Using PostGre SQL

The Easy set ------

**/\* Q1. Who is the senior most employee based on the job title \*/**

**select \* from employee**

**order by levels desc**

**limit 1**

**/\* Q2 which countries have the most Invoices \*/**

**select count(\*) as Count, billing\_country**

**from invoice**

**group by billing\_country**

**order by Count desc**

**/\***

**3. What are the top 5 values of Total Invoice**

**\*/**

**--select \* from invoice**

**select total from invoice**

**order by total desc**

**limit 5**

**/\***

**4. which city has the best customer? write a query that returns one city that has highest sum of invoice totals.**

**return both city name and sum of all invoice total**

**\*/**

**select sum(total) as Total\_Invoice, billing\_city**

**from invoice**

**group by billing\_city**

**order by Total\_Invoice desc**

**/\***

**5. Who is the best customer? The customer who has spent most money will be declared as the best customer.**

**write a query who have spent most pf the money**

**\*/**

**--select \* from invoice**

**--select \* from customer**

**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**

**Moderate Type Question**

**/\***

**1. Write a query to return the email, first name, last name, and Genre of all music listeners**

**Return your list ordered alphabetically by email starting with 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 like 'Rock'**

**)**

**order by email;**

**/\***

**2. Lets invite the artist 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 brands**

**\*/**

**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;**

**/\***

**3. Return the name of all tracks name that have a song length longer than average song length. Return the name**

**and miliseconds for each track. Order the song length with longest songs listed first.**

**\*/**

**select \* from track;**

**select name, milliseconds**

**from track**

**where milliseconds > (**

**select avg(milliseconds) as avg\_track\_length**

**from track)**

**order by milliseconds DESC;**

**Advance Set**

**/\***

**1. Find out how much amount spent by each customers on the artist? write a query to return the 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\_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 customer.customer\_id, customer.first\_name, customer.last\_name, best\_selling\_artist.Artist\_Name,**

**sum (invoice\_line.unit\_price \* invoice\_line.quantity) as amount\_spent**

**from invoice**

**join customer on customer.customer\_id = invoice.customer\_id**

**join invoice\_line on invoice\_line.invoice\_id = invoice.invoice\_id**

**join track on track.track\_id = invoice\_line.track\_id**

**join album on album.album\_id = track.album\_id**

**join best\_selling\_artist on best\_selling\_artist.artist\_id = album.artist\_id**

**group by 1, 2, 3, 4**

**order by 5 DESC;**

**/\***

**2. We want to find out the most popular music genre for each country.**

**we determine the most popular genre as 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 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 RowNum**

**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 RowNum <= 1**

**/\***

**3. write a query that determines the customer that has spent the most on music for each country.**

**write the query that returns the country along with the top customers and how much they spent.**

**for countries where the top amount is shared provide all customers who spent this amount**

**\*/**

**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 2, 3 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;**