

SQL PROJECT- MUSIC STORE DATA ANALYSIS

Question Set 1 - Easy

1. Who is the senior most employee based on job title?

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure, including the 'public' schema and various objects like Aggregates, Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (11). The main pane shows a SQL query editor with the following query:

```
1 Who is the senior most employee based on job title?
2
3 SELECT * FROM EMPLOYEE
4 ORDER BY levels DESC
5 LIMIT 1
6
```

The 'Data Output' tab shows the result of the query:

employee_id	last_name	first_name	title	reports_to	levels	birth_date
9	Madan	Mohan	Senior General Manager	[null]	L7	15

The status bar at the bottom indicates 'Total rows: 1 of 1' and 'Query complete 00:00:00.344'.

2. Which countries have the most Invoices?

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure, including the 'public' schema and various objects like Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (11). The main pane shows a SQL query editor with the following query:

```
1 2. Which countries have the most Invoices?
2
3 SELECT COUNT(*) AS C, billing_country
4 FROM invoice
5 GROUP BY billing_country
6 ORDER BY C DESC
7
```

The 'Data Output' tab shows the result of the query:

c	billing_country
131	USA
76	Canada
61	Brazil
50	France
41	Germany
30	Czech Republic
29	Portugal

The status bar at the bottom indicates 'Total rows: 24 of 24' and 'Query complete 00:00:00.137'.

3. What are top 3 values of total invoice?

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure, with the 'invoice' table selected under the 'Tables (11)' category. The main query editor contains the following SQL query:

```

1  3. What are top 3 values of total invoice?
2
3  SELECT total FROM invoice
4  ORDER BY total DESC
5  LIMIT 3
6

```

The 'Data Output' tab shows the results of the query:

	total	double precision
1	23.759999999999998	
2	19.8	
3	19.8	

The status bar at the bottom indicates 'Total rows: 3 of 3' and 'Query complete 00:00:00.093'.

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

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure, with the 'customer' table selected under the 'Tables (11)' category. The main query editor contains the following SQL query:

```

1  4. Which city has the best customers?
2  We would like to throw a promotional Music Festival in the city we made the most money.
3  Write a query that returns one city that has the highest sum of invoice totals.
4  Return both the city name & sum of all invoice totals
5
6  SELECT SUM(total) AS invoice_total, billing_city
7  FROM invoice
8  GROUP BY billing_city
9  ORDER BY invoice_total desc
10
11

```

The 'Data Output' tab shows the results of the query:

	invoice_total	double precision	billing_city	character varying (30)
1	273.240000000000007		Prague	
2	169.29		Mountain View	
3	166.32		London	
4	158.4		Berlin	
5	151.47		Paris	
6	129.69		São Paulo	
7	114.839999999999997		Dublin	

The status bar at the bottom indicates 'Total rows: 53 of 53' and 'Query complete 00:00:00.109'.

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

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure, with the 'customer' table selected. The main pane shows a SQL query:

```

1 5. Who is the best customer?
2 The customer who has spent the most money will be declared the best customer.
3 Write a query that returns the person who has spent the most money
4
5 SELECT customer.customer_id, customer.first_name, customer.last_name, SUM(invoice.total
6 FROM customer
7 JOIN invoice ON customer.customer_id = invoice.customer_id
8 GROUP BY customer.customer_id
9 ORDER BY total DESC
10 LIMIT 1

```

The 'Data Output' tab shows the result of the query:

customer_id	first_name	last_name	total
5	R	Madhav	144.54000000000002

Total rows: 1 of 1. Query complete 00:00:00.144. Ln 10, Col 8.

Question Set 2 – Moderate

- 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

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure, with the 'customer' table selected. The main pane shows a SQL query:

```

4 SELECT DISTINCT email, first_name, last_name
5 FROM customer
6 JOIN invoice ON customer.customer_id = invoice.customer_id
7 JOIN invoice_line ON invoice.invoice_id = invoice_line.invoice_id
8 WHERE track_id IN(
9     SELECT track_id FROM track
10    JOIN genre ON track.genre_id = genre.genre_id
11    WHERE genre.name = 'Rock'
12 )
13 ORDER BY email;
14

```

The 'Data Output' tab shows the result of the query:

email	first_name	last_name
aaronmitchell@yahoo.ca	Aaron	Mitchell
alero@uol.com.br	Alexandre	Rocha
astrid.gruber@apple.at	Astrid	Gruber
bjorn.hansen@yahoo.no	Bjorn	Hansen
camille.bernard@yahoo.fr	Camille	Bernard
daan.peeters@apple.be	Daan	Peeters
diego.gutierrez@yahoo.ar	Diego	Gutiérrez

Total rows: 59 of 59. Query complete 00:00:00.273. Ln 12, Col 2.

- Let's invite the artists who have written the most rock music in our dataset. Write a

@mrijitesh

query that returns the Artist name and total track count of the top 10 rock bands

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure, with 'Tables (11)' expanded. The main pane shows a SQL query in the 'Query' tab. The query is as follows:

```

1 2. Let's invite the artists who have written the most rock music in our dataset.
2 Write a query that returns the Artist name and total track count of the top 10 rock bands
3
4 SELECT artist.artist_id, artist.name, COUNT(track.track_id) AS number_of_songs
5 FROM track
6 JOIN album ON album.album_id = track.album_id
7 JOIN artist ON artist.artist_id = album.artist_id
8 JOIN genre ON genre.genre_id = track.genre_id
9 WHERE genre.name LIKE 'Rock'
10 GROUP BY artist.artist_id, artist.name
11 ORDER BY number_of_songs DESC
12 LIMIT 10;

```

The 'Data Output' tab shows the results of the query:

	artist_id [FK] character varying (50)	name character varying (120)	number_of_songs bigint
1	22	Led Zeppelin	114
2	150	U2	112
3	58	Deep Purple	92
4	90	Iron Maiden	81
5	118	Pearl Jam	54
6	152	Van Halen	52
7	51	Queen	45

Total rows: 10 of 10 Query complete 00:00:00.095 Ln 12, Col 10

- 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

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure, with 'Tables (11)' expanded. The main pane shows a SQL query in the 'Query' tab. The query is as follows:

```

1 3. Return all the track names that have a song length longer than the average song len
2 Return the Name and Milliseconds for each track. Order by the song length with the long
3
4 SELECT track.name AS Name, track.milliseconds AS Milliseconds
5 FROM track
6 WHERE track.milliseconds > (SELECT AVG(milliseconds) FROM track)
7 ORDER BY track.milliseconds DESC;
8

```

The 'Data Output' tab shows the results of the query:

	name character varying (150)	milliseconds integer
1	Occupation / Precipice	5286953
2	Through a Looking Glass	5088838
3	Greetings from Earth, Pt. 1	2960293
4	The Man With Nine Lives	2956998
5	Battlestar Galactica, Pt. 2	2956081
6	Battlestar Galactica, Pt. 1	2952702
7	Murder On the Rising Star	2935894

Total rows: 494 of 494 Query complete 00:00:00.317 Ln 7, Col 34

Question Set 3 – Advance

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

The screenshot shows the pgAdmin 4 interface with a SQL query executed in the 'Query' tab. The query is as follows:

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

SELECT c.first_name || ' ' || c.last_name AS customer_name, a.name AS artist_name, SUM(i.invoice_line.quantity * t.track.price) AS total_spent
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 il.track_id = t.track_id
JOIN album al ON t.album_id = al.album_id
JOIN artist a ON al.artist_id = a.artist_id
GROUP BY c.customer_id, customer_name, artist_name
ORDER BY total_spent DESC;
```

The 'Data Output' tab shows the results of the query:

customer_name	artist_name	total_spent
Wyatt Girard	Frank Sinatra	570.23999999999999
Robert Brown	Creedence Clearwater Revival	396.00000000000001
R Madhav	Kiss	396.00000000000001
François Tremblay	The Who	396.00000000000001
Aaron Mitchell	James Brown	396.00000000000001
Hugh O'Reilly	Queen	392.03999999999998
Heather Leacock	House Of Pain	357.39

Total rows: 1000 of 2189 Query complete 00:00:00.264 Ln 12, Col 27

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

The screenshot shows the pgAdmin 4 interface with a SQL query executed in the 'Query' tab. The query is as follows:

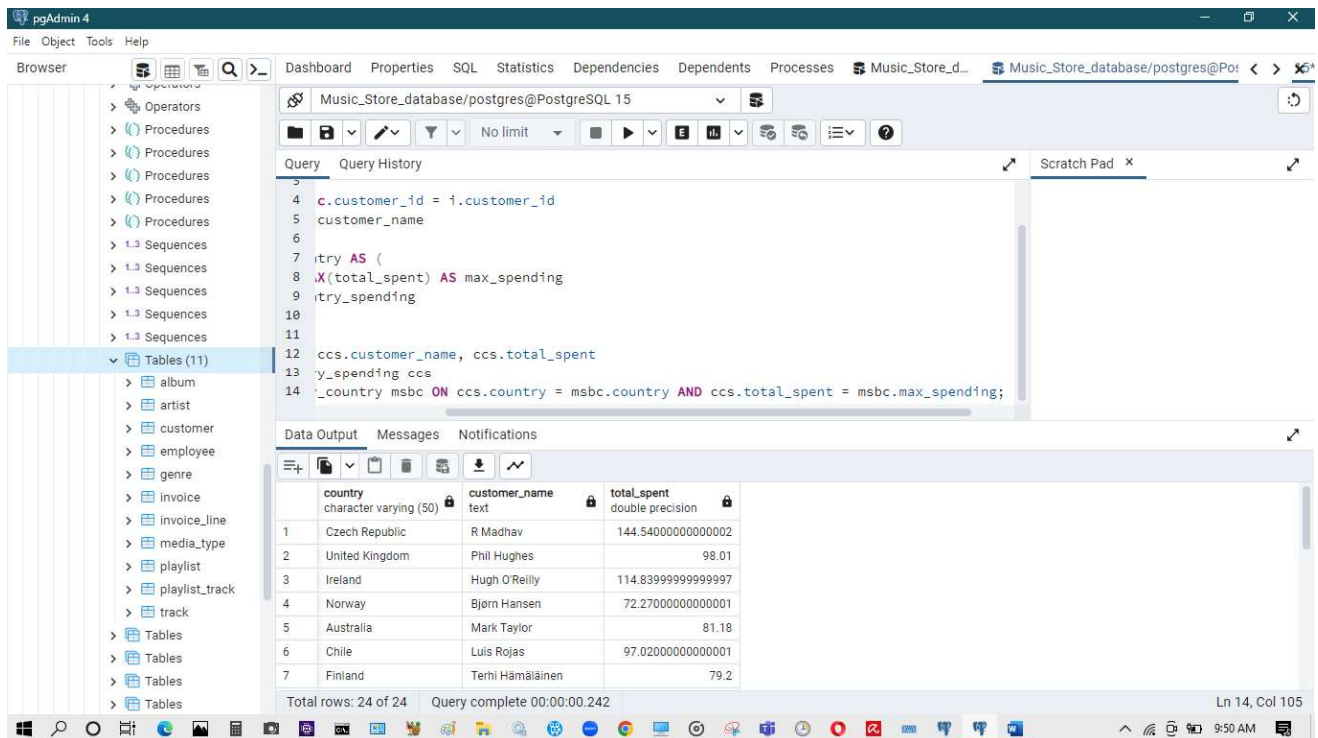
```
WITH country_genre_purchases AS (
SELECT c.country AS country, g.name AS genre, SUM(il.quantity) AS total_purchases
FROM invoice i
JOIN customer c ON i.customer_id = c.customer_id
JOIN invoice_line il ON i.invoice_id = il.invoice_id
JOIN track t ON il.track_id = t.track_id
JOIN genre g ON t.genre_id = g.genre_id
GROUP BY c.country, g.name
),
max_purchases_by_country AS (
SELECT country, MAX(total_purchases) AS max_purchases
FROM country_genre_purchases
)
SELECT country, genre
FROM country_genre_purchases
JOIN max_purchases_by_country ON country_genre_purchases.country = max_purchases_by_country.country
AND country_genre_purchases.total_purchases = max_purchases_by_country.max_purchases;
```

The 'Data Output' tab shows the results of the query:

country	genre
India	Rock
Ireland	Rock
Finland	Rock
Spain	Rock
Argentina	Alternative & Punk
USA	Rock
Poland	Rock

Total rows: 24 of 24 Query complete 00:00:00.123 Ln 1, Col 1

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



The screenshot shows the pgAdmin 4 interface with a SQL query executed in the 'Query' tab. The query is as follows:

```
4 c.customer_id = i.customer_id
5 customer_name
6
7 ctry AS (
8   X(total_spent) AS max_spending
9   ctry_spending
10
11
12 ccs.customer_name, ccs.total_spent
13 y_spending ccs
14 _country msbc ON ccs.country = msbc.country AND ccs.total_spent = msbc.max_spending;
```

The 'Data Output' tab shows the results of the query, which are displayed in a table with 3 columns: country, customer_name, and total_spent. The table contains 7 rows of data, representing the top customer for each country.

	country	customer_name	total_spent
1	Czech Republic	R Madhav	144.54000000000002
2	United Kingdom	Phil Hughes	98.01
3	Ireland	Hugh O'Reilly	114.83999999999997
4	Norway	Bjorn Hansen	72.27000000000001
5	Australia	Mark Taylor	81.18
6	Chile	Luis Rojas	97.02000000000001
7	Finland	Terhi Hämäläinen	79.2

The status bar at the bottom indicates 'Total rows: 24 of 24' and 'Query complete 00:00:00.242'.