

EASY QUESTION SET

QUES. 1

Who is the senior most employee based on job title?

```
select employee_id, last_name, first_name, title, levels from employee
ORDER BY levels desc
LIMIT 5
```

QUES. 2

Which countries have the most Invoices?

```
SELECT billing_country, COUNT(billing_country)as "No." from invoice
GROUP BY billing_country
ORDER BY "No." DESC
```

QUES.3

What are top 3 values of total invoice?

```
SELECT total from invoice
limit 5
```

QUES. 4

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, sum(total) as total_money FROM invoice
GROUP BY (billing_city)
ORDER BY (total_money) DESC
```

QUES. 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 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 QUESTION SET

QUES. 1

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 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
join track on invoice_line.track_id = track.track_id
join genre on track.genre_id = genre.genre_id
where genre.name = 'Rock'
order by email
```

QUES. 2

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 "No. 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 LIKE 'Rock'
GROUP BY artist.artist_id
ORDER BY "No. of Songs" DESC
LIMIT 10
```

QUES.3

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 track_id, name, milliseconds FROM track
WHERE milliseconds >
    ( SELECT AVG (milliseconds)
      FROM track
    )
ORDER BY milliseconds DESC
```

ADVANCE QUESTION SET

QUES. 1

Find how much amount is 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 AS artist_name, SUM (invoice_line.unit_price * invoice_line.quantity) as
    "total_earnings"
    FROM artist
    JOIN album ON artist.artist_id = album.artist_id
    JOIN track ON album.album_id = track.album_id
    JOIN invoice_line ON track.track_id = invoice_line.track_id
    GROUP BY 1
    ORDER BY 3 DESC
    LIMIT 1
)
SELECT c.customer_id, c.first_name, c.last_name, bsa.artist_name, bsa.artist_id, SUM(il.unit_price * il.quantity) AS
amount_spent
FROM customer c
JOIN invoice AS i ON c.customer_id = i.customer_id
JOIN invoice_line AS il ON i.invoice_id = il.invoice_id
JOIN track t ON il.track_id = t.track_id
JOIN album a ON t.album_id = a.album_id
JOIN best_selling_artist bsa ON a.artist_id = bsa.artist_id
GROUP BY 1, 2, 3, 4, 5
ORDER BY 6 DESC
```

QUES. 2

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_music_genre as (
    SELECT COUNT (il.quantity) AS purchases, c.country, g.genre_id, g.name,
    ROW_NUMBER () OVER (PARTITION BY c.country ORDER BY COUNT(il.quantity) DESC) AS row_no
    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 t ON t.track_id = il.track_id
    JOIN genre g ON t.genre_id = g.genre_id
    GROUP BY 2, 3, 4
    ORDER BY 1 DESC
)

SELECT * FROM popular_music_genre WHERE row_no = 1
```

QUES. 3

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 top_customers AS (
    SELECT c.customer_id, c.first_name, c.last_name, i.billing_country, SUM(i.total) as total_spending,
    ROW_NUMBER() OVER (PARTITION BY i.billing_country ORDER BY SUM(i.total) DESC) AS row_no
    FROM customer c
    JOIN invoice i ON c.customer_id = i.customer_id
    GROUP BY 1,2,3,4
    ORDER BY 4 ASC, 5 DESC
)
SELECT * FROM top_customers WHERE row_no =1
```