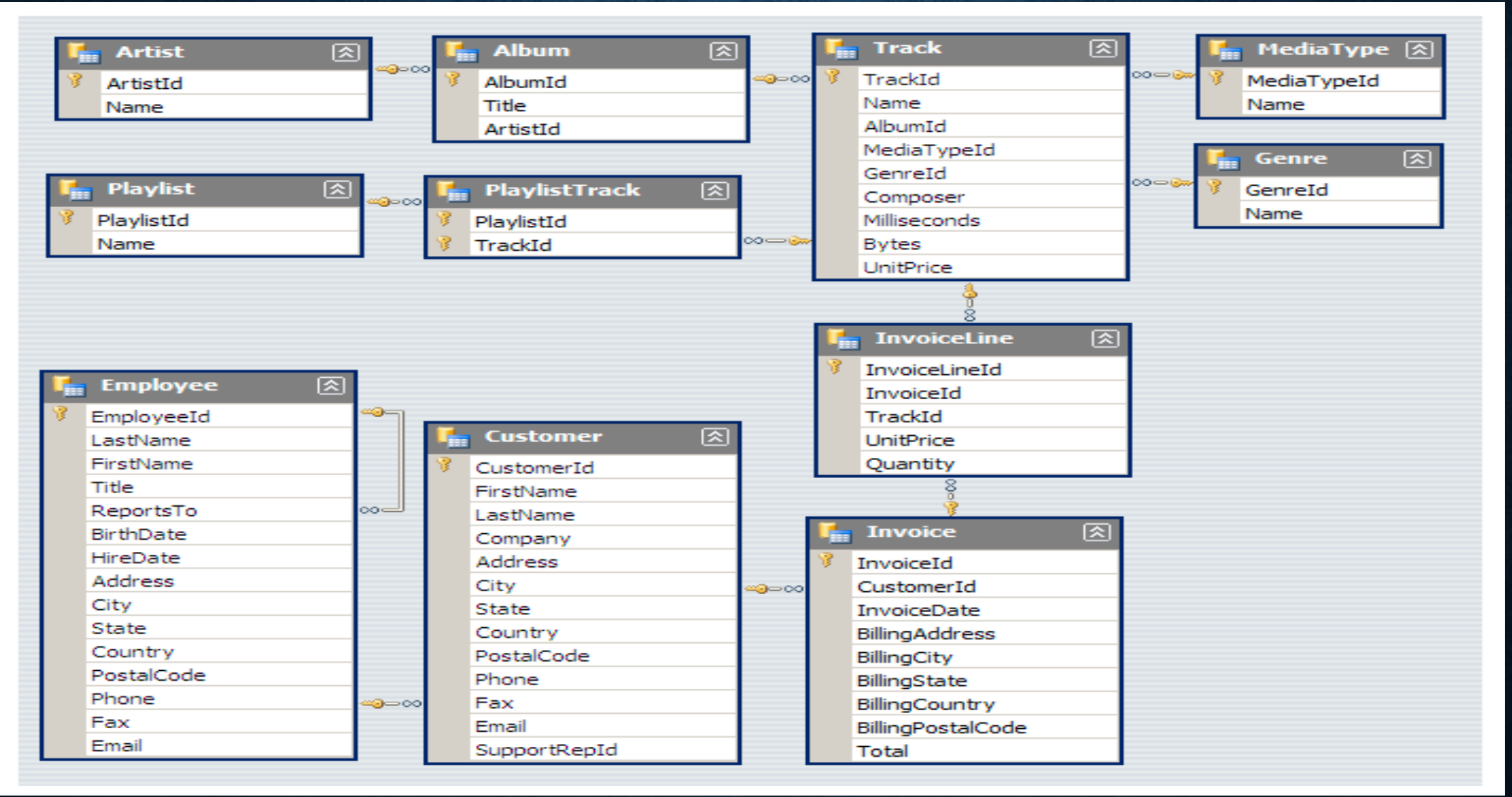
DIGITAL MUSIC STORE ANALYSIS BY USING SQL

SCHEMAS


Artist
ArtistId
Name
Album
AlbumId
Title
ArtistId
Track
TrackId
Name
AlbumId
MediaTypeId
GenreId
Composer
Milliseconds
Bytes
UnitPrice
MediaType
MediaTypeId
Name
Genre
GenreId
Name
Playlist
PlaylistId
Name
PlaylistTrack
PlaylistId
TrackId
InvoiceLine
InvoiceLineId
InvoiceId
TrackId
UnitPrice
Quantity
Employee
EmployeeId
LastName
FirstName
Title
ReportsTo
BirthDate
HireDate
Address
City
State
Country
PostalCode
Phone
Fax
Email
Customer
CustomerId
FirstName
LastName
Company
Address
City
State
Country
PostalCode
Phone
Fax
Email
SupportRepId
Invoice
InvoiceId
CustomerId
InvoiceDate
BillingAddress
BillingCity
BillingState
BillingCountry
BillingPostalCode
Total

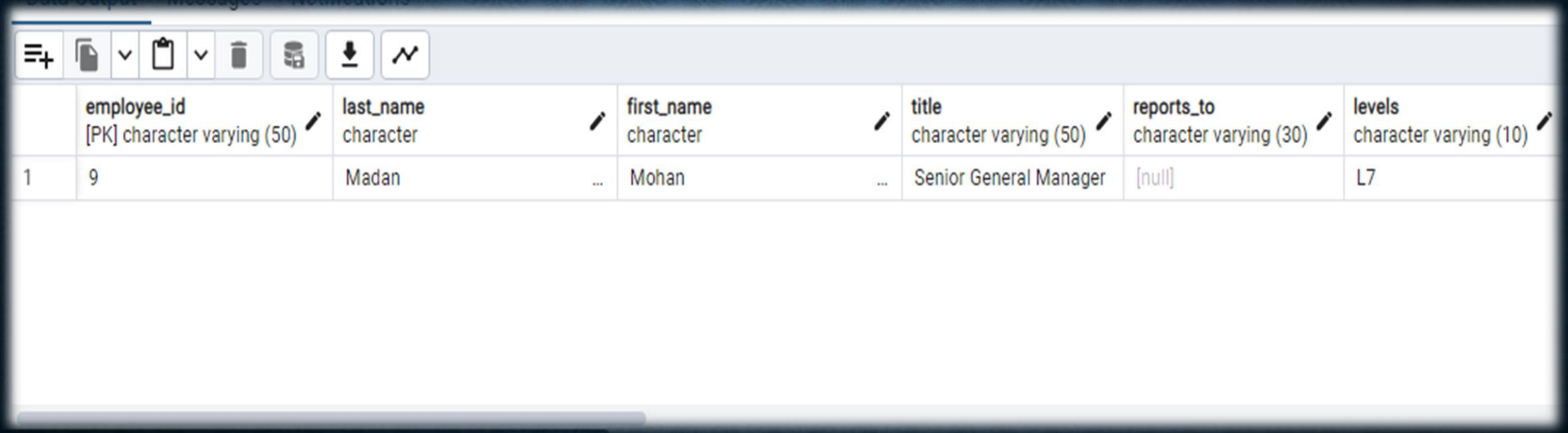
QUESTION SET 1 - EASY

1. who is the senior most employee based on job title?

Query-

```
select * from employee
    order by levels desc
    limit 1;
```

Out Put-



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

QUESTION SET 1 - EASY

2. Which country have the most invoice.

Query-

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

Out Put-

Output | Messages | Notifications

c bigint	billing_country character varying (30)
131	USA
76	Canada
61	Brazil
50	France
41	Germany
30	Czech Republic
29	Portugal
28	United Kingdom
21	India
13	Chile
13	Ireland
11	Spain

QUESTION SET 1 - EASY

3. what are top 3 values of total invoice.

Query-

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

Out Put-

Data Output | Messages | Notifications

	invoice_id [PK] integer	customer_id integer	invoice_date timestamp without time zone	billing_address character varying (120)	billing_city character varying (30)	billing_state character varying (30)	billing_country character varying (30)
1	183	42	2018-02-09 00:00:00	9, Place Louis Barthou	Bordeaux	None	France
2	92	32	2017-07-02 00:00:00	696 Osborne Street	Winnipeg	MB	Canada
3	31	3	2017-02-21 00:00:00	1498 rue Bélanger	Montréal	QC	Canada

QUESTION SET 1 - EASY

4. 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-

```
select billing_city, sum(total) as Total_Invoice from invoice
    group by billing_city order by Total_Invoice desc;
```

Out Put-

Data Output | Messages | Notifications

	billing_city character varying (30)	total_invoice double precision
1	Prague	273.24000000000007
2	Mountain View	169.29
3	London	166.32
4	Berlin	158.4
5	Paris	151.47
6	São Paulo	129.69
7	Dublin	114.83999999999997
8	Delhi	111.86999999999999
9	São José dos Campos	108.89999999999998
10	Brasília	106.91999999999999
11	Lisbon	102.96000000000001

QUESTION SET 1 - EASY

5. who is the best customer? the customer who has spent the most money will be declared the best customer .write a query that return the person who has spent the most money.

Query-

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

Out Put-

Data Output | Messages | Notifications

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

QUESTION SET 2 - MODERATE

1: 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.

Query-

```
select  distinct c.email ,c.first_name,c.last_name
    from customer c
    join invoice i on c.customer_id= i.customer_id
    join invoice_line on i.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 email;
```

Out Put-

Data Output | Messages | Notifications

	email character varying (50)	first_name character	last_name character
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

QUESTION SET 2 - MODERATE

2: 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.

Query-

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

Out Put-

Data Output | Messages | Notifications

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

QUESTION SET 2 - MODERATE

3: 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-

```
select distinct email, first_name, last_name, genre.name from customer
    join invoice on customer.customer_id = invoice.customer_id
    join invoice_line on invoice.invoice_id = invoice_line.invoice_id
    join track on invoice_line.track_id = track.track_id
    join genre on track.genre_id = genre.genre_id
    where genre.name like 'Rock'
    order by email;
```

Out Put-

Data Output | Messages | Notifications

	email character varying (50)	first_name character	last_name character	name character varying (120)
1	aaronmitchell@yahoo.ca	Aaron	Mitchell	Rock
2	alero@uol.com.br	Alexandre	Rocha	Rock
3	astrid.gruber@apple.at	Astrid	Gruber	Rock
4	bjorn.hansen@yahoo.no	Bjørn	Hansen	Rock
5	camille.bernard@yahoo.fr	Camille	Bernard	Rock
6	daan_peeters@apple.be	Daan	Peeters	Rock
7	diego.gutierrez@yahoo.ar	Diego	Gutiérrez	Rock
8	dmiller@comcast.com	Dan	Miller	Rock
9	dominiquelefebvre@gmail.c...	Dominique	Lefebvre	Rock
10	edfrancis@yachoo.ca	Edward	Francis	Rock
11	eduardo@woodstock.com.br	Eduardo	Martins	Rock

QUESTION SET 3 - ADVANCE

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

Query-

```
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 invoice_line.track_id = track.track_id
    join album on track.album_id = album.album_id
    join artist on album.artist_id = artist.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;
```

Out Put-

	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
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

Total rows: 43 of 43 Query complete 00:00:00.563

QUESTION SET 3 - ADVANCE

2: 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.

Query-

```
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;
```

Out Put-

	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
8	143	Czech Republic	Rock	1	1
9	24	Denmark	Rock	1	1
10	46	Finland	Rock	1	1
11	211	France	Rock	1	1
12	194	Germany	Rock	1	1
13	44	Hungary	Rock	1	1
14	102	India	Rock	1	1

Total rows: 24 of 24 Query complete 00:00:00.333

QUESTION SET 3 - ADVANCE

3: Write a query that determines the customer that has spentthe 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-

```
with Recursive
customer_with _country as (
    select customer.customer_id, first_name,last_name, billing_country,
    sum(total) as total_spending
    from invoice
    join customer on customer.customer_id = invoice.customer_id
    group by 1,2,3,4
    order by 1,5 desc),

country_max_spending as (
    select billing_country, max(total_spending) as max_spending
    from customer_with_country
    group by billing_country)

select cc.billing_country,cc.total_spending,cc.first_name,cc.last_name,cc.customer_id
from customer_with_country cc
join country_max_spending ms
on cc.billing_country = ms.billing_country
where cc.total_spending= ms.max_spending
order by 1;
```

Out Put-

	billing_country character varying (30)	total_spending double precision	first_name character	last_name character	customer_id integer
1	Argentina	39.6	Diego	Gutiérrez	56
2	Australia	81.18	Mark	Taylor	55
3	Austria	69.3	Astrid	Gruber	7
4	Belgium	60.38999999999999	Daan	Peeters	8
5	Brazil	108.89999999999998	Luís	Gonçalves	1
6	Canada	99.99	François	Tremblay	3
7	Chile	97.02000000000001	Luis	Rojas	57
8	Czech Republic	144.54000000000002	R	Madhav	5
9	Denmark	37.61999999999999	Kara	Nielsen	9
10	Finland	79.2	Terhi	Hämäläinen	44
11	France	99.99	Wyatt	Girard	42
12	Germany	94.05000000000001	Fynn	Zimmermann	37
13	Hungary	78.21	Ladislav	Kovács	45
14	India	111.86999999999999	Manoj	Pareek	58
15	Ireland	114.62000000000007	Hugh	O'Reilly	46

Total rows: 24 of 24 | Query complete 00:00:00.156

SPECIAL THANKS TO **RISHABH MISHRA SIR** FOR HIS EXCEPTIONAL GUIDANCE AND INVALUABLE SUPPORT THROUGHOUT THIS PROJECT.

GUIDED BY – RISHABH MISHRA

PREPARED BY – VIKAS KUMAR MAURYA