



# **DIGITAL MUSIC STORE**

## **DATA ANALYSIS PROJECT USING SQL**

# OBJECTIVE

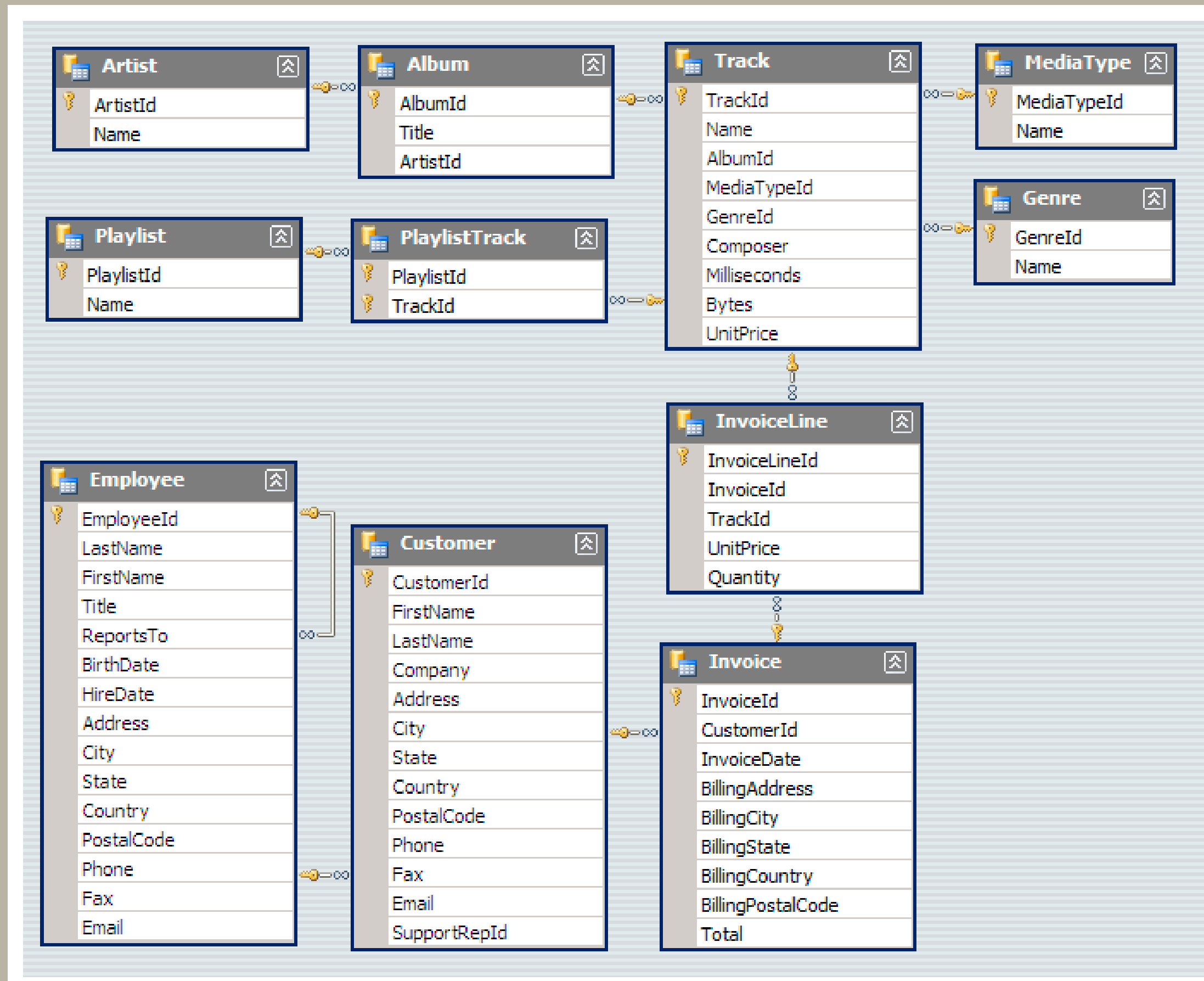
This project is use to analyze the music playlist database. You can examine the dataset with SQL and help the store to understand the business growth by answering simple questions.

# DATABASE AND TOOLS



- PostgreSQL
- PgAdmin4

# Schema- Music Store Database



# **QUESTIONS BASED ON SQL**

# BASIC QUESTIONS

Who is the senior most employee based on job title?

```
-- Ques1: Who is the senior most employee based on job title?  
SELECT  
    *  
FROM  
    EMPLOYEE  
ORDER BY  
    LEVELS DESC  
LIMIT  
    1
```

Data Output Messages Notifications										
	employee_id [PK] character varying (50)	last_name character	first_name character	title character varying (50)	reports_to character varying (30)	levels character var	birthdate timestamp without tim	hire_date timestamp without time	address character varying (120)	city charac
1	9	Madan ...	Mohan ...	Senior General Manager	[null]	L7	1961-01-26 00:00:00	2016-01-14 00:00:00	1008 Vrinda Ave MT	Edmo

# Which countries have the most Invoices?

--Ques2: Which countries have the most Invoices?

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

Data Output

Messages

Notifications

billing\_country

character varying (30)

count\_invoices

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

Total rows: 24 of 24

Query complete 00:00:00.110

# What are top 3 values of total invoice?

--Ques3: What are top 3 values of total invoice?

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

Data Output

Messages

Notifications

	<div>total</div> <div>double precision</div> <div></div>
1	23.759999999999998
2	19.8
3	19.8



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.

```
SELECT
    BILLING_CITY,
    SUM(TOTAL) AS INVOICE_TOTAL
FROM
    INVOICE
GROUP BY
    BILLING_CITY
ORDER BY
    INVOICE_TOTAL DESC
LIMIT
    1
```

Data Output			Messages	Notifications
<div><div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div></div><div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div></div><div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div></div><div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div></div></div>				
	<div><div>billing_city</div><div>character varying (30)</div><div></div></div>	<div><div>invoice_total</div><div>double precision</div><div></div></div>		
1	Prague	273.2400000000000007		

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 AS FULL_NAME,
    SUM(INVOICE.TOTAL) AS TOTAL_SPEND
FROM
    CUSTOMER
    JOIN INVOICE ON CUSTOMER.CUSTOMER_ID = INVOICE.CUSTOMER_ID
GROUP BY
    CUSTOMER.CUSTOMER_ID
ORDER BY
    TOTAL_SPEND DESC
LIMIT
    1
```

Data Output				Messages	Notifications
<div><div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div></div>					
	customer_id [PK] integer	full_name text	total_spend double precision		
1	5	R Madhav	144.540000000000002		

# MODERATE QUESTIONS

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
WHERE
  TRACK_ID IN (
    SELECT
      TRACK_ID
    FROM
      TRACK
      JOIN GENRE ON TRACK.GENRE_ID = GENRE.GENRE_ID
    WHERE
      GENRE.NAME = 'Rock'
  )
ORDER BY
  EMAIL;
```

Data Output Messages Notifications			
	email character varying (50)	first_name character	last_name character
1	aaronmitchell@yahoo.ca	Aaron	Mitchell
2	alero@uol.com.br	Alexandre	Rocha
3	astrid.gruber@apple.at	Astrid	Gruber
4	bjorn.hansen@yahoo.no	Bjørn	Hansen
5	camille.bernard@yahoo.fr	Camille	Bernard
6	daan.peeters@apple.be	Daan	Peeters
7	diego.gutierrez@yahoo.ar	Diego	Gutiérrez
8	dmiller@comcast.com	Dan	Miller
Total rows: 59 of 59		Query complete 00:00:00.088	

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 (TRACK.TRACK_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
ORDER BY
  NUMBER_OF_SONGS DESC
LIMIT
  10
```

Data Output Messages Notifications			
	artist_id [PK] 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
8	142	The Rolling Stones	41
9	76	Creedence Clearwater Revival	40
10	52	Kiss	35

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_LENGTH
        FROM
            TRACK
    )
ORDER BY
    MILLISECONDS DESC
```

Data Output			Messages	Notifications
<div><div><div>≡+</div><div></div><div>▼</div><div></div><div>▼</div><div></div><div></div><div></div><div></div></div></div>				
	name	milliseconds		
	character varying (150)	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		
8	Battlestar Galactica, Pt. 3	2927802		
9	Take the Celestra	2927677		
10	Fire In Space	2926593		
Total rows: 494 of 494			Query complete 00:00:00.081	

# ADVANCE QUESTIONS

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 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;
```

Data Output

Messages

Notifications



	customer_id integer	first_name character	last_name character	artist_name character var	amount_spent double precision
1	46	Hugh	O'Reilly	Queen	27.719999999999985
2	38	Niklas	Schröder	Queen	18.81
3	3	François	Tremblay	Queen	17.82
4	34	João	Fernandes	Queen	16.830000000000002
5	53	Phil	Hughes	Queen	11.88
6	41	Marc	Dubois	Queen	11.88
7	47	Lucas	Mancini	Queen	10.89
8	33	Ellie	Sullivan	Queen	10.89
9	20	Dan	Miller	Queen	3.96

Total rows: 43 of 43

Query complete 00:00:00.088

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 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
```



Data Output						Messages	Notifications
<div><div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div><div><div></div><div></div></div></div></div>							
	purchases bigint	country character varying (50)	name character varying (120)	genre_id character varying (50)	rowno bigint		
1	17	Argentina	Alternative & Punk	4	1		
2	34	Australia	Rock	1	1		
3	40	Austria	Rock	1	1		
4	26	Belgium	Rock	1	1		
5	205	Brazil	Rock	1	1		
6	333	Canada	Rock	1	1		
7	61	Chile	Rock	1	1		
8	143	Czech Republic	Rock	1	1		
9	24	Denmark	Rock	1	1		
10	46	Finland	Rock	1	1		
11	211	France	Rock	1	1		
Total rows: 24 of 24		Query complete 00:00:00.142					

**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 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
```

Data Output

Messages

Notifications

	customer_id integer	first_name character	last_name character	billing_country character varying (30)	total_spending double precision	rowno bigint
1	56	Diego	Gutiérrez	Argentina	39.6	1
2	55	Mark	Taylor	Australia	81.18	1
3	7	Astrid	Gruber	Austria	69.3	1
4	8	Daan	Peeters	Belgium	60.38999999999999	1
5	1	Luis	Gonçalves	Brazil	108.89999999999998	1
6	3	François	Tremblay	Canada	99.99	1
7	57	Luis	Rojas	Chile	97.02000000000001	1
8	5	R	Madhav	Czech Republic	144.54000000000002	1
9	9	Kara	Nielsen	Denmark	37.61999999999999	1
10	44	Terhi	Hämäläinen	Finland	79.2	1
11	42	Wyatt	Girard	France	99.99	1
12	37	Fynn	Zimmermann	Germany	94.05000000000001	1

Total rows: 24 of 24

Query complete 00:00:00.052



**THANK YOU!**