Music Store Data Analysis
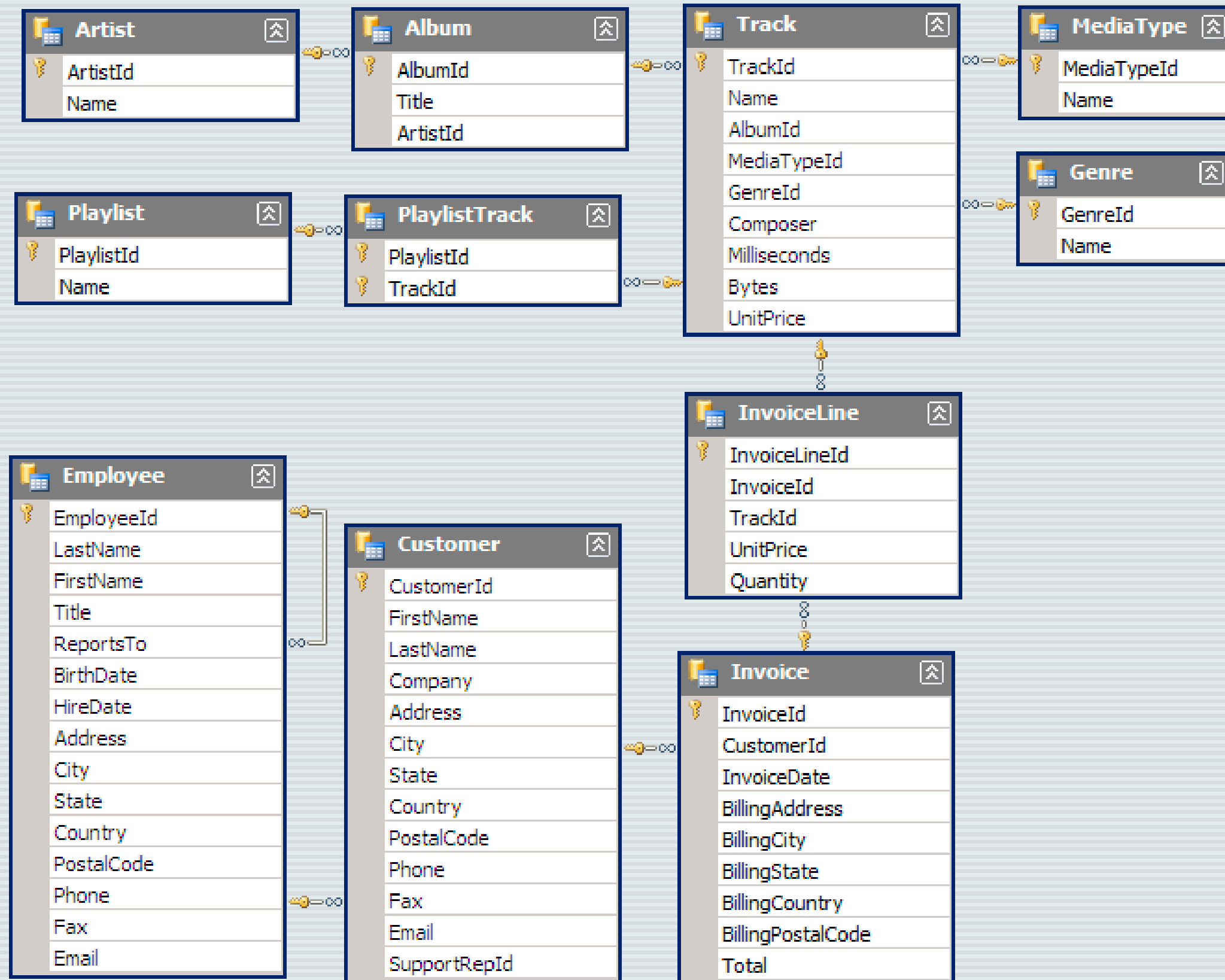
using SQL

Hello!

I'm Monika, In this project I've utilized SQL to solve questions that were related to music store.

Artist
- ArtistId
- Name

Album
- AlbumId
- Title
- ArtistId

Track
- TrackId
- Name
- AlbumId
- MediaTypeId
- GenreId
- Composer
- Milliseconds
- Bytes
- UnitPrice

MediaType
- MediaTypeId
- Name

Genre
- GenreId
- Name

Playlist
- PlaylistId
- Name

PlaylistTrack
- PlaylistId
- TrackId

InvoiceLine
- InvoiceLineId
- InvoiceId
- TrackId
- UnitPrice
- Quantity

Employee
- EmployeeId
- LastName
- FirstName
- Title
- ReportsTo
- BirthDate
- HireDate
- Address
- City
- State
- Country
- PostalCode
- Phone
- Fax
- Email

Customer
- CustomerId
- FirstName
- LastName
- Company
- Address
- City
- State
- Country
- PostalCode
- Phone
- Fax
- Email
- SupportRepId

Invoice
- InvoiceId
- CustomerId
- InvoiceDate
- BillingAddress
- BillingCity
- BillingState
- BillingCountry
- BillingPostalCode
- Total

Who is the senior most employee based on job title?

```sql
select top (1) last_name,first_name,title,levels
from employee
order by levels desc;
```

	last_name	first_name	title	levels
1	Madan	Mohan	Senior General Manager	7.00

Which countries have the most Invoices?

```sql
select count(billing_country),billing_country
from invoice
group by billing_country
order by count(billing_country) desc;
```

	(No column name)	billing_country
1	131	USA
2	76	Canada
3	61	Brazil
4	50	France
5	41	Germany
6	30	Czech Republic
7	29	Portugal
8	28	United Kingdom
9	21	India
10	13	Ireland
11	13	Chile
12	11	Finland
13	11	Spain
14	10	Sweden
15	10	Poland
16	10	Netherlands
17	10	Hungary
18	10	Denmark
19	10	Australia
20	9	Austria
21	9	Italy
22	9	Norway
23	7	Belgium
24	5	Argentina

What are top 3 values of total invoice?

```
select top (3) total
from invoice
order by total desc;
```

	total
1	23.7600002288818
2	19.7999992370605
3	19.7999992370605

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.

```sql
select top(1) sum(total),city
from customer
inner join invoice on customer.customer_id=invoice.customer_id
group by city
order by sum(total) desc;
```

	(No column name)	city
1	273.24000247955	Prague

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

```sql
select top(1) 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,customer.first_name, customer.last_name
order by total desc;
```

	customer_id	first_name	last_name	total
1	5	František	Wichterlová	144.539998054504

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

```sql
select distinct customer.email,customer.first_name,customer.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 track.track_id=invoice_line.track_id
join genre on genre.genre_id=track.genre_id
where genre.name = 'Rock'
order by email;
```

	email	first_name	last_name
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
9	dominiquelefebvre@gmail.com	Dominique	Lefebvre
10	edfrancis@yachoo.ca	Edward	Francis
11	eduardo@woodstock.com.br	Eduardo	Martins
12	ellie.sullivan@shaw.ca	Ellie	Sullivan
13	emma_jones@hotmail.com	Emma	Jones
14	enrique_munoz@yahoo.es	Enrique	Muñoz
15	fernadaramos4@uol.com.br	Fernanda	Ramos
16	fharris@google.com	Frank	Harris
17	fralston@gmail.com	Frank	Ralston
18	frantisekw@jetbrains.com	František	Wichterl...
19	ftremblay@gmail.com	François	Tremblay
20	fzimmermann@yahoo.de	Fynn	Zimmer...
21	hannah.schneider@yahoo.de	Hannah	Schneider
22	hholy@gmail.com	Helena	Holý
23	hleacock@gmail.com	Heather	Leacock
24	hughoreilly@apple.ie	Hugh	O'Reilly
25	isabelle_mercier@apple.fr	Isabelle	Mercier
26	jacksmith@microsoft.com	Jack	Smith
27	jenniferp@rogers.ca	Jennifer	Peterson
28	jfernandes@yahoo.pt	João	Fernandes
29	joakim.johansson@yahoo.se	Joakim	Johanss...
30	johavanderberg@yahoo.nl	Johannes	Van der ...
31	johngordon22@vahoo.com	John	Gordon

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

```sql
select top(10) count(track.name) as total_track_count, artist.name from artist
join album on artist.artist_id=album.artist_id
join track on track.album_id=album.album_id
join genre on genre.genre_id=track.genre_id
where genre.name = 'Rock'
group by artist.name
order by count(track.name) desc;
```

	total_track_count	name
1	114	Led Zeppelin
2	112	U2
3	92	Deep Purple
4	81	Iron Maiden
5	54	Pearl Jam
6	52	Van Halen
7	45	Queen
8	41	The Rolling Stones
9	40	Creedence Clearwater Revival
10	35	Kiss

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) from track)
order by milliseconds;
```

	name	milliseconds
1	Wicked Ways	393691
2	Concerto for Clarinet in A Major, K. 622: II. Ada...	394482
3	The Shortest Straw	395389
4	The Unforgiven II	395520
5	22 Acacia Avenue	395572
6	King For A Day	395859
7	The Thing That Should Not Be	396199
8	Save Me (Remix)	396303
9	Deep Waters	396460
10	Creeping Death	396878
11	Astronomy	397531
12	Ride The Lightning	397740
13	Still Of The Night	398210
14	Thank You	398262
15	Helpless	398315
16	Jizzlobber	398341
17	The House Jack Built	398942
18	The Edge Of Darkness	399333
19	Soot & Stars	399986
20	Terra	401319
21	Babe I'm Gonna Leave You	401475
22	Dazed and Confused	401920
23	Jerusalem	402390
24	Blackened	403382
25	The Small Hours	403435
26	Wherever I May Roam	404323
27	Computadores Fazem Arte	404323

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

```sql
SELECT c.customer_id,c.first_name,c.last_name,a.name AS artist_name,
SUM(il.unit_price * il.quantity) AS amount_spent
FROM invoice_line il
JOIN invoice i ON il.invoice_id = i.invoice_id
JOIN customer c ON i.customer_id = c.customer_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
WHERE a.artist_id = (
    SELECT TOP 1 a2.artist_id
    FROM invoice_line il2
    JOIN track t2 ON il2.track_id = t2.track_id
    JOIN album al2 ON t2.album_id = al2.album_id
    JOIN artist a2 ON al2.artist_id = a2.artist_id
    GROUP BY a2.artist_id
    ORDER BY SUM(il2.unit_price * il2.quantity) DESC
)
GROUP BY c.customer_id,c.first_name,c.last_name,a.name
ORDER BY amount_spent DESC;
```

	customer_id	first_name	last_name	artist_name	amount_spent
1	46	Hugh	O'Reilly	Queen	27.7200002670288
2	38	Niklas	Schröder	Queen	18.8100001811981
3	3	François	Tremblay	Queen	17.8200001716614
4	34	João	Fernandes	Queen	16.8300001621246
5	41	Marc	Dubois	Queen	11.8800001144409
6	53	Phil	Hughes	Queen	11.8800001144409
7	47	Lucas	Mancini	Queen	10.8900001049042
8	33	Ellie	Sullivan	Queen	10.8900001049042
9	5	František	Wichterlová	Queen	3.96000003814697
10	20	Dan	Miller	Queen	3.96000003814697
11	23	John	Gordon	Queen	2.97000002861023
12	31	Martha	Silk	Queen	2.97000002861023
13	54	Steve	Murray	Queen	2.97000002861023
14	57	Luis	Rojas	Queen	1.98000001907349
15	1	Luís	Gonçalves	Queen	1.98000001907349
16	35	Madalena	Sampaio	Queen	1.98000001907349
17	36	Hannah	Schneider	Queen	1.98000001907349
18	42	Wyatt	Girard	Queen	1.98000001907349
19	52	Emma	Jones	Queen	1.98000001907349
20	44	Terhi	Hämäläin...	Queen	1.98000001907349
21	48	Johannes	Van der B...	Queen	1.98000001907349
22	49	Stanisław	Wójcik	Queen	1.98000001907349
23	24	Frank	Ralston	Queen	1.98000001907349
24	28	Julia	Barnett	Queen	1.98000001907349
25	30	Edward	Francis	Queen	1.98000001907349
26	8	Daan	Peeters	Queen	1.98000001907349
27	11	Alexandre	Rocha	Queen	1.98000001907349
28	15	Jennifer	Peterson	Queen	1.98000001907349
29	16	Frank	Harris	Queen	1.98000001907349

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

```sql
WITH popular_genre AS
(SELECT COUNT(il.quantity) AS purchases, c.country, g.name AS genre_name, g.genre_id,
 ROW_NUMBER() OVER (PARTITION BY c.country ORDER BY COUNT(il.quantity) DESC) AS RowNo
    FROM invoice_line il
    JOIN invoice i ON il.invoice_id = i.invoice_id
    JOIN customer c ON c.customer_id = i.customer_id
    JOIN track t ON t.track_id = il.track_id
    JOIN genre g ON g.genre_id = t.genre_id
    GROUP BY c.country, g.name, g.genre_id
)SELECT country,genre_name,genre_id,purchases
FROM popular_genre
WHERE RowNo = 1
ORDER BY country;
```

	country	genre_name	genre_id	purchases
1	Argentina	Alternative & Punk	4	17
2	Australia	Rock	1	34
3	Austria	Rock	1	40
4	Belgium	Rock	1	26
5	Brazil	Rock	1	205
6	Canada	Rock	1	333
7	Chile	Rock	1	61
8	Czech Republic	Rock	1	143
9	Denmark	Rock	1	24
10	Finland	Rock	1	46
11	France	Rock	1	211
12	Germany	Rock	1	194
13	Hungary	Rock	1	44
14	India	Rock	1	102
15	Ireland	Rock	1	72
16	Italy	Rock	1	35
17	Netherlands	Rock	1	33
18	Norway	Rock	1	40
19	Poland	Rock	1	40
20	Portugal	Rock	1	108
21	Spain	Rock	1	46
22	Sweden	Rock	1	60
23	United Kingdo...	Rock	1	166
24	USA	Rock	1	561

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

```sql
WITH Customer_with_country 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 RowNo
    FROM invoice i
    JOIN customer c ON c.customer_id = i.customer_id
    GROUP BY c.customer_id, c.first_name, c.last_name, i.billing_country
)
SELECT customer_id,first_name,last_name,billing_country,total_spending
FROM Customer_with_country
WHERE RowNo = 1
ORDER BY billing_country;
```

	customer_id	first_name	last_name	billing_country	total_spending
1	56	Diego	Gutiérrez	Argentina	39.5999991893768
2	55	Mark	Taylor	Australia	81.1800003051758
3	7	Astrid	Gruber	Austria	69.299998998642
4	8	Daan	Peeters	Belgium	60.3899998664856
5	1	Luís	Gonçalves	Brazil	108.899999380112
6	3	François	Tremblay	Canada	99.9899981021881
7	57	Luis	Rojas	Chile	97.0199997425079
8	5	František	Wichterlová	Czech Republic	144.539998054504
9	9	Kara	Nielsen	Denmark	37.6199996471405
10	44	Terhi	Hämäläinen	Finland	79.2000005245209
11	42	Wyatt	Girard	France	99.9899990558624
12	37	Fynn	Zimmermann	Germany	94.0499994754791
13	45	Ladislav	Kovács	Hungary	78.2099997997284
14	58	Manoj	Pareek	India	111.869999647141
15	46	Hugh	O'Reilly	Ireland	114.839999437332
16	47	Lucas	Mancini	Italy	50.4900009632111
17	48	Johannes	Van der Berg	Netherlands	65.3400001525879
18	4	Bjørn	Hansen	Norway	72.2699995040894
19	49	Stanisław	Wójcik	Poland	76.2300002574921
20	34	João	Fernandes	Portugal	102.95999956131
21	50	Enrique	Muñoz	Spain	98.0099990367889
22	51	Joakim	Johansson	Sweden	75.2399995326996
23	53	Phil	Hughes	United Kingdo...	98.0099992752075
24	17	Jack	Smith	USA	98.0099999904633

Thank you!