INTRODUCTION

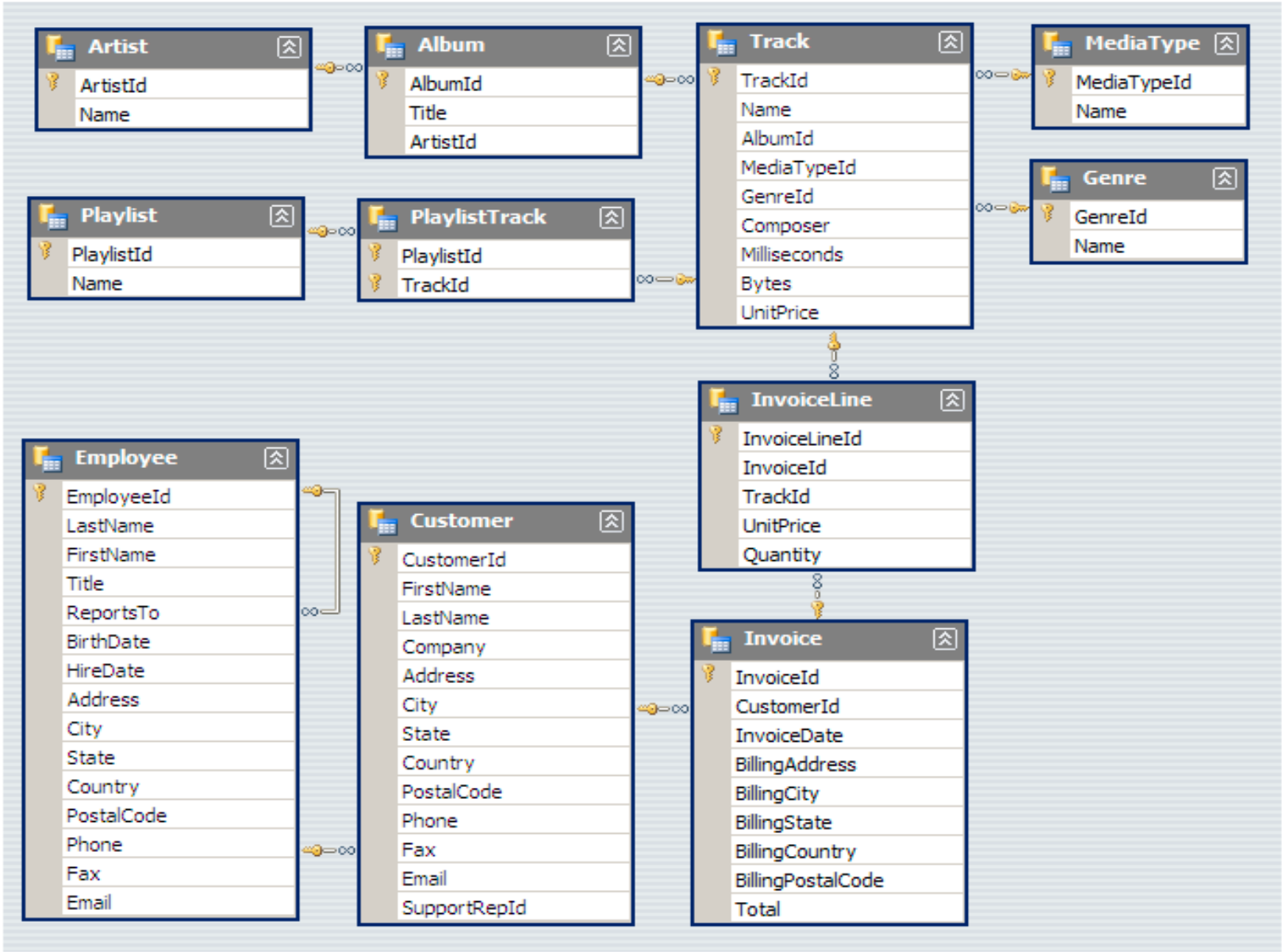
My name is M Fawad ,working on SQL skill for searching data analyst role inn future

Total eleven tables used in this project which are linked through relationship

A music store is seeking to pinout crucial factors that can boosting its business, finding issues within the current business landscape, recognizing high-value customers, and gaining insights into new promotions and initiatives.

We will employ SQL to address these challenges

Schema or Entity Relationship Model



Presentation

Music Store Analysis

- Muhammad Fawad

EASY QUERIES

USING SIMPLE QUERIES AND JOINS

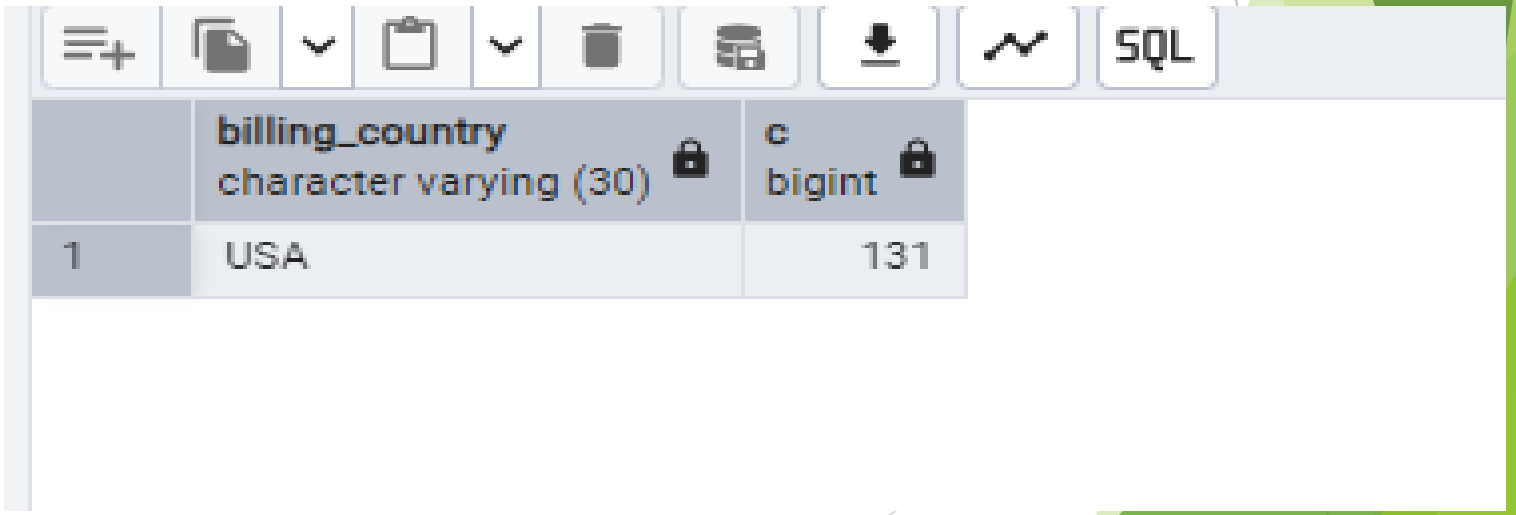
Who is the most senior employee base on the job tittle?

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

	employee_id [PK] character varying (50)	first_name character (50)	last_name character (50)	title character varying (50)
1	9	Mohan	Madan	Senior General Manager

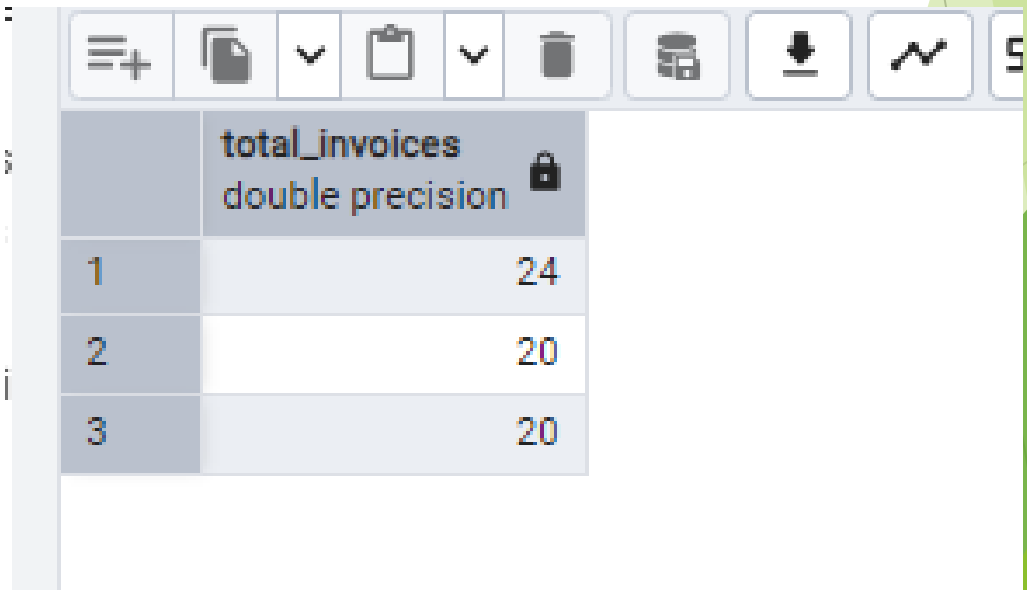
2: Which country is the most invoices?

```
SELECT billing_country,COUNT(*) AS C FROM
invoice
GROUP BY billing_country
ORDER BY C DESC
LIMIT 1;
```



Q3: What are the three values of total invoices?

```
SELECT ROUND(total) AS total_invoices FROM
invoice
ORDER BY total_invoices DESC
LIMIT 3;
```



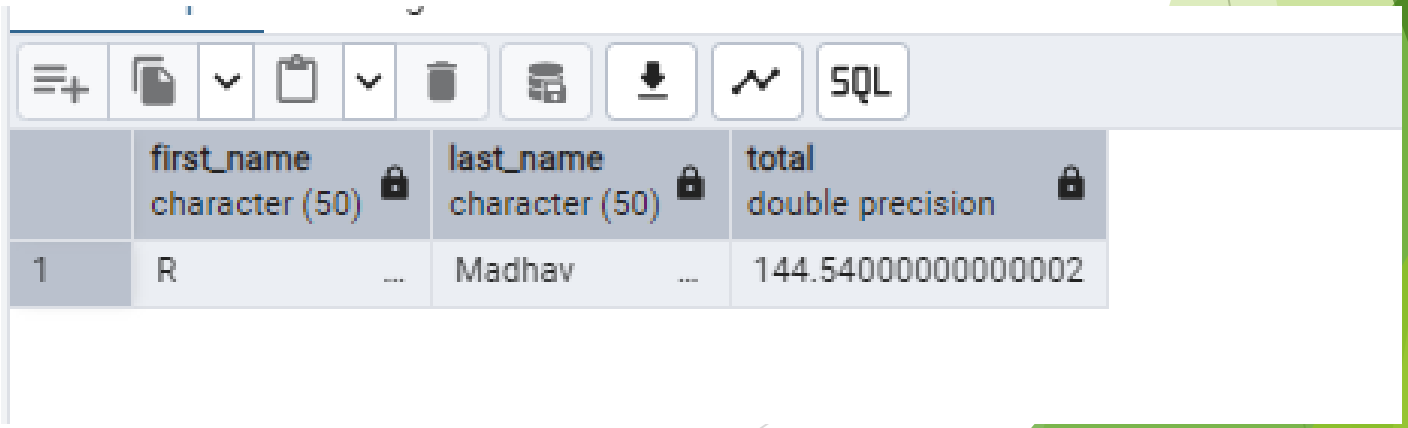
	total_invoices double precision
1	24
2	20
3	20

Music Store Analysis
The purpose of this project

- The output of this project is
- We can easily given direction base on the analysis we done through different queries

Q5 who is the best customer? The customer who has spent the most money will be declared the best customer

```
SELECT
  c.first_name,
  c.last_name,
  SUM(i.total) as total
FROM
  customer c
  INNER JOIN invoice i ON c.customer_id = i.customer_id
GROUP BY
  c.customer_id
ORDER BY
  total DESC
LIMIT 1;
```



	first_name character (50)	last_name character (50)	total double precision
1	R …	Madhav …	144.54000000000002

Questions Moderate level part 2

Q1: Write a query that return the email, first name, last name& Genre of all Rock Music listeners.

Returns your list ordered alphabetically

```
SELECT email, first_name,  last_name
FROM  customer c
 JOIN invoice i ON c.customer_id = i.customer_id
  JOIN invoice_line inl ON i.invoice_id = inl.invoice_id
WHERE track_id IN( SELECT
      track_id  FROM
      track t
 JOIN genre g ON t.genre_id = g.genre_id WHERE
      g.name LIKE 'Rock'
  )ORDER BY
  email;
```

Data Output | Messages | Notifications

SQL

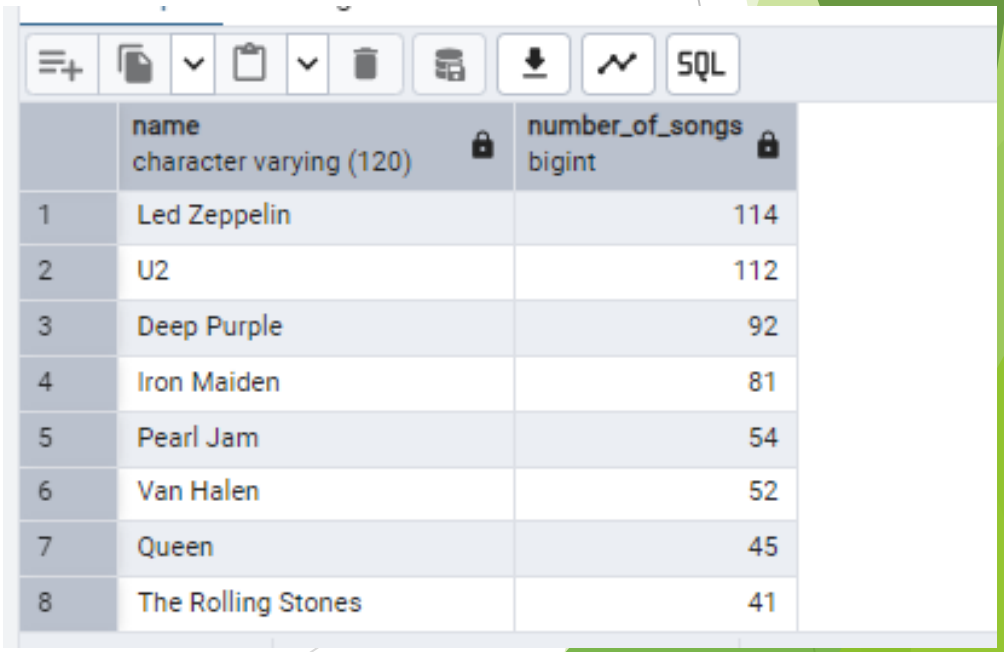
	email character varying (50)	first_name character (50)	last_name character (50)
1	aaronmitchell@yahoo.ca	Aaron	Mitchell
2	aaronmitchell@yahoo.ca	Aaron	Mitchell
3	aaronmitchell@yahoo.ca	Aaron	Mitchell
4	aaronmitchell@yahoo.ca	Aaron	Mitchell
5	aaronmitchell@yahoo.ca	Aaron	Mitchell
6	aaronmitchell@yahoo.ca	Aaron	Mitchell
7	aaronmitchell@yahoo.ca	Aaron	Mitchell
8	aaronmitchell@yahoo.ca	Aaron	Mitchell

Total rows: 2635 Query complete 00:00:00.161

Q2; let's invite the artists who have written the most rock music in our data set

write a query that returns the artist name and track count of the top 10 rock music.

```
SELECT  ar.name, COUNT(ar.artist_id) AS number_of_songs
FROM
  track t  JOIN album a ON t.album_id = a.album_id
  JOIN artist ar ON ar.artist_id = a.artist_id
  JOIN genre g ON t.genre_id = g.genre_id WHERE
  g.name LIKE 'Rock'
GROUP BY
  ar.artist_id
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

Q 3 Return all the tracks names that have a song length longer than the average song length.

Return the Names and Milliseconds for each track. Order by the song

```
-- Listed first:

 SELECT  name,  milliseconds FROM  track
 WHERE  milliseconds > ( SELECT  AVG(milliseconds) as Avg_track_length
 from   track
) ORDER BY
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

Advanced Questions

- Q 1; Find how much amount spent by each customer on artists? Write a query 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_spent  FROM
    invoice_line  JOIN track ON track.track_id = invoice_line.track_id
    JOIN album ON track.album_id = album.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 total_spent FROM  invoice i
  JOIN customer c ON c.customer_id = i.customer_id  JOIN invoice_line il ON i.invoice_id = il.invoice_id
  JOIN track t ON t.track_id = il.track_id
  JOIN album alb ON t.album_id = alb.album_id JOIN best_selling_artist bsa ON bsa.artist_id = alb.artist_id
GROUP BY  1,  2,  3,  4 ORDER BY  5 DESC;
```

Showing rows: 1 to 43

	customer_id integer	first_name character (50)	last_name character (50)	artist_name character varying (120)	total_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

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.

```
-- METHOD ONE
WITH popular_genre AS (
  SELECT  COUNT(quantity), country, g.name, g.genre_id,
    ROW_NUMBER() OVER( PARTITION BY country ORDER BY
  COUNT(quantity) DESC ) AS RowNo  FROM
    customer c
    JOIN invoice i 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 genre g ON t.genre_id = g.genre_id
  GROUP BY  2,  3,   4 ORDER BY   2 ASC,   1 DESC
)
SELECT  * FROM  popular_genre WHERE ROWNo <= 1;
```

nre. For countries

	count 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
8	143	Czech Republic	Rock	1	1

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

```
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
    customer c
    JOIN invoice i 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;
```

Showing ro

	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
6	3	François	Tremblay	Canada	99.99	1
7	57	Luis	Rojas	Chile	97.02000000000001	1
8	5	R	Madhav	Czech Republic	144.54000000000002	1

Total rows: 24 | Query complete 00:00:00.146