

QUERIES

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

```
SELECT TITLE, LAST_NAME, FIRST_NAME  
FROM employee  
ORDER BY LEVELS DESC  
LIMIT 1;
```

Q2: Which countries have the most Invoices?

```
SELECT BILLING_COUNTRY, COUNT(*) AS TOTAL_INVOICE  
FROM INVOICE  
GROUP BY BILLING_COUNTRY  
ORDER BY TOTAL_INVOICE DESC;
```

Q3: What are top 3 values of total invoice?

```
SELECT TOTAL  
FROM INVOICE  
ORDER BY TOTAL DESC  
LIMIT 3;
```

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

```
SELECT BILLING_CITY, ROUND(SUM(TOTAL), 3) AS TOTAL_INVOICE  
FROM INVOICE  
GROUP BY BILLING_CITY  
ORDER BY TOTAL_INVOICE DESC  
LIMIT 1;
```

Question 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.CUSTOMER_ID, C.FIRST_NAME, C.LAST_NAME, SUM(I.TOTAL) TOTAL_SPENDING  
FROM CUSTOMER C  
JOIN INVOICE I ON C.CUSTOMER_ID = I.CUSTOMER_ID  
GROUP BY C.CUSTOMER_ID , C.FIRST_NAME , C.LAST_NAME  
ORDER BY TOTAL_INVOICE DESC  
LIMIT 1;
```

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 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 IL ON I.INVOICE_ID = IL.INVOICE_ID  
WHERE TRACK_ID IN (SELECT TRACK_ID  
FROM TRACK T  
JOIN GENRE G ON T.GENRE_ID = G.GENRE_ID  
WHERE G.NAME = 'ROCK')
```

ORDER BY EMAIL;

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 ARTIST.ARTIST_ID, ARTIST.NAME,COUNT(ARTIST.ARTIST_ID) AS NUMBER_OF_SONGS  
FROM ARTIST  
JOIN ALBUM ON ARTIST.ARTIST_ID = ALBUM.ARTIST_ID  
JOIN TRACK ON ALBUM.ALBUM_ID = TRACK.ALBUM_ID  
JOIN GENRE ON TRACK.GENRE_ID = GENRE.GENRE_ID  
WHERE GENRE.NAME = 'ROCK'  
GROUP BY ARTIST.ARTIST_ID , ARTIST.NAME  
ORDER BY NUMBER_OF_SONGS DESC  
LIMIT 10;
```

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 NAME, MILLISECONDS  
FROM TRACK  
WHERE MILLISECONDS > (SELECT  
    AVG(MILLISECONDS) AS AVG_TRACK_LENGTH  
    FROM TRACK)  
ORDER BY MILLISECONDS DESC;
```

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

```
WITH BEST_SELLING_ARTIST AS(  
    SELECT ARTIST.ARTIST_ID ,ARTIST.NAME,  
        SUM(IL.UNIT_PRICE*IL.QUANTITY) AS TOTAL_SALES  
    FROM INVOICE_LINE IL  
    JOIN TRACK ON IL.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 1,2  
    ORDER BY 3 DESC  
    LIMIT 1  
)  
SELECT C.CUSTOMER_ID,C.FIRST_NAME,C.LAST_NAME,BSA.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;
```

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 COUNT(INVOICE_LINE.QUANTITY) AS PURCHASE, 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;
```

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

```
WITH CUSTOMER_WITH_COUNTRY AS(
SELECT C.CUSTOMER_ID,C.FIRST_NAME,C.LAST_NAME,BILLING_COUNTRY,SUM(TOTAL) AS
TOTAL_SPENDING,
ROW_NUMBER()OVER(PARTITION BY BILLING_COUNTRY ORDER BY SUM(TOTAL) DESC) AS
ROWNO
FROM INVOICE I
JOIN CUSTOMER C ON C.CUSTOMER_ID=I.CUSTOMER_ID
GROUP BY 1,2,3,4
ORDER BY 4 ASC,5 DESC)
SELECT * FROM CUSTOMER_WITH_COUNTRY WHERE ROWNO=1;
```