



# Audio Store Analysis and Insights

SQL | PostgreSQL Database



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



```
1 SELECT *
2 FROM employee
3 WHERE levels = (SELECT MAX(levels) FROM employee);
```



employee_id [PK] character varying (50)	first_name character	last_name character	title character varying (50)
9	Mohan	... Madan	Senior General Manager

- Mohan Madan is the most senior employee at the level of Senior General Manager.

## 2. Which countries have the most Invoices?



```
1 SELECT  
2     billing_country,  
3     COUNT(*) AS invoice_count  
4 FROM invoice  
5 GROUP BY billing_country  
6 ORDER BY invoice_count DESC;
```

	billing_country character varying (30)	invoice_count bigint
1	USA	131
2	Canada	76
3	Brazil	61
4	France	50
5	Germany	41
6	Czech Republic	30
7	Portugal	29
8	United Kingdom	28
9	India	21
10	Chile	13
11	Ireland	13
12	Spain	11
13	Finland	11
14	Australia	10
15	Netherlands	10
16	Sweden	10
17	Poland	10
18	Hungary	10
19	Denmark	10
20	Austria	9
21	Norway	9
22	Italy	9



- The USA, Canada and Brazil have the most Invoices.

### 3. What are top 3 values of total invoice?

```
1 SELECT total  
2 FROM invoice  
3 ORDER BY total DESC  
4 LIMIT 3;
```



	total	
	double precision	🔒
1	23.759999999999998	
2		19.8
3		19.8

- Top invoices are of approx 24K\$ and 20K\$

4. Which city has the best customers? We would like to throw a promotional Music Festival in the city we made the most money.



```
1 SELECT
2     billing_city,
3     ROUND(SUM(total)::numeric,2) AS invoice_total
4 FROM invoice
5 GROUP BY billing_city
6 ORDER BY invoice_total DESC
7 LIMIT 1;
```



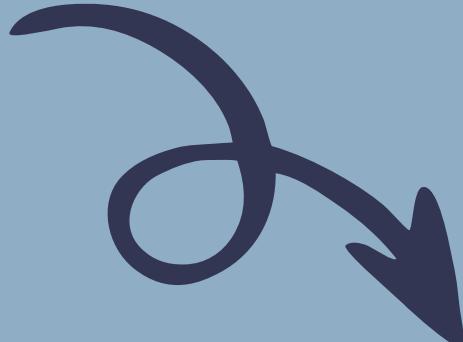
billing_city	invoice_total
character varying (30)	numeric
Prague	273.24

- Prague, the capital Czech Republic has the best customer base with total invoices of around 273K \$.

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



```
1  SELECT
2      customer.customer_id,
3      first_name,
4      last_name,
5      SUM(total) AS total_invoice
6  FROM invoice
7  JOIN customer
8  USING(customer_id)
9  GROUP BY customer.customer_id
10 ORDER BY total_invoice DESC;
```



	customer_id [PK] integer	first_name character	last_name character	total_invoice numeric
1	5	R	Madhav	144.54
2	6	Helena	Holý	128.70
3	46	Hugh	O'Reilly	114.84
4	58	Manoj	Pareek	111.87
5	1	Luís	Gonçalves	108.90

- R Madhav has spent the most amount of money.

## 6. Write a 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

```
● ● ●  
1 -- Joining track and genre table and wrapping it up in CTE  
2 WITH track_genre AS  
3 (  
4     SELECT  
5         track_id,  
6         track.name,  
7         genre.name AS genre  
8     FROM track  
9     JOIN genre  
10    USING(genre_id)  
11 )  
12 /* Joining all the relevant tables to get to the genre, Selecting all the distinct customers filtering genre so that all the genres name containing rock shows up*/  
13 SELECT DISTINCT  
14     first_name,  
15     last_name,  
16     email,  
17     genre  
18 FROM customer  
19 JOIN invoice USING(customer_id)  
20 JOIN invoice_line USING(invoice_id)  
21 JOIN track_genre USING(track_id)  
22 WHERE genre LIKE '%Rock%'  
23 ORDER BY email;
```



	first_name character	last_name character	email character varying (50)	genre character varying (120)
1	Aaron	Mitchell	aaronmitchell@yahoo.ca	Rock
2	Alexandre	Rocha	alero@uol.com.br	Rock
3	Astrid	Gruber	astrid.gruber@apple.at	Rock
4	Bjørn	Hansen	bjorn.hansen@yahoo.no	Rock
5	Camille	Bernard	camille.bernard@yahoo.fr	Rock
6	Daan	Peeters	daan_peeters@apple.be	Rock
7	Diego	Gutiérrez	diego.gutierrez@yahoo.ar	Rock
8	Dan	Miller	dmiller@comcast.com	Rock

Total rows: 59 of 59

Query complete 00:00:00.107

## 7. 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.

```
1 --Selecting artist_name, and couting total track
2   s
3   SELECT
4       artist.name as artist,
5       COUNT(DISTINCT track_id) as total_tracks
6
7   -- Joining track table with album and artist
8   FROM track
9   JOIN album USING(album_id)
10  JOIN artist USING(artist_id)
11
12  -- Filtering genre_id where genre is Rock
13  WHERE genre_id IN
14  (
15      SELECT genre_id
16      FROM genre
17      WHERE name = 'Rock'
18  )
19
20  -- Grouping by artist_name and showing only top
21  10 rows
22  GROUP BY artist.name
23  ORDER BY total_tracks DESC
24  LIMIT 10;
```



	artist character varying (120)	total_tracks bigint
1	Led Zeppelin	114
2	U2	112
3	Deep Purple	92
4	Iron Maiden	81
5	Pearl Jam	54
6	Van Halen	52
7	Queen	45
8	The Rolling Stones	41
9	Creedence Clearwater Revival	40
10	Kiss	35

- Led Zeppelin and U2 have the most number of music tracks

8. 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.

```
● ● ●  
1  SELECT DISTINCT  
2      name,  
3      ROUND(  
4          milliseconds::numeric/(1000*60)  
5          ,2) as minutes  
6  FROM track  
7  WHERE milliseconds > (  
8      SELECT AVG(milliseconds)  
9      FROM track  
10 )  
11 ORDER BY minutes DESC;
```



	name character varying (150)	minutes numeric
1	Occupation / Precipice	88.12
2	Through a Looking Glass	84.81
3	Greetings from Earth, Pt. 1	49.34
4	The Man With Nine Lives	49.28
5	Battlestar Galactica, Pt. 2	49.27
6	Battlestar Galactica, Pt. 1	49.21
7	Murder On the Rising Star	48.93
8	Battlestar Galactica, Pt. 3	48.80
9	Take the Celestra	48.79
10	Fire In Space	48.78
11	The Long Patrol	48.75
12	The Magnificent Warriors	48.75
13	The Gun On Ice Planet Zero, Pt. 2	48.74

Total rows: 491 of 491    Query complete 00:00:00.119

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

```
● ● ●  
1  -- Creating a CTE for joining track,album and artist table  
2  WITH track_details AS  
3  (  
4      SELECT  
5          track_id,  
6          track.name AS track_name,  
7          artist.name AS artist_name  
8      FROM track  
9      JOIN album USING(album_id)  
10     JOIN artist USING(artist_id)  
11 ),  
12  -- CTE for joining customer, invoice and invoice_line table  
13 customer_invoice AS  
14 (  
15      SELECT  
16          *  
17      FROM customer  
18      JOIN invoice USING(customer_id)  
19      JOIN invoice_line USING(invoice_id)  
20 )  
21  
22  -- Now Joining both results of CTEs and calculating total sum of spent and grouping my customer details and artist_name  
23 SELECT  
24     c.customer_id,  
25     c.first_name,  
26     c.last_name,  
27     t.artist_name,  
28     ROUND(SUM(c.unit_price*c.quantity)::numeric,2)  
     AS total_spent  
29 FROM customer_invoice AS c  
30 JOIN track_details AS t  
31 USING(track_id)  
32 GROUP BY 1,2,3,4  
33 ORDER BY total_spent DESC;
```



	customer_id	first_name	last_name	artist_name	total_spent
	integer	character	character	character varying (120)	numeric
1	46	Hugh	O'Reilly	Queen	27.72
2	42	Wyatt	Girard	Frank Sinatra	23.76
3	3	François	Tremblay	The Who	19.80
4	6	Helena	Holý	Red Hot Chili Peppers	19.80
5	5	R	Madhav	Kiss	19.80
6	29	Robert	Brown	Creedence Clearwater Revival	19.80
7	32	Aaron	Mitchell	James Brown	19.80
8	22	Heather	Leacock	House Of Pain	18.81
9	46	Hugh	O'Reilly	Nirvana	18.81
10	38	Niklas	Schröder	Queen	18.81

- Hugh O'Reilly has spent the most on band Queen.

# 10. 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.

```
● ● ●  
1  -- Using subquery in FROM clause  
2  SELECT *  
3  FROM  
4  (  
5      SELECT  
6          billing_country,  
7          genre.name AS genre,  
8          -- Count the invoice lines to calculate total purchases  
9          COUNT(invoice_line_id) AS total_purchases,  
10         -- Add ranking at country level on total purchases, so  
11         that max will be ranked as 1  
12         RANK() OVER(PARTITION BY billing_country  
13                         ORDER BY COUNT(invoice_line_id) DESC) AS ranking  
14  
15         -- Join invoice, invoice_line, track and genre table on res  
16         pective id columns  
17         FROM invoice  
18         JOIN invoice_line USING(invoice_id)  
19         JOIN track USING(track_id)  
20         JOIN genre USING(genre_id)  
21  
22         -- Group the data by country and genre  
23         GROUP BY genre.name, billing_country  
24         ORDER by billing_country, total_purchases DESC  
25     )  
26  
27     -- Filtered the data where rank is 1, in this way we determine t  
28     he most popular genre as the genre with the highest amount of p  
29     urchases.  
30 WHERE ranking = 1;
```



	billing_country character varying (30) 	genre character varying (120) 	total_purchases bigint 
1	Argentina	Alternative & Punk	17
2	Australia	Rock	34
3	Austria	Rock	40
4	Belgium	Rock	26
5	Brazil	Rock	205
6	Canada	Rock	333
7	Chile	Rock	61
8	Czech Republic	Rock	143
9	Denmark	Rock	24
10	Finland	Rock	46
11	France	Rock	211
12	Germany	Rock	194
13	Hungary	Rock	44
		--	--
Total rows: 24 of 24		Query complete 00:00:00.199	

- Most of the countries have Rock genre as the most popular.

# 11. 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.



```
1 -- Creating CTE for customer rankings
2 WITH customer_rankings_by_country AS
3 (
4     SELECT
5         country,
6         customer_id,
7         first_name,
8         last_name,
9         -- Summing spent amount
10        SUM(total) AS total_spent,
11        -- Ranking by total spent by country level
12        RANK() OVER(PARTITION BY country
13                                         ORDER BY SUM(total) DESC) AS ranking
14     FROM invoice
15     JOIN customer USING(customer_id)
16     GROUP BY 1,2,3,4
17     ORDER BY country, total_spent DESC
18 )
19
20 -- Selecting only rows with ranking 1, so that top customers are selected
21 SELECT
22     country,
23     customer_id,
24     first_name,
25     last_name,
26     ROUND(total_spent::numeric,2) AS spent
27     FROM customer_rankings_by_country
28     WHERE ranking = 1
29     ORDER BY spent DESC;
```

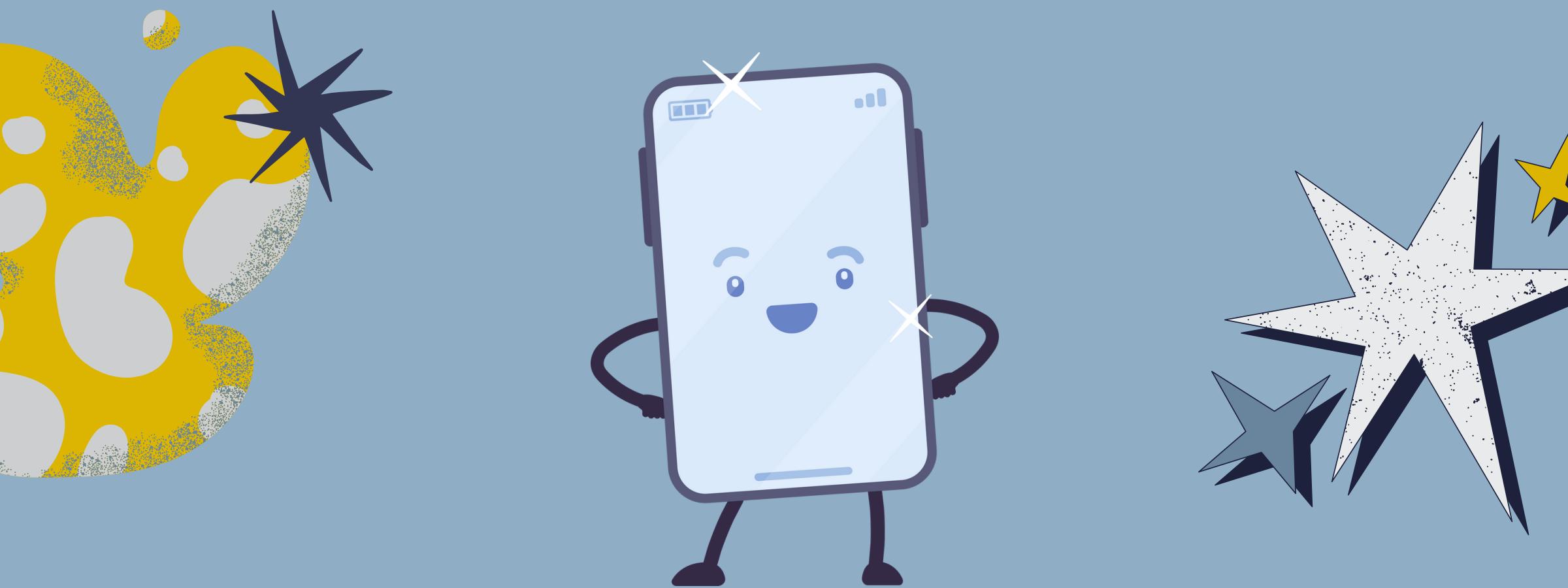


	country character varying (50)	customer_id integer	first_name character	last_name character	spent numeric
1	Czech Republic	5	R	Madhav	144.54
2	Ireland	46	Hugh	O'Reilly	114.84
3	India	58	Manoj	Pareek	111.87
4	Brazil	1	Luis	Gonçalves	108.90
5	Portugal	34	João	Fernandes	102.96
6	France	42	Wyatt	Girard	99.99
7	Canada	3	François	Tremblay	99.99
8	Spain	50	Enrique	Muñoz	98.01
9	United Kingdom	53	Phil	Hughes	98.01

Total rows: 24 of 24    Query complete 00:00:00.163

- Showing all countries with their top customer by spent.

# THANK YOU !



ALI BIN KASHIF – DATA ANALYST