
```sql
use chinook;

# 1. SELECT, WHERE, ORDER BY, GROUP BY
# To get the number of invoices per TOP 10 country, sorted from highest to lowest.
# Answer: Shows how many purchases (invoices) came from TOP 10 country.

select Billingcountry,
    count(*) as Total_Invoices
from invoice
group by BillingCountry
order by Total_invoices desc
LIMIT 10;
```

Billingcountry	Total_Invoices
USA	91
Canada	56
France	35
Brazil	35
Germany	28
United Kingdom	21
Czech Republic	14
Portugal	14
India	13
Norway	7


```
use chinook;

# 2. INNER JOIN, LEFT JOIN, RIGHT JOIN
# Question: Get the first name of customers and their invoice totals. Use INNER JOIN.
# Answer: Only customers who made at least one invoice are shown.

select c.firstname,
       i.Total as invoice_Total
from invoice i
left join customer c
    using(CustomerId)

```

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

firstname	invoice_Total
Leonie	1.98
Bjørn	3.96
Daan	5.94
Mark	8.91
John	13.86
Fynn	0.99
Niklas	1.98
Dominique	1.98
Wyatt	3.96
Hugh	5.94
Emma	8.91
Leonie	13.86
Frank	0.99
Jack	1.98
Tim	1.98
Kathy	3.96
Victor	5.94
Martha	8.91


```
1  use chinook;
2
3  # 3. SUBQUERIES
4  # Find customers who have spent more than $30 in total.
5  # Returns only those customers who have spent more than $30 total.
6
7  select firstname,
8         lastname
9  from customer c
10 where CustomerId in (
11     select CustomerId
12     from invoice
13     group by CustomerId
14     having sum(Total) > 30)
```

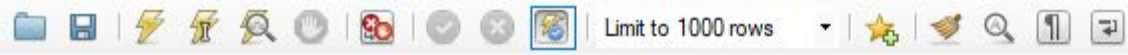
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

firstname	lastname
Luís	Gonçalves
Leonie	Köhler
François	Tremblay
Bjørn	Hansen
František	Wichterlová
Helena	Holý
Astrid	Gruber
Daan	Peeters
Kara	Nielsen
Eduardo	Martins
Alexandre	Rocha
Roberto	Almeida
Fernanda	Ramos
Mark	Philips
Jennifer	Peterson
Frank	Harris
Jack	Smith
Michelle	Brooks

```sql
use chinook;

--  4. Aggregate Functions - SUM, AVG, COUNT.
# What is the average and total invoice value based on each country?
# Gives overall business revenue and average per country.

select BillingCountry,
        avg(total) as Average_Invoice,
        sum(total) as Total_Revenue

from invoice
group by BillingCountry
```

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

BillingCountry	Average_Invoice	Total_Revenue
Norway	5.660000	39.62
Belgium	5.374286	37.62
Canada	5.427857	303.96
USA	5.747912	523.06
France	5.574286	195.10
Ireland	6.517143	45.62
United Kingdom	5.374286	112.86
Australia	5.374286	37.62
Chile	6.660000	46.62
India	5.789231	75.26
Brazil	5.431429	190.10
Portugal	5.517143	77.24
Netherlands	5.802857	40.62
Spain	5.374286	37.62
Sweden	5.517143	38.62
Czech Republic	6.445714	90.24
Finland	5.945714	41.62
Denmark	5.374286	37.62

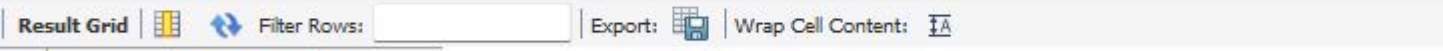



```
1  use chinook;
2
3  # 5. Create a view to show all Canadian customers.
4  # After creating view we can run (SELECT * FROM Canadian_Customers); like a real table.
5
6  create view Canadian_Customers as
7  select * from customer
8  where Country = 'canada';
9
10 create view Ind_Customers as
11 select * from customer
12 where Country = 'India';
13
14 select * from Canadian_Customers;
15 select * from ind_Customers;
16
```

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

CustomerId	FirstName	LastName	Company	Address	City	State	Country	PostalCode	Phone
3	François	Tremblay	NULL	1498 rue Bélanger	Montréal	QC	Canada	H2G 1A7	+1 (514)
14	Mark	Philips	Telus	8210 111 ST NW	Edmonton	AB	Canada	T6G 2C7	+1 (780)
15	Jennifer	Peterson	Rogers Canada	700 W Pender Street	Vancouver	BC	Canada	V6C 1G8	+1 (604)
29	Robert	Brown	NULL	796 Dundas Street West	Toronto	ON	Canada	M6J 1V1	+1 (416)
30	Edward	Francis	NULL	230 Elgin Street	Ottawa	ON	Canada	K2P 1L7	+1 (613)
31	Martha	Silk	NULL	194A Chain Lake Drive	Halifax	NS	Canada	B3S 1C5	+1 (902)
32	Aaron	Mitchell	NULL	696 Osborne Street	Winnipeg	MB	Canada	R3L 2B9	+1 (204)
33	Ellie	Sullivan	NULL	5112 48 Street	Yellowknife	NT	Canada	X1A 1N6	+1 (867)


```
1  use chinook;
2  # Handling NULL Values
3  # Replace null values in the State field with 'Not Provided'.
4
5  -- For diplaying NULL values
6  select firstname,lastname,state
7  from customer
8  where State is null;
9
10 -- To delete NULL values
11 delete from customers
12 where State IS NULL;
13
14 # How to Replace NULL Values with Custom Text
15 # Replace NULL values in the State column with "Unknown".
16 select firstname, lastname, coalesce(State, 'Unknown') as State
17 from customer;
18 # COALESCE() works in MySQL, PostgreSQL, and SQLite.)
19
```



firstname	lastname	State
Luís	Gonçalves	SP
Leonie	Köhler	Unknown
François	Tremblay	QC
Bjørn	Hansen	Unknown
František	Wichterlová	Unknown
Helena	Holý	Unknown
Astrid	Gruber	Unknown
Daan	Peeters	Unknown
Kara	Nielsen	Unknown
Eduardo	Martins	SP
Alexandre	Rocha	SP
Roberto	Almeida	RJ
Fernanda	Ramos	DF


```
1  use chinook;
2
3  #  Query Optimization with Indexes
4  # Add an index on the CustomerId column of the invoices table to optimize lookup.
5  # Speeds up queries using WHERE CustomerId = , or joins with customers.
6
7
8  SELECT * FROM invoice
9  WHERE CustomerId = 10;
10
11 create index id_cust on invoice(customerid);
12
13 SELECT * FROM invoice
14 WHERE CustomerId = 10;
15
16 SHOW INDEX FROM invoice;
17 drop index id_cust on invoice;
18
```

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

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null
invoice	0	PRIMARY	1	InvoiceId	A	412	NULL	NULL	
invoice	1	IFK_InvoiceCustomerId	1	CustomerId	A	59	NULL	NULL	
invoice	1	id_cust	1	CustomerId	A	59	NULL	NULL	