

SQL for Data Analyst 102

Function

- AS = ตั้งชื่อ Table ใหม่
 - SQL สามารถใช้เคาะ Space bar แทนเขียน AS ได้
- AND = ทั้งสองเงื่อนไขต้องตรงกันถึงจะถึง
- OR = หาก 1 เงื่อนไขตรง จะถึงข้อมูล
- NOT = ไม่ถึงข้อมูล
- IN = ใ้ย่อ Code กรณีต้องถึงข้อมูล Column เดียวกันหลายๆตัว
- BETWEEN = ถึงข้อมูลระหว่าง x ถึง x
- IS NOT = ใช้กรณีไม่ต้องการค่า NULL โดยเขียนเป็น IS NOT NULL
- % = Match character ยกตัวอย่าง '@gmail.com' ถึงข้อมูลด้านหน้าที่เป็นอะไรก็ได้ แต่ต้องลงท้ายด้วย @gmail.com
- LIKE = SQL จะใช้ LIKE เป็นการ Match สามารถใช้ _ ในการ Match กรณีที่เราไม่ทราบตัวนั้นเช่น Leonie เป็น Leon__
- COALESCE = ใช้ในการ Clean ค่า Null
- GROUP BY = จับกลุ่ม
- ORDER BY = เรียงข้อมูล
- NTILE = แบ่งกลุ่ม
 - NTLIE ใช้คู่กับ ORDER BY เสมอ
- WHERE เกิดขึ้นก่อน GROUP BY , HAVING และ ORDER ตามลำดับ
 - WHERE ใช้ Filter Table
 - HAVING ใช้ Filter GROUP (Column)
- SUBSTR = สามารถใช้คู่กับการสร้าง Column ใหม่
 - ตัวอย่าง ต้องการสร้าง e-mail จาก Lastname โดยเริ่มจาก character ที่ 1 ออกมา 1 ตัว
- STRFTIME = ใช้ในการถึงข้อมูล Date
- TYPEOF = ใ้เช็ค Type
- CAST = ใช้ในการเปลี่ยน Data Type

- CASE, WHEN = ใช้เขียนเงื่อนไขเหมือน IF เป็นการสร้าง Column ใหม่
- LIMIT = แถวบน
- OFFSET = 5 แถวล่าง (*LIMIT 5 OFFSET 5)

Use WHERE to filter data

เราใช้ **WHERE** clause เวลาที่เราต้องการจะ filter เฉพาะข้อมูลแถวที่เราต้องการจาก database tables

99.99% ของ query ที่เราเขียนในชีวิตจริงส่วนใหญ่ต้องเขียน **WHERE** เสมอเลย ยิ่งเราเขียน query ได้ specific มากเท่าไร ยิ่งช่วยให้ analysis เราตอบโจทย์ได้แม่นยำขึ้นเท่านั้น

```
SELECT * FROM customers WHERE country = 'USA';
```

Common WHERE usage

ตัวอย่างด้านล่างคือตัวการเขียน **WHERE** ที่เราใช้กันบ่อยๆ เวลาเขียน **SQL** query

🌱 **LIKE** คือการเขียน pattern matching ปกติเราจะใช้ **LIKE** กับ wildcard คือ **%** หรือ **_**

- **%** ใช้ match any character ที่ตัวก็ได้
- **_** ใช้ match single character ตัวเดียว

ตัวอย่างเช่น **WHERE country LIKE 'U%'** จะฟิเตอร์เฉพาะประเทศที่ขึ้นต้นด้วยตัว U ทั้งหมด

หรือ **WHERE firstname LIKE 'J_hn'** จะฟิเตอร์ชื่อ firstname ลูกคำขึ้นต้นด้วยตัว J ตามด้วยตัวอักษรอะไรก็ได้หนึ่งตัวและปิดท้ายด้วย hn เช่น **John Jahn Jihn Jehn** เป็นต้น

🌱 ใน SQLite ตัว **LIKE** operator จะเป็นแบบ case insensitive ไม่สนตัวพิมพ์เล็กใหญ่ เวลาเขียน **LIKE 'J_hn'** สามารถ match ได้ทั้ง **JOHN john** หรือ **JohN** ไม่แตกต่างกัน

TEXT

```
SELECT * FROM customers
WHERE country = 'USA';
```

```
SELECT * FROM customers
WHERE country = 'USA' AND state = 'CA';
```

```
SELECT * FROM customers
WHERE country = 'USA' OR country = 'United Kingdom';
```

```
SELECT * FROM customers
WHERE country IN ('USA', 'United Kingdom');
```

```
SELECT * FROM customers
WHERE country NOT IN ('USA', 'United Kingdom');
```

```
SELECT * FROM customers
WHERE email LIKE '%@gmail.com';
```

```
SELECT * FROM customers
WHERE email NOT LIKE '%gmail.com';
```

```
SELECT * FROM customers
WHERE company IS NULL;
```

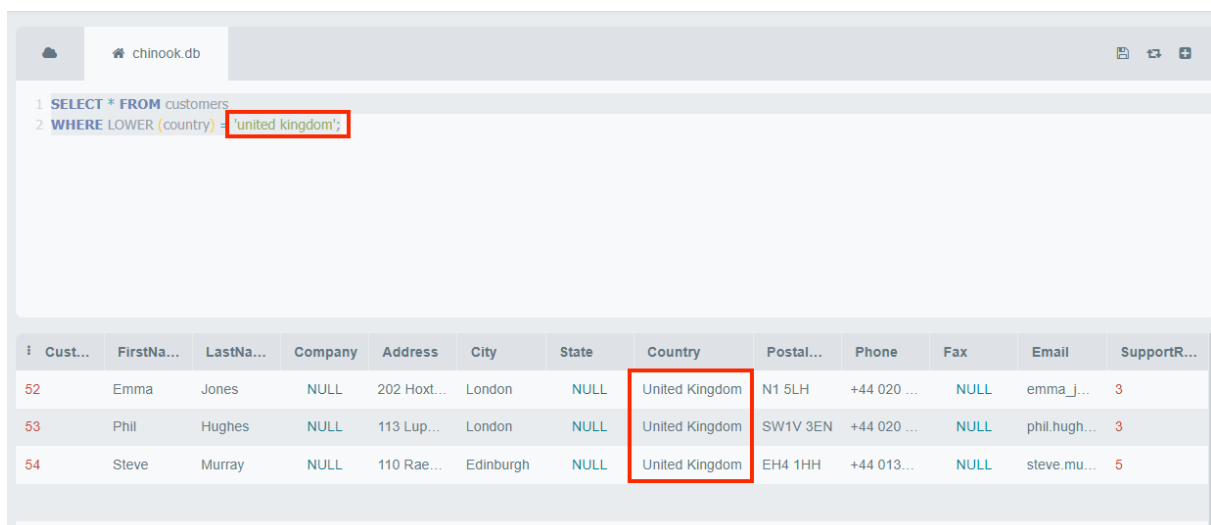
```
SELECT * FROM customers
WHERE company IS NOT NULL;
```

```
SELECT * FROM customers
WHERE customerid BETWEEN 10 AND 15;
```

Filter Data Part 1

โดยปกติการดึงข้อมูลจาก Column ตัวพิมพ์เล็ก-ใหญ่ต้องตรง

สามารถใช้ Function LOWER เพื่อให้สามารถดึงข้อมูลได้ ไม่ว่าข้อมูลจะเป็นพิมพ์เล็กหรือใหญ่



The screenshot shows a database interface with a query editor and a results table. The query is:

```
1 SELECT * FROM customers
2 WHERE LOWER(country) = 'united kingdom';
```

The results table has the following columns: Cust..., FirstNa..., LastNa..., Company, Address, City, State, Country, Postal..., Phone, Fax, Email, and SupportR... The table contains three rows of data where the Country is 'United Kingdom':

Cust...	FirstNa...	LastNa...	Company	Address	City	State	Country	Postal...	Phone	Fax	Email	SupportR...
52	Emma	Jones	NULL	202 Hox...	London	NULL	United Kingdom	N1 5LH	+44 020 ...	NULL	emma_j...	3
53	Phil	Hughes	NULL	113 Lup...	London	NULL	United Kingdom	SW1V 3EN	+44 020 ...	NULL	phil.hugh...	3
54	Steve	Murray	NULL	110 Rae...	Edinburgh	NULL	United Kingdom	EH4 1HH	+44 013...	NULL	steve.mu...	5

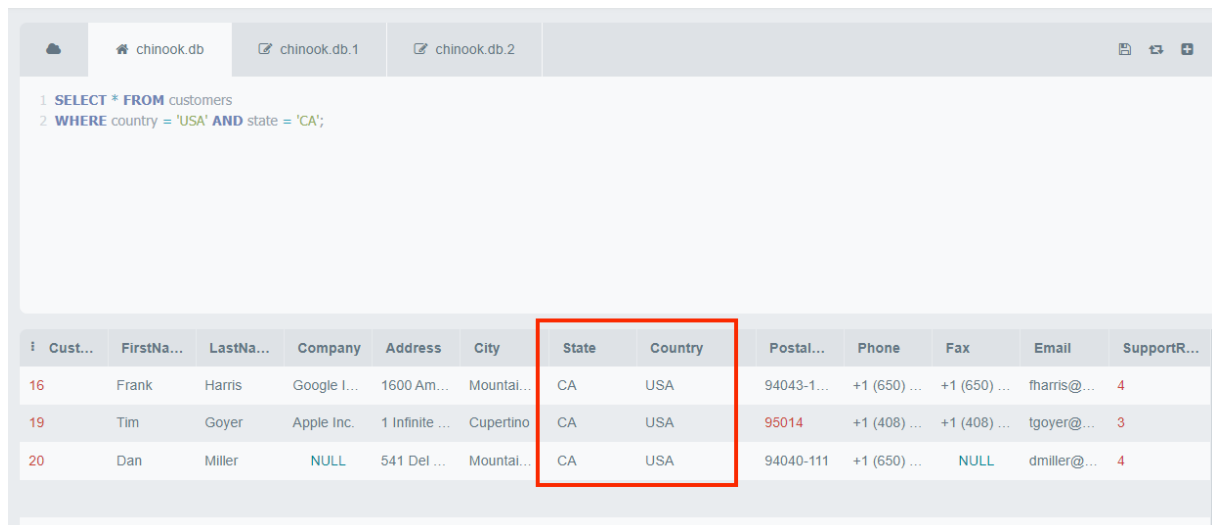
```
SELECT * FROM customers
WHERE LOWER (country) = 'united kingdom';
```

Function AND , OR , NOT

- AND = ทั้งสองเงื่อนไขต้องตรงกันถึงจะถึง
- OR = หาก 1 เงื่อนไขตรง จะถึงข้อมูล
- NOT = ไม่ถึงข้อมูล
- IN = ใ้ย่อ Code กรณีต้องถึงข้อมูล Column เดียวกันหลายๆตัว
- BETWEEN = ถึงข้อมูลระหว่าง x ถึง x
- IS NOT = ใช้กรณีไม่ต้องการค่า NULL โดยเขียนเป็น IS NOT NULL

ถึงข้อมูล Table Customers โดยเงื่อนไขถึง Country และ state ต้องเป็น CA

```
SELECT * FROM customers
WHERE country = 'USA' AND state = 'CA';
```



	Cust...	FirstNa...	LastNa...	Company	Address	City	State	Country	Postal...	Phone	Fax	Email	SupportR...
16		Frank	Harris	Google I...	1600 Am...	Mountai...	CA	USA	94043-1...	+1 (650) ...	+1 (650) ...	fharris@...	4
19		Tim	Goyer	Apple Inc.	1 Infinite ...	Cupertino	CA	USA	95014	+1 (408) ...	+1 (408) ...	tgoyer@...	3
20		Dan	Miller	NULL	541 Del ...	Mountai...	CA	USA	94040-111	+1 (650) ...	NULL	dmiller@...	4

ถึงข้อมูล USA หรือ Canada

```
SELECT * FROM customers
WHERE country = 'USA' OR country = 'Canada';
```

chinook.db

chinook.db.1

chinook.db.2

1 SELECT * FROM customers

2 WHERE country = 'USA' OR country = 'Canada';

i	Cust...	FirstNa...	LastNa...	Company	Address	City	State	Country	Postal...	Phone	Fax	Email	SupportR...
3		François	Tremblay	NULL	1498 rue...	Montréal	QC	Canada	H2G 1A7	+1 (514) ...	NULL	ftrembla...	3
14		Mark	Phillips	Telus	8210 111...	Edmonton	AB	Canada	T6G 2C7	+1 (780) ...	+1 (780) ...	mphillips...	5
15		Jennifer	Peterson	Rogers ...	700 W P...	Vancouver	BC	Canada	V6C 1G8	+1 (604) ...	+1 (604) ...	jenniferp...	3
16		Frank	Harris	Google I...	1600 Am...	Mountai...	CA	USA	94043-1...	+1 (650) ...	+1 (650) ...	fharris@...	4
17		Jack	Smith	Microsoft...	1 Micros...	Redmond	WA	USA	98052-8...	+1 (425) ...	+1 (425) ...	jacksmi...	5
18		Michelle	Brooks	NULL	627 Broa...	New York	NY	USA	10012-2...	+1 (212) ...	+1 (212) ...	michelle...	3
19		Tim	Goyer	Apple Inc.	1 Infinite ...	Cupertino	CA	USA	95014	+1 (408) ...	+1 (408) ...	tgoyer@...	3

ไม่ถึงข้อมูลประเทศ USA , Canada และ France

chinook.db

chinook.db.1

chinook.db.2

chinook.db.3

1 SELECT * FROM customers

2 WHERE NOT (country = 'USA' OR country = 'Canada' OR country = 'France');

i	Cust...	FirstNa...	LastNa...	Company	Address	City	State	Country	Postal...	Phone	Fax	Email	SupportR...
1		Luis	Gonçalves	Embraer...	Av. Briga...	São Jos...	SP	Brazil	12227-000	+55 (12) ...	+55 (12) ...	luisg@e...	3
2		Leonie	Köhler	NULL	Theodor...	Stuttgart	NULL	Germany	70174	+49 071...	NULL	leonekoh...	5
4		Bjørn	Hansen	NULL	Ullevålsv...	Oslo	NULL	Norway	0171	+47 22 4...	NULL	bjorn.ha...	4
5		František	Wichterl...	JetBrain...	Klanova ...	Prague	NULL	Czech Republic	14700	+420 2 4...	+420 2 4...	frantisek...	4
6		Helena	Holý	NULL	Rilská 3...	Prague	NULL	Czech Republic	14300	+420 2 4...	NULL	hholy@g...	5
7		Astrid	Gruber	NULL	Rotentur...	Vienne	NULL	Austria	1010	+43 01 5...	NULL	astrid.gr...	5
8		Daan	Peeters	NULL	Grétrystr...	Brussels	NULL	Belgium	1000	+32 02 2...	NULL	daan_pe...	4
9		Kara	Nielsen	NULL	Sønder ...	Copenha...	NULL	Denmark	1720	+453 33...	NULL	kara.niel...	4
10		Eduardo	Martins	Woodsto...	Rua Dr. ...	São Paulo	SP	Brazil	01007-010	+55 (11) ...	+55 (11) ...	eduardo...	4
11		Alexandre	Rocha	Banco d...	Av. Pauli...	São Paulo	SP	Brazil	01310-200	+55 (11) ...	+55 (11) ...	alero@u...	5
12		Roberto	Almeida	Riotur	Praça Pl...	Rio de J...	RJ	Brazil	20040-020	+55 (21) ...	+55 (21) ...	roberto.a...	3
13		Fernanda	Ramos	NULL	Qe 7 Blo...	Brasília	DF	Brazil	71020-677	+55 (61) ...	+55 (61) ...	fernadar...	4

```

SELECT * FROM customers
WHERE NOT (country = 'USA' OR country = 'Canada' OR country = 'France');

```

ดึงหลายๆประเทศโดยใช้ IN

chinook.db

chinook.db.3

chinook.db.4

chinook.db.5

chinook.db.6

chinook.db.8

1 SELECT * FROM customers

2 WHERE UPPER(country) IN ('UNITED KINGDOM', 'USA', 'CANADA');

i	Cust...	FirstNa...	LastNa...	Company	Address	City	State	Country	Postal...	Phone	Fax	Email	SupportR...
3		François	Tremblay	NULL	1498 rue...	Montréal	QC	Canada	H2G 1A7	+1 (514) ...	NULL	ftrembla...	3
14		Mark	Phillips	Telus	8210 111...	Edmonton	AB	Canada	T6G 2C7	+1 (780) ...	+1 (780) ...	mphilips...	5
15		Jennifer	Peterson	Rogers ...	700 W P...	Vancouver	BC	Canada	V6C 1G8	+1 (604) ...	+1 (604) ...	jenniferp...	3
16		Frank	Harris	Google I...	1600 Am...	Mountai...	CA	USA	94043-1...	+1 (650) ...	+1 (650) ...	fharris@...	4
17		Jack	Smith	Microsoft...	1 Micros...	Redmond	WA	USA	98052-8...	+1 (425) ...	+1 (425) ...	jacksmi...	5

```
SELECT * FROM customers
WHERE UPPER(country) IN ('UNITED KINGDOM' , 'USA', 'CANADA');
```

Filter Data Part 2

การย่อ Code กรณีต้องดึงข้อมูล Column เดียวกันหลายๆตัว

ใช้ Function IN เข้าช่วย

chinook.db

```

1 SELECT * FROM customers
2 WHERE country IN ('Brazil' , 'Germany' , 'Norway')

```

i	Cust...	FirstNa...	LastNa...	Company	Address	City	State	Country	Postal...	Phone	Fax	Email	SupportR...
1		Luís	Gonçalves	Embraer...	Av. Briga...	São Jos...	SP	Brazil	12227-000	+55 (12) ...	+55 (12) ...	luisg@e...	3
2		Leonie	Köhler	NULL	Theodor-...	Stuttgart	NULL	Germany	70174	+49 071...	NULL	leonekoh...	5
4		Bjørn	Hansen	NULL	Ullevålsv...	Oslo	NULL	Norway	0171	+47 22 4...	NULL	bjorn.ha...	4
10		Eduardo	Martins	Woodsto...	Rua Dr. ...	São Paulo	SP	Brazil	01007-010	+55 (11) ...	+55 (11) ...	eduardo...	4
11		Alexandre	Rocha	Banco d...	Av. Pauli...	São Paulo	SP	Brazil	01310-200	+55 (11) ...	+55 (11) ...	alero@u...	5
12		Roberto	Almeida	Riotur	Praça Pl...	Rio de J...	RJ	Brazil	20040-020	+55 (21) ...	+55 (21) ...	roberto.a...	3
13		Fernanda	Ramos	NULL	Qe 7 Blo...	Brasília	DF	Brazil	71020-677	+55 (61) ...	+55 (61) ...	fernadar...	4
36		Hannah	Schneider	NULL	Tauentzi...	Berlin	NULL	Germany	10789	+49 030 ...	NULL	hannah....	5
37		Fynn	Zimmer...	NULL	Berger S...	Frankfurt	NULL	Germany	60316	+49 069 ...	NULL	fzimmer...	3
38		Niklas	Schröder	NULL	Barbaros...	Berlin	NULL	Germany	10779	+49 030 ...	NULL	nschrod...	3

```

SELECT * FROM customers
WHERE country IN ('Brazil' , 'Germany' , 'Norway');

```

หรือใช้กับ NOT

```

SELECT * FROM customers
WHERE country NOT IN ('Brazil' , 'Germany' , 'Norway');

```

chinook.db

```
1 SELECT * FROM customers
2 WHERE country NOT IN ('Brazil', 'Germany', 'Norway')
```

i	Cust...	FirstNa...	LastNa...	Company	Address	City	State	Country	Postal...	Phone	Fax	Email	SupportR...
3		François	Tremblay	NULL	1498 rue...	Montréal	QC	Canada	H2G 1A7	+1 (514) ...	NULL	ftrembla...	3
5		František	Wichterl...	JetBrain...	Klanova ...	Prague	NULL	Czech Republic	14700	+420 2 4...	+420 2 4...	frantisek...	4
6		Helena	Holý	NULL	Rilská 3...	Prague	NULL	Czech Republic	14300	+420 2 4...	NULL	hholy@g...	5
7		Astrid	Gruber	NULL	Rotentur...	Vienne	NULL	Austria	1010	+43 01 5...	NULL	astrid.gr...	5
8		Daan	Peeters	NULL	Grétrystr...	Brussels	NULL	Belgium	1000	+32 02 2...	NULL	daan_pe...	4
9		Kara	Nielsen	NULL	Sender ...	Copenha...	NULL	Denmark	1720	+453 33...	NULL	kara.niel...	4
14		Mark	Philips	Telus	8210 111...	Edmonton	AB	Canada	T6G 2C7	+1 (780) ...	+1 (780) ...	mphilips...	5
15		Jennifer	Peterson	Rogers ...	700 W P...	Vancouver	BC	Canada	V6C 1G8	+1 (604) ...	+1 (604) ...	jenniferp...	3
16		Frank	Harris	Google I...	1600 Am...	Mountai...	CA	USA	94043-1...	+1 (650) ...	+1 (650) ...	fharris@...	4

Function BETWEEN

ดึงข้อมูล Customer ID 5-10

```

SELECT * FROM customers
WHERE customerid BETWEEN 5 AND 10;

```

chinook.db

1 SELECT * FROM customers

2 WHERE customerid BETWEEN 5 AND 10;

i	Cust...	FirstNa...	LastNa...	Company	Address	City	State	Country	Postal...	Phone	Fax	Email	SupportR...
5		František	Wichterl...	JetBrain...	Klanova ...	Prague	NULL	Czech Republic	14700	+420 2 4...	+420 2 4...	frantisek...	4
6		Helena	Holý	NULL	Rilská 3...	Prague	NULL	Czech Republic	14300	+420 2 4...	NULL	hholy@g...	5
7		Astrid	Gruber	NULL	Rotentur...	Vienne	NULL	Austria	1010	+43 01 5...	NULL	astrid.gr...	5
8		Daan	Peeters	NULL	Grétrystr...	Brussels	NULL	Belgium	1000	+32 02 2...	NULL	daan_pe...	4
9		Kara	Nielsen	NULL	Sønder ...	Copenha...	NULL	Denmark	1720	+453 33...	NULL	kara.niel...	4
10		Eduardo	Martins	Woodsto...	Rua Dr. ...	São Paulo	SP	Brazil	01007-010	+55 (11) ...	+55 (11) ...	eduardo...	4

ดึงข้อมูล Table Customers โดย Company ไม่เอาจค่า NULL

chinook.db

chinook.db.4

chinook.db.5

1 SELECT * FROM customers

2 WHERE company IS NOT NULL;

i	Cust...	FirstNa...	LastNa...	Company	Address	City	State	Country	Postal...	Phone	Fax	Email	SupportR...
1		Luís	Gonçalves	Embraer...	Av. Briga...	São Jos...	SP	Brazil	12227-000	+55 (12) ...	+55 (12) ...	luisg@e...	3
5		František	Wichterl...	JetBrain...	Klanova ...	Prague	NULL	Czech Republic	14700	+420 2 4...	+420 2 4...	frantisek...	4
10		Eduardo	Martins	Woodsto...	Rua Dr. ...	São Paulo	SP	Brazil	01007-010	+55 (11) ...	+55 (11) ...	eduardo...	4
11		Alexandre	Rocha	Banco d...	Av. Pauli...	São Paulo	SP	Brazil	01310-200	+55 (11) ...	+55 (11) ...	alero@u...	5
12		Roberto	Almeida	Riotur	Praça Pi...	Rio de J...	RJ	Brazil	20040-020	+55 (21) ...	+55 (21) ...	roberto.a...	3
14		Mark	Phillips	Telus	8210 111...	Edmonton	AB	Canada	T6G 2C7	+1 (780) ...	+1 (780) ...	mphilips...	5
15		Jennifer	Peterson	Rogers ...	700 W P...	Vancouver	BC	Canada	V6C 1G8	+1 (604) ...	+1 (604) ...	jenniferp...	3
16		Frank	Harris	Google I...	1600 Am...	Mountai...	CA	USA	94043-1...	+1 (650) ...	+1 (650) ...	fharris@...	4
17		Jack	Smith	Microsoft...	1 Micros...	Redmond	WA	USA	98052-8...	+1 (425) ...	+1 (425) ...	jacksmi...	5
19		Tim	Goyer	Apple Inc.	1 Infinite ...	Cupertino	CA	USA	95014	+1 (408) ...	+1 (408) ...	tgoyer@...	3

```
SELECT * FROM customers
WHERE company IS NOT NULL;
```

Filter Data Part 3

การใช้ Pattern matching

ดึงข้อมูลที่ใช้ @gmail.com

chinook.db			
<pre>1 SELECT firstname, lastname, country, email FROM customers 2 WHERE email LIKE '%@gmail.com';</pre>			
FirstName	LastName	Country	Email
François	Tremblay	Canada	ftremblay@gmail.com
Helena	Holý	Czech Republic	hholy@gmail.com
Heather	Leacock	USA	hleacock@gmail.com
Frank	Ralston	USA	fralston@gmail.com
Julia	Barnett	USA	jubarnett@gmail.com
Martha	Silk	Canada	marthasilk@gmail.com
Dominique	Lefebvre	France	dominiquelefebvre@gmail.com
Phil	Hughes	United Kingdom	phil.hughes@gmail.com

```
SELECT firstname, lastname, country, email FROM customers
WHERE email LIKE '%@gmail.com';
```



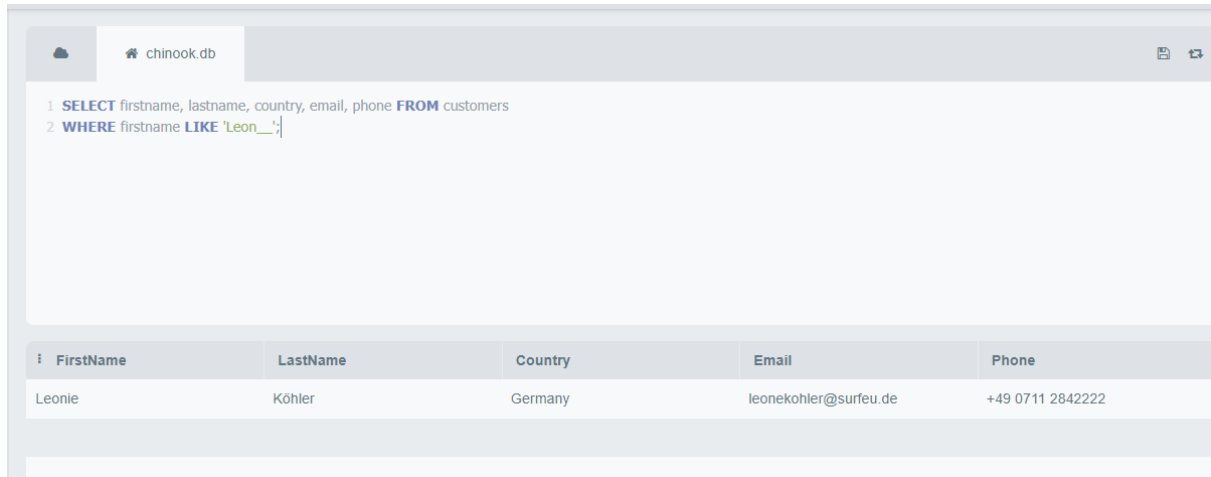
หากไม่ต้องการดึงข้อมูลที่ @gmail ใช้ NOT LIKE

ดึงข้อมูลที่มีเลข 99 โดยเลขด้านหน้าและหลังจะเป็นอะไรก็ได้

chinook.db				
<pre>1 SELECT firstname, lastname, country, email, phone FROM customers 2 WHERE phone LIKE '%99%'</pre>				
FirstName	LastName	Country	Email	Phone
Kara	Nielsen	Denmark	kara.nielsen@jubii.dk	+453 3331 9991
Tim	Goyer	USA	tgoyer@apple.com	+1 (408) 996-1010
Heather	Leacock	USA	hleacock@gmail.com	+1 (407) 999-7788
Isabelle	Mercier	France	isabelle_mercier@apple.fr	+33 03 80 73 66 99
Puja	Srivastava	India	puja_srivastava@yahoo.in	+91 080 22289999

```
SELECT firstname, lastname, country, email, phone FROM customers
WHERE phone LIKE '%99%';
```

ดึงข้อมูลชื่อโดยใช้ _ แทน Character ที่ไม่ทราบ
เช่น Leonie



The screenshot shows a database interface with a query editor and a results table. The query editor contains the following SQL query:

```
1 SELECT firstname, lastname, country, email, phone FROM customers
2 WHERE firstname LIKE 'Leon_';
```

The results table displays the following data:

FirstName	LastName	Country	Email	Phone
Leonie	Köhler	Germany	leonekohler@surfeu.de	+49 0711 2842222

```
SELECT firstname, lastname, country, email, phone FROM customers
WHERE firstname LIKE 'Leon_';
```

COALESCE

การจัดการค่า Null

ดึงข้อมูล Table Customers Column company

ใช้ COALESCE หาค่า null ใน Column Company และเปลี่ยนชื่อเป็น End Customer ให้ขึ้น Column ใหม่ว่า Company Clean

ใช้ CASE WHEN ในการเขียนเงื่อนไข หาก Company เป็นค่า null ให้มัน(THEN) โชว์ End Customer

ELSE (หากเป็นเงื่อนไขอื่นๆ) ให้โชว์ Corporate

และสร้าง Column ใหม่เป็น Segment

chinook.db		
<pre> 1 SELECT 2 company, 3 COALESCE(company, 'End Customer') AS 'Company Clean', 4 CASE WHEN company IS NULL THEN 'End Customer' 5 ELSE 'Corporate' 6 END AS 'Segment' 7 8 FROM customers; </pre>		
Company	Company Clean	Segment
Embraer - Empresa Brasileira de Aeronáutica S.A.	Embraer - Empresa Brasileira de Aeronáutica S.A.	Corporate
NULL	End Customer	End Customer
NULL	End Customer	End Customer
NULL	End Customer	End Customer
JetBrains s.r.o.	JetBrains s.r.o.	Corporate
NULL	End Customer	End Customer
NULL	End Customer	End Customer
NULL	End Customer	End Customer

```


SELECT
    company,
    COALESCE(company, 'End Customer') AS 'Company Clean',
    CASE WHEN company IS NULL THEN 'End Customer'
         ELSE 'Corporate'
    END AS 'Segment'


FROM customers;


```

JOIN Data using WHERE

ดึงข้อมูลจากหลาย Table โดยใช้ WHERE



 chinook.db

 chinook.db.8

```
1 SELECT * FROM artists,albums
2 WHERE artists.artistid = albums.artistid;
3 |
```


i	ArtistId	Name	AlbumId	Title	ArtistId
1		AC/DC	1	For Those About To Rock We S...	1
2		Accept	2	Balls to the Wall	2
2		Accept	3	Restless and Wild	2
1		AC/DC	4	Let There Be Rock	1
3		Aerosmith	5	Big Ones	3

```


SELECT * FROM artists,albums
WHERE artists.artistid = albums.artistid;

```

ดึงข้อมูลจากหลาย Table โดยใช้ INNER JOIN



chinook.db

 chinook.db.8

```
1 SELECT * FROM artists JOIN albums
2 ON artists.artistid = albums.artistid;
```

i	ArtistId	Name	AlbumId	Title	ArtistId
1		AC/DC	1	For Those About To Rock We S...	1
2		Accept	2	Balls to the Wall	2
2		Accept	3	Restless and Wild	2
1		AC/DC	4	Let There Be Rock	1
3		Aerosmith	5	Big Ones	3

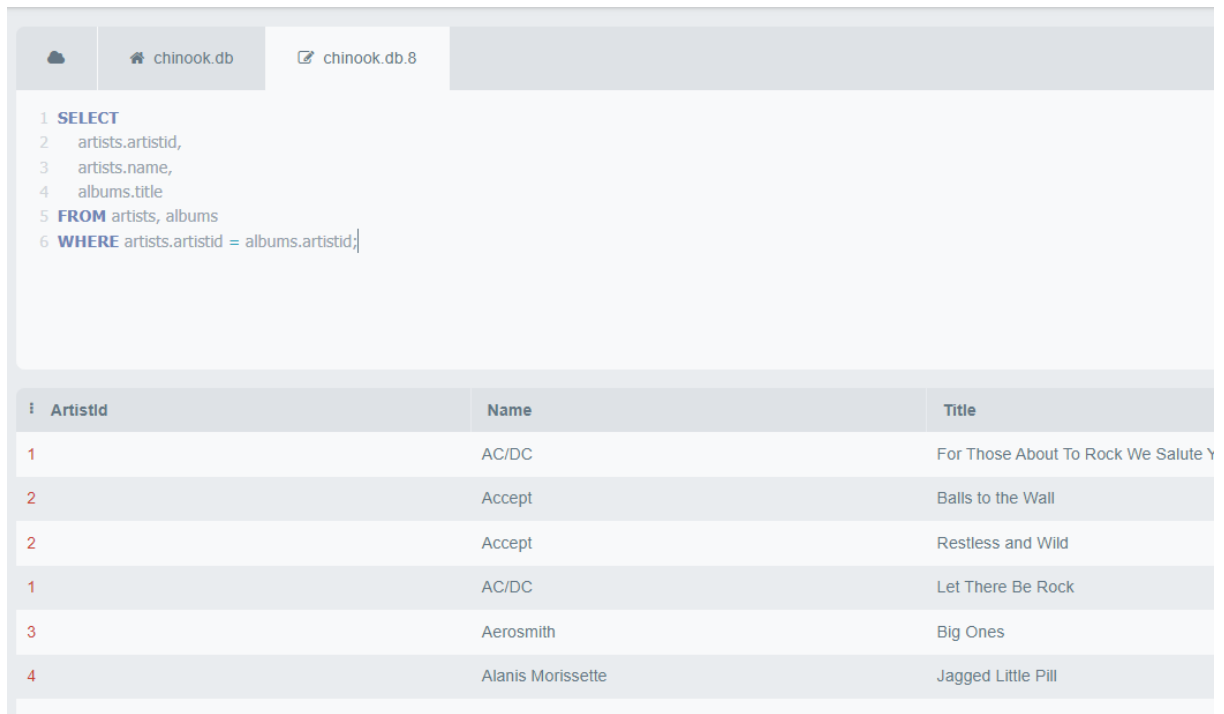
```

SELECT * FROM artists JOIN albums

```

```
ON artists.artistid = albums.artistid;
```

จะเห็นว่าทั้ง WHERE และ INNER JOIN ได้ผลลัพธ์เหมือนกันจะเขียนแบบไหนก็ได้
หากต้องการดึงแค่ Column Name และ Title



The screenshot shows a database interface with a query editor and a results table. The query editor contains the following SQL code:

```
1 SELECT
2   artists.artistid,
3   artists.name,
4   albums.title
5 FROM artists, albums
6 WHERE artists.artistid = albums.artistid;
```

The results table has three columns: Artistid, Name, and Title. It displays the following data:

Artistid	Name	Title
1	AC/DC	For Those About To Rock We Salute Y
2	Accept	Balls to the Wall
2	Accept	Restless and Wild
1	AC/DC	Let There Be Rock
3	Aerosmith	Big Ones
4	Alanis Morissette	Jagged Little Pill

```
SELECT
    artists.artistid,
    artists.name,
    albums.title
FROM artists, albums
WHERE artists.artistid = albums.artistid;
```

เปลี่ยนชื่อโดยใช้ AS

cloud	chinook.db	chinook.db.8	
-------	------------	--------------	--

```

1 SELECT
2   artists.artistid,
3   artists.name AS artist_name,
4   albums.title AS album_name
5 FROM artists, albums
6 WHERE artists.artistid = albums.artistid;

```

i	Artistid	artist_name	album_name
1		AC/DC	For Those About To Rock We Salute You
2		Accept	Balls to the Wall
2		Accept	Restless and Wild
1		AC/DC	Let There Be Rock
3		Aerosmith	Big Ones
4		Alanis Morissette	Jagged Little Pill
5		Alice In Chains	Facelift

```

SELECT
    artists.artistid,
    artists.name AS artist_name,
    albums.title AS album_name
FROM artists, albums
WHERE artists.artistid = albums.artistid;

```

เพิ่มเงื่อนไขอยากรู้ว่า Artist คนที่ 100 ออกมาแล้วกี่ Album
ตามรูปจะทราบได้ว่าออกมาแค่ 1 Album

cloud	chinook.db	chinook.db.8	
-------	------------	--------------	--

```

1 SELECT
2   artists.artistid,
3   artists.name AS artist_name,
4   albums.title AS album_name
5 FROM artists, albums
6 WHERE artists.artistid = albums.artistid
7   AND artists.artistid = 100;

```

i	Artistid	artist_name	album_name
100		Lenny Kravitz	Greatest Hits

```

SELECT
    artists.artistid,
    artists.name AS artist_name,
    albums.title AS album_name
FROM artists, albums
WHERE artists.artistid = albums.artistid
    AND artists.artistid = 100;

```

ใช้ Function IN ในการดึงข้อมูลหลายๆคน
 จะเห็นได้ว่าคนที่ 8 ออกมาแล้ว 3 Album

cloud	chinook.db	chinook.db.8
-------	------------	--------------

```

1 SELECT
2   artists.artistid,
3   artists.name AS artist_name,
4   albums.title AS album_name
5 FROM artists, albums
6 WHERE artists.artistid = albums.artistid
7   AND artists.artistid IN (8 , 100, 120);

```

Artistid	artist_name	album_name
8	Audioslave	Audioslave
8	Audioslave	Out Of Exile
8	Audioslave	Revelations
100	Lenny Kravitz	Greatest Hits
120	Pink Floyd	Dark Side Of The Moon

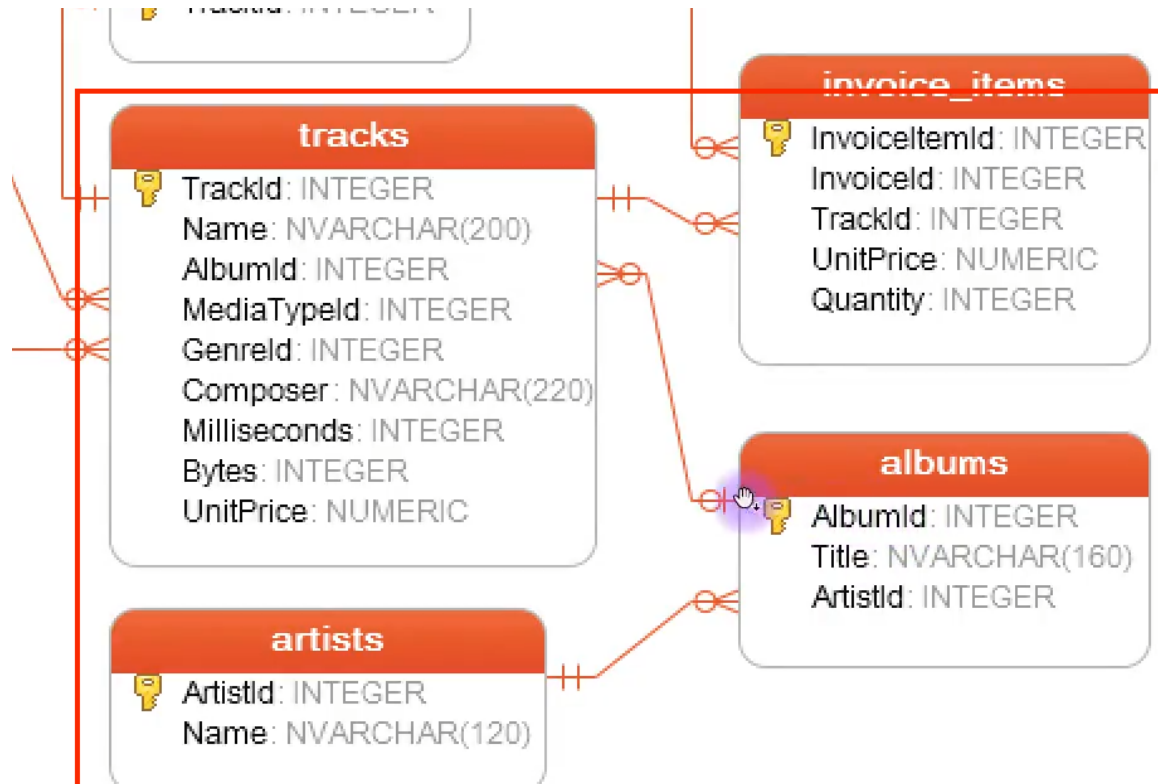
```

SELECT
    artists.artistid,
    artists.name AS artist_name,
    albums.title AS album_name
FROM artists, albums
WHERE artists.artistid = albums.artistid
    AND artists.artistid IN (8 , 100, 120);

```


Join 3 Table

เพิ่มเติมไขถึง Table Tracks Column ชื่อเพลง(Name)ไปด้วย



chinook.db chinook.db.8

```

1 SELECT
2   artists.artistid,
3   artists.name AS artist_name,
4   albums.title AS album_name,
5   tracks.name AS song_name
6 FROM artists, albums, tracks
7 WHERE artists.artistid = albums.artistid
8   AND albums.albumid = tracks.albumid
9   AND artists.artistid IN (8, 100, 120);
    
```

ArtistId	artist_name	album_name	song_name
8	Audioslave	Audioslave	Like a Stone
8	Audioslave	Audioslave	Set It Off
8	Audioslave	Audioslave	Shadow on the Sun
8	Audioslave	Audioslave	I am the Highway
8	Audioslave	Audioslave	Exploder
8	Audioslave	Audioslave	Hypnotize
8	Audioslave	Audioslave	Bring'em Back Alive

```

SELECT
    artists.artistid,
    
```

```

    artists.name AS artist_name,
    albums.title AS album_name,
    tracks.name AS song_name
FROM artists, albums, tracks
WHERE artists.artistid = albums.artistid
    AND albums.albumid = tracks.albumid
    AND artists.artistid IN (8 , 100, 120);

```

Aggregate Functions

Aggregate functions คือฟังก์ชันสถิติเบื้องต้นไว้สรุปผลข้อมูล ฟังก์ชันที่เราเป็นประจำใน standard SQL จะมีอยู่ 5 functions คือ

- COUNT
- AVG
- SUM
- MIN
- MAX

🔊 สิ่งที่ทุกคนต้องรู้คือ aggregate functions ไม่สนใจค่า NULL ในคอลัมน์นั้นๆ เช่น

```
SELECT COUNT(company) FROM customers;
```

Query นี้เราต้องการนับจำนวน company ใน customers table

ถ้าข้อมูลใน customers table มีทั้งหมด 59 rows แต่เรา COUNT(company) ได้แค่ 10 rows แปลว่าคอลัมน์ company มีค่า NULL ทั้งหมด 49 rows (คิดจาก 59 - 10)

The screenshot shows a SQL query execution interface for a database named 'chinook.db'. The query is as follows:

```

1 SELECT
2   AVG(milliseconds),
3   SUM(milliseconds),
4   MIN(milliseconds),
5   MAX(milliseconds),
6   COUNT(milliseconds)
7 FROM tracks;

```

The results are displayed in a table with the following columns and values:

AVG(milliseconds)	SUM(milliseconds)	MIN(milliseconds)	MAX(milliseconds)	count(milliseconds)
393599.2121039109	1378778040	1071	5286953	3503


```

SELECT
    AVG(milliseconds),
    SUM(milliseconds),
    MIN(milliseconds),
    MAX(milliseconds),
    count(milliseconds)
FROM tracks;


```

เปลี่ยนชื่อ Column โดยการใช้ AS

แต่ SQL สามารถใช้เคาะ Space bar แทนเขียน AS ได้



chinook.db

 chinook.db.9

```
1 SELECT
2   AVG(milliseconds) AS avg_mill,
3   SUM(milliseconds) AS sum_mill,
4   MIN(milliseconds) sum_mill,
5   MAX(milliseconds) min_mill,
6   COUNT(milliseconds) count_mill
7 FROM tracks;
```

avg_mill	sum_mill	sum_mill	min_mill	count_mill
393599.2121039109	1378778040	1071	5286953	3503

สามารถเปลี่ยนหลักทศนิยมโดยใช้ ROUND

chinook.db

chinook.db.9

```

1 SELECT
2   ROUND(AVG(milliseconds),2) AS avg_mill,
3   SUM(milliseconds) AS sum_mill,
4   MIN(milliseconds) sum_mill,
5   MAX(milliseconds) min_mill,
6   COUNT(milliseconds) count_mill
7 FROM tracks;

```

avg_mill	sum_mill	sum_mill	min_mill	count_mill
393599.21	1378778040	1071	5286953	3503

```

SELECT
    ROUND(AVG(milliseconds),2) AS avg_mill,
    SUM(milliseconds) AS sum_mill,
    MIN(milliseconds) sum_mill,
    MAX(milliseconds) min_mill,
    count(milliseconds) count_mill
FROM tracks;

```

ตัวอย่างการใช้ ROUND ในการ SUM ยอดแล้วปัดทศนิยม

Query the following two values from the **STATION** table:

1. The sum of all values in LAT_N rounded to a scale of 2 decimal places.
2. The sum of all values in LONG_W rounded to a scale of 2 decimal places.

Input Format

The **STATION** table is described as follows:

STATION	
Field	Type
ID	NUMBER
CITY	VARCHAR2 (21)
STATE	VARCHAR2 (2)
LAT_N	NUMBER
LONG_W	NUMBER

```

1 SELECT ROUND(SUM(LAT_N), 2),
2        ROUND(SUM(LONG_W), 2)
3 FROM STATION;

```

Upload Code as File

```

SELECT  ROUND(SUM(LAT_N), 2),
        ROUND(SUM(LONG_W), 2)

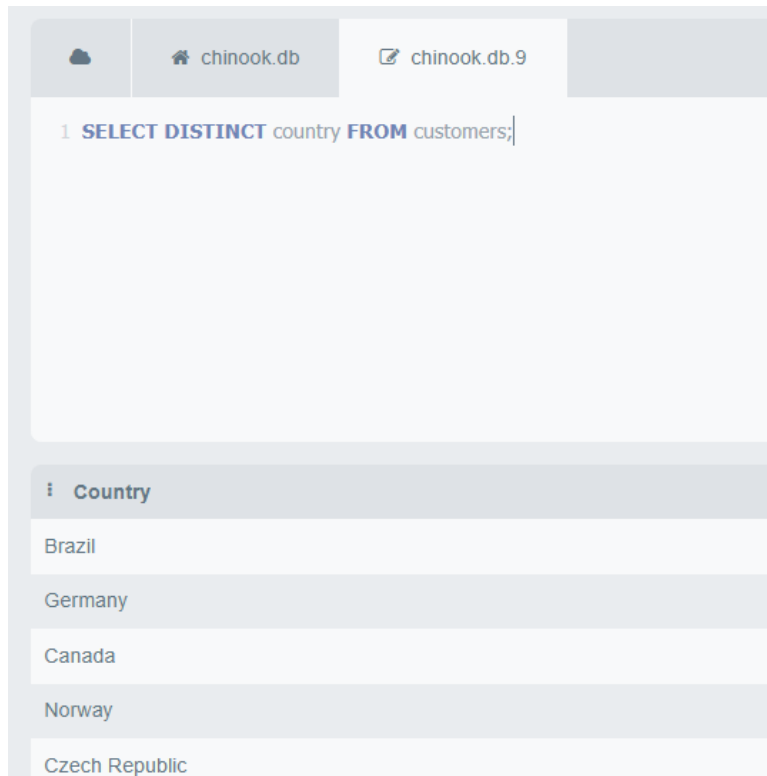
```

```
FROM STATION;
```

COUNT Distinct

ดึงข้อมูลแบบ Unique (ไม่ซ้ำ)

ใช้ Function DISTINCT



```
SELECT DISTINCT country FROM customers;
```

ใช้ COUNT เพื่อนับจำนวน

มี 24 ประเทศ

	chinook.db	chinook.db.9
1 SELECT COUNT(DISTINCT country) FROM customers;		
COUNT(DISTINCT country)		
24		

```
SELECT COUNT(DISTINCT country) FROM customers;
```

รับจำนวนคนของใน 24 ประเทศ

	chinook.db	chinook.db.9
1 SELECT COUNT(DISTINCT country), COUNT(*) FROM customers;		
COUNT(DISTINCT country)	COUNT(*)	
24	59	

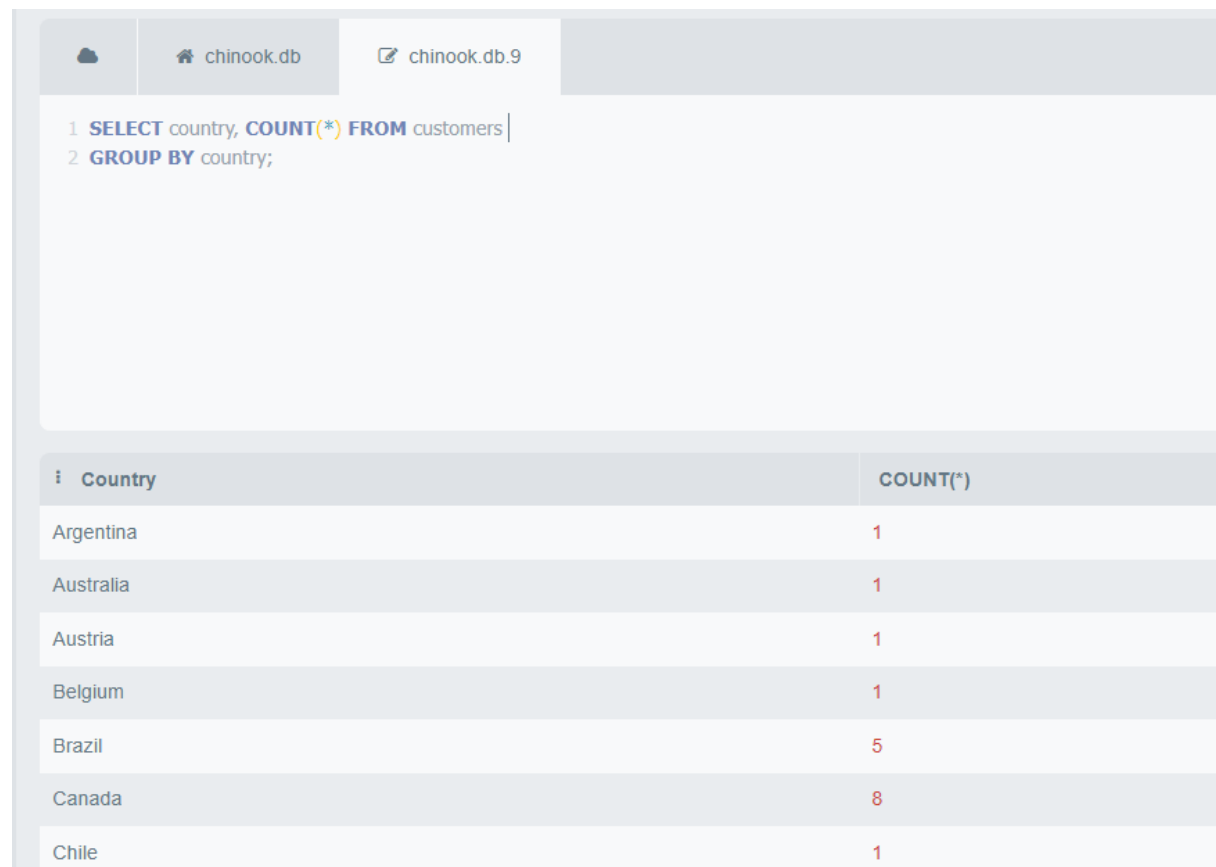
```
SELECT COUNT(DISTINCT country), COUNT(*) FROM customers;
```

GROUP BY

เราใช้ **GROUP BY** กับ **Aggregate Functions** เพื่อสรุปข้อมูลแบ่งตามกลุ่มที่เราต้องการ ตัวอย่าง เช่น การนับจำนวนลูกค้าในแต่ละประเทศ

```
SELECT country, COUNT(*) FROM customers GROUP BY country;
```

Tip - คอลัมน์ไหนที่อยู่ **GROUP BY** ให้เราเขียนคอลัมน์นั้นใน **SELECT** ด้วย 😊



Country	COUNT(*)
Argentina	1
Australia	1
Austria	1
Belgium	1
Brazil	5
Canada	8
Chile	1

ดึงข้อมูลประเภทของเพลงว่า แต่ละประเภทมีกี่ประเภท

chinook.db		chinook.db.9
<pre> 1 /*SELECT country, COUNT(*) AS count_country FROM customers 2 GROUP BY country;*/ 3 4 SELECT genres.name, COUNT(*) AS count_songs FROM genres, tracks 5 WHERE genres.genreid = tracks.genreid 6 GROUP BY genres.name; </pre>		
i	Name	count_songs
	Alternative	40
	Alternative & Punk	332
	Blues	81
	Bossa Nova	15
	Classical	74
	Comedy	17
	Drama	64
	Easy Listening	24

```

SELECT genres.name, COUNT(*) AS count_songs FROM genres, tracks
WHERE genres.genreid = tracks.genreid
GROUP BY genres.name;

```

HAVING

HAVING เขียนเหมือนกับ **WHERE** clause

แต่เราใช้ **HAVING** ในการกรองข้อมูลที่ผ่านการ **GROUP BY** มาแล้วนะครับ 😊

อธิบายง่ายๆคือการกรองกลุ่มที่เราต้องการนั่นเอง

ดึงข้อมูลประเภทของเพลงว่า แต่ละประเภทมีกี่ประเภท

เพิ่มเงื่อนไขดึงประเภทที่มีมากกว่า 100 เพลง

```

SELECT
    genres.name,
    COUNT(*) AS count_songs
FROM genres, tracks

```



```
WHERE genres.genreid = tracks.genreid
GROUP BY genres.name
HAVING COUNT(*) >= 100;
```

เพิ่มเงื่อนไขไม่ดังเพลง Rock

	chinook.db	chinook.db.9
<pre>1 /*SELECT country, COUNT(*) AS count_country FROM customers 2 GROUP BY country;*/ 3 4 SELECT 5 genres.name, 6 COUNT(*) AS count_songs 7 FROM genres, tracks 8 WHERE genres.genreid = tracks.genreid AND genres.name <> 'Rock' 9 GROUP BY genres.name 10 HAVING COUNT(*) >= 100;</pre>		
Name		count_songs
Alternative & Punk		332
Jazz		130
Latin		579
Metal		374




```
SELECT
    genres.name,
    COUNT(*) AS count_songs
FROM genres, tracks
WHERE genres.genreid = tracks.genreid AND genres.name <> 'Rock'
GROUP BY genres.name
HAVING COUNT(*) >= 100;
```

ORDER BY

ORDER BY ใช้เพื่อ sort data เรียงข้อมูลจากน้อยไปมาก (default, ascending order) หรือถ้าอยากเรียงจากมากไปน้อยให้ใส่ **DESC** ต่อท้ายชื่อคอลัมน์

ตัวอย่าง Query ด้านล่างเราดึงข้อมูลทุกคอลัมน์จาก customers table และเรียงข้อมูลตามประเทศตามตัวอักษร Z ไป A (descending order)

```
SELECT * FROM customers ORDER BY country DESC;
```

	 chinook.db	 chinook.db.9
<pre>1 SELECT genres.NAME , COUNT(*) FROM genres, tracks 2 WHERE genres.genreid = tracks.genreid 3 GROUP BY genres.name 4 ORDER BY COUNT(*) LIMIT 5;</pre>		
i Name		COUNT(*)
Opera		1
Rock And Roll		12
Science Fiction		13
Bossa Nova		15
Comedy		17

```
SELECT genres.NAME , COUNT(*) FROM genres, tracks
WHERE genres.genreid = tracks.genreid
GROUP BY genres.name
ORDER BY COUNT(*) LIMIT 5;
```