
Search
Home
Library
Our Music Playlists
SQL Project:
Mayank's
Music Company
A Project on Music Store Sales Analysis
https://www.linkedin.com/in/mayank-srivastava-6a8421105/

Project Summary

01 OBJECTIVE

3:15min

To analyze sales data from a music store to gain insights into sales performance, customer behavior, and product trends.

02 ABOUT THE DATASET

3:15min

- Company has 3500+ tracks, 4757 transactions, across 700 invoices.
- The company has shared 11 csv files to complete the required analysis.

03 FEATURES OF THE TOPIC

3:15min

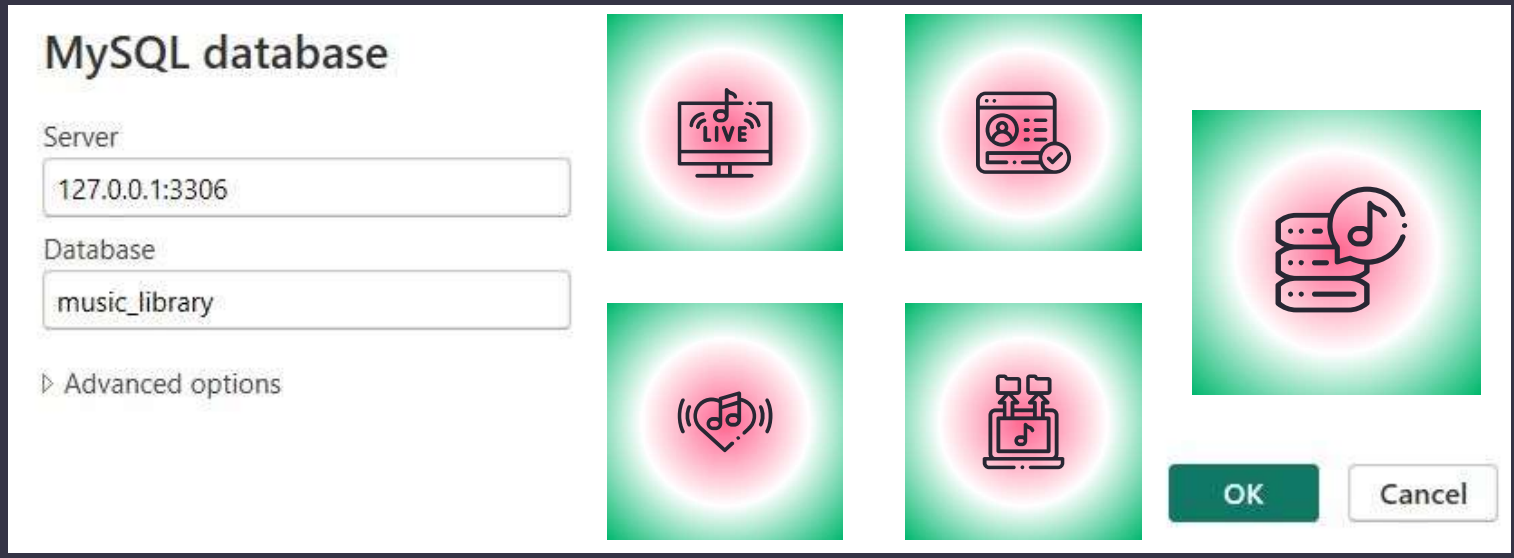
- Sub Query
- Group By, Joins
- Rank
- Visualizations

04 TOOLS USED

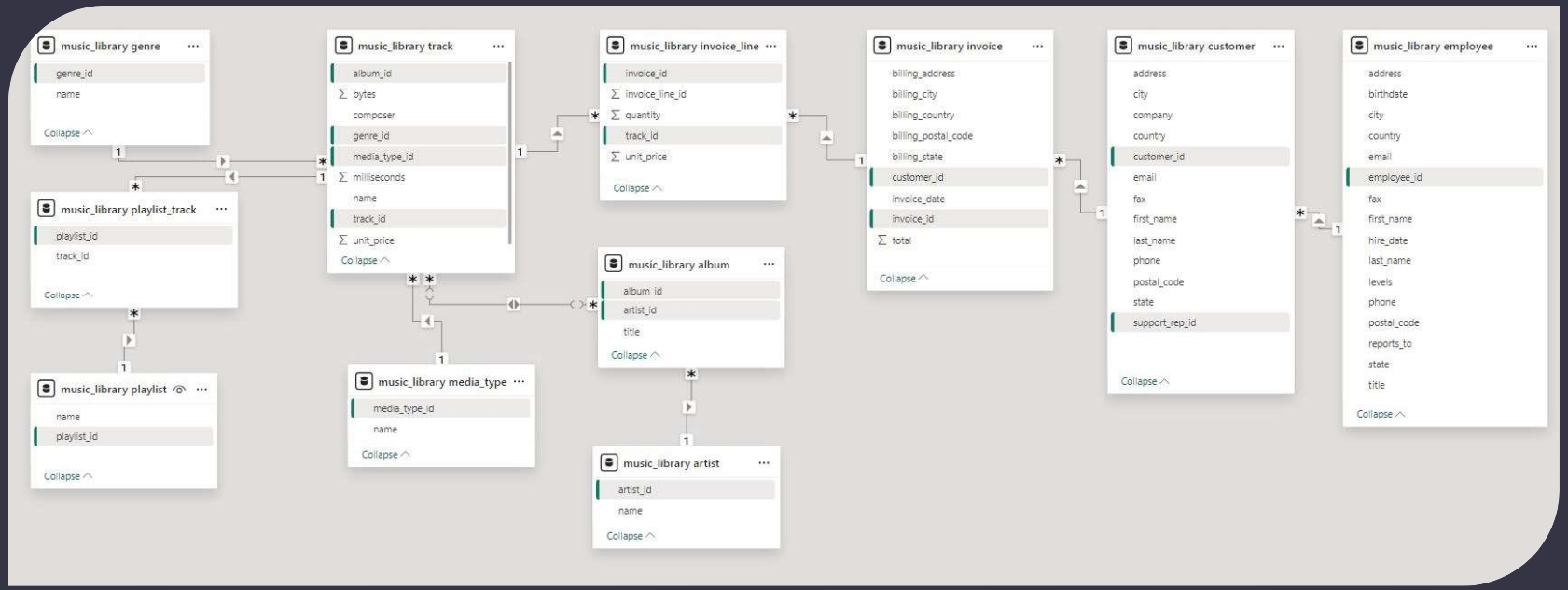
3:15min

- **MySQL** for data querying and manipulation
- **Python** (Pandas, Matplotlib, Seaborn)

Connecting to Music Co. Database



Data Schema


music_library genre
genre_id
name
Collapse
music_library track
album_id
bytes
composer
genre_id
media_type_id
milliseconds
name
track_id
unit_price
Collapse
music_library invoice_line
invoice_id
invoice_line_id
quantity
track_id
unit_price
Collapse
music_library invoice
billing_address
billing_city
billing_country
billing_postal_code
billing_state
customer_id
invoice_date
invoice_id
total
Collapse
music_library customer
address
city
company
country
customer_id
email
fax
first_name
last_name
phone
postal_code
state
support_rep_id
Collapse
music_library employee
address
birthdate
city
country
email
employee_id
fax
first_name
hire_date
last_name
levels
phone
postal_code
reports_to
state
title
Collapse
music_library playlist_track
playlist_id
track_id
Collapse
music_library playlist
name
playlist_id
Collapse
music_library media_type
media_type_id
name
Collapse
music_library album
album_id
artist_id
title
Collapse
music_library artist
artist_id
name
Collapse

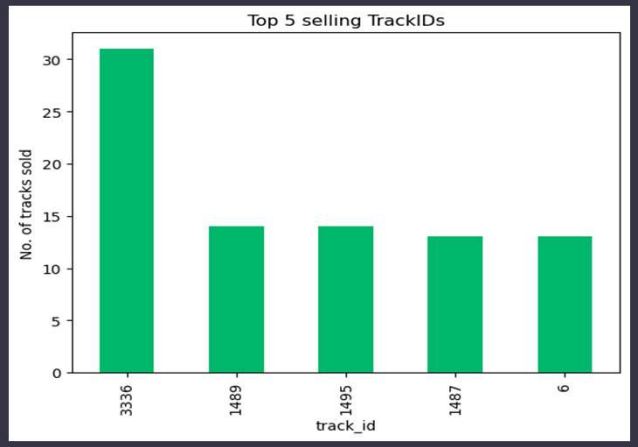

Search
Home
Library
Our Music Playlists
BASIC
QUESTIONS

Calculate the total revenue generated from all sales. Retrieve the total number of sales transactions.

```sql
SELECT
    ROUND(SUM(total), 2) AS Total_Sales,
    COUNT(invoice_id) AS Transactions
FROM
    invoice;
```

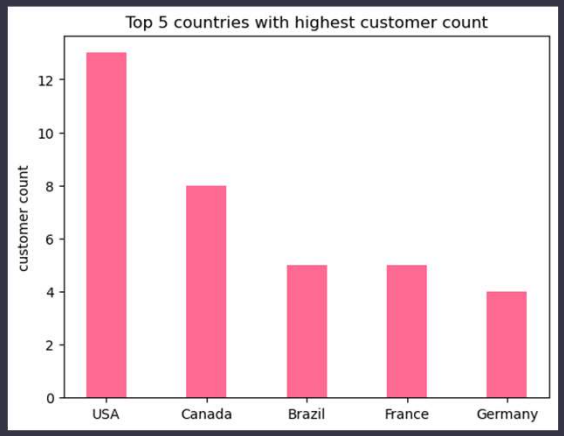
Result Grid | Filter Rows:

	Total_Sales	Transactions
▶	4709.43	614



Identify the country with the most customers.

```sql
SELECT
    billing_country,
    COUNT(DISTINCT (customer_id)) AS customer_count
FROM
    invoice
GROUP BY billing_country
ORDER BY customer_count DESC
LIMIT 1;
```



Result Grid | Filter Rows:

billing_country	customer_count
USA	13

Who is the senior most employee based on job title?

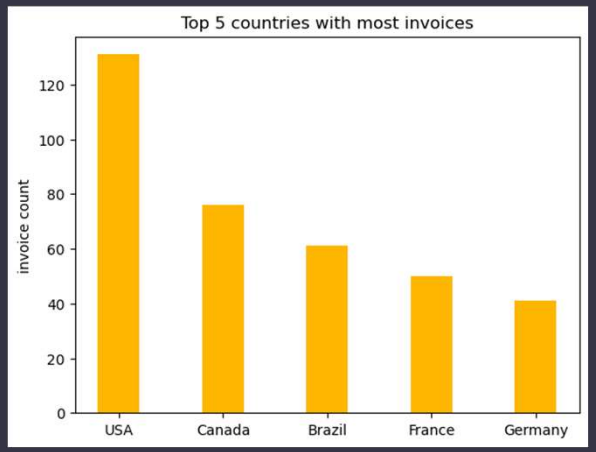
```sql
SELECT
    first_name, last_name, title, levels
FROM
    employee
ORDER BY levels DESC
LIMIT 1;
```

Result Grid | Filter Rows:

	first_name	last_name	title	levels
▶	Andrew	Adams	General Manager	L6

Which countries have the most Invoices?

```sql
SELECT
    billing_country, COUNT(invoice_id) AS invoice_count
FROM
    invoice
GROUP BY billing_country
ORDER BY invoice_count DESC
LIMIT 5;
```

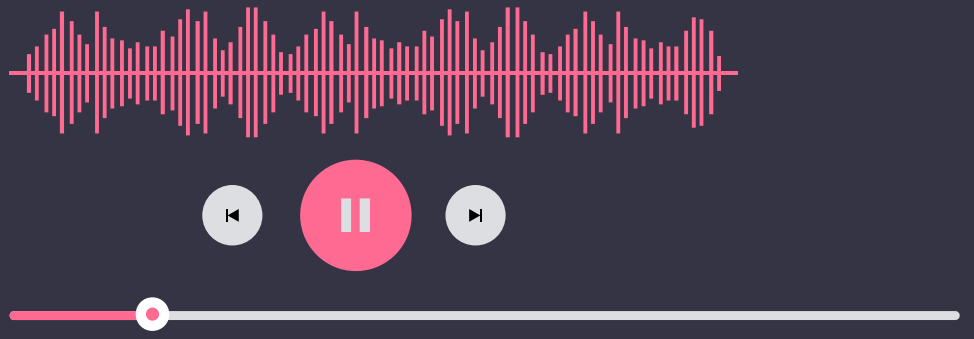


Result Grid | Filter Rows:

billing_country	invoice_count
USA	131
Canada	76
Brazil	61
France	50
Germany	41

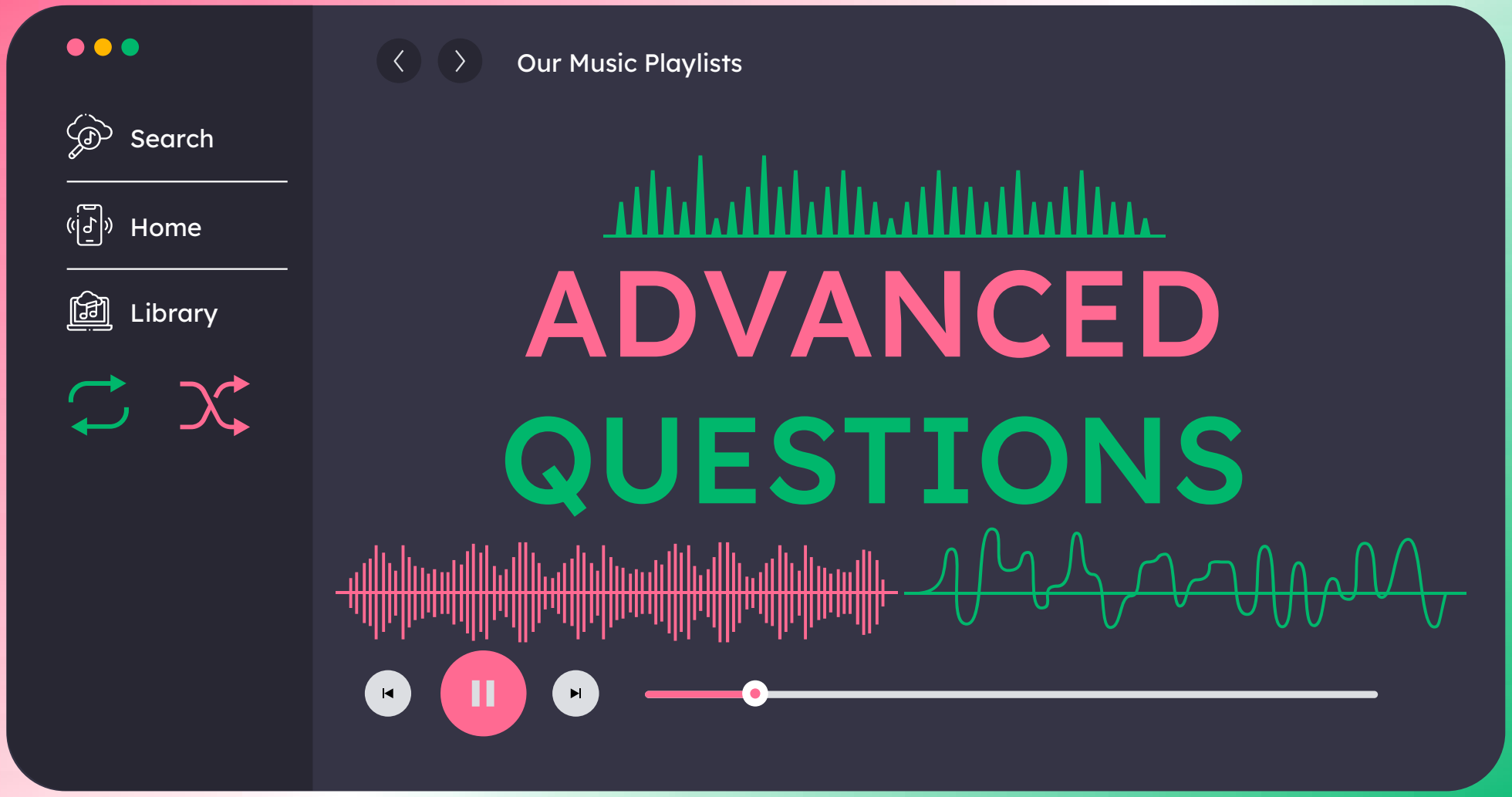
What are top 3 values of total invoice?

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



Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

invoice_id	customer_id	invoice_date	billing_address	billing_city	billing_state	billing_country	billing_postal_code	total
183	42	2018-02-09 00:00:00	9, Place Louis Barthou	Bordeaux	None	France	33000	23.759999999999998
92	32	2017-07-02 00:00:00	696 Osborne Street	Winnipeg	MB	Canada	R3L 2B9	19.8
526	5	2020-06-08 00:00:00	Klanova 9/506	Prague	None	Czech Republic	14700	19.8

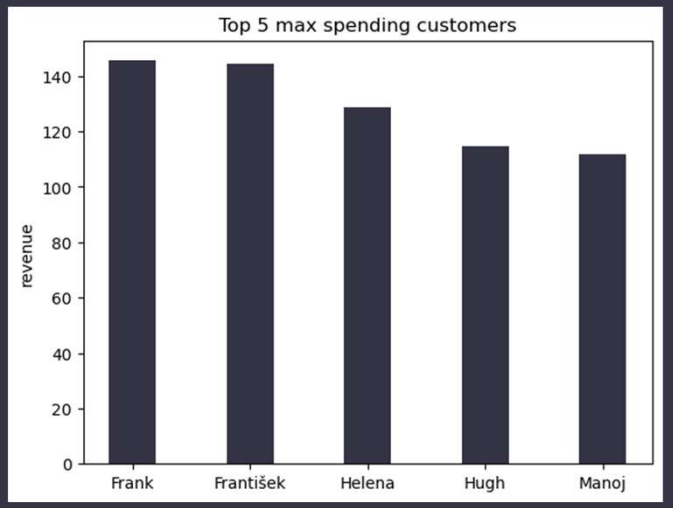

Search
Home
Library
Our Music Playlists
ADVANCED
QUESTIONS

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.

```sql
SELECT
    billing_city, round(SUM(total),2) AS revenue
FROM
    invoice
GROUP BY billing_city
ORDER BY revenue DESC
LIMIT 1;
```

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.

```sql
SELECT
    customer.customer_id,
    CONCAT(first_name, ' ', last_name) AS name,
    SUM(total) AS spent
FROM
    invoice
        JOIN
    customer ON invoice.customer_id = customer.customer_id
GROUP BY customer_id , first_name , last_name
ORDER BY spent DESC
LIMIT 1;
```



Result Grid | Filter Rows: | Export:

customer_id	name	spent
5	FrantiÅiek WichterlovÃ¡	144.54000000...

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’.

```sql
SELECT DISTINCT
    email, first_name, last_name, genre.name
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
WHERE
    genre.name = 'Rock'
ORDER BY email ASC;
```

Result Grid | Filter Rows: | Export:

email	first_name	last_name	name
aaronmitchell@yahoo.ca	Aaron	Mitchell	Rock
alero@uol.com.br	Alexandre	Rocha	Rock
astrid.gruber@apple.at	Astrid	Gruber	Rock
bjorn.hansen@yahoo.no	BjÃ¸rn	Hansen	Rock
camille.bernard@yahoo.fr	Camille	Bernard	Rock
daan_peeters@apple.be	Daan	Peeters	Rock
diego.gutierrez@yahoo.ar	Diego	GutiÃ©rrez	Rock
.			
.			
.			
roberto.almeida@riotur.gov.br	Roberto	Almeida	Rock
stanisÅ‚aw.wÃ³jcik@wp.pl	StanisÅ‚aw	WÃ³jcik	Rock
steve.murray@yahoo.uk	Steve	Murray	Rock
terhi.hamalainen@apple.fi	Terhi	HÃ¤mÃ¤lÃ...	Rock
tgoyer@apple.com	Tim	Goyer	Rock
vstevens@yahoo.com	Victor	Stevens	Rock
wyatt.girard@yahoo.fr	Wyatt	Girard	Rock

Revenue Contribution by genre: Calculate the percentage of total revenue contributed by each genre.

```sql
SELECT
    genre.name AS genre,
    ROUND(SUM(invoice_line.quantity * invoice_line.unit_price) * 100 / (SELECT
                SUM(total)
            FROM
                invoice),
        2) AS 'revenue%'
FROM
    invoice
        JOIN
    invoice_line ON invoice.invoice_id = invoice_line.invoice_id
        JOIN
    track ON track.track_id = invoice_line.track_id
        JOIN
    genre ON genre.genre_id = track.genre_id
GROUP BY genre.name;
```

Result Grid | Filter Rows:

genre	revenue%
Rock	55.39
Latin	3.51
Blues	2.61
Alternative & Punk	10.34
Pop	1.32
R&B/Soul	3.34
Metal	13.01
Alternative	2.46
Classical	0.99
Easy Listening	1.56
Jazz	2.54
Electronica/Dance	1.16
Heavy Metal	0.17
Reggae	0.74
Drama	0.02
Hip Hop/Rap	0.69
TV Shows	0.04

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

```sql
SELECT
    artist.name AS artist_name, COUNT(genre_id) AS track_count
FROM
    artist
        JOIN
    album ON artist.artist_id = album.artist_id
        JOIN
    track ON album.album_id = track.album_id
GROUP BY artist.name , genre_id
HAVING genre_id IN (SELECT
        genre_id
    FROM
        genre
    WHERE
        name = 'Rock')
ORDER BY track_count DESC
LIMIT 10;
```

Result Grid | Filter Rows:

artist_name	track_count
Led Zeppelin	114
U2	112
Deep Purple	92
Iron Maiden	81
Pearl Jam	54
Van Halen	52
Queen	45
The Rolling Stones	41
Creedence Clearwater Revival	40
Kiss	35

Return count of track_ids that have a song length longer than the average song length. Return the track_id and Milliseconds for each track. Order by the song length with the longest songs listed first

```sql
SELECT
    AVG(milliseconds)
FROM
    track AS x;
-- 251177.7432
-- count
SELECT
    count(track_id)
FROM
    track
WHERE
    milliseconds > (SELECT
            AVG(milliseconds)
        FROM
            track)
ORDER BY milliseconds DESC;
```

```sql
-- all track id's
SELECT
    track_id, milliseconds
FROM
    track
WHERE
    milliseconds > (SELECT
            AVG(milliseconds)
        FROM
            track)
ORDER BY milliseconds DESC;
```

Result Grid

count(track_id)
494

Result Grid Filter

track_id	milliseconds
2820	5286953
3224	5088838
3244	2960293
3242	2956998
3227	2956081
3226	2952702
3243	2935894
3228	2927802
3248	2927677
3239	2926593
3232	2925008
3235	2924716
3237	2924507
3234	2924341
3249	2924007
3247	2923548
3241	2923381

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

```
select concat(customer.first_name," ", customer.last_name) as customer, artist.name as artist,
sum(invoice_line.quantity* invoice_line.unit_price) as spent
from invoice join customer 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 album on album.album_id = track.album_id
join artist on artist.artist_id = album.artist_id
group by customer.customer_id, first_name, last_name, artist.name order by customer;
```

Result Grid | Filter Rows: | Export:

customer	artist	spent
Aaron Mitchell	Red Hot Chili Peppers	0.99
Aaron Mitchell	Pink Floyd	0.99
Aaron Mitchell	AC/DC	0.99
Aaron Mitchell	Rush	0.99
Aaron Mitchell	Black Sabbath	0.99
Aaron Mitchell	Green Day	0.99
Aaron Mitchell	Creedence Clearwater Revival	1.98
Aaron Mitchell	The Rolling Stones	0.99
Aaron Mitchell	Chris Cornell	13.860000000...
Aaron Mitchell	Jota Quest	0.99
Aaron Mitchell	Def Leppard	0.99
Aaron Mitchell	House Of Pain	0.99
Aaron Mitchell	Godsmack	0.99
Aaron Mitchell	Kiss	0.99
Aaron Mitchell	James Brown	19.799999999...
Aaron Mitchell	Metallica	0.99
Aaron Mitchell	Guns N' Roses	0.99

Result Grid | Filter Rows: | Export: | Wrap

customer	artist	spent
Alexandre Rocha	Scorpions	0.99
Alexandre Rocha	R.E.M.	0.99
Alexandre Rocha	JET	12.870000000...
Alexandre Rocha	Jorge Ben	0.99
Alexandre Rocha	Nirvana	0.99
Alexandre Rocha	MotÃ¶rhead	0.99
Alexandre Rocha	Chico Buarque	0.99
Alexandre Rocha	Pink Floyd	0.99
Alexandre Rocha	Led Zeppelin	0.99
Alexandre Rocha	Aerosmith	0.99
Alexandre Rocha	System Of A Down	0.99
Alexandre Rocha	Queen	1.98
Alexandre Rocha	Judas Priest	0.99
Alexandre Rocha	The Who	1.98
Alexandre Rocha	Frank Sinatra	0.99
Alexandre Rocha	The Rolling Stones	1.98
Alexandre Rocha	Guns N' Roses	0.99

. . .

Result Grid | Filter Rows: | Export: | Wrap

customer	artist	spent
Jennifer Peterson	Raul Seixas	0.99
Jennifer Peterson	Alanis Morissette	0.99
Jennifer Peterson	Velvet Revolver	1.98
Jennifer Peterson	Kiss	0.99
Jennifer Peterson	Iron Maiden	0.99
Jennifer Peterson	Black Label Society	0.99
Jennifer Peterson	Nirvana	11.88
Jennifer Peterson	Godsmack	0.99
Jennifer Peterson	Tim Maia	0.99
Jennifer Peterson	Deep Purple	0.99
Jennifer Peterson	The Doors	0.99
Jennifer Peterson	Metallica	1.98
Jennifer Peterson	Guns N' Roses	0.99
Jennifer Peterson	MÃ¶tley CrÃ¼e	0.99
Jennifer Peterson	Alice In Chains	1.98
Jennifer Peterson	AC/DC	0.99
Jennifer Peterson	Aerosmith	14.850000000...

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.

```sql
-- RANK
select billing_country, genre from
(select billing_country, genre, purchases,
rank() over(partition by billing_country
order by billing_country asc, purchases desc) as class from
(select billing_country, genre.name as genre, sum(quantity) as purchases
from invoice 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 billing_country, genre.name) as a) as b
where class = 1;
```

Result Grid | Filter Rows:

billing_country	genre
Argentina	Alternative & Punk
Australia	Rock
Austria	Rock
Belgium	Rock
Brazil	Rock
Canada	Rock
Chile	Rock
Czech Republic	Rock
Denmark	Rock
Finland	Rock
France	Rock
Germany	Rock
Hungary	Rock
India	Rock
Ireland	Rock
Italy	Rock
Netherlands	Rock

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

```
select billing_country, customer_id, first_name,last_name, spent from
(select billing_country, customer_id, first_name,last_name, spent,
rank() over(partition by billing_country order by spent desc) as class from
(select billing_country, customer.customer_id, first_name,last_name, sum(total) as spent
from customer join invoice on customer.customer_id  =invoice.customer_id
group by customer.customer_id, first_name,last_name, billing_country
order by billing_country asc, spent desc) as a) as b where class =1
```

Result Grid | Filter Rows: | Export: | Wrap Cell Conte

billing_country	customer_id	first_name	last_name	spent
Argentina	56	Diego	GutiÃ©rrez	39.6
Australia	55	Mark	Taylor	81.18
Austria	7	Astrid	Gruber	69.3
Belgium	8	Daan	Peeters	60.389999999...
Brazil	1	LuÃ-s	GonÃ§alves	108.89999999...
Canada	3	FranÃ§ois	Tremblay	99.99
Chile	57	Luis	Rojas	97.020000000...
Czech Republic	5	FrantiÅiek	WichterlovÃi	144.54000000...
Denmark	9	Kara	Nielsen	37.619999999...
Finland	44	Terhi	HÃ¤mÃ¤lÃ¤inen	79.2
France	42	Wyatt	Girard	99.99
Germany	37	Fynn	Zimmermann	94.050000000...
Hungary	45	Ladislav	KovÃ¡cs	78.21
India	58	Manoj	Pareek	111.86999999...
Ireland	46	Hugh	O'Reilly	114.83999999...
Italy	47	Lucas	Mancini	50.49
Netherlands	48	Johannes	Van der Berg	65.34

Search

Home

Library

Our Music Playlists

Thanks!

Do you have any questions?

- mayanksri461994@gmail.com
- https://www.linkedin.com/in/mayank-srivastava-6a84211054
- https://www.hackerrank.com/profile/mayanksri461994
- https://github.com/Mayank4694
- https://www.kaggle.com/mayanksrivastava469