



# Q1 : Who is the senior most employee based on job titile?

```
select * from employee
ORDER BY levels desc
limit 1;
```

Result Grid   Filter Rows:  | Export:  | Wrap Cell Content:  | Fetch rows: 

	employee_id	last_name	first_name	title	reports_to	levels	birthdate	hire_date	address	city
	1	Adams	Andrew	General Manager	9	L6	18-02-1962 00:00	14-08-2016 00:00	11120 Jasper Ave NW	Edmonton

# Q2 : Which Country have the most invoices?

```
select COUNT(*), billing_country
from invoice
GROUP BY billing_country
ORDER BY billing_country desc;
```

	COUNT(*)	billing_country
▶	131	USA
	28	United Kingdom
	10	Sweden
	11	Spain
	29	Portugal
	10	Poland
	9	Norway
	10	Netherlands
	9	Italy
	13	Ireland
	21	India
	10	Hungary

# Q3 : What are top 3 values of total invoice?

```
select total from invoice
ORDER BY total desc
Limit 3;
```

Result Grid



Filter Rows:

Export:



	total
▶	23.759999999999998
	19.8
	19.8

```
/* 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 invoices totals.Return both the city name & sum of all invoice */
```

```
select SUM(total) as invoice_total, billing_city  
from invoice  
GROUP BY billing_city  
ORDER BY invoice_total desc;
```

Result Grid |   Filter Rows: | Export:  | Wrap


	invoice_total	billing_city
▶	273.240000000000007	Prague
	169.29	Mountain View
	166.32	London
	158.4	Berlin
	151.47	Paris
	129.69	SÃ£o Paulo
	114.839999999999997	Dublin
	111.869999999999999	Delhi
	108.899999999999998	SÃ£o JosÃ© dos Campos
	106.919999999999999	BrasÃ-lia
	102.960000000000001	Lisbon
	99.99	Bordeaux





/\* Q5: Who is the best customer? The customer who has spent the most money will declare 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, 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
LIMIT 1;
```


Result Grid



 Filter Rows:

Export: 

Wrap Cell Content: 

Fetch rows: 

	customer_id	first_name	last_name	total
▶	5	František	Wichterlově	144.54000000000002

.

Q6 : 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 LIKE 'Rock'
)
ORDER BY email;
```

Result Grid





Filter Rows:

Export:



Wrap Cell Content:



	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	Björn	Hansen
	camille.bernard@yahoo.fr	Camille	Bernard
	daan_peeters@apple.be	Daan	Peeters
	diego.gutierrez@yahoo.ar	Diego	Gutiérrez
	dmiller@comcast.com	Dan	Miller
	dominiquedefebvre@gmail.com	Dominique	Lefebvre
	edfrancis@yahoo.ca	Edward	Francis
	eduardo@woodstock.com.br	Eduardo	Martins
	ellie.sullivan@shaw.ca	Ellie	Sullivan

/\* Q7 : 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 AS name, COUNT(track.track_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 = 'Rock'
GROUP BY artist.artist_id, artist.name
ORDER BY number_of_songs DESC
LIMIT 10;
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	artist_id	name	number_of_songs
▶	1	AC/DC	18
	3	Aerosmith	15
	8	Audioslave	14
	22	Led Zeppelin	14
	4	Alanis Morissette	13
	5	Alice In Chains	12
	23	Frank Zappa & Captain Beefheart	9

Q8 : 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_track_length
    FROM track)
ORDER BY milliseconds DESC;
```

Result Grid |   Filter Rows: | Export:  | Wrap Cell Content: 

	name	milliseconds
▶	How Many More Times	711836
	Advance Romance	677694
	Sleeping Village	644571
	You Shook Me(2)	619467
	Talkin' 'Bout Women Obviously	589531
	Stratus	582086
	No More Tears	555075
	The Alchemist	509413
	Wheels Of Confusion / The Straightener	494524
	Book Of Thel	494393
	You Oughta Know (Alternate)	491885
	Terra	482429