: Records of purchases made by customers.
- Invoice_Line: Details of each item included in an invoice.
- Media_Type: Information on the different media formats available.
- Playlist: Data on playlists created within the store.
- Playlist_Track: Links between tracks and playlists.
- Track: Details of individual music tracks available in the store.



LEARNING OBJECTIVES

Throughout this project, you will:

1. Understand the Database Structure: Learn about the relationships between different tables and how to navigate a complex database.
2. Write SQL Queries: Develop your ability to write SQL queries to retrieve specific information from the database.
3. Analyze Business Metrics: Use SQL to answer key business questions, helping the store understand its sales, customer preferences, and overall business growth.



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

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

```
Select  
title,  
first_name,  
last_name,  
levels  
from employee  
order by levels desc  
limit 1
```



	title character varying (50)	first_name character	last_name character	levels character varying (10)
1	Senior General Manager	Mohan	Madan	L7

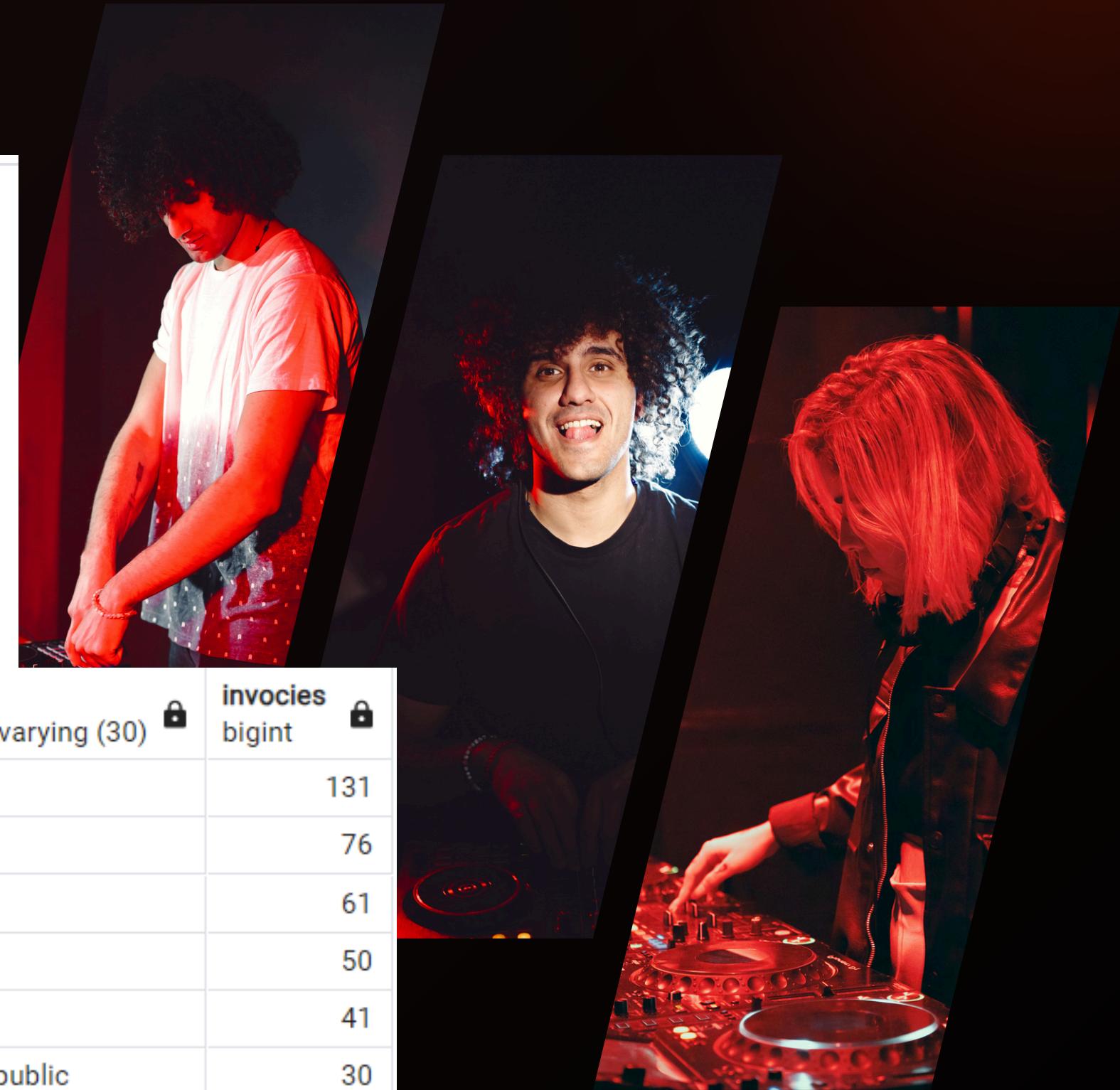
Which countries have the most invocies?

--Q2: Which countries have the most invocies?

Select

```
billing_country as Country,  
count (*) as invocies  
from invoice  
Group by Country  
Order by invocies DESC;
```

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



What are the top 3 values of total invoice?

--Q3: What are the top 3 values of total invoice?

Select

```
Round(total::numeric,2) AS total_invoice  
from invoice  
order by total DESC  
LIMIT 3;
```



	total_invoice numeric
1	23.76
2	19.80
3	19.80

Which cities has the best customers?

--Q4: Which city has the best customers?

Select

```
billing_city,  
ROUND(SUM(total::numeric),2) as invoice_total  
from invoice  
group by billing_city  
order by invoice_total DESC  
Limit 1;
```



	billing_city character varying (30)	invoice_total numeric
1	Prague	273.24

Write a query to return the email, first name, last name and gener of all rock music listeners. Return your list in alphabetically by email starting with A.

--Q5:Write a query to return the email, first name, last name & gener of all rock music listeners.
-- Return your list in alphabetically by email starting with A.

```
Select DISTINCT  
email as Email,  
first_name as Firstname,  
last_name as Lastname,  
genre.name as Name  
from customer  
JOIN invoice on invoice.customer_id = customer.customer_id  
JOIN invoice_line on invoice_line.invoice_id = invoice.invoice_id  
JOIN track on track.track_id = invoice_line.track_id  
JOIN genre on genre.genre_id = track.genre_id  
WHERE genre.name LIKE 'Rock'  
Order by email;
```



Write a query to return the email, first name, last name and genre of all rock music listeners. Return your list in alphabetically by email starting with A.

	email character varying (50)	firstname character	lastname character	name character varying (120)
1	aaronmitchell@yahoo.ca	Aaron	Mitchell	Rock
2	alero@uol.com.br	Alexandre	Rocha	Rock
3	astrid.gruber@apple.at	Astrid	Gruber	Rock
4	bjorn.hansen@yahoo.no	Bjørn	Hansen	Rock
5	camille.bernard@yahoo.fr	Camille	Bernard	Rock
6	daan_peeters@apple.be	Daan	Peeters	Rock
7	diego.gutierrez@yahoo.ar	Diego	Gutiérrez	Rock
8	dmiller@comcast.com	Dan	Miller	Rock
9	dominiquelefebvre@gmail.c...	Dominique	Lefebvre	Rock
10	edfrancis@yachoo.ca	Edward	Francis	Rock
11	eduardo@woodstock.com.br	Eduardo	Martins	Rock
12	ellie.sullivan@shaw.ca	Ellie	Sullivan	Rock
13	emma_jones@hotmail.com	Emma	Jones	Rock
14	enrique_munoz@yahoo.es	Enrique	Muñoz	Rock
15	fernadaramos4@uol.com.br	Fernanda	Ramos	Rock
16	fharris@google.com	Frank	Harris	Rock





Return all the track names that have a song lenght longer than the average song lenght.
Return the name and miliseconds for each track. Order by the song lenght
with the longest song listed first.

```
/*Q6: Return all the track names that have a song lenght longer than the average song lenght.  
Return the name and miliseconds for each track. Order by the song lenght  
with the longest song listed first.
```

```
*/
```

Select

```
name as Track_name,  
milliseconds  
From track  
Where milliseconds > (  
    Select AVG(milliseconds) as avg_track_leght  
    From track)  
  
order by milliseconds DESC;
```

Return all the track names that have a song lenght longer than the average song lenght.
Return the name and miliseconds for each track. Order by the song lenght
with the longest song listed first.

	track_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
12	The Magnificent Warriors	2924716
13	The Living Legend, Pt. 1	2924507
14	The Gun On Ice Planet Zero, Pt. 2	2924341
15	The Hand of God	2924007
16	Experiment In Terra	2923548
17	War of the Gods, Pt. 2	2923381



Find the amount spent by each customer on artist? Write a query to return the customer name, artist name and total spent.

```
/*Q7: Find the amount spent by each customer on artist? Write a query to return customer name,  
artist name and total spent
```

```
*/  
  
WITH best_selling_artist 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  
        artist.artist_id, artist.name  
    ORDER BY  
        Total_Sales DESC  
    LIMIT 1  
)
```

```
SELECT  
    c.customer_id,  
    c.first_name,  
    c.last_name,  
    bsa.artist_name,  
    CAST(SUM(il.unit_price * il.quantity) AS DECIMAL(10, 2)) AS Total_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  
    c.customer_id, c.first_name, c.last_name, bsa.artist_name  
ORDER BY  
    Total_Spent DESC;
```

Find the amount spent by each customer on artist? Write a query to return the customer name, artist name and total spent.

	customer_id	first_name	last_name	artist_name	total_spent
	integer	character	character	character varying (120)	numeric (10,2)
1	46	Hugh	O'Reilly	Queen	27.72
2	38	Niklas	Schröder	Queen	18.81
3	3	François	Tremblay	Queen	17.82
4	34	João	Fernandes	Queen	16.83
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	2.97
12	54	Steve	Murray	Queen	2.97
13	31	Martha	Silk	Queen	2.97
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



Find 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.

```
/*Q7: Find the most popular music gener for each country. We detrmine 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 purchaises is shared return  
all genres.  
*/  
  
WITH most_popular_genre AS(  
SELECT customer.country AS Country, COUNT(invoice_line.quantity) AS Purchases,  
genre.genre_id AS Genre_ID, genre.name AS Genre_Name,  
ROW_NUMBER() OVER(PARTITION BY customer.country ORDER BY COUNT(invoice_line.quantity) DESC) as Row_No  
FROM customer  
  
JOIN invoice ON invoice.customer_id = customer.customer_id  
JOIN invoice_line ON invoice_line.invoice_id = invoice.invoice_id  
JOIN track ON track.track_id = invoice_line.track_id  
JOIN genre ON genre.genre_id= track.genre_id  
GROUP BY Country, genre.genre_id, genre.name  
ORDER BY Country, Purchases DESC  
)  
SELECT Country, Purchases, Genre_ID, Genre_Name  
FROM most_popular_genre WHERE Row_No <= 1;
```



Find 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.

	country character varying (50)	purchases bigint	genre_id character varying (50)	genre_name character varying (120)
1	Argentina	17	4	Alternative & Punk
2	Australia	34	1	Rock
3	Austria	40	1	Rock
4	Belgium	26	1	Rock
5	Brazil	205	1	Rock
6	Canada	333	1	Rock
7	Chile	61	1	Rock
8	Czech Republic	143	1	Rock
9	Denmark	24	1	Rock
10	Finland	46	1	Rock
11	France	211	1	Rock
12	Germany	194	1	Rock
13	Hungary	44	1	Rock
14	India	102	1	Rock
15	Ireland	72	1	Rock
16	Italy	35	1	Rock
17	Netherlands	33	1	Rock

