

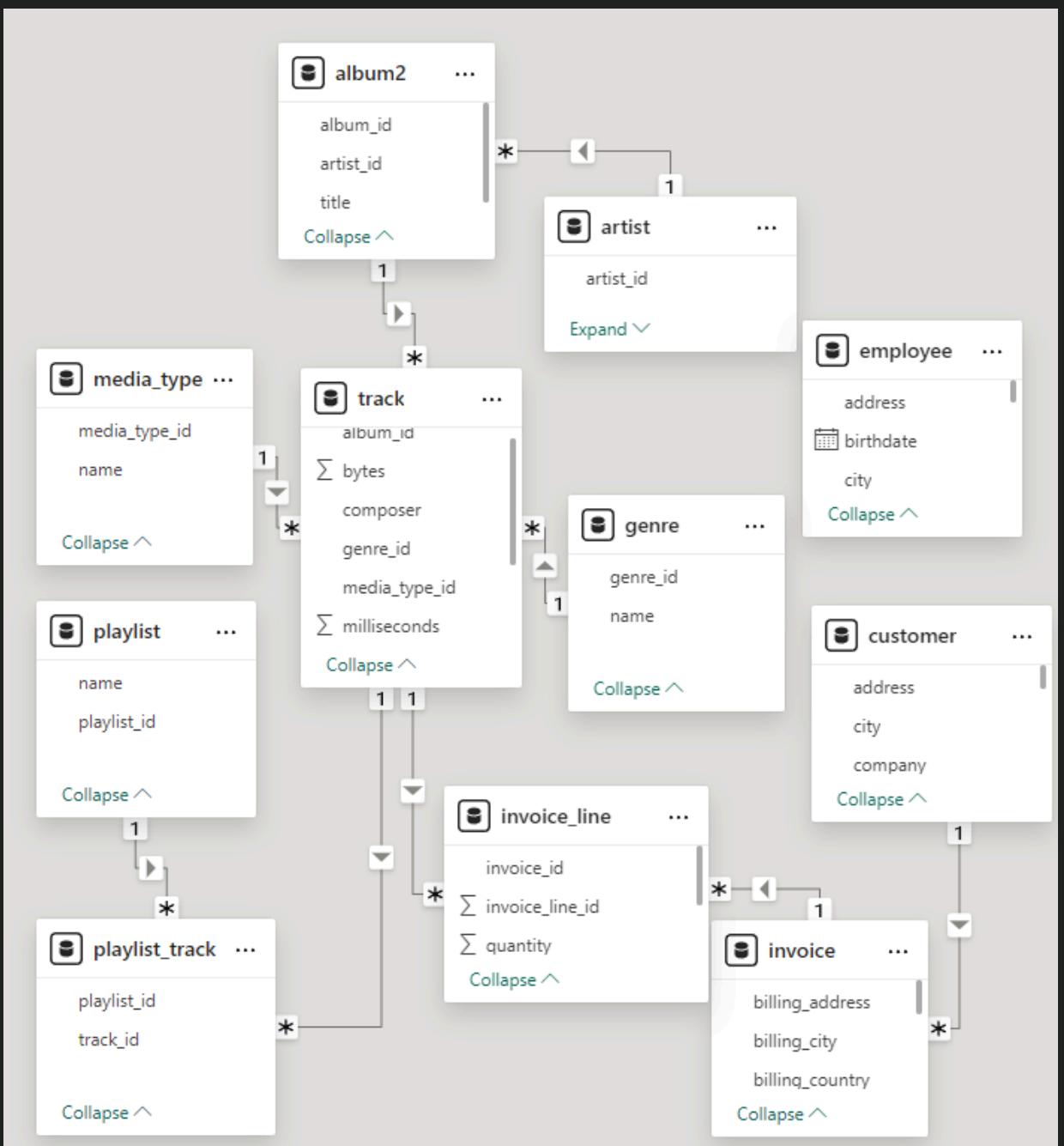


# MUSIC STORE ANALYSIS with mySQL

---



# DATASET ER DIAGRAM





Q1: Who is the senior most employee based on job title?

```
select first_name,last_name,title from employee order by levels desc limit 1;
```

**Result Grid** | Filter Rows:

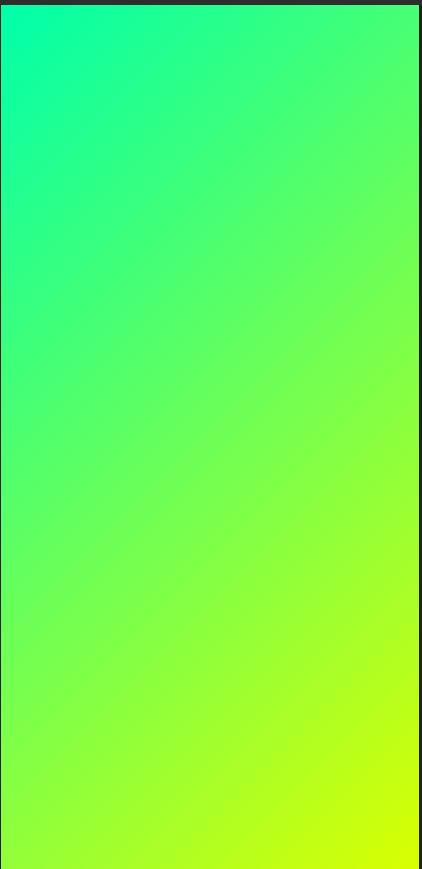
	first_name	last_name	title
▶	Andrew	Adams	General Manager

Q2: Which countries have the most Invoices?

```
select billing_country ,count(invoice_id) as no_of_invoice  
from invoice group by billing_country  
order by no_of_invoice desc limit 10;
```

**Result Grid** | Filter Rows:

	billing_country	no_of_invoice
▶	USA	131
	Canada	76
	Brazil	61
	France	50
	Germany	41
	Czech Republic	30
	Portugal	29
	United Kingdom	28
	India	21
	Ireland	13



Q3: What are top 3 values of total invoice?

```
delimiter //
create procedure total_n(in N int)
begin
select * from invoice order by total desc limit N;
end //
delimiter ;
```

	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

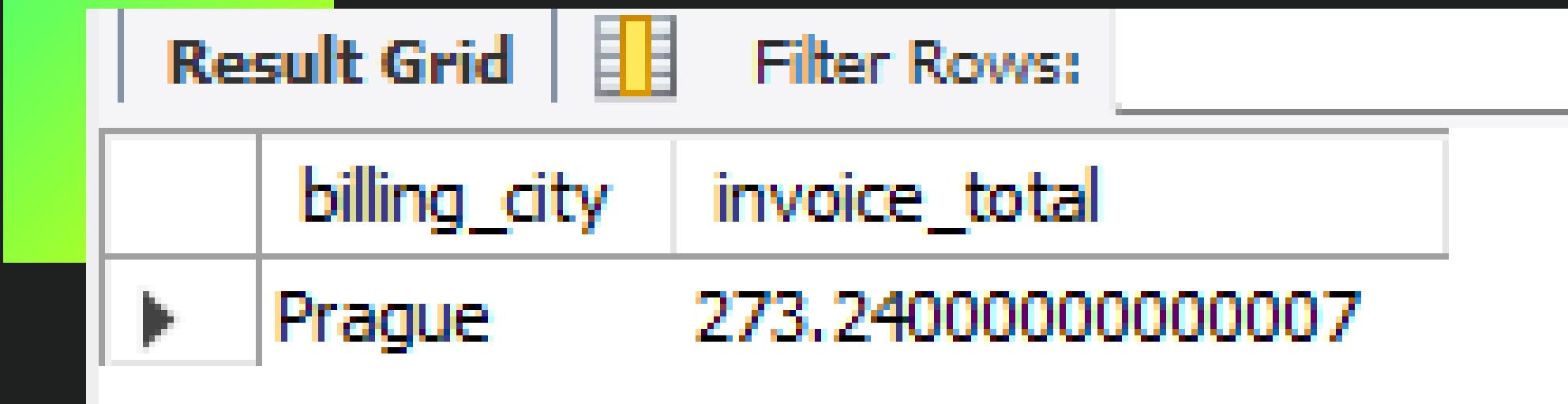


# OUR MISSION

 Q4: 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

 delimiter //  
create procedure best\_city(in N int)  
begin  
select billing\_city,sum(total) as invoice\_total  
from invoice group by billing\_city  
order by invoice\_total desc limit N;  
end //  
delimiter ;  
call best\_city(1);

Prague is the city which has best customers the store should throw a promotional music festival in Prague.



Result Grid	Filter Rows:	
	billing_city	invoice_total
	Prague	273.24000000000007



Q5: 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 c.first_name,c.last_name,c.city,sum(i.total) as total_spend  
from customer as c join invoice as i on c.customer_id=i.customer_id  
group by c.first_name,c.last_name,c.city order by total_spend desc limit 1;
```

# František is the best customer from Prague with total spending of ~144.54

The screenshot shows a database query results grid. The grid has four columns: first\_name, last\_name, city, and total\_spend. The data row shows a single result: František Wichterlová from Prague with a total spend of 144.54.

	first_name	last_name	city	total_spend
▶	František	Wichterlová	Prague	144.54000000000002



Q6: 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 c.email,c.first_name,c.last_name,g.name  
from customer as c join invoice as i on c.customer_id=i.customer_id  
join invoice_line il on i.invoice_id=il.invoice_id  
join track as t on il.track_id=t.track_id  
join genre as g on t.genre_id=g.genre_id  
where g.name="Rock" order by c.email;
```

	email	first_name	last_name	name
▶	aaronmitchell@yahoo.ca	Aaron	Mitchell	Rock
	aaronmitchell@yahoo.ca	Aaron	Mitchell	Rock
	aaronmitchell@yahoo.ca	Aaron	Mitchell	Rock
	aaronmitch aaronmitchell@yahoo.ca		Mitchell	Rock
	alero@uol.com.br	Alexandre	Rocha	Rock
	astrid.gruber@apple.at	Astrid	Gruber	Rock
	astrid.gruber@apple.at	Astrid	Gruber	Rock
	astrid.gruber@apple.at	Astrid	Gruber	Rock
	astrid.gruber@apple.at	Astrid	Gruber	Rock
	astrid.gruber@apple.at	Astrid	Gruber	Rock
	astrid.gruber@apple.at	Astrid	Gruber	Rock
	bjorn.hansen@yahoo.no	BjÃ¸rn	Hansen	Rock
	camille.bernard@yahoo.fr	Camille	Bernard	Rock
	camille.bernard@yahoo.fr	Camille	Bernard	Rock
	camille.bernard@yahoo.fr	Camille	Bernard	Rock
	daan_peeters@apple.be	Daan	Peeters	Rock

Q7: 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 a.name,g.name,count(t.track_id) as total_track  
from artist as a join album2 as al on a.artist_id = al.artist_id  
join track as t on al.album_id = t.album_id  
join genre as g on t.genre_id = g.genre_id  
where g.name="Rock" group by a.name,g.name  
order by total_track desc limit 10;
```

	name	name	total_track
▶	AC/DC	Rock	18
	Aerosmith	Rock	15
	Audioslave	Rock	14
	Led Zeppelin	Rock	14
	Alanis Morissette	Rock	13
	Alice In Chains	Rock	12
	Frank Zappa & Captain Beefheart	Rock	9
	Accept	Rock	4





Q8: 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.

```
select t.name,g.name,t(milliseconds) from track as t
join genre as g on t.genre_id = g.genre_id
where t(milliseconds) > ( select avg(milliseconds) from track)
group by t.name,g.name,t(milliseconds)
order by t(milliseconds) desc;
```

	name	name	milliseconds
▶	How Many More Times	Rock	711836
	Advance Romance	Rock	677694
	Sleeping Village	Metal	644571
	You Shook Me(2)	Rock	619467
	Talkin' 'Bout Women Obviously	Blues	589531
	Stratus	Jazz	582086
	No More Tears	Metal	555075
	The Alchemist	Metal	509413
	Wheels Of Confusion / The Straightener	Metal	494524
	Book Of Thel	Metal	494393
	You Oughta Know (Alternate)	Rock	491885
	Terra	Latin	482429
	Snoopy's search-Red baron	Jazz	456071
	Sozinho (Hitmakers Classic Mix)	Latin	436636
	Master Of Puppets	Metal	436453
	Stone Crazy	Blues	433397

Q9: Find how much amount spent by each customer on artists?  
Write a query to return customer name, artist name and total spent.

```
select c.first_name,c.last_name,a.name as artist_name,sum(in_l.unit_price*in_l.quantity) as total_sales
from customer as c join invoice as i on c.customer_id=i.customer_id
join invoice_line as in_l on i.invoice_id=in_l.invoice_id
join track as t on in_l.track_id=t.track_id
join album2 as a2 on t.album_id=a2.album_id
join artist as a on a2.artist_id=a.artist_id
group by c.first_name,c.last_name,a.name;
```

	first_name	last_name	artist_name	total_sales
▶	Martha	Silk	Aerosmith	0.99
	StanisÅaw	WÅ³jciech	Alanis Morissette	0.99
	LuÅ-s	GonÅsalves	AC/DC	7.920000000000001
	LuÅ-s	GonÅsalves	Aerosmith	2.9699999999999998
	Dan	Miller	Alice In Chains	2.9699999999999998
	Isabelle	Mercier	Alice In Chains	1.98
	Kara	Nielsen	AC/DC	1.98
	Phil	Hughes	AC/DC	10.89
	Camille	Bernard	Alanis Morissette	0.99
	Camille	Bernard	Aerosmith	0.99
	Roberto	Almeida	Aerosmith	1.98
	Hugh	O'Reilly	Aerosmith	0.99
	Edward	Francis	Alanis Morissette	12.870000000000001





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 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 c.country,g.name,count(in_l.quantity)as purchases,
row_number() over(partition by c.country order by count(in_l.quantity) desc) as row_no from
customer as c join invoice as i on c.customer_id=i.customer_id
join invoice_line as in_l on i.invoice_id=in_l.invoice_id
join track as t on in_l.track_id=t.track_id
join genre as g on t.genre_id=g.genre_id group by c.country,
g.name order by c.country asc, purchases desc)
select * from popular_genre where row_no <=1;
```



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.

```
select c.first_name,c.last_name,c.country,sum(i.total)as total_sepending
from customer as c join invoice as i
group by c.first_name,c.last_name,c.country
order by c.country asc,
total_sepending desc;
```

Result Grid				Filter Rows:	Export:
	first_name	last_name	country	total_sepending	
▶	Diego	GutiÃ©rrez	Argentina	4709.429999999994	
▶	Mark	Taylor	Australia	4709.429999999994	
▶	Astrid	Gruber	Austria	4709.429999999994	
▶	Daan	Peeters	Belgium	4709.429999999994	
▶	LuÃ­s	GonÃ¡lves	Brazil	4709.429999999994	
▶	Eduardo	Martins	Brazil	4709.429999999994	
▶	Alexandre	Rocha	Brazil	4709.429999999994	



THANK YOU



---