



MUSIC STORE ANALYSIS USING SQL

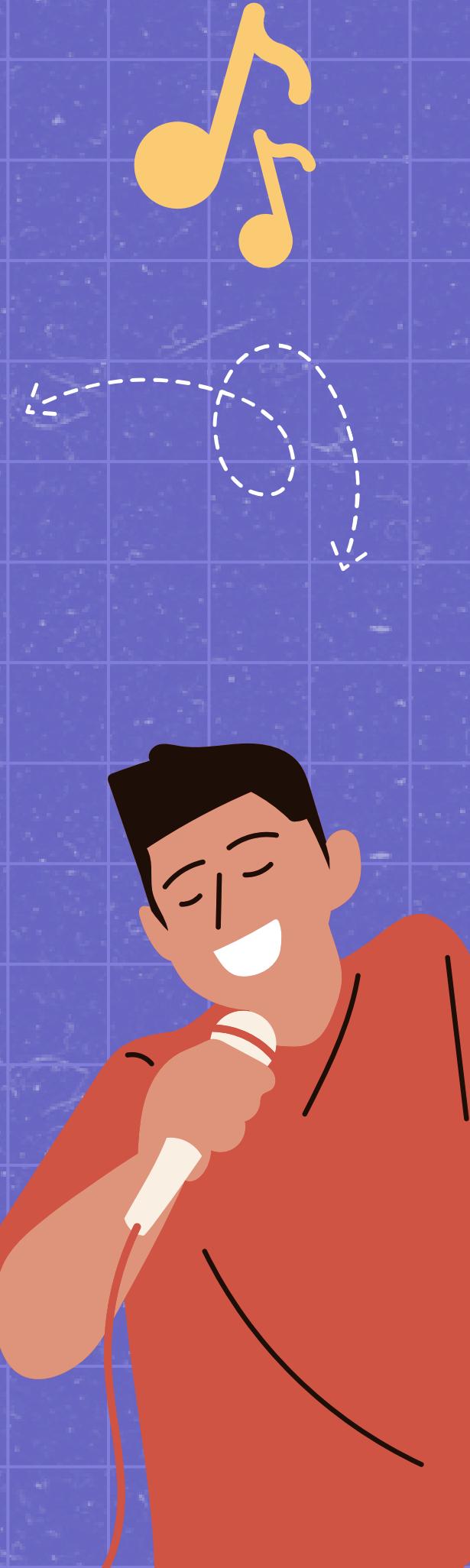
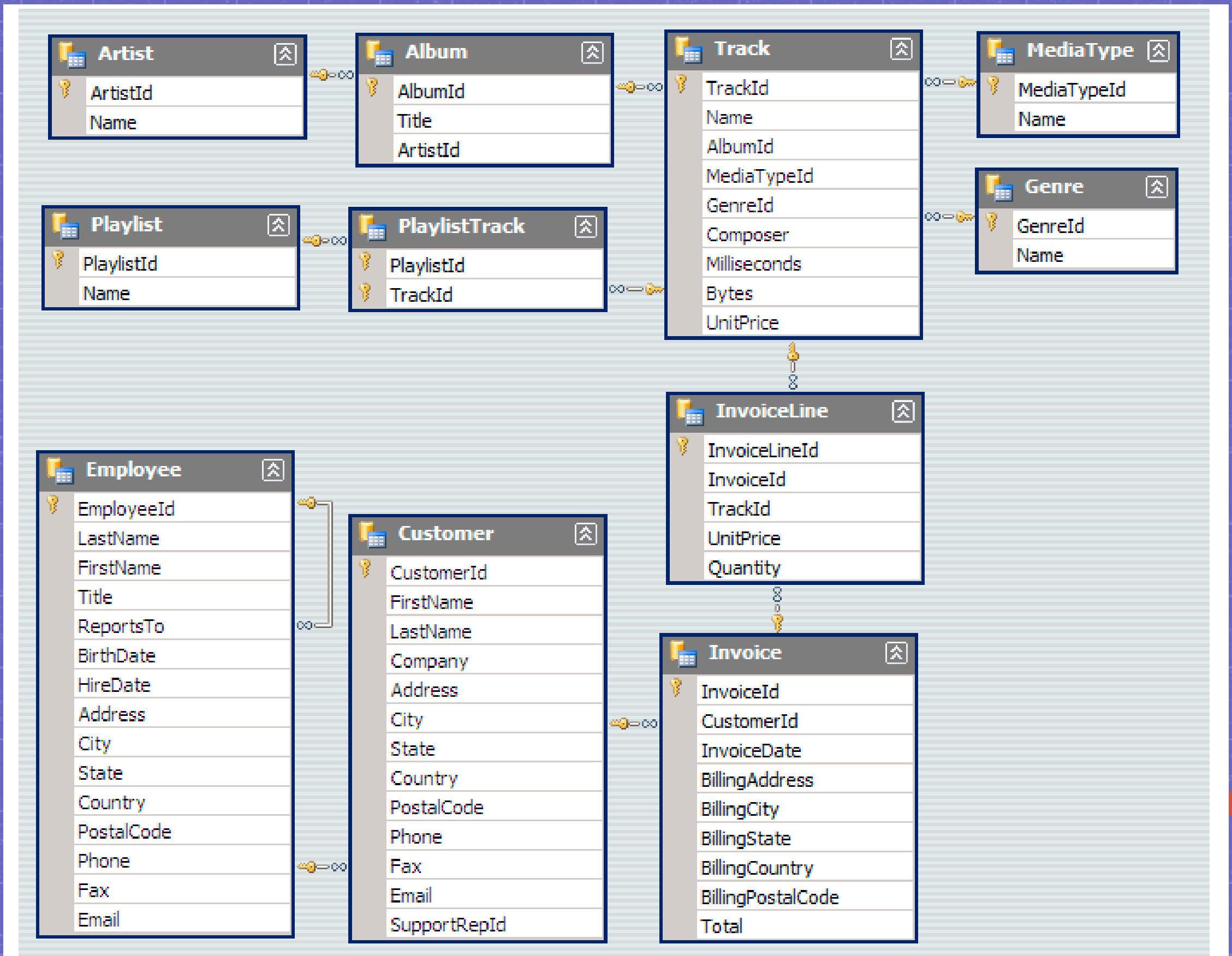


Introduction



This project delves into music data to assess the performance of various tracks, artists, and genres. By analyzing patterns in listener behavior, it aims to identify what resonates most with audiences and what trends are emerging in the industry. The insights gained will be used to make informed decisions on marketing strategies, playlist curation, and artist promotions.

ER Diagram



Who is the senior most employee based on job title?

```
SELECT * FROM employee  
ORDER BY levels DESC  
LIMIT 1;
```

Data Output Messages Notifications

SQL

	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

Which countries have the most Invoices?

```
SELECT COUNT(*) as cnt, billing_country  
FROM invoice  
GROUP BY billing_country ORDER BY cnt DESC;
```

	cnt 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

What are top 3 values of total invoice?

```
SELECT total FROM invoice  
ORDER BY total DESC  
LIMIT 3
```

Data Output Messages Notifications

	total	
1	23.759999999999998	
2	19.8	
3	19.8	



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 SUM(total) as invoice_total, billing_city  
2   FROM invoice  
3 GROUP BY billing_city ORDER BY invoice_total DESC;
```

Data Output Messages Notifications

	invoice_total double precision	billing_city character varying (30)
1	273.24000000000007	Prague
2	169.29	Mountain View
3	166.32	London
4	158.4	Berlin
5	151.47	Paris
6	129.69	São Paulo
7	114.83999999999997	Dublin



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;
```

Data Output Messages Notifications

SQL

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



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_line.invoice_id = invoice.invoice_id
WHERE track_id IN(
    SELECT track_id
    FROM track JOIN genre
    ON genre.genre_id = track.genre_id
    WHERE genre.name LIKE 'Rock'
)
ORDER BY email;
```

Data Output Messages Notifications

SQL

	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



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_song
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_song DESC
LIMIT 10;
```

Data Output Messages Notifications

SQL

	artist_id [PK] character varying (50)	name character varying (120)	number_of_song 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

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
  FROM track
 WHERE milliseconds > (
   SELECT AVG(milliseconds) AS avg_track_length
   FROM track
 )
 ORDER BY milliseconds DESC;
```

Data Output Messages Notifications

SQL

	name	milliseconds
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



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

```

1  WITH best_selling_artist AS (
2      SELECT artist.artist_id AS artist_id, artist.name AS artist_name,
3          SUM(invoice_line.unit_price*invoice_line.quantity) AS total_sales
4      FROM invoice_line
5      JOIN track ON track.track_id = invoice_line.track_id
6      JOIN album ON album.album_id = track.album_id
7      JOIN artist ON artist.artist_id = album.artist_id
8      GROUP BY 1
9      ORDER BY 3 DESC
10     LIMIT 1
11 )
12     SELECT c.customer_id, c.first_name, c.last_name, bsa.artist_name,
13         SUM(il.unit_price*il.quantity) AS amount_spent
14     FROM invoice i
15     JOIN customer c ON c.customer_id = i.customer_id
16     JOIN invoice_line il ON il.invoice_id = i.invoice_id
17     JOIN track t ON t.track_id = il.track_id
18     JOIN album alb ON alb.album_id = t.album_id
19     JOIN best_selling_artist bsa ON bsa.artist_id = alb.artist_id
20     GROUP BY 1,2,3,4
21     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



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(invoice_line.quantity) AS purchases, customer.country, genre.name, genre.genre_id,
    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
```

	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
6	333	Canada	Rock	1	1
7	61	Chile	Rock	1	1





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

```
1 ✓ WITH Customer_with_country AS (
2     SELECT customer.customer_id, first_name, last_name, billing_country, SUM(total) AS total_spending,
3         ROW_NUMBER() OVER(PARTITION BY billing_country ORDER BY SUM(total) DESC) AS RowNo
4     FROM invoice
5     JOIN customer ON customer.customer_id = invoice.customer_id
6     GROUP BY 1,2,3,4
7     ORDER BY 4 ASC,5 DESC)
8 SELECT * FROM Customer_with_country WHERE RowNo <= 1
```



Data Output Messages Notifications

customer_id first_name last_name billing_country total_spending rowno

	customer_id	first_name	last_name	billing_country	total_spending	rowno
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	108.8999999999998	1
6	3	François	Tremblay	Canada	99.99	1
7	57	Luis	Rojas	Chile	97.02000000000001	1

Thank You

Keep up to date with the latest music news