```
--Q. Who is the most senior employeed based on job title--
select * from employee
order by levels desc
limit 1

--Q. Which countries have the most # of invoices--
select count(*) as c, billing_country
from invoice
group by billing_country
order by c desc

--Q. What are top 3 value of total invoice--
select total from invoice
order by total desc
limit 3

--Q.Which city has the best customer, we would like to throw promotional
music festival in the city
---   we made the most money. Write a query that return one city that has
highest sum of invoices
--- total. Return both city & name of all invoices
select  sum(total) as invoice_total,billing_city
from invoice
group by billing_city
order by invoice_total desc

--Q.Who is the best customer?The customer who has spent the most money
will be declared as best
--customer write a query that return the person who has spent most money-
-
select customer.customer_id,customer.first_name,customer.last_name,
SUM(invoice.total) as total
from customer
join invoice on customer.customer_id = invoice.customer_id
group by customer.customer_id
order by total desc
limit 1

--Q.Write a query to return the email, first name, lastname, & Genre of
all rock music listener.
----Return your list alphabetically email starting with aplhabet A.
select distinct email, first_name, last_name
from customer
join invoice on customer.customer_id = invoice.customer_id
join invoice_line on invoice.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 = 'Rock'
)
order by email;

--Q.Lets invite who have writeen the most rock music in our dataset.Write
a query that return artist name & total track count
----of top 10 rock bands.

select artist.artist_id, artist.name,COUNT(artist.artist_id) AS
number_of_songs
```

```
from track
join album on album.album_id = track.album_id
join artist on artist.artist_id = album.artist_id
join genre on genre.genre_id = track.genre_id
WHERE genre.name LIKE 'Rock'
group by artist.artist_id
order by number_of_songs DESC
LIMIT 10;

--Q. Return all the track name that have song length longer than average
song length
---- Return name & miliseconds of each track. Order by song length with
longest song listed first
select name, milliseconds
from track
where milliseconds > (
            SELECT AVG(milliseconds) AS avg_track_length
            from track)
ORDER BY milliseconds DESC;


--Q. Find how much amount spent by each customer on artist? Write query
to return artist name,
----- customer name & 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_sales
      from invoice_line
      JOIN track on track.track_id = invoice_line.track_id
      JOIN album on album.album_id = track.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 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;
```