Title: "Music Store Data Analysis Project"

Subtitle: "Exploring Customer Behaviors, Sales Trends, and Popular Genres"

Overview:-

This project tackles real-world business questions through SQL queries, analyzing a fictional music store's data to extract insights into customer behaviors, music trends, and sales performance.

Project Sections:-

- **Basic Analysis**:
  - Exploring employee roles, invoice counts, and high-level customer metrics.
- **Moderate Analysis**:
  - Delving into specific genres, top artists, and detailed track information.
- **Advanced Analysis**:
  - Performing complex calculations on customer spending by genre and artist, identifying country-based trends.

Tools and Techniques:

- SQL Queries for Data Extraction and Transformation
- Data Aggregation, Joins, and Window Functions

Problem Solving Objectives:

1. **Identify Key Customer Metrics**
    - Who are the top customers based on spending?
    - Which countries generate the most revenue?
    - Who is the "Best Customer" with the highest lifetime value?
2. **Analyze Popular Music and Genre Trends**
    - Which genres are most popular among customers?
    - Which artists and songs have the highest play and purchase rates?
    - What are the top genres for different countries?
3. **Evaluate Sales and Customer Patterns**
    - What are the total sales for each customer?
    - How much is spent by each customer on specific artists?
    - Which customers are most active in purchasing music?

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	first_name character	title character varying (50)
9	Madan	Mohan	Senior General Manager

Q2 : Which countries have the most Invoices

```
select  count(*) as c , billing_country
from invoice
group by billing_country
order by c desc
limit 1
```

	c bigint	billing_country character varying (30)
1	131	USA

Q3 : What are top 3 values of total invoice

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

total double precision
23.759999999999998
19.8
19.8

Q4 : Which city has the best cutomers ? 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 sum(total) as invoice_total , billing_city
from invoice
group by billing_city
order by invoice_total desc
limit 1
```

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

Q5: Who is the best customer? The customer who have spent most money will be declared the best customer write a query that returns the persom who have spend the most money

```
select c.customer_id,c.first_name , c.last_name ,
sum(i.total) as s_total
from
customer c
inner join
invoice i on i.customer_id = c.customer_id
group by c.customer_id
order by s_total desc
limit 1
```

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

```sql
-- Q6 : Write query to return the email , first name
-- last name and genre of all Rock Music listeners.
-- Return your list ordered alphabatically by email
-- starting with A

select distinct c.first_name , c.last_name ,
c.email ,  g.name as genre
from customer c
inner join
invoice i on i.customer_id= c.customer_id
inner join
invoice_line il on il.invoice_id = i.invoice_id
inner join
track t on t.track_id = il.track_id
inner join
genre g on g.genre_id = t.genre_id
where  g.name = 'Rock'
order by c.email
```

	first_name character	last_name character	email character varying (50)	genre character varying (120)
1	Aaron	Mitchell	aaronmitchell@yahoo.ca	Rock
2	Alexandre	Rocha	alero@uol.com.br	Rock
3	Astrid	Gruber	astrid.gruber@apple.at	Rock
4	Bjørn	Hansen	bjorn.hansen@yahoo.no	Rock
5	Camille	Bernard	camille.bernard@yahoo.fr	Rock
6	Daan	Peeters	daan_peeters@apple.be	Rock
7	Diego	Gutiérrez	diego.gutierrez@yahoo.ar	Rock
8	Dan	Miller	dmiller@comcast.com	Rock

Total rows: 59 of 59 | Query complete 00:00:00.086

```sql
-- Q7 : Let's invitet the artists who have written the most
-- rock music in our dataset. Write a Query that return the
-- Artist name and total track count of the top 10 rock band

select a.name,
count(a.name) as number_of_songs
from
artist a
inner join
album al on al.artist_id= a.artist_id
inner join
track t on t.album_id = al.album_id
inner join
genre g on g.genre_id = t.genre_id
where g.name ='Rock'
group by a.name
order by number_of_songs desc
limit 10
```

	name character varying (120)	number_of_songs bigint
1	Led Zeppelin	114
2	U2	112
3	Deep Purple	92
4	Iron Maiden	81
5	Pearl Jam	54
6	Van Halen	52
7	Queen	45
8	The Rolling Stones	41
9	Creedence Clearwater Revival	40
10	Kiss	35

Total rows: 10 of 10 | Query complete 00:00:00.106

```sql
--Q8 : Return all the track names that have a song length
-- longer than the average song length . Return the name
-- and the millisecond for each track . order by the song
-- length with the lingest songs listed first

select name ,milliseconds
from track
where milliseconds >(

    select avg(milliseconds) as average
    from track
)
order by milliseconds Desc;
```

name character varying (150)	milliseconds integer
Occupation / Precipice	5286953
Through a Looking Glass	5088838
Greetings from Earth, Pt. 1	2960293
The Man With Nine Lives	2956998
Battlestar Galactica, Pt. 2	2956081
Battlestar Galactica, Pt. 1	2952702
Murder On the Rising Star	2935894

```sql
--Q9: Find how much amound spent by each customer on artists?
-- Write a query to return customer name, artist name and
-- total spent

with best_selling_artist as
(
   select a.artist_id , a.name as artist_name,
   sum(il.unit_price* il.quantity)as total_sales
   from invoice_line il
   inner join
   track t on t.track_id = il.track_id
    inner join
   album al on al.album_id = t.album_id
    inner join
   artist a on a.artist_id = al.artist_id
    group by 1
    order by 3 desc
    limit 1
)
```

```sql
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
inner join
customer c on c.customer_id= i.customer_id
inner join
invoice_line il on il.invoice_id = i.invoice_id
inner join
track t on t.track_id = il.track_id
inner join
album al on al.album_id = t.album_id
inner join
best_selling_artist bsa on bsa.artist_id= al.artist_id
group by 1,2,3,4
order by 5 Desc
```

	customer_id integer	first_name character	last_name character	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

```sql
--Q10: 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 amout of purchases. Write
--a query that returns each country along with the top
--Genre. For countries where the maximum number of purchase
--is shared return all Genres.

with popular_genre as
(
select count(il.quantity) as purchase , 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 il
join invoice i on i.invoice_id= il.invoice_id
join customer c on c.customer_id= i.customer_id
join track t on t.track_id= il.track_id
join genre 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
```

Data Output | Messages | Notifications

	purchase bigint	country character varying (50)	name character varying (120)	genre_id character varying (5
1	17	Argentina	Alternative & Punk	4
2	34	Australia	Rock	1
3	40	Austria	Rock	1
4	26	Belgium	Rock	1
5	205	Brazil	Rock	1
6	333	Canada	Rock	1
7	61	Chile	Rock	1
8	143	Czech Republic	Rock	1

```sql
--Q11 : 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 top_customer as
(
 select sum(il.unit_price*il.quantity) as total_spent ,
 c.country,c.first_name,c.customer_id,
 Row_number() over (partition by c.country order by
    sum(il.unit_price*il.quantity) desc)
    as rowno
 from invoice_line il
 join invoice i on i.invoice_id= il.invoice_id
 join customer c on c.customer_id = i.customer_id
 group by 2,3,4
 order by 2 Asc , 1 Desc

)
select * from top_customer where rowno<=1
```

	total_spent double precision	country character varying (50)	first_name character	customer_id [PK] integer	rowno bigint
1	39.599999999999994	Argentina	Diego	56	1
2	81.17999999999995	Australia	Mark	55	1
3	69.30000000000001	Austria	Astrid	7	1
4	60.390000000000036	Belgium	Daan	8	1
5	108.8999999999998	Brazil	Luís	1	1
6	99.98999999999985	Canada	François	3	1