```
/*    Question Set 1 - Easy */

/Q1: Who is the senior most employee based on job title?
 HINT (Levels column has the senoirity level for the employee)

SELECT * FROM employee
ORDER BY levels DESC
LIMIT 1


/* Q2: Which are top 5 countries have the most Invoices? */
 HINT (table Invoice and billing_country column has the countries with Invoices
information)

SELECT COUNT(*) AS c, billing_country
FROM invoice
GROUP BY billing_country
ORDER BY c DESC
LIMIT 5


/* Q3: What are top 3 values of total invoice? */

SELECT total
FROM invoice
ORDER BY total DESC


/* 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 */
HINT (Use Table Inovice and 'billing_city' and 'total' column)

SELECT SUM(total) AS Total_invoices, billing_city
FROM invoice
GROUP BY billing_city
ORDER BY Total_invoices DESC
LIMIT 1;


/* Q5: 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.*/
HINT (Use 'Customer' and 'Invoice' table to JOIN on Customer_ID)

SELECT customer.customer_id, customer.first_name, customer.last_name,
SUM(invoice.total) AS total_spent
from customer
JOIN invoice
ON customer.customer_id = invoice.customer_id
GROUP BY customer.customer_id
ORDER BY total_spent DESC
LIMIT 1




/* Question Set 2 - Moderate */
```

```
/* Q1: 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. */

/*Method 1 */

SELECT DISTINCT email,first_name, last_name
FROM customer
JOIN invoice ON customer.customer_id = invoice.customer_id
JOIN invoiceline ON invoice.invoice_id = invoiceline.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;


/* Method 2 */

SELECT DISTINCT email AS Email,first_name AS FirstName, last_name AS LastName,
genre.name AS Name
FROM customer
JOIN invoice ON invoice.customer_id = customer.customer_id
JOIN invoiceline ON invoiceline.invoice_id = invoice.invoice_id
JOIN track ON track.track_id = invoiceline.track_id
JOIN genre ON genre.genre_id = track.genre_id
WHERE genre.name LIKE 'Rock'
ORDER BY email;


/* Q2: 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 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;


/* Q3: 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,miliseconds
FROM track
WHERE miliseconds > (
      SELECT AVG(miliseconds) AS avg_track_length
      FROM track )
ORDER BY miliseconds DESC;
```

```
/* Question Set 3 - Advance */

/* Q1: Find how much amount spent by each customer on artists? Write a query to
return customer name, artist name and total spent */

/* Steps to Solve: First, find which artist has earned the most according to the
InvoiceLines. Now use this artist to find
which customer spent the most on this artist. For this query, you will need to use
the Invoice, InvoiceLine, Track, Customer,
Album, and Artist tables. Note, this one is tricky because the Total spent in the
Invoice table might not be on a single product,
so you need to use the InvoiceLine table to find out how many of each product was
purchased, and then multiply this by the price
for each artist. */

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;


/* Q2: 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. */

/* Steps to Solve:  There are two parts in question- first most popular music genre
and second need data at country level. */

/* Method 1: Using CTE */

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 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


/* Method 2: : Using Recursive */

WITH RECURSIVE
      sales_per_country AS(
            SELECT COUNT(*) AS purchases_per_genre, customer.country, genre.name,
genre.genre_id
            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
      ),
      max_genre_per_country AS (SELECT MAX(purchases_per_genre) AS
max_genre_number, country
            FROM sales_per_country
            GROUP BY 2
            ORDER BY 2)

SELECT sales_per_country.*
FROM sales_per_country
JOIN max_genre_per_country ON sales_per_country.country =
max_genre_per_country.country
WHERE sales_per_country.purchases_per_genre =
max_genre_per_country.max_genre_number;


/* Q3: 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. */

/* Steps to Solve:  Similar to the above question. There are two parts in question-

first find the most spent on music for each country and second filter the data for
respective customers. */

/* Method 1: using CTE */

WITH Customter_with_country AS (
            SELECT
customer.customer_id,first_name,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
            JOIN customer ON customer.customer_id = invoice.customer_id
            GROUP BY 1,2,3,4
            ORDER BY 4 ASC,5 DESC)
SELECT * FROM Customter_with_country WHERE RowNo <= 1


/* Method 2: Using Recursive */

WITH RECURSIVE
      customter_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 customter_with_country
            GROUP BY billing_country)

SELECT cc.billing_country, cc.total_spending, cc.first_name, cc.last_name,
cc.customer_id
FROM customter_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;
```