Digital Music Store Insights: A Comprehensive Business Intelligence Report

Date: June 19, 2025

1. Synopsis

This report details a comprehensive data analysis project conducted on a fictional music store's sales and customer database. Leveraging SQL for robust data extraction and transformation, and Power BI for interactive visualization, the project aims to uncover key business insights regarding customer purchasing behaviour, product performance (tracks & albums), and regional sales trends. The findings presented herein provide actionable intelligence for strategic decision-making, including targeted marketing, inventory optimization, and customer relationship management.

2. Project Goals & Objectives

The primary objective of this project was to transform raw transactional data into meaningful business insights. This was achieved by addressing the following specific questions:

- **Employee Structure:** Identify the most senior employee based on organizational hierarchy.
- **Geographical Sales Performance:** Determine which countries and cities contribute most to revenue.
- **Customer Segmentation:** Identify high-value customers and analyse their spending habits.
- **Content Performance:** Uncover best-selling tracks, albums, and the most popular music genres.
- **Artist Performance:** Identify top-performing artists and their contribution to specific genres.
- **Personalized Insights:** Understand customer spending per artist and popular genres per country for targeted campaigns.

3. Data Source & Schema

The analysis utilized a relational database representing the music store's operations. The database comprises several interconnected tables, enabling a holistic view of sales, customer, and music catalog data.

Key Tables Include:

- **Employee**: Employee hierarchy and contact information.
- **Customer**: Customer demographics and contact details.
- **Invoice**: Transaction-level data (invoice ID, total amount, billing address).

- **InvoiceLine**: Details of each item within an invoice (quantity, unit price, track ID).
- **Track**: Song details (name, genre, album, duration).
- **Album**: Album details (title, associated artist).
- **Artist**: Artist names and IDs.
- **Genre**: Music genre classifications.
- **MediaType**: Format of the music tracks.
- **PlaylistTrack:** Playlist with tracks.

(Refer to Database Schema (ERD) file in the repository for a visual representation of table relationships.)

4. Methodology

The project followed a two-phase analytical approach:

4.1. Data Extraction & Transformation (SQL)

All data manipulation and insight generation were performed using **PostgreSQL** queries within **PgAdmin4**. Key SQL techniques employed include:

- **Aggregations (SUM, COUNT, AVG):** To calculate totals, counts, and averages for various metrics (e.g., total spending, number of invoices, average song length).
- **Joins (INNER JOIN):** To combine data from multiple tables based on common columns (e.g., linking customer purchases to specific tracks and artists).
- **Subqueries:** Used to filter data based on the results of another query (e.g., finding rock music listeners by first identifying rock track IDs).
- **Common Table Expressions (CTEs):** Employed for complex, multi-step queries, improving readability and maintainability (e.g., identifying the best-selling artist first, then analysing customer spending on that artist).
- **Window Functions (ROW_NUMBER()):** Critical for ranking and partitioning data, enabling the identification of top performers within specific groups (e.g., most popular genre per country, top-spending customer per country).
- **GROUP BY and ORDER BY:** Essential for grouping data for aggregations and sorting results for easy interpretation.

The SQL script (Music Store Analysis-Queries.sql in the repository) contains all queries used to derive the insights.

4.2. Data Visualization & Reporting (Power BI)

The processed data and generated insights from SQL queries were then imported into **Power BI Desktop**. Power BI was utilized to create an interactive and dynamic dashboard, transforming raw numbers into compelling visual narratives. This phase involved:

- **Data Model Design:** Establishing relationships between imported tables to facilitate cross-filtering and accurate calculations.
- **Measure & Column Creation:** Defining custom calculations (DAX formulas) to derive additional metrics not directly available from SQL queries.
- **Interactive Visualizations:** Employing a variety of visuals suitable for different data insights:
    - **Bar Charts:** For comparing values across categories (e.g., sales by country, top artists).
    - **Tables:** For displaying detailed data (e.g., customer spending).
    - **Cards:** For showing KPIs (e.g. Senior most employee, Best-selling artist name).
    - **Maps:** For geographical insights (e.g., sales distribution across countries).
- **Dashboard Layout & Design:** Creating an intuitive and user-friendly interface that facilitates exploration and understanding of the data story.

The Power BI file (Music Store Dashboard.pbix in repository) provides the interactive dashboard.

5. Key Insights & Findings

The analysis yielded several actionable insights, categorized by the business questions addressed:

5.1. Organizational & Market Overview

- **Senior Employee Identification:** The query successfully identified the most senior employee based on their levels field, providing clarity on internal hierarchy.
- **Top Invoice Countries:** Analysis revealed the leading countries by invoice count, indicating primary markets. This was visualized in Power BI using a bar chart showing country vs. number of invoices, immediately highlighting top geographies.
- **High-Value Invoices:** The top invoice totals were identified, showcasing the scale of individual high-value transactions.
- **Best Performing City (Revenue):** The city generating the highest total revenue was pinpointed, which is a critical insight for local promotional strategies. The Power BI dashboard features a clear KPI card displaying this city and its total revenue.

5.2. Customer Behaviour & Content Performance

- **Best Customer:** The customer with the highest overall spending was identified, enabling focused loyalty initiatives. The name and total spending are prominently displayed in the Power BI dashboard.
- **Rock Music Listeners:** A distinct list of all customers who have purchased Rock music was compiled, providing a target audience for genre-specific marketing campaigns.
- **Top 10 Rock Artists:** The artists with the most rock music tracks were identified, crucial for understanding genre-specific content popularity and potential partnerships. This is presented as a bar chart in Power BI, allowing users to see top artists within selected genre.
- **Longer-Than-Average Tracks:** All tracks exceeding the average song length were identified, providing insights into content duration preferences.

5.3. Advanced Customer & Genre Analysis

- **Customer Spending per Artist:** Detailed analysis was performed to show how much each customer spent on each artist. While granular, this data can feed personalized recommendation engines.
- **Spending on Best-Selling Artist:** The project successfully identified the overall best-selling artist and calculated each customer's expenditure specifically on that artist. This highlights the reach and impact of the top performer across the customer base.
- **Most Popular Genre Per Country:** A crucial insight revealing the top music genre (by purchase count) for each country. This is visualized effectively in Power BI using interactive map, where clicking on a country reveals its top genre, for quick country comparisons of genre popularity. This supports geographically tailored content promotion and inventory management.
- **Top Customer Per Country:** The customer who spent the most within each country was identified. This allows for localized VIP customer programs. The Power BI dashboard includes a table visualization that lists each country and its top-spending customer along with their total spend, offering direct insights for regional marketing efforts.

6. Tools & Technologies Utilized

- **Database Management System:** PostgreSQL
- **Database Client:** PgAdmin4
- **Query Language:** SQL
- **Version Control:** Git & GitHub
- **Data Visualization & Business Intelligence:** Power BI Desktop

7. Skills Demonstrated

This project showcases a robust set of skills critical for a data analyst:

- **Advanced SQL Querying:** Proficiency in joins, subqueries, CTEs, aggregations, and window functions.
- **Database Understanding:** Ability to navigate and query relational databases effectively.
- **Business Acumen:** Translating abstract business questions into concrete data analysis tasks.
- **Data Cleaning & Transformation:** Preparing raw data for analysis.
- **Data Visualization:** Designing and building interactive dashboards in Power BI.
- **Data Storytelling:** Communicating complex insights clearly and concisely through visuals and reports.
- **Version Control:** Managing project code and documentation using Git/GitHub.
- **Problem-Solving:** Analytical thinking to derive actionable insights from data.

8. Conclusion & Future Scope

This Music Store Analysis project successfully extracted valuable business intelligence from transactional data, providing insights into customer behaviour, content popularity, and geographical trends. The interactive Power BI dashboard serves as a powerful tool for business stakeholders to explore these insights dynamically.

Potential Future Enhancements and Analysis:

- **Time-Series Analysis:** Analyse sales trends over time (monthly, quarterly, yearly) to identify seasonal patterns or growth.
- **Customer Lifetime Value (CLTV):** Calculate CLTV to better segment and target customers.
- **Churn Prediction:** Identify factors that might lead to customer churn.
- **Recommendation Engine:** Develop a more sophisticated system for personalized track/artist recommendations based on purchase history and genre preferences.
- **Inventory Optimization:** Analyse track/album availability against sales data to optimize stock levels.

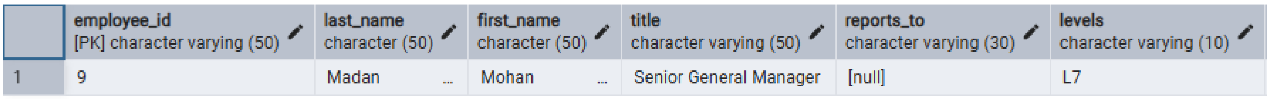
This project serves as a strong foundation for continued data-driven decision-making within the music store context.

Questions, Queries & Output:

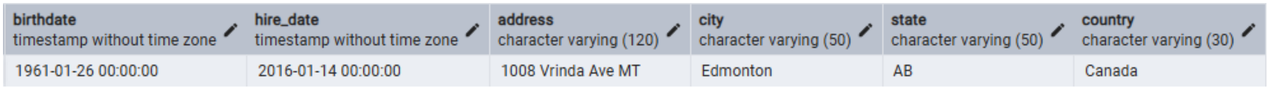
--SET 1

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

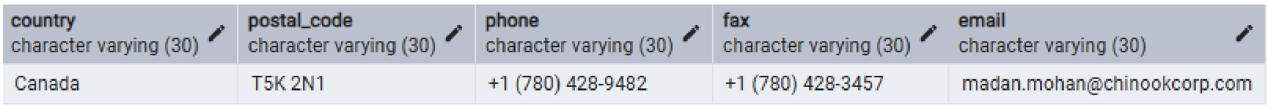
select * from employee
order by levels desc
limit 1;



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



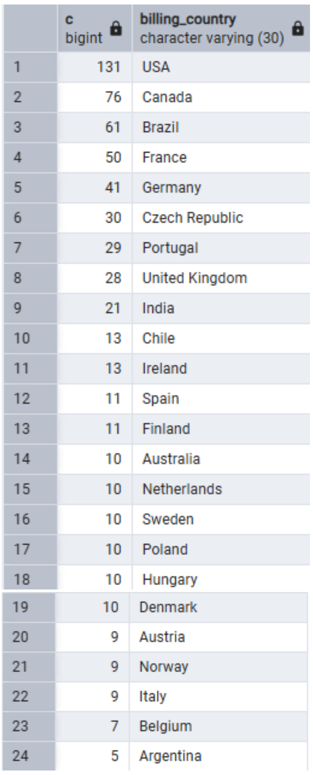
birthdate timestamp without time zone	hire_date timestamp without time zone	address character varying (120)	city character varying (50)	state character varying (50)	country character varying (30)
1961-01-26 00:00:00	2016-01-14 00:00:00	1008 Vrinda Ave MT	Edmonton	AB	Canada



country character varying (30)	postal_code character varying (30)	phone character varying (30)	fax character varying (30)	email character varying (30)
Canada	T5K 2N1	+1 (780) 428-9482	+1 (780) 428-3457	madan.mohan@chinookcorp.com

Q2: Which countries have the most Invoices?

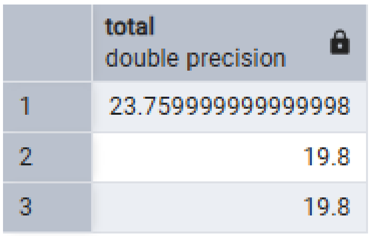
select count(*) as c, billing_country
from invoice
group by billing_country
order by c desc;



	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
9	21	India
10	13	Chile
11	13	Ireland
12	11	Spain
13	11	Finland
14	10	Australia
15	10	Netherlands
16	10	Sweden
17	10	Poland
18	10	Hungary
19	10	Denmark
20	9	Austria
21	9	Norway
22	9	Italy
23	7	Belgium
24	5	Argentina

Q3: What are top 3 values of total invoice?

```
select total from invoice
order by total desc
limit 3;
```

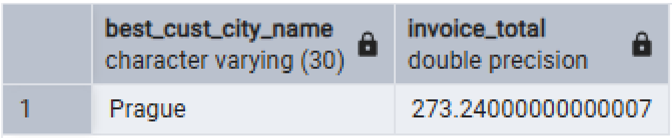


	total double precision
1	23.759999999999998
2	19.8
3	19.8

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 the highest sum of invoice totals.
Return both the city name & sum of all invoice totals.

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

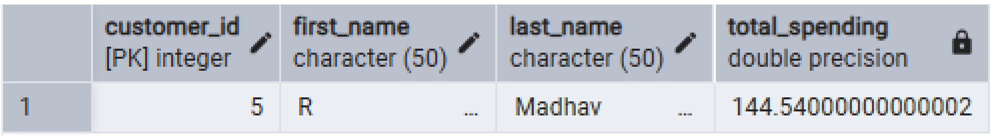


	best_cust_city_name character varying (30)	invoice_total double precision
1	Prague	273.24000000000007

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_spending
from customer
JOIN invoice on customer.customer_id=invoice.customer_id
group by customer.customer_id
order by total_spending desc
limit 1;
```



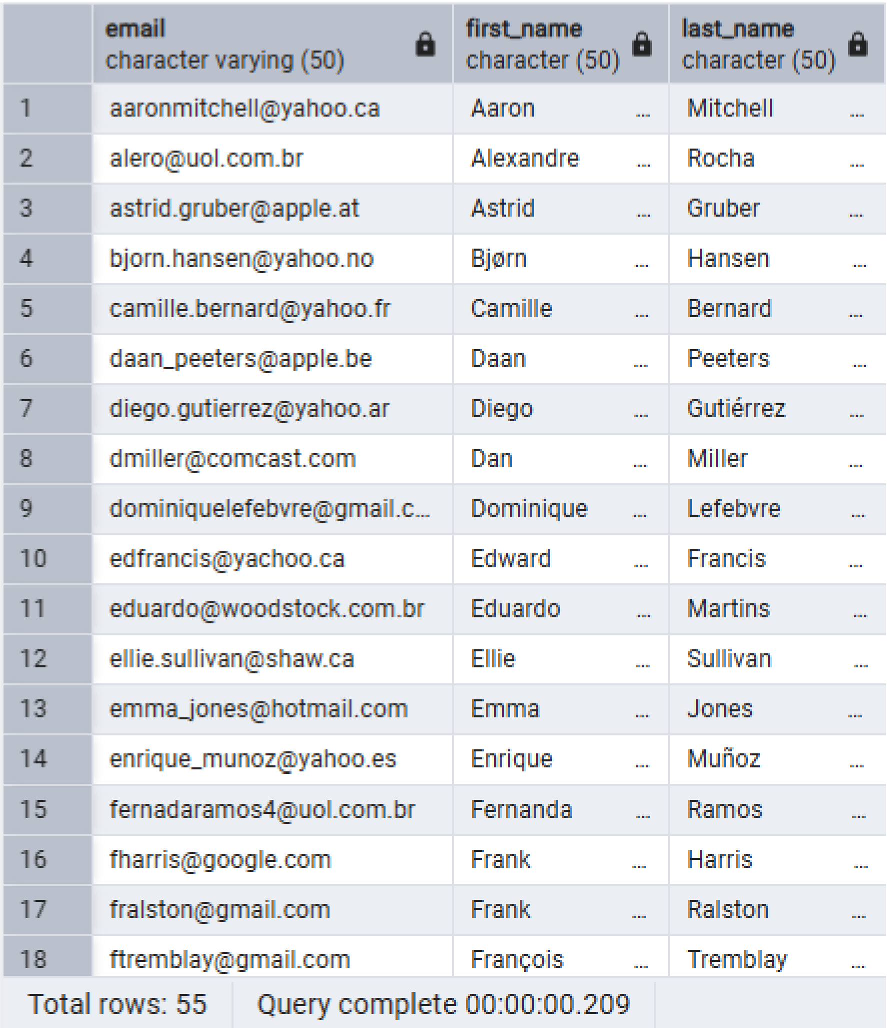
	customer_id [PK] integer	first_name character (50)	last_name character (50)	total_spending double precision
1	5	R ...	Madhav ...	144.54000000000002

--SET 2

Q1: 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 customer.email, customer.first_name, customer.last_name
from customer
join invoice on customer.customer_id=invoice.invoice_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 customer.email;
```



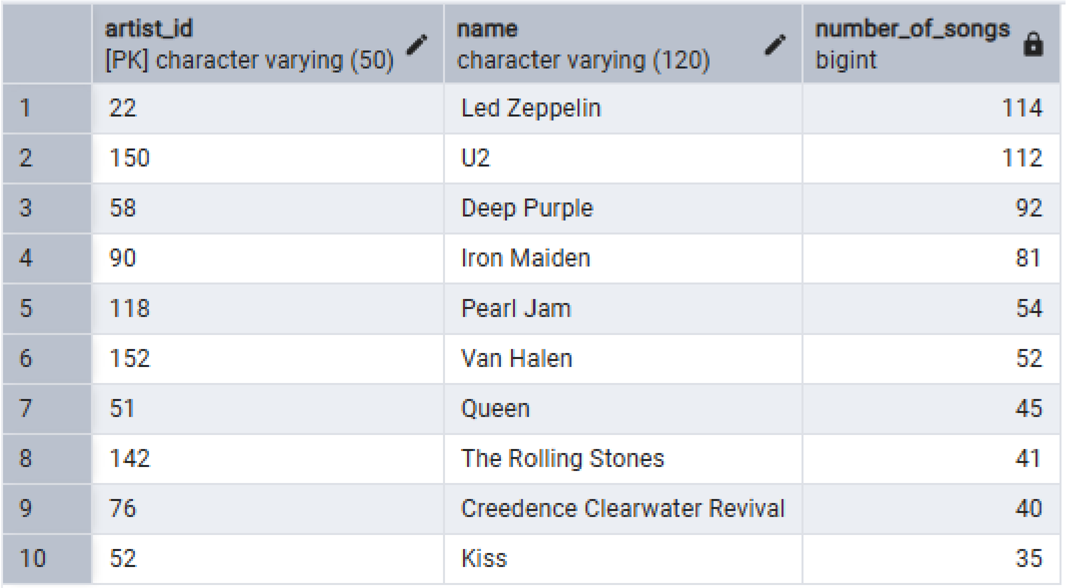
	email character varying (50)	first_name character (50)	last_name character (50)
1	aaronmitchell@yahoo.ca	Aaron	Mitchell
2	alero@uol.com.br	Alexandre	Rocha
3	astrid.gruber@apple.at	Astrid	Gruber
4	bjorn.hansen@yahoo.no	Bjørn	Hansen
5	camille.bernard@yahoo.fr	Camille	Bernard
6	daan_peeters@apple.be	Daan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

Total rows: 55   Query complete 00:00:00.209

Q2: 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.

```
select artist.artist_id, artist.name, count(artist.artist_id) as number_of_songs
from track
join album on album.album_id=track.album_id
join artist on artist.artist_id=album.artist_id
join genre on genre.genre_id=track.genre_id
where genre.name like 'Rock'
group by artist.artist_id
order by number_of_songs desc
limit 10;
```

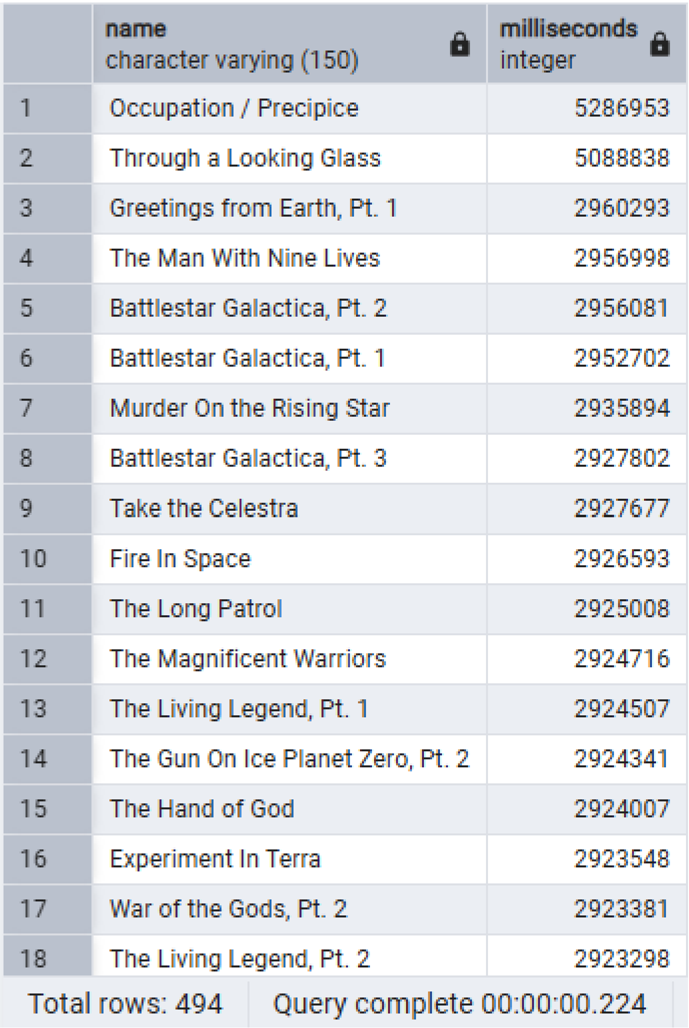


	artist_id [PK] character varying (50)	name character varying (120)	number_of_songs bigint
1	22	Led Zeppelin	114
2	150	U2	112
3	58	Deep Purple	92
4	90	Iron Maiden	81
5	118	Pearl Jam	54
6	152	Van Halen	52
7	51	Queen	45
8	142	The Rolling Stones	41
9	76	Creedence Clearwater Revival	40
10	52	Kiss	35

Q3: 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.

```
select name,track.milliseconds
from track
where track.milliseconds>
(
select avg(track.milliseconds) as avg_track_length
from track
)
order by track.milliseconds desc;
```



	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
18	The Living Legend, Pt. 2	2923298

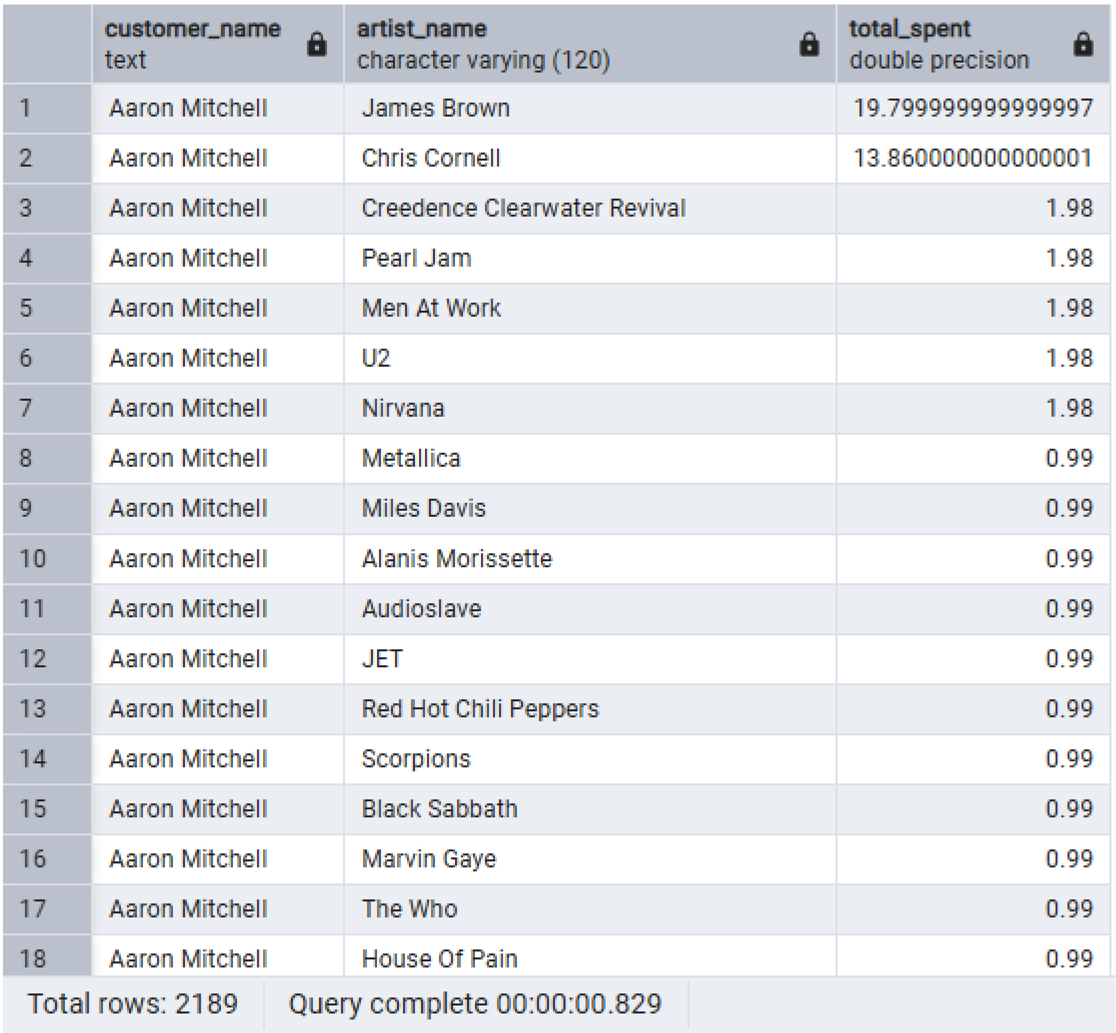
Total rows: 494 Query complete 00:00:00.224

--SET 3

Q1a: Find how much amount spent by each customer on artists?

Write a query to return customer name, artist name and total spent.

```
select c.first_name || ' ' || c.last_name as customer_name,
   ar.name as artist_name, sum(il.unit_price * il.quantity) as total_spent
from customer as c
join invoice as i on c.customer_id = i.customer_id
join invoice_line as il on i.invoice_id = il.invoice_id
join track as t on il.track_id = t.track_id
join album as al on t.album_id = al.album_id
join artist as ar on al.artist_id = ar.artist_id
group by c.customer_id, customer_name, ar.name
order by customer_name, total_spent desc;
```



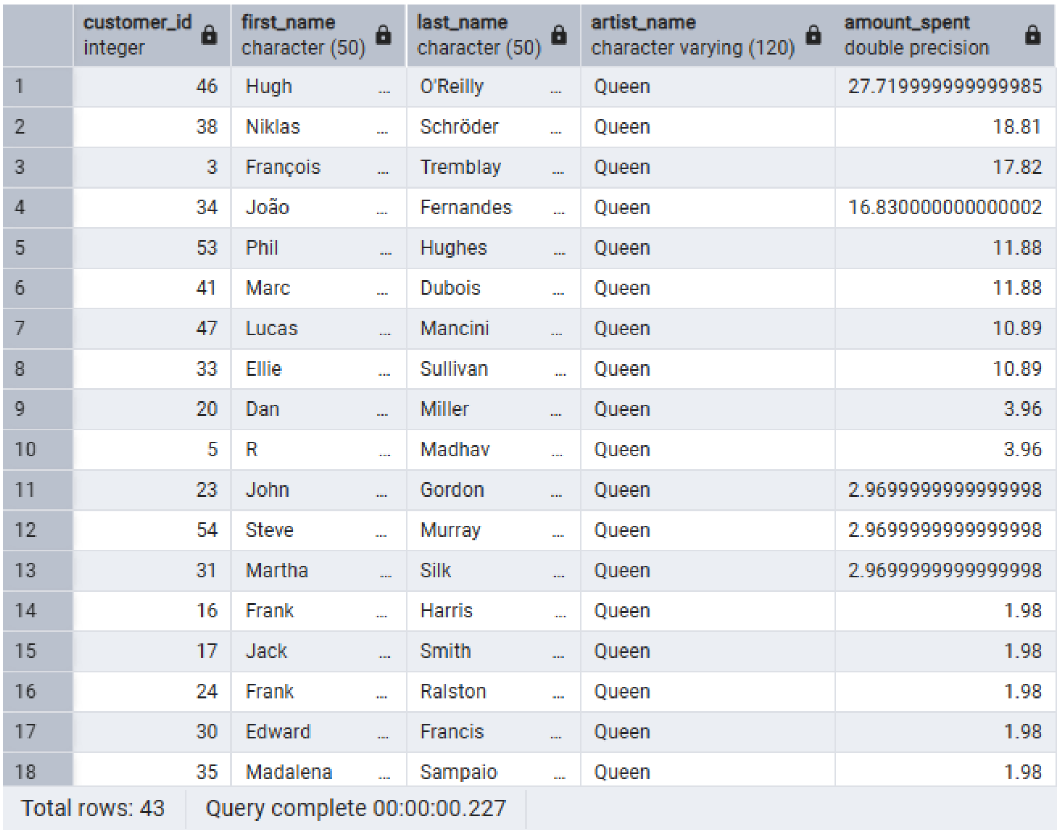
	customer_name text	artist_name character varying (120)	total_spent double precision
1	Aaron Mitchell	James Brown	19.799999999999997
2	Aaron Mitchell	Chris Cornell	13.860000000000001
3	Aaron Mitchell	Creedence Clearwater Revival	1.98
4	Aaron Mitchell	Pearl Jam	1.98
5	Aaron Mitchell	Men At Work	1.98
6	Aaron Mitchell	U2	1.98
7	Aaron Mitchell	Nirvana	1.98
8	Aaron Mitchell	Metallica	0.99
9	Aaron Mitchell	Miles Davis	0.99
10	Aaron Mitchell	Alanis Morissette	0.99
11	Aaron Mitchell	Audioslave	0.99
12	Aaron Mitchell	JET	0.99
13	Aaron Mitchell	Red Hot Chili Peppers	0.99
14	Aaron Mitchell	Scorpions	0.99
15	Aaron Mitchell	Black Sabbath	0.99
16	Aaron Mitchell	Marvin Gaye	0.99
17	Aaron Mitchell	The Who	0.99
18	Aaron Mitchell	House Of Pain	0.99

Total rows: 2189 Query complete 00:00:00.829

Q1b: Find how much amount spent by each customer on best selling artist?

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;
```

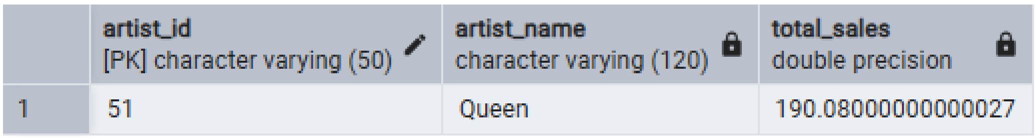


	customer_id integer	first_name character (50)	last_name character (50)	artist_name character varying (120)	amount_spent double precision
1	46	Hugh	O'Reilly	Queen	27.719999999999985
2	38	Niklas	Schröder	Queen	18.81
3	3	François	Tremblay	Queen	17.82
4	34	João	Fernandes	Queen	16.830000000000002
5	53	Phil	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
18	35	Madalena	Sampaio	Queen	1.98

Total rows: 43 Query complete 00:00:00.227

Q1c: Best Selling Artist

```
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;
```

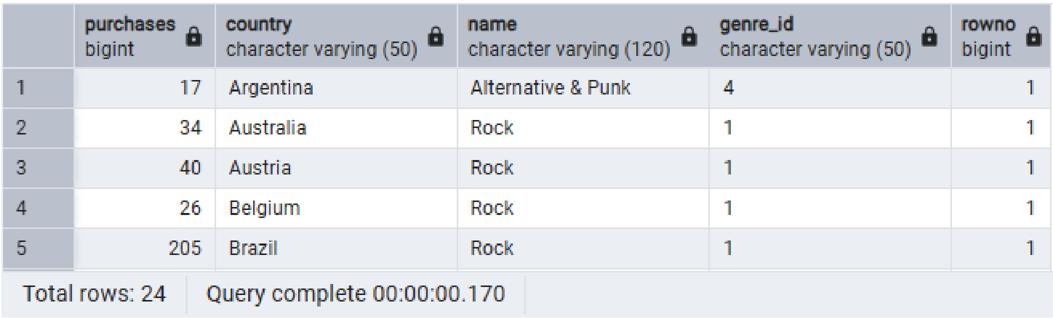


	artist_id [PK] character varying (50)	artist_name character varying (120)	total_sales double precision
1	51	Queen	190.08000000000027

Q2: 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.

```
with popular_genre as
(
  select count(il.quantity) as purchases, c.country, g.name, g.genre_id,
        row_number() over(partition by c.country order by count(il.quantity) desc) as rowno
  from invoice_line as il
        join invoice as i on i.invoice_id = il.invoice_id
        join customer as c on c.customer_id = i.customer_id
        join track as t on t.track_id = il.track_id
        join genre as g on g.genre_id = t.genre_id
        group by 2,3,4
        order by 2 asc, 1 desc
)
select * from popular_genre where rowno <= 1;
```



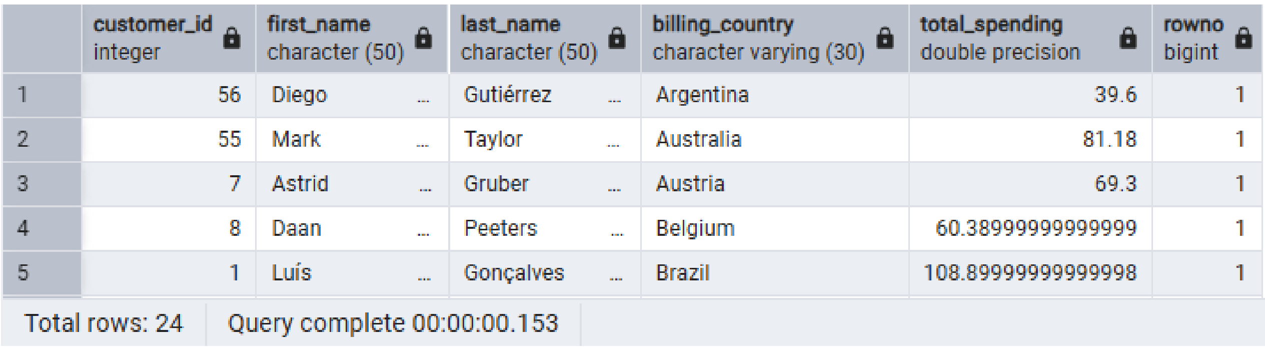
	purchases bigint	country character varying (50)	name character varying (120)	genre_id character varying (50)	rowno bigint
1	17	Argentina	Alternative & Punk	4	1
2	34	Australia	Rock	1	1
3	40	Austria	Rock	1	1
4	26	Belgium	Rock	1	1
5	205	Brazil	Rock	1	1

Total rows: 24 Query complete 00:00:00.170

Q3: 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 customer_with_country as (
                    select c.customer_id,first_name,last_name,billing_country,sum(total) as
total_spending,
             row_number() over(partition by billing_country order by sum(total) desc) as rowno
                    from invoice as i
                    join customer as c on c.customer_id = i.customer_id
                    group by 1,2,3,4
                    order by 4 asc,5 desc)
select * from customer_with_country where rowno <= 1;
```



	customer_id integer	first_name character (50)	last_name character (50)	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

Total rows: 24   Query complete 00:00:00.153