



SQL for Data Analyst 102

☰ Tags	Foundation	SQL
↗ Class		
☑ Finished Yet?	<input checked="" type="checkbox"/>	
↗ Knowledge	 The Second Sprint: SQL	

How to use WHERE clause?

- WHERE clause ใช้ในการ Filter เอพะข้อมูลจาก row กี่ต้องการ
- LIKE ใช้คู่กับ Wildcard 2 ตัวคือ % (ใช้ Match กี่ Character ก็ได้) และ _ (ใช้ Match 1 Character) ยกตัวอย่างเช่น:

```
SELECT * FROM customers WHERE country LIKE '%U';
```

[ดึงข้อมูล column กั้งหมดจากตาราง customers โดยดึงเฉพาะข้อมูลที่ column country ขึ้นต้นด้วย U]

- หรือ 'J_hn' จะกรองให้ชื่อขึ้นต้นด้วยตัว J และวตามด้วยตัวอักษรอะไรก็ได้ และลงท้ายด้วย hn
- ใน SQLite ตัว operator LIKE นั้น case insensitive ไม่สนใจตัวพิมพ์ใหญ่-ตัวพิมพ์เล็ก
- ตัวอย่างการดึงข้อมูลด้วย WHERE และ LIKE:

```

#All rows that country = USA
SELECT * FROM customers
WHERE country = 'USA';

#All rows that country = USA and state = CA
SELECT * FROM customers
WHERE country = 'USA' AND state = 'CA';

#All rows that country = USA or country = United Kingdom
SELECT * FROM customers
WHERE country = 'USA' OR country = 'United Kingdom';

#All rows that country = USA or country = United Kingdom (Use IN)
SELECT * FROM customers
WHERE country IN ('USA', 'United Kingdom');

#All rows that country != USA and United Kingdom
SELECT * FROM customers
WHERE country NOT IN ('USA', 'United Kingdom');

#All rows that email contains @gmail.com
SELECT * FROM customers
WHERE email LIKE '%@gmail.com';

#All rows that email does not contain @gmail.com
SELECT * FROM customers
WHERE email NOT LIKE '%gmail.com';

#All rows that value in the company column is NULL
SELECT * FROM customers
WHERE company IS NULL;

#All rows that value in the company column is not NULL
SELECT * FROM customers
WHERE company IS NOT NULL;

#All rows that customerid's value is between 10 and 15, includes 10 and 15
SELECT * FROM customers
WHERE customerid BETWEEN 10 AND 15;

```

Lesson 1: Filter Data Part 1

-การกรองข้อมูลที่เราต้องการจะดึง สามารถทำได้ด้วย WHERE clause เป็นการสร้างเงื่อนไขในการดึงข้อมูล เช่น:

```
1 SELECT * FROM customers WHERE Country = 'USA';
```

	CustomerId	FirstName	LastName	Company	Address	City	State	Country	PostalCode	Phone	Fax	Email	SupportRepId
1	16	Frank	Harris	Google Inc.	1600 Amphitheatre Parkway	Mountain View	CA	USA	94043-1351	+1 (650) 253-0000	+1 (650) 253-0000	farris@google.com	
2	17	Jack	Smith	Microsoft Corporation	1 Microsoft Way	Redmond	WA	USA	98052-8300	+1 (425) 882-8080	+1 (425) 882-8081	jacksmith@microsoft.com	
3	18	Michelle	Brooks	NULL	627 Broadway	New York	NY	USA	10012-2612	+1 (212) 221-3546	+1 (212) 221-4679	michelleb@comcast.net	
4	19	Tim	Goyer	Apple Inc.	1 Infinite Loop	Cupertino	CA	USA	95014	+1 (408) 996-1010	+1 (408) 996-1011	tgoyer@apple.com	
5	20	Dan	Miller	NULL	541 Del Medio Avenue	Mountain View	CA	USA	94040-111	+1 (650) 644-3358	NULL	dmillier@comcast.net	
6	21	Kathy	Chase	NULL	801 W 4th Street	Reno	NV	USA	89503	+1 (775) 223-7665	NULL	kachase@kachase.com	
7	22	Heather	Leacock	NULL	120 S Orange Ave	Orlando	FL	USA	32801	+1 (407) 999-7788	NULL	hleacock@comcast.net	
8	23	John	Gordon	NULL	69 Salem Street	Boston	MA	USA	2113	+1 (617) 522-1333	NULL	johnordor@gmail.com	

[ดึงข้อมูล column กั้งหมดจากตาราง customers โดยดึงเฉพาะข้อมูลที่ column country มีค่าเท่ากับ USA]

-ในกรณีที่เราไม่รู้ว่าค่าที่เราต้องการจะกรองเป็นตัวพิมพ์เล็กหรือตัวพิมพ์ใหญ่ เราสามารถใช้ LOWER ครอบ column ที่เราต้องการกรองเพื่อให้ค่ากั้งหมดใน column นั้นเป็นตัวพิมพ์เล็ก ก่อน แล้วพิมพ์ค่าที่เราต้องการกรองเป็นตัวพิมพ์เล็กได้ เช่น:

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

	CustomerId	FirstName	LastName	Company	Address	City	State	Country	PostalCode	Phone	Fax	Email	SupportRepId
1	52	Emma	Jones	NULL	202 Hoxton Street	London	NULL	United Kingdom	N1 5LH	+44 020 7707 0707	NULL	emma_jones@hotmail.com	3
2	53	Phil	Hughes	NULL	113 Lopus St	London	NULL	United Kingdom	SW1V 3EN	+44 020 7976 5722	NULL	phil.hughes@gmail.com	3
3	54	Steve	Murray	NULL	110 Raeburn Pl	Edinburgh	NULL	United Kingdom	EH4 1HH	+44 0131 315 3300	NULL	steve.murray@yahoo.uk	5

[ดึงข้อมูล column กั้งหมดจากตาราง customers โดยดึงเฉพาะข้อมูลที่ column country มีค่าเท่ากับ United Kingdom]

-เราสามารถใช้ OR แทน ‘หรือ’ และ NOT แทน ‘ไม่’ ร่วมกับการใช้ WHERE clause ได้ เช่น:

```
1 SELECT * FROM customers WHERE NOT Country IN ('USA', 'Canada');
```

	CustomerId	FirstName	LastName	Company	Address	City	State	Country	PostalCode	Phone
1	1	Luis	Gonçalves	Embraer - Empresa Brasileira de ...	Av. Brigadeiro Faria Lima, 2170	São José dos Campos	SP	Brazil	12227-000	+55 (11) 3923-5555
2	2	Leonie	Köhler	NULL	Theodor-Heuss-Straße 34	Stuttgart	NULL	Germany	70174	+49 0711 2842222
3	4	Bjørn	Hansen	NULL	Ullevålsveien 14	Oslo	NULL	Norway	0171	+47 22 44 22 22
4	5	František	Wichterlová	JetBrains s.r.o.	Klanova 9/506	Prague	NULL	Czech Republic	14700	+420 2
5	6	Helena	Holý	NULL	Rilská 3174/6	Prague	NULL	Czech Republic	14300	+420 2
6	7	Astrid	Gruber	NULL	Rotenturmstraße 4, 1010 Innere Stadt	Vienne	NULL	Austria	1010	+43 01
7	8	Daan	Peeeters	NULL	Grétrystraat 63	Brussels	NULL	Belgium	1000	+32 02
8	9	Kara	Nielsen	NULL	Sønder Boulevard 51	Copenhagen	NULL	Denmark	1720	+45 33 32 00

[ดึงข้อมูล column กั้งหมดจากตาราง customers โดยดึงเฉพาะข้อมูลที่ column country ไม่ได้มีค่าเท่ากับ USA หรือ Canada]

Lesson 2: Filter Data Part 2

- เราสามารถใช้ IN แทนการใช้ OR ซ้ำซ้อนในการเลือกค่าจาก column เดียวกัน ทำให้ Query เราสั้นลง เช่น:

```
1 SELECT *
2 FROM customers
3 WHERE Country = 'Brazil' OR Country = 'Germany' OR Country = 'Norway';
```

	CustomerId	FirstName	LastName	Company	Address	City	State	Country	PostalCode	Phone
1	1	Luis	Gonçalves	Embraer - Empresa Brasileira de ...	Av. Brigadeiro Faria Lima, 2170	São José dos Campos	SP	Brazil	12227-000	+55 (11) 3923-5555
2	2	Leonie	Köhler	NULL	Theodor-Heuss-Straße 34	Stuttgart	NULL	Germany	70174	+49 0711 2842222
3	4	Bjørn	Hansen	NULL	Ullevålsveien 14	Oslo	NULL	Norway	0171	+47 22 44 22 22
4	10	Eduardo	Martins	Woodstock Discos	Rua Dr. Falcão Filho, 155	São Paulo	SP	Brazil	01007-010	+55 (11) 3033-5446
5	11	Alexandre	Rocha	Banco do Brasil S.A.	Av. Paulista, 2022	São Paulo	SP	Brazil	01310-200	+55 (11) 3055-3278
6	12	Roberto	Almeida	Riotur	Praça Pio X, 119	Rio de Janeiro	RJ	Brazil	20040-020	+55 (21) 2271-7000
7	13	Fernanda	Ramos	NULL	Qe 7 Bloco G	Brasília	DF	Brazil	71020-677	+55 (61) 3363-5547
8	36	Hannah	Schneider	NULL	Tauentzienstraße 8	Berlin	NULL	Germany	10789	+49 030 26550280

Execution finished without errors.
Result: 10 rows returned in 5ms

```

1 SELECT *
2 FROM customers
3 WHERE Country IN ('Brazil', 'Germany', 'Norway');

```

	CustomerId	FirstName	LastName	Company	Address	City	State	Country	PostalCode	Phone	
1	1	Luís	Gonçalves	Embraer - Empresa Brasileira de ...	Av. Brigadeiro Faria Lima, 2170	São José dos Campos	SP	Brazil	12227-000	+55 (12) 3923-5555	...
2	2	Leони	Кöhler	NULL	Theodor-Heuss-Straße 34	Stuttgart	NULL	Germany	70174	+49 0711 2842222	/
3	4	Bjørn	Hansen	NULL	Ullevålsveien 14	Oslo	NULL	Norway	0171	+47 22 44 22 22	/
4	10	Eduardo	Martins	Woodstock Discos	Rua Dr. Falcão Filho, 155	São Paulo	SP	Brazil	01007-010	+55 (11) 3033-5446	...
5	11	Alexandre	Rocha	Banco do Brasil S.A.	Av. Paulista, 2022	São Paulo	SP	Brazil	01310-200	+55 (11) 3055-3278	...
6	12	Roberto	Almeida	Riotur	Praça Pio X, 119	Rio de Janeiro	RJ	Brazil	20040-020	+55 (21) 2271-7000	...
7	13	Fernanda	Ramos	NULL	Qe 7 Bloco G	Brasília	DF	Brazil	71020-677	+55 (61) 3363-5547	...
8	36	Hannah	Schneider	NULL	Tautentienstraße 8	Berlin	NULL	Germany	10789	+49 030 26550280	>

Execution finished without errors.
Result: 10 rows returned in 5ms

[ดึงข้อมูล column กึ่งหนดจากตาราง customers โดยดึงเฉพาะข้อมูลที่ column country มีค่า เก่ากับ Brazil, Germany และ Norway]

-ถ้าอยากรู้ด้วยข้อมูลจากลูกค้าคนอื่น ๆ ที่ไม่ได้มาจาก 3 ประเทศที่เราเลือก เราสามารถพิมพ์ NOT หน้า IN เพื่อกลับผลลัพธ์ในการกรองได้

-Between แปลว่า อุปะระหว่าง เราสามารถใช้ Clause นี้ในการหาค่าที่อยู่ในช่วงที่เราต้องการได้ เช่น:

```

1 SELECT *
2 FROM customers
3 WHERE CustomerId BETWEEN 5 AND 10;

```

	CustomerId	FirstName	LastName	Company	Address	City	State	Country	PostalCode	Phone	Fax
1	5	František	Wichterlová	JetBrains s.r.o.	Klanova 9/506	Prague	NULL	Czech Republic	14700	+420 2 4172 5555	+420 2 4172 5555
2	6	Helena	Holý	NULL	Rilská 3174/6	Prague	NULL	Czech Republic	14300	+420 2 4177 0449	NULL
3	7	Astrid	Gruber	NULL	Rotenturmstraße 4, 1010 Innere Stadt	Vienne	NULL	Austria	1010	+43 01 5134505	NULL
4	8	Daan	Peeters	NULL	Grétrystraat 63	Brussels	NULL	Belgium	1000	+32 02 219 03 03	NULL
5	9	Kara	Nielsen	NULL	Sønder Boulevard 51	Copenhagen	NULL	Denmark	1720	+453 3331 9991	NULL
6	10	Eduardo	Martins	Woodstock Discos	Rua Dr. Falcão Filho, 155	São Paulo	SP	Brazil	01007-010	+55 (11) 3033-5446	+55 (11) 3033-4564

[ดึงข้อมูล column กึ่งหนดจากตาราง customers โดยดึงเฉพาะข้อมูลที่ CustomerId อยู่ระหว่าง 5 และ 10 (ເອົາເລີх 5 และ 10 ດັ່ງຍ)]

-เราสามารถใช้ Between กับข้อมูลที่เป็นວັນກີ່ແລະເວລາໄດ້ ເຊັ່ນ:

```

1 SELECT InvoiceDate
2 FROM invoices
3 WHERE InvoiceDate BETWEEN '2009-01-01 00:00:00' AND '2009-01-19 00:00:00';

```

	InvoiceDate
1	2009-01-01 00:00:00
2	2009-01-02 00:00:00
3	2009-01-03 00:00:00
4	2009-01-06 00:00:00
5	2009-01-11 00:00:00
6	2009-01-19 00:00:00

[ดึงข้อมูล column InvoiceDate จากตาราง invoices โดยที่เวลาอยู่ระหว่าง 00:00:00 ของวันที่ 2009-01-01 และ 00:00:00 ของวันที่ 2009-01-19]

-ถ้าเราต้องการดึงค่า NULL เราจะใช้ WHERE column IS NULL ไม่ใช่ = เช่น:

```

1 SELECT *
2 FROM customers
3 WHERE Company IS NULL

```

	CustomerId	FirstName	LastName	Company	Address	City	State	Country	PostalCode	Phone	Fax
1	2	Leonie	Köhler	NULL	Theodor-Heuss-Straße 34	Stuttgart	NULL	Germany	70174	+49 0711 2842222	NULL
2	3	François	Tremblay	NULL	1498 rue Bélanger	Montréal	QC	Canada	H2G 1A7	+1 (514) 721-4711	NULL
3	4	Bjørn	Hansen	NULL	Ullevålsveien 14	Oslo	NULL	Norway	0171	+47 22 44 22 22	NULL
4	6	Helena	Holý	NULL	Rilská 3174/6	Prague	NULL	Czech Republic	14300	+420 2 4177 0449	NULL
5	7	Astrid	Gruber	NULL	Rotenturmstraße 4, 1010 Innere Stadt	Vienne	NULL	Austria	1010	+43 01 5134505	NULL
6	8	Daan	Peeters	NULL	Grétrystraat 63	Brussels	NULL	Belgium	1000	+32 02 219 03 03	NULL
7	9	Kara	Nielsen	NULL	Sønder Boulevard 51	Copenhagen	NULL	Denmark	1720	+453 3331 9991	NULL
8	13	Fernanda	Ramos	NULL	Oe 7 Bloco G	Brasília	DF	Brazil	71020-677	+55 (61) 3363-5547	+55 (61) 3363-7855

[ดึงข้อมูล column กึ่งหมดจากตาราง customers โดยดึงเฉพาะข้อมูลที่ค่าใน column Company มีค่า NULL]

-ถ้าไม่อยากได้ค่า NULL ให้พิมพ์ NOT หน้า NULL (เป็น WHERE column IS NOT NULL)

Lesson 3: Filter Data Part 3

-การเขียน Pattern Matching ใช้ในการหา Pattern ใน column ที่เราสนใจ เช่น:

```
1 SELECT FirstName, LastName, Country, Email  
2 FROM customers  
3 WHERE Email LIKE '%gmail.com';
```

	FirstName	LastName	Country	Email
1	François	Tremblay	Canada	ftremblay@gmail.com
2	Helena	Holý	Czech Republic	hholý@gmail.com
3	Heather	Leacock	USA	hleacock@gmail.com
4	Frank	Ralston	USA	fralston@gmail.com
5	Julia	Barnett	USA	jubarnett@gmail.com
6	Martha	Silk	Canada	marthasilk@gmail.com
7	Dominique	Lefebvre	France	dominiquelefebvre@gmail.com
8	Phil	Hughes	United Kingdom	phil.hughes@gmail.com

[ดึงข้อมูลจาก column FirstName, LastName, Country, และ Email จากตาราง customers โดยที่ค่าใน column Email ข้างหน้าเป็นค่าอะไรก็ได้ แต่ข้างหลังต้องลงท้ายด้วย @gmail.com]

-เราสามารถใช้ในการหา Pattern ของตัวเลขได้ด้วย เช่น:

```

1 SELECT FirstName, LastName, Country, Email, Phone
2 FROM customers
3 WHERE Phone LIKE '%99%';

```

	FirstName	LastName	Country	Email	Phone
1	Kara	Nielsen	Denmark	kara.nielsen@jubii.dk	+453 3331 9991
2	Tim	Goyer	USA	tgoyer@apple.com	+1 (408) 996-1010
3	Heather	Leacock	USA	hleacock@gmail.com	+1 (407) 999-7788
4	Isabelle	Mercier	France	isabelle_mercier@apple.fr	+33 03 80 73 66 99
5	Puja	Srivastava	India	puja_srivastava@yahoo.in	+91 080 22289999

[ดึงข้อมูลจาก column FirstName, LastName, Country, Email, และ Phone จากตาราง customers โดยที่ค่าใน column phone มีค่า 99 อยู่ ส่วนค่าข้างหน้าและข้างหลังจะเป็นค่าใดก็ได้]

-ในกรณีที่เราต้องการ Match แค่ character ตัวเดียว เราจะใช้ Underscore (_) เช่น

```

1 SELECT FirstName, LastName, Country, Email, Phone
2 FROM customers
3 WHERE FirstName LIKE 'Leon__';

```

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

[ดึงข้อมูลจาก column Firstname, Lastname, Country, Email, และ Phone จากตาราง customers โดยที่ค่าใน column FirstName ข้างหน้าต้องเป็น Leon แต่ค่า 2 ค่าหลังจะเป็นค่า [ได้ก็ได้ 2 ตัวอักษร]

*สมมติว่าเราไม่รู้ว่า Leonie ชื่อสะกดยังไง เรารู้แค่ว่าชื่อขึ้นต้นด้วย Leon__ เราสามารถหาข้อมูลของ Leonie จากฐานข้อมูลด้วยวิธีนี้ได้

Lesson 4: COALESCE

-COALESCE ใช้ในการ replace missing values (ค่า NULL) ใน column ที่มีค่า NULL เช่น:

```
1 SELECT
2     Company,
3     COALESCE(Company, 'End Customer') AS 'Company Clean'
4 FROM customers;
```

	Company	Company Clean
1	Embraer - Empresa Brasileira de ...	Embraer - Empresa Brasileira de ...
2	NULL	End Customer
3	NULL	End Customer
4	NULL	End Customer
5	JetBrains s.r.o.	JetBrains s.r.o.
6	NULL	End Customer
7	NULL	End Customer
8	NULL	End Customer
9	NULL	End Customer

[ดึงข้อมูลจาก column Company และสร้าง column ใหม่ชื่อ Company Clean ที่แทนค่า NULL ด้วย End Customer จากตาราง customers]

-เราสามารถใช้ CASE ใน การสร้างเงื่อนไขให้แทนค่า column ที่มีค่า NULL ได้เช่นกัน เช่น:

```

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 'Company Clean 2'
7  FROM customers;

```

	Company	Company Clean	Company Clean 2
1	Embraer - Empresa Brasileira de ...	Embraer - Empresa Brasileira de ...	Corporate
2	NULL	End Customer	End Customer
3	NULL	End Customer	End Customer
4	NULL	End Customer	End Customer
5	JetBrains s.r.o.	JetBrains s.r.o.	Corporate
6	NULL	End Customer	End Customer
7	NULL	End Customer	End Customer
8	NULL	End Customer	End Customer

[ดึงข้อมูลจาก column Company และสร้าง column ใหม่คือ Company Clean กี่แทนค่า NULL ด้วย End Customer และ Company Clean 2 กี่ตั้งเงื่อนไขไว้ว่า ถ้าค่าเป็น Null ให้แทน NULL ด้วย End Customer แต่ถ้าไม่ใช่ค่า NULL ให้แทนด้วย Corporate จากตาราง customers]

Lesson 5: JOIN Data using WHERE

- เราสามารถใช้ WHERE แทน INNER JOIN ได้ เช่น:

```

1 SELECT *
2 FROM artists, albums
3 WHERE artists.ArtistId = albums.ArtistId;

```

	ArtistId	Name	AlbumId	Title	ArtistId
1	1	AC/DC	1	For Those About To Rock We Salute ...	1
2	2	Accept	2	Balls to the Wall	2
3	2	Accept	3	Restless and Wild	2
4	1	AC/DC	4	Let There Be Rock	1
5	3	Aerosmith	5	Big Ones	3
6	4	Alanis Morissette	6	Jagged Little Pill	4
7	5	Alice In Chains	7	Facelift	5
8	6	Antônio Carlos Jobim	8	Warner 25 Anos	6

[ดึงข้อมูลจาก column กั้งหนดจากตาราง artists และ albums โดยที่ artist.ArtistId (Primary Key) = albums.ArtistID (Foreign Key)]

```

#WHERE
SELECT * FROM artists, albums
WHERE artists.ArtistId = albums.ArtistId;

#INNER JOIN
SELECT * FROM artists JOIN albums
ON artists.ArtistId = albums.ArtistId;

```

-ตัวอย่างเพิ่มเติม:

```

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
1	8	Audioslave	Audioslave
2	8	Audioslave	Out Of Exile
3	8	Audioslave	Revelations
4	100	Lenny Kravitz	Greatest Hits
5	120	Pink Floyd	Dark Side Of The Moon

[ดึงข้อมูลจาก column ArtistId และ Name (เปลี่ยนชื่อเป็น Artist_Name) จากตาราง artists และ Title (เปลี่ยนชื่อเป็น Album_Name) จากตาราง albums โดยที่ artist.ArtistId (Primary Key) = albums.ArtistID (Foreign Key) และ ArtistId คือ 8, 100, และ 120]

- เราสามารถ JOIN ได้มากกว่า 2 ตาราง เช่น:

```

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
1	8	Audioslave	Audioslave	Cochise
2	8	Audioslave	Audioslave	Show Me How to Live
3	8	Audioslave	Audioslave	Gasoline
4	8	Audioslave	Audioslave	What You Are
5	8	Audioslave	Audioslave	Like a Stone
6	8	Audioslave	Audioslave	Set It Off
7	8	Audioslave	Audioslave	Shadow on the Sun
8	8	Audioslave	Audioslave	I am the Highway

[ดึงข้อมูลจาก column ArtistId, Name(ของ artist), Title, และ Name(ของ tracks) จากตาราง artists, albums, และ tracks โดยที่ artist.ArtistId (Primary Key) = albums.ArtistID (Foreign Key) และ albums.AlbumId = tracks.AlbumID และ ArtistId คือ 8, 100, และ 120]

Lesson 6: Aggregate Functions

-Aggregate Functions คือ function ทางสถิติเบื้องต้นไว้สรุปผลข้อมูล ที่ใช้เป็นประจำใน SQL จะมี COUNT (นับจำนวน), AVG (ค่าเฉลี่ย), SUM (ค่ารวม), MIN (ค่าต่ำสุด), MAX (ค่าสูงสุด)

*Aggregate จะไม่สนใจค่า NULL ใน column นั้น ๆ เช่น ถ้าเราต้องการใช้ COUNT นับจำนวน row แล้ว COUNT จะนับเฉพาะ row ที่ไม่ใช่ค่า NULL ส่วน row ที่ค่าเป็น NULL มันจะไม่บันทึก

-ตัวอย่างการใช้ Aggregate Functions:

```

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

```

	Avg(Milliseconds)	Sum(Milliseconds)	Min(Milliseconds)	Max(Milliseconds)	Count(Milliseconds)
1	393599.212103911	1378778040	1071	5286953	3503

[ดึงข้อมูลจาก column Milliseconds เป็นค่าเฉลี่ย, ค่ารวม, ค่าต่ำสุด, ค่าสูงสุด, และนับจำนวนค่าทั้งหมด(ที่ไม่ใช่ค่า NULL)ใน column Milliseconds จากตาราง tracks]

-เราสามารถใช้ AS ในการตั้งชื่อ column ใหม่ที่เราทำการ Aggregate ได้ โดยพิมพ์ต่อท้าย AVG(Milliseconds) เช่น AVG(Milliseconds) AS avg_mill [เราไม่จำเป็นต้องพิมพ์ AS ถ้าคือว่าเป็น optional clause]

```

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

```

[ROUND 2 คือการทำให้แสดงค่าอออกมาเป็นทศนิยม 2 ตำแหน่ง]

Lesson 7: COUNT Distinct

-การ COUNT Distinct คือการนับค่าแบบไม่ซ้ำกัน ปกติแล้วการใช้ COUNT จะนับค่าที่ซ้ำกัน การใช้ COUNT Distinct จะทำให้นับค่าแบบไม่ซ้ำกัน เช่น:

```
1 SELECT  
2      Country  
3 FROM customers;
```

```
1 SELECT DISTINCT  
2      Country  
3 FROM customers;
```

	Country
1	Brazil
2	Germany
3	Canada
4	Norway
5	Czech Republic
6	Czech Republic

	Country
1	Brazil
2	Germany
3	Canada
4	Norway
5	Czech Republic
6	Austria

[จะสังเกตว่า ในรูปข้างบน เรา SELECT แบบปกติ จะมี Czech Republic โผล่มา 2 ชื่อ แต่ในรูปขว่า เรา SELECT DISTINCT จะทำให้ค่าที่โผล่มาไม่ซ้ำกันแล้ว]

```
1 SELECT  
2     COUNT(Country)  
3 FROM customers;
```

```
1 SELECT  
2     COUNT(DISTINCT Country)  
3 FROM customers;
```

COUNT(DISTINCT Country)

1	24
---	----

COUNT(Country)

1	59
---	----

[จะสังเกตว่า ในรูปข้างมือ เรา COUNT แบบปกติ เราจะนับค่าซ้ำ แต่ในรูปขว่า จำนวนที่นับได้จะน้อยลง เพราะเราใช้ COUNT DISTINCT ในการนับค่าแบบไม่ซ้ำ]

Lesson 8: GROUP BY

- เราใช้ GROUP BY จับกลุ่มข้อมูลที่เราสรุปผลด้วย Aggregate Functions แบ่งเป็นกลุ่มตามต้องการ เช่น การนับจำนวนลูกค้าในแต่ละประเทศ

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

[ดึงข้อมูลจาก column country พร้อมกับนับจำนวน row ก็ต้องมา จากตาราง customers จัดกลุ่มอิงตาม column country]

*Column ไหนที่อยู่ใน GROUP BY ให้พิมพ์ column บันทึกใน SELECT ด้วย

```

1   SELECT
2       Country,
3           COUNT(*) AS 'Count_Country'
4   FROM customers
5   GROUP BY Country;

```

	Country	Count_Country
1	Argentina	1
2	Australia	1
3	Austria	1
4	Belgium	1
5	Brazil	5
6	Canada	8
7	Chile	1
8	Czech Republic	2

[ดึงข้อมูลจาก column country พร้อมกับนับจำนวน row ก็ต้องมาแต่ละตัวซึ่ง column ใหม่ว่า Count_Country จากตาราง customers จัดกลุ่มอิงตาม column country]

-เราสามารถ GROUP BY ได้มากกว่า 1 ตาราง เช่น:

```

1 SELECT
2     genres.Name,
3         COUNT(*) AS 'Count_Song'
4 FROM genres, tracks
5 WHERE genres.GenreId = tracks.GenreId
6 GROUP BY genres.name;

```

	Name	Count_Song
1	Alternative	40
2	Alternative & Punk	332
3	Blues	81
4	Bossa Nova	15
5	Classical	74
6	Comedy	17
7	Drama	64
8	Easy Listening	24
9	Folk	20

[ดึงข้อมูลจาก column Name จากตาราง genres พร้อมกับนับจำนวนเพลงในแต่ละ genres เรา เชื่อมตาราง genres กับ tracks ด้วย column GenreId และจัดกลุ่มตาม name ของ genres เพราะเราต้องการรู้ว่าเพลงในแต่ละ genre มีจำนวนเท่าใด (ดึงชื่อจาก genres และดึงจำนวน เพลงจาก tracks เเลยต้องใช้ 2 ตาราง)]

Lesson 9: HAVING

-HAVING ใช้เหมือน WHERE clause แต่ใช้ในการกรองข้อมูลที่ผ่านการ GROUP BY มาแล้ว เช่น:

```

1   SELECT
2       genres.Name,
3           COUNT(*) AS 'Count_Song'
4   FROM genres, tracks
5   WHERE genres.GenreId = tracks.GenreId
6   GROUP BY genres.name
7   HAVING COUNT(*) >= 100
8   ORDER BY COUNT(*) DESC;

```

	Name	Count_Song
1	Rock	1297
2	Latin	579
3	Metal	374
4	Alternative & Punk	332
5	Jazz	130

[ดึงข้อมูลจาก column Name จากตาราง genres พร้อมกับบันทึกจำนวนเพลงในแต่ละ genres เรา เชื่อมตาราง genres กับ tracks ด้วย column GenreId และจัดกลุ่มตาม name ของ genres โดยมีเงื่อนไขว่าจำนวนเพลงมากกว่าหรือเท่ากับ 100 และเรียงค่าตามจำนวนเพลงมากไปน้อย]

-เราสามารถกรองค่าก่อนการจัดกลุ่มได้ด้วยการเพิ่มเงื่อนไขตรง WHERE เช่น genres.name <> 'Rock' ก็จะเอา genre ที่เป็น Rock ออกไป

Lesson 10: ORDER BY

-ORDER BY ใช้ในการเรียงข้อมูลจากน้อยไปมาก (ASC/Default) หรือมากไปน้อย (DESC)
เช่น:

```

1 SELECT genres.Name, COUNT(*)
2 FROM genres, tracks
3 WHERE genres.GenreId = tracks.GenreId
4 GROUP BY genres.Name
5 ORDER BY COUNT(*) ;

```

	Name	COUNT(*)
1	Opera	1
2	Rock And Roll	12
3	Science Fiction	13
4	Bossa Nova	15
5	Comedy	17
6	Easy Listening	24
7	Sci Fi & Fantasy	26
8	Heavy Metal	28

[ดึงข้อมูลจาก column Name จากตาราง genres พร้อมกับนับจำนวนเพลงในแต่ละ genres เรา เชื่อมตาราง genres กับ tracks ด้วย column GenreId และจัดกลุ่มตาม name ของ genres เรียงค่าตามจำนวนเพลงจากน้อยไปมาก]

-เราสามารถใช้ LIMIT X เพื่อจำกัดการแสดงผลให้เหลือแค่ X แถวแรกเท่านั้น เช่น:

```
1 SELECT genres.Name, COUNT(*)
2 FROM genres, tracks
3 WHERE genres.GenreId = tracks.GenreId
4 GROUP BY genres.Name
5 ORDER BY COUNT(*) DESC
6 LIMIT 5;
```

	Name	COUNT(*)
1	Rock	1297
2	Latin	579
3	Metal	374
4	Alternative & Punk	332
5	Jazz	130

[ดึงข้อมูลจาก column Name จากตาราง genres พร้อมกับบันทึกจำนวนเพลงในแต่ละ genres เรา เชื่อมตาราง genres กับ tracks ด้วย column GenreId และจัดกลุ่มตาม name ของ genres เรียงค่าตามจำนวนเพลงจากมากไปน้อย และแสดงผลเฉพาะ 5 แนวแรกเท่านั้น]