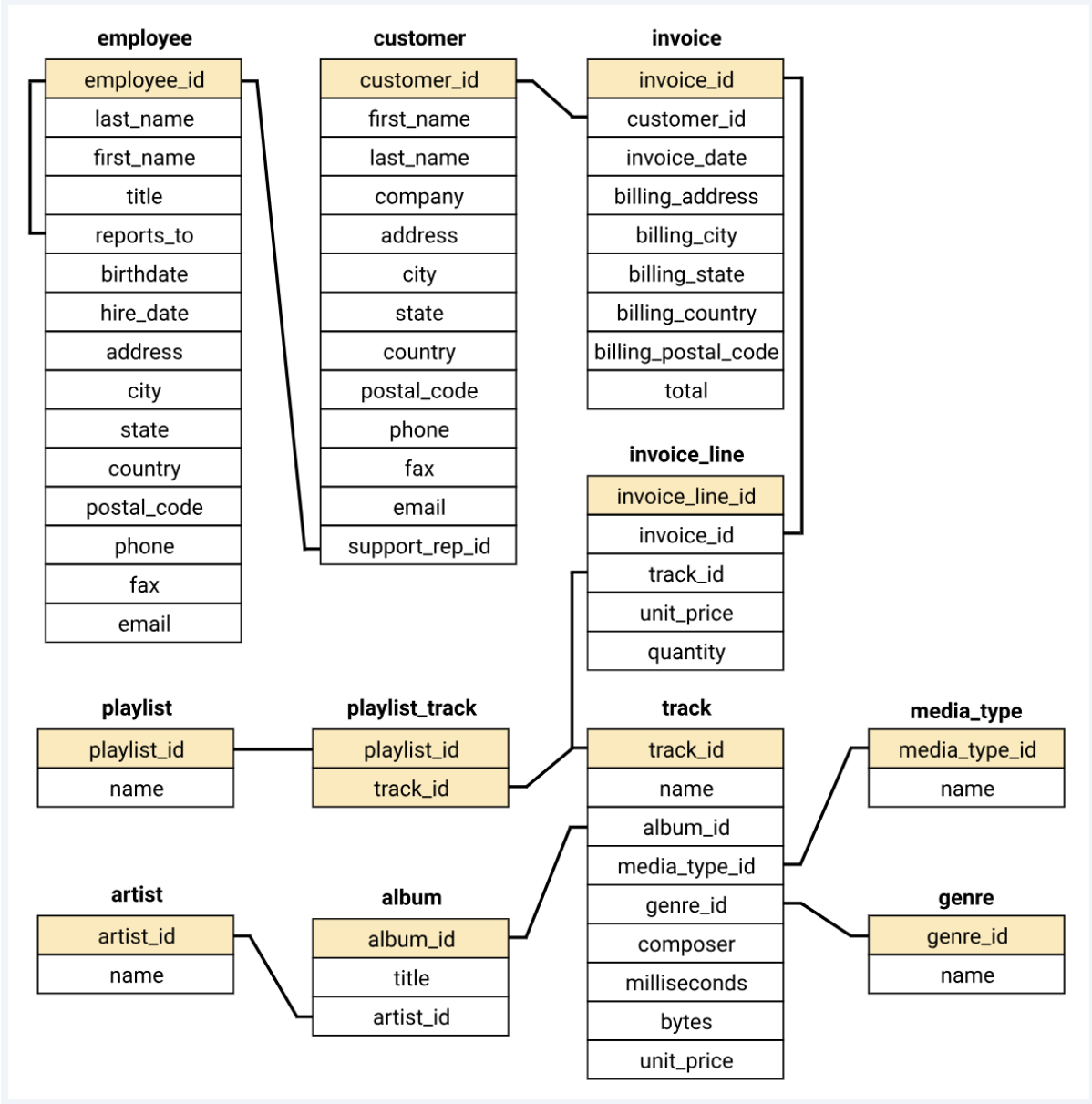
SQL Project – Music Store Data Analysis

Schema



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

```
select title,max(extract(year from age(birthdate))) as Senior from employee
group by title
order by title desc;
```

2. Which countries has the most invoices?

```
select billing_country,count(*) from invoice
group by billing_country
order by 2 desc
;
```

3. What are the top 3 values of total invoice?

```
select billing_country,sum(total) as Total from invoice
group by billing_country
order by 2 desc
limit 3;
```

4. Which city has the best customer? 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 billing_city,sum(total) from invoice
group by billing_city
order by 2 desc
limit 1
```

5. 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,sum(i.total) as total from customer c
inner join invoice i on c.customer_id = i.customer_id
group by c.first_name
order by 2 desc
limit 1
```

6. Write a query to return email, firstname, lastname and genre of all Rock Music listeners. Return your list ordered alphabetically by email starting with A.

```
select distinct first_name, last_name,email from customer c
inner join invoice i on c.customer_id=i.invoice_id
inner join invoice_line l on i.invoice_id=l.invoice_id
where track_id in (select track_id from track as tr
          join genre as g on tr.genre_id=g.genre_id where g.name like '%Rock%')
order by email
```

7. Lets 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,count(a.artist_id)as no_of_songs from artist a
join album al on a.artist_id=al.artist_id
join track t on al.album_id=t.album_id
where t.genre_id::int=1
group by a.artist_id
order by no_of_songs desc
limit 10
```

8. Return all the track names that have a song length longer than average long length. Return the Name and Milliseconds of each track. Order by song length with longest songs listed first.

```
select name, milliseconds as average_second from track
where milliseconds > (select avg(milliseconds) from track)
order by average_second desc
```

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

```
with CTE as(
          select a.artist_id,a.name,sum(il.unit_price*il.quantity) from artist a
          join album al on a.artist_id=al.artist_id
          join track tr on al.album_id=tr.album_id
          join invoice_line il on il.track_id=tr.track_id
          group by 1,2
          order by 3 desc
          limit 1
)

select c.first_name,c.last_name,mt.name,sum(il.unit_price*il.quantity) as total from customer c
join invoice i on c.customer_id=i.customer_id
join invoice_line il on i.invoice_id=il.invoice_id
join track tr on tr.track_id = il.track_id
join album al on tr.album_id=al.album_id
join CTE mt on mt.artist_id=al.artist_id
group by 1,2,3
order by total desc
```