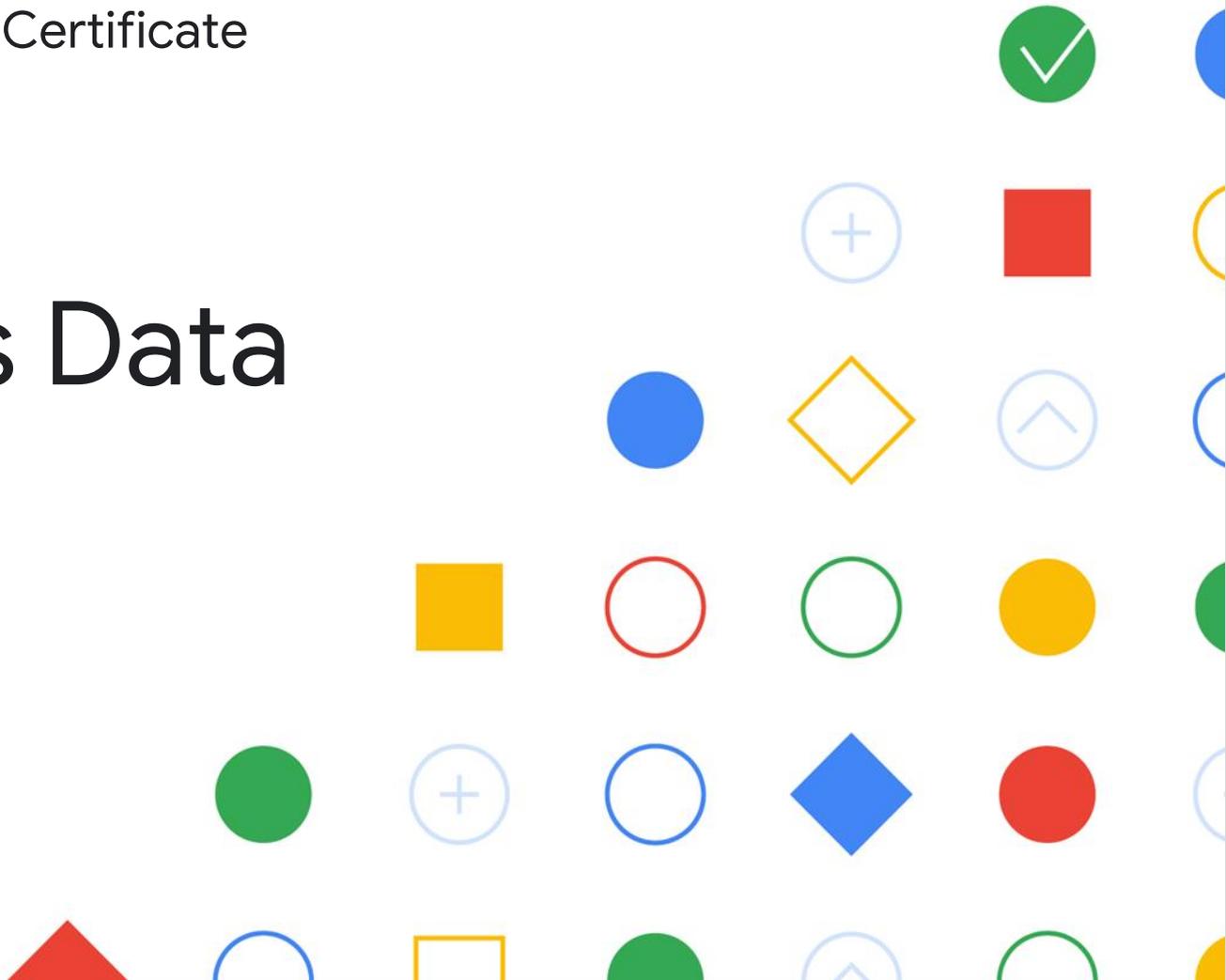


4. Process Data

from Dirty to Clean



Overview:

01

The Importance of Integrity

02

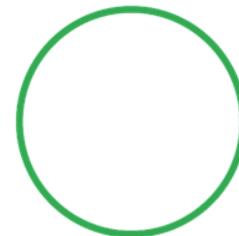
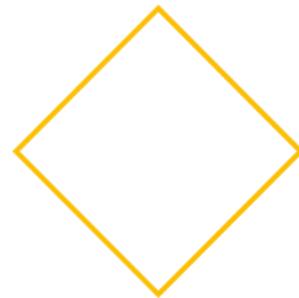
Sparkling-clean Data

03

Cleaning Data with SQL

04

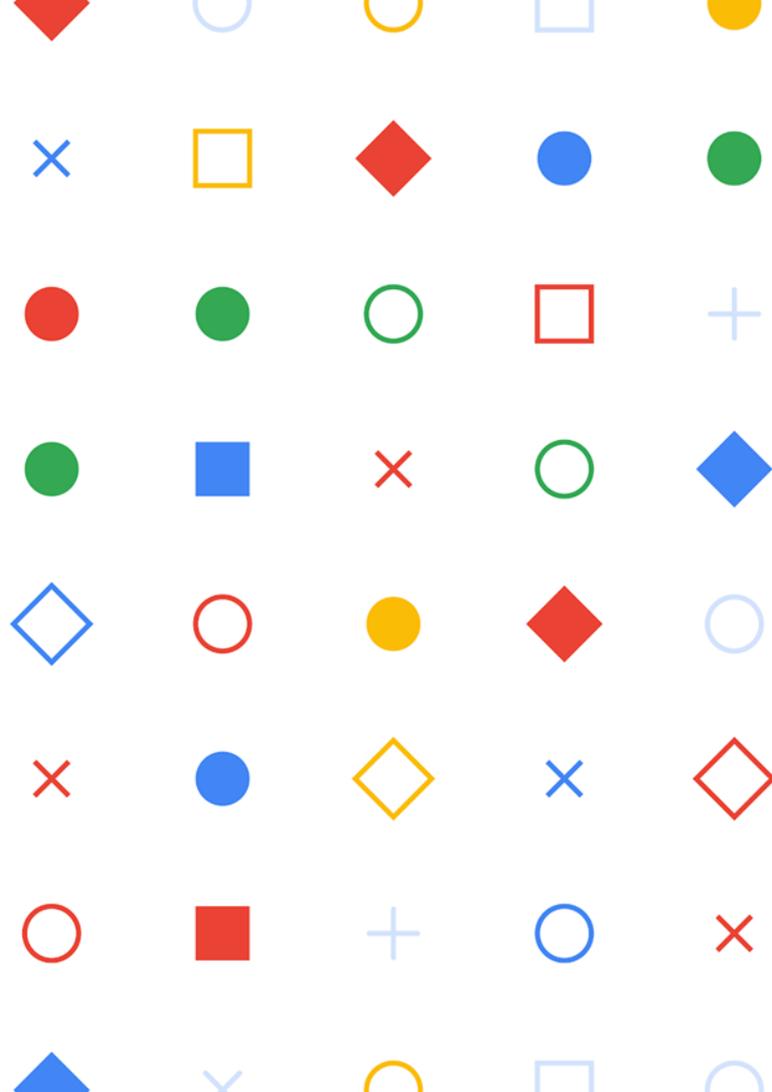
Verify and Report on Cleaning Results



The Importance of Integrity

— *Part 1:*

1. Data Integrity
2. Insufficient Data
3. Sampling: Technical Terms
4. Sample Size and Margin of Error



Data Integrity

Data Integrity

= Accuracy + Completeness + Consistency + Trustworthiness of Data

= ความแม่นยำ + ความครบถ้วน + ความสม่ำเสมอ + ความน่าเชื่อถือ ของข้อมูล

ปัจจัยที่อาจกระทบ Data Integrity

Data Replication

การเก็บข้อมูลเดียวกันไว้หลายๆ ที่

แต่ละที่ข้อมูลอาจไม่ตรงกัน

Data Transfer

การ move/copy ข้อมูลจากที่หนึ่งไปยังอีกที่หนึ่ง

อาจเกิดการตกหล่น ขัดข้องหรือคลาดเคลื่อนระหว่างโอนย้าย เช่น ตั้งค่าประเภทข้อมูลผิดตอนนำเข้าข้อมูล

Data Manipulation

การแปลงข้อมูลเพื่อจัดระเบียบหรือทำให้ใช้ง่ายขึ้น

อาจเกิดข้อมูลตกหล่นเช่นตั้งใจจะลบแถวซ้ำแต่ความจริงไม่ซ้ำ หรือข้อมูลคลาดเคลื่อนจากการคำนวณที่ผิด

Data Integrity

ข้อผิดพลาดที่พบบ่อยและทางแก้ไข

Problems

ข้อมูลประเภท Date

อเมริกาใช้ MMDDYYYY ประเทศอื่นใช้ DDMMYYYY เช่น
07-01-1991 อาจเป็น 1 ก.ค. 1991 หรือ 7 ม.ค. 1991

บันทึกหรือนำเข้าข้อมูลผิดประเภท เช่น นำเข้าข้อมูลตัวเลขเป็น
ข้อความโดยไม่รู้ตัว (1234 ไม่เท่ากับ "1234")

ข้อมูลมีแถวซ้ำ (Duplicated rows) โดยไม่ตั้งใจ

ข้อมูลผิดจากความผิดพลาดของมนุษย์ (Human errors) เช่น
กรอกหมายเลขบัตรประชาชนไม่ครบหลัก หรือกรอกข้อมูลเดียวกัน
ไม่เหมือนกัน เช่น กทม. กรุงเทพฯ กรุงเทพมหานคร

Solutions

ดู Data Dictionary (ถ้ามี) ว่าใช้แบบไหน และควรปรับเป็น
รูปแบบ YYYYMMDD เพื่อให้เรียงลำดับก่อนหลังง่ายขึ้น

มีสัปดาห์บันทึกหรือนำเข้าข้อมูลว่าประเภทของข้อมูลถูกต้อง
แล้วหรือยัง ถ้ายัง แปลงประเภทให้เรียบร้อยก่อน

อย่าลืมลบแถวซ้ำให้เหลือแถวเดียว (Deduplicate)
ก่อนทำการคำนวณต่อ

กำหนด "ข้อจำกัด" (Constraints) ในการกรอกข้อมูล เช่น เลข
บัตรประชาชนต้องมี 13 หลัก มี Drop-down list ให้เลือก
จังหวัดแทนที่จะให้พิมพ์เอง เป็นต้น

Insufficient Data

ประเภทของปัญหาข้อมูลไม่พอมีดังนี้

Problem Types

ข้อมูลมาจากแค่แหล่งเดียว
(Data from only one source)

ข้อมูลมีการอัปเดตอยู่ตลอดเวลา
(Data that keeps updating)

ข้อมูลล้าสมัยแล้ว
(Outdated data)

ข้อมูลถูกจำกัดด้วยภูมิศาสตร์
(Geographically-limited data)

Examples

ต้องการวิเคราะห์แนวโน้มตลาดรองเท้าวิ่ง แต่มีข้อมูลยอดขายย้อนหลังแค่
ของบริษัทตนเอง

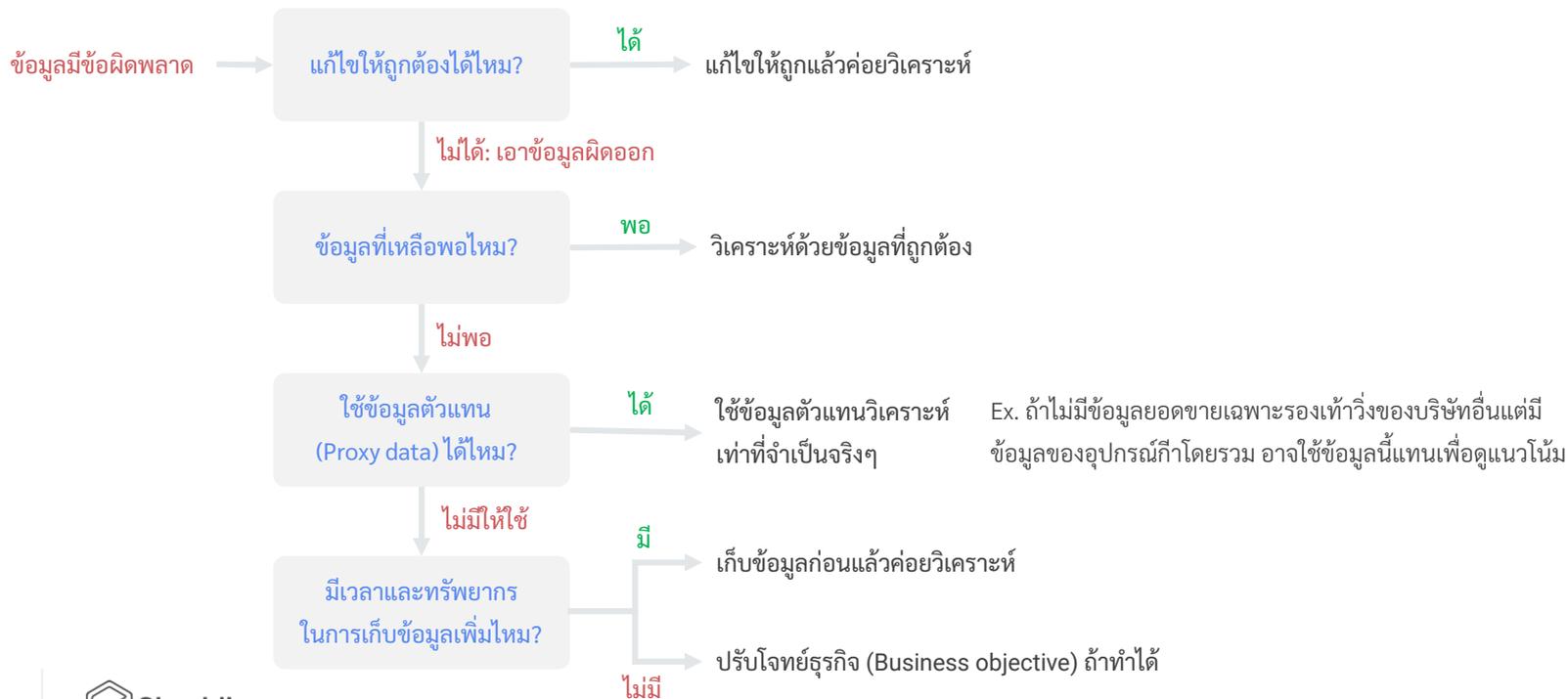
ต้องการวิเคราะห์ว่าแคมเปญการตลาดลด 20% ตลอดเดือน มิ.ย. ได้ผลหรือไม่
แต่นี้เพิ่งผ่านไปแค่ 5 วัน

ต้องการหาไอเดียออกรองเท้าวิ่งรุ่นใหม่ แต่มีแค่ข้อมูลผลสำรวจความชอบ
ของนักวิ่งที่เก็บเมื่อ 5 ปีที่แล้ว

ต้องการขยายตลาดไปยุโรป แต่มีข้อมูลแค่ของประเทศไทย

Insufficient Data

แนวทางแก้ไขเมื่อข้อมูลไม่พอ



Sampling: Technical Terms

คำศัพท์ที่ต้องรู้เกี่ยวกับการเลือกตัวอย่างจากประชากร

Definitions

Population

ประชากร

Sample

กลุ่มตัวอย่าง

Margin of Error (MOE)

ขอบเขตความคลาดเคลื่อน

Confidence Level

ระดับความเชื่อมั่น

Confidence Interval (CI)

ช่วงความเชื่อมั่น

Statistical significance

นัยสำคัญทางสถิติ

ทุกคนจากทุกกลุ่มในโรงเรียนที่จะศึกษา

ส่วนหนึ่ง (Subset) ของประชากรที่จะศึกษา

ความคลาดเคลื่อนของผลลัพธ์
กลุ่มตัวอย่างจากค่าจริงของประชากร

ความมั่นใจในความแม่นยำของผลลัพธ์
ที่คาดหวัง

ช่วงของผลลัพธ์ที่เราคาดว่าค่าจริงของ
ประชากรจะอยู่ในช่วงนี้

การสรุปว่าผลลัพธ์ของเราเกิดจากความ
บังเอิญเฉยๆ หรือไม่

Examples

ต้องการออกโรงเต๋าวิ่งสำหรับคนอายุ 20-35 ปี

Population : คนไทยอายุ 20-35 ปีทุกคนที่ชอบวิ่ง

Sample : สุ่มเลือกคนไทยอายุ 20-35 ปีที่ชอบวิ่งมา
ทำแบบสำรวจออนไลน์ 400 คน

กลุ่มตัวอย่าง 80% บอกว่าชอบโรงเต๋าวิ่ง ถ้า MOE = 5%
แปลว่าประชากรอาจชอบโรงเต๋าวิ่ง 75%-85%

คาดหวัง Confidence level = 95% แปลว่าถ้าทำการสำรวจ 100 ครั้ง
จะมีราว ๆ 95 ครั้งที่ได้ผลลัพธ์ใกล้เคียงกัน

เรามั่นใจ 95% ว่าประชากรจะชอบโรงเต๋าวิ่งในช่วง 75%-85%
แปลว่า CI = 75%-85%

ถ้าผู้ชายชอบโรงเต๋าวิ่งสีแดง 20% ส่วนผู้หญิงชอบ 25% แต่ถ้า MOE = 5%
แปลว่าความต่างไม่ได้มีนัยสำคัญทางสถิติเพราะช่วง 15-25% (ชาย) กับ
20-30% (หญิง) คาบเกี่ยวกัน

Sample Size and Margin of Error

- ขนาดกลุ่มตัวอย่าง (Sample size) เล็กที่สุดที่พอรับได้ในทางทฤษฎีคือ 30 คน
- เราคำนวณ Sample size (n) ที่เหมาะสมได้จาก Population size (N), Confidence Level (CL) และ Margin of Error (MOE) ที่คาดหวัง เช่น ถ้าเราประมาณว่าคนไทยช่วงอายุ 25-40 ปีที่ชอบวิ่งคือ 300,000 คน (สมมติ) คาดหวังความมั่นใจ 95% คาดเคลื่อนไม่เกิน 5% แปลว่าเราต้องใช้กลุ่มตัวอย่างอย่างน้อย 384 คน
[Note: ในคอร์สมีไฟล์ Excel สำหรับใช้คำนวณให้แบบสำเร็จรูป]

Population Size	Confidence Level (%)	Margin of Error (%)
300,000	95	5
Sample Size		
384		

Population Size	Confidence Level (%)	Sample Size
300,000	95	384
Margin of Error (%)		
5		

ในทางกลับกัน เราหา MOE ได้จาก N , CL และ n

- ยิ่งต้องการความมั่นใจสูงขึ้น กลุ่มตัวอย่างต้องใหญ่ขึ้น กลับกัน หากกลุ่มตัวอย่างเล็กลง ความคลาดเคลื่อนก็จะสูงขึ้น

Population Size	Confidence Level (%)	Margin of Error (%)
300,000	99	5
Sample Size		
665		

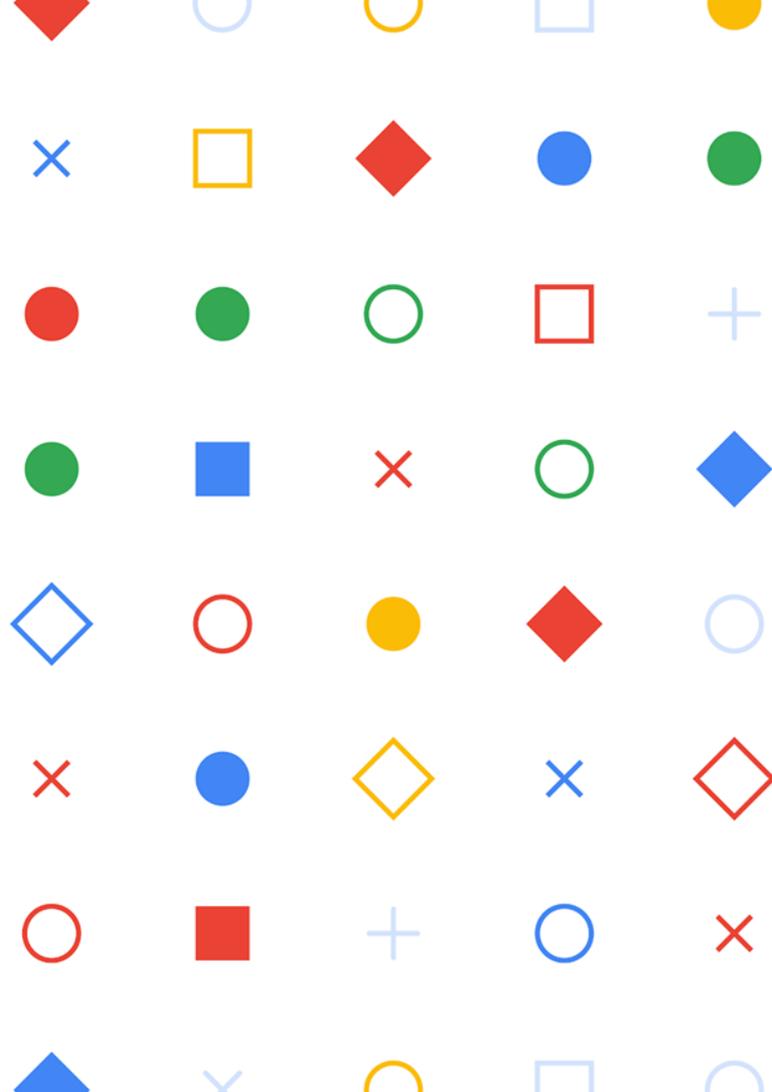
ข้อสังเกตอีกอย่างคือ CL + MOE ไม่จำเป็นต้องเท่ากับ 100%

Population Size	Confidence Level (%)	Sample Size
300,000	95	30
Margin of Error (%)		
17.9		

Sparkling-clean Data

— *Part2:*

1. Five Types of Dirty Data
2. Data Team to Ensure Data Integrity
3. Data Merging
4. Cleaning Data with Spreadsheets



5 Types of Dirty Data

ข้อมูลที่จัดว่าสกปรก คือข้อมูลที่ไม่ครบถ้วน (Incomplete), ผิด (Incorrect) หรือไม่เกี่ยวข้องกับโจทย์ที่จะแก้ (Irrelevant) ซึ่งอาจมีลักษณะ 5 แบบดังต่อไปนี้



Duplicate data

ข้อมูลแถวซ้ำ เช่น คนๆ เดียวกันแต่ตอบแบบสอบถามสองรอบ



Outdated data

ข้อมูลที่ล้าสมัย เช่น เบอร์โทรเก่าที่ตอนนี้เปลี่ยนไปแล้ว ข้อมูลพนักงานที่ลาออกไปแล้ว



Incomplete

ข้อมูลไม่ครบถ้วนหรือสูญหายเป็น “ค่าว่าง” (Null/NaN/NA) เช่น คนตอบแบบสอบถามลืมตอบคำถามในหน้าสุดท้าย



Incorrect/inaccurate data

ข้อมูลผิดจากความจริง เช่น พิมพ์อายุผิดจาก 31 มีอีลันกลายเป็น 321



Inconsistent data

ข้อมูลของสิ่งเดียวกันแต่กลับมีค่าไม่เท่ากันหรือรูปแบบไม่ตรงกัน เช่น วันเกิดในฐานข้อมูลหนึ่งเป็น 01-07-1991 (MM-DD-YYYY) แต่อีกฐานข้อมูลเป็น 07-01-1991 (DD-MM-YYYY)

Data Team to Ensure Data Integrity

ในทีม Data จะมีหลักๆ อยู่ 2 ตำแหน่งที่ช่วยเพิ่ม Data Integrity ให้กับองค์กรคือ

Data Engineers

วิศวกรข้อมูล

- แปลงข้อมูลในฐานข้อมูลให้มีรูปแบบที่ทีมนำไปใช้ต่อได้ง่าย
- สร้างโครงสร้างพื้นฐานที่เชื่อถือได้ (Reliable infrastructure) ให้ข้อมูล
- พัฒนา (Develop) ดูแลรักษา (Maintain) และทดสอบ (Test) ฐานข้อมูลและระบบที่เกี่ยวข้อง

Data Warehousing Specialists

ผู้เชี่ยวชาญระบบคลังข้อมูล

- พัฒนาระบบการและระเบียบการ (Processes and procedures) ในการจัดเก็บและจัดการข้อมูล
- ทำให้มั่นใจว่าข้อมูลคงอยู่ (Available) ปลอดภัย (Secure) และมีสำรอง (Backed up) เพื่อป้องกันการสูญหาย

Note :อย่างไรก็ตาม Data Analysts ก็ยังต้องตรวจสอบความสะอาดของข้อมูลเองอยู่ดี

โดยเฉพาะกรณีข้อมูลมาจากภายนอกองค์กร (External data)

Data Merging

บางครั้งเราต้องรวมข้อมูลจากหลายแหล่งมาอยู่ในตารางเดียวกัน ซึ่งการทำเช่นนี้ได้ ข้อมูลจากแต่ละแหล่งจะต้องเข้ากันได้ (Compatible) ซึ่งสิ่งที่เราต้องตรวจสอบคือ

- ข้อมูลเดียวกัน ต้องมีค่า รูปแบบที่ตรงกัน (Consistent)
- ชื่อคอลัมน์ของข้อมูลต้องปรับให้ตรงกัน

Company A					
Name	Age	Height	Gender	Birthday	Nationality
Great	31	172	M	1991-01-07	Thai
Cherprang	26	160	F	1996-05-02	Thai
Jennie	26	163	F	1996-01-16	Korean
Bambam	25	178	M	1997-05-02	Thai
Sergey	48	173	M	1973-06-21	American
Larry	49	181	M	1973-03-26	American

Company B					
name	age	height	sex	birthday	nation
Great	31	1.72	Male	1991-01-07	TH
Cherprang	26	1.6	Female	1996-05-02	TH
Jisoo		1.62	Female	1995-01-03	KR
Elon	50	1.87	Male	1971-06-28	



Company A+B					
Name	Age	Height	Gender	Birthday	Nationality
Bambam	25	178	M	1997-05-02	Thai
Cherprang	26	160	F	1996-05-02	Thai
Jennie	26	163	F	1996-01-16	Korean
Jisoo		162	F	1995-01-03	Korean
Great	31	172	M	1991-01-07	Thai
Sergey	48	173	M	1973-06-21	American
Larry	49	181	M	1973-03-26	American
Elon	50	187	M	1971-06-28	

Cleaning Data with Spreadsheets

Conditional formatting

ปรับผลการแสดงเช่นสีช่องตามเงื่อนไขของค่าข้อมูล

ประโยชน์

- ช่วยทำให้ข้อมูลที่สนใจหรือแปลกประหลาดเด่นขึ้น เช่น ค่าว่าง (Blank/Null)
- ช่วยให้เห็น Pattern ของข้อมูล เช่น กำไรสีเขียว ขาดทุนสีแดง

Ex. เปลี่ยนให้ช่องที่แสดงอายุเกิน 30 ปีเป็นสีแดง

Conditional format rules

Single color | Color scale

Apply to range: B1:B9

Format rules: Greater than 30

Formatting style: Custom (Red)

Cancel Done

+ Add another rule

1 2 3

Result

	A	B	C	D	E
1	Name	Age	Height	Gender	Birthday
2	Bambam	25	178	M	1997-05-02
3	Cherprang	26	160	F	1996-05-02
4	Elon	50	187	M	1971-06-28
5	Great	31	172	M	1991-01-07
6	Jennie	26	163	F	1996-01-16
7	Jisoo		162	F	1995-01-03
8	Larry	49	181	M	1973-03-26
9	Sergey	48	173	M	1973-06-21

Cleaning Data with Spreadsheets

Remove duplicates

ลบแถวที่มีข้อมูลซ้ำกันออก

เมื่อไหร่ต้องลบแถวซ้ำ?

- คน ๆ เดียวกัน ตอบแบบสอบถามมาหลายครั้ง ถ้าจะคำนวณค่าทางสถิติ อาจทำให้ตีความคลาดเคลื่อนได้
- แต่ต้องระวังว่า บางที่เราอาจไม่ต้องลบแถวซ้ำ เช่น ในตารางบันทึกรายการซื้อของ คน ๆ เดียวกัน ชื่อของสิ่งเดียวกัน แต่ซื้อซ้ำหลายที แล้วตารางไม่ได้มีการระบุหมายเลขการสั่งซื้อหรือบันทึกเวลาที่แตกต่างกันในการซื้อแต่ละครั้ง

New Members of Our Company (2022)

File Edit View Insert Format Data Tools Extensions Help Last edit was

Sort sheet
Sort range
Create a filter
Filter views
Add a slicer
Protect sheets and ranges
Named ranges
Randomize range
Column stats
Data validation
Data cleanup
Split text to columns
Data connectors

Remove duplicates

12 rows and 6 columns selected

Data has header row

Columns to analyze

Select all

Column A - Name

Column B - Age

Column C - Height

Column D - Gender

Column E - Birthday

Column F - Nationality

Cancel Remove duplicates

Cleanup suggestions

Remove duplicates

Trim whitespace

Result

	A	B	C	D	E	F
1	Name	Age	Height	Gender	Birthday	Nationality
2	Bam Bam	25	178	M	1997-05-02	Thai
3	Cherprang	26	160	F	1996-05-02	Thai
4	Elon	50	187	M	1971-06-28	
5	Great	31	172	M	1991-01-07	Thai
6	Jennie	26	163	F	1996-01-16	Korean
7	Jisoo		162	F	1995-01-03	
8	Larry	49	181	M	1973-03-26	American
9	Sergey	48	173	M	1973-06-21	American

Cleaning Data with Spreadsheets

Split text to columns

แยกข้อมูลที่คั่นด้วยตัวคั่นออกเป็นคอลัมน์

Note: ตัวคั่น (Delimiter/Separator) ที่ Google Sheets มีให้แล้วคือ

- Comma ,
- Semicolon ;
- Period .
- Space (ช่องว่าง)

ถ้านอกเหนือจากนี้ ต้องระบุเอง เช่น

- Dash -
- Pipe |

Ex. แยกคอลัมน์ Birthday ออกเป็น ปี เดือน วัน อย่างละคอลัมน์

New Members of Our Company (2022) ☆ 📁 🗑️
File Edit View Insert Format Data Tools Extensions Help Last edit was sa

100% \$ % .0 ↕ Sort sheet
↕ Sort range

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

B1:B9 Birthday

Birthday

1997-05-02
1996-05-02
1971-06-28
1991-01-07
1996-01-16
1995-01-03
1973-03-26
1973-06-21

Create a filter
Filter views
Add a slicer
Protect sheets and ranges
Named ranges
Randomize range
Column stats
Data validation
Data cleanup
Split text to columns
Data connectors

Separator: Detect automatically
Comma
Semicolon
Period
Space
Custom

Birthday	C	D	E
1997	5	2	
1996	5	2	
1971	6	28	
1991	1	7	
1996	1	16	
1995	1	3	
1973	3	26	
1973	6	21	

Separator: Custom -

Result

Year	Month	Day
1997	5	2
1996	5	2
1971	6	28
1991	1	7
1996	1	16
1995	1	3
1973	3	26
1973	6	21

Cleaning Data with Spreadsheets

COUNTIF

นับจำนวนช่องถ้าค่าในช่องตรงกับเงื่อนไขที่ต้องการ

Syntax:

```
=COUNTIF(range, "criterion")
```

Notes:

- เงื่อนไข (Criterion) ต้องใส่เครื่องหมาย quotes คร่อม
- มีฟังก์ชันที่คล้ายๆ กันเช่น
 - COUNTIFS: ใส่ได้หลายเงื่อนไข
 - SUMIF: เปลี่ยนจากนับช่องเป็นหาผลรวม

<https://support.google.com/docs/table/25273?hl=en>

Ex. นับจำนวนคนที่มีอายุเกิน 30 ปี

	A	B	C	D	E	F
1	Name	Age	Height	Gender	Birthday	Natio
2	Bambam	25	178	M	1997-05-02	Th
3	Cherprang	26	160	F	1996-05-02	Th
4	Elon	50	187	M	1971-06-28	
5	Great	31	172	M	1991-01-07	Th
6	Jennie	26	163	F	1996-01-16	Kor
7	Jisoo		162	F	1995-01-03	
8	Larry	49	181	M	1973-03-26	Ame
9	Sergey	48	173	M	1973-06-21	Ame
10						
11	Number of people with Age > 30					
12						
13						
14						
15						

4 x

```
=COUNTIF(B2:B9, ">30")
```

COUNTIF(range, criterion)

Cleaning Data with Spreadsheets

LEN

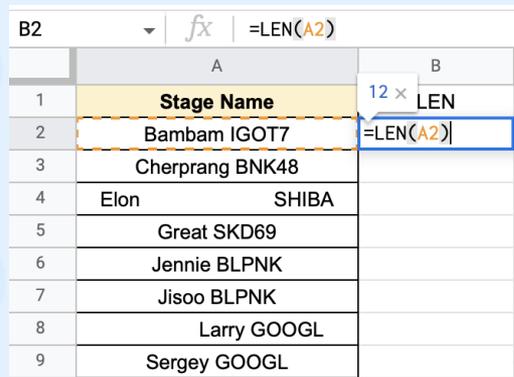
หาจำนวนตัวอักษรและช่องว่าง (ถ้ามี) ของข้อความ

Syntax:

```
=LEN(text)
```

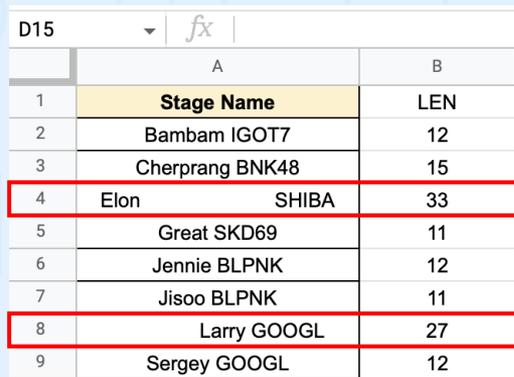
Note: สังเกตว่า ช่องว่าง (Whitespaces) ถ้ามีก็จะถูกนับด้วย โดยนับแต่ละช่องว่าง เช่น ถ้าเคาะ Spacebar 10 ครั้ง ก็จะนับเป็น 10 ตัวอักษร

Ex. นับจำนวนตัวอักษรของชื่อในวงการ



	A	B
1	Stage Name	LEN
2	Bambam IGOT7	=LEN(A2)
3	Cherprang BNK48	
4	Elon SHIBA	
5	Great SKD69	
6	Jennie BLPNK	
7	Jisoo BLPNK	
8	Larry GOOGL	
9	Sergey GOOGL	

Result



	A	B
1	Stage Name	LEN
2	Bambam IGOT7	12
3	Cherprang BNK48	15
4	Elon SHIBA	33
5	Great SKD69	11
6	Jennie BLPNK	12
7	Jisoo BLPNK	11
8	Larry GOOGL	27
9	Sergey GOOGL	12

Cleaning Data with Spreadsheets

TRIM

ตัดช่องว่างส่วนเกินทิ้งให้เหลือที่จำเป็น

Syntax:

`=TRIM(text)`

Note: ฟังก์ชัน TRIM จะทำการตัด

- ช่องว่างด้านหน้า (Leading spaces) ออกทั้งหมด
- ช่องว่างด้านหลัง (Trailing spaces) ออกทั้งหมด
- ช่องว่างระหว่างข้อความ เหลือแค่ช่องเดียว

Ex. ตัดช่องว่างส่วนเกินทิ้งจากชื่อในวงการ

	A	B
1	Stage Name	Trimmed Name
2	Bambam IGOT7	
3	Cherprang BNK48	
4	Elon SHIBA	<code>=TRIM(A4)</code>
5	Great SKD69	
6	Jennie BLPNK	
7	Jisoo BLPNK	
8	Larry GOOGL	
9	Sergey GOOGL	

Result

	A	B
1	Stage Name	Trimmed Name
2	Bambam IGOT7	Bambam IGOT7
3	Cherprang BNK48	Cherprang BNK48
4	Elon SHIBA	Elon SHIBA
5	Great SKD69	Great SKD69
6	Jennie BLPNK	Jennie BLPNK
7	Jisoo BLPNK	Jisoo BLPNK
8	Larry GOOGL	Larry GOOGL
9	Sergey GOOGL	Sergey GOOGL

Cleaning Data with Spreadsheets

LEFT / RIGHT

ดึงตัวอักษรจากทางซ้ายหรือขวาตามจำนวนที่ต้องการ

Syntax:

=LEFT(text, จำนวนตัวอักษรที่ต้องการ)

=RIGHT(text, จำนวนตัวอักษรที่ต้องการ)

Ex. ดึงชื่อวงซึ่งเป็นตัวอักษร 5 ตัวทำออกจากชื่อในวงการ

	A	B
1	Stage Name	
2	Bambam IGOT7	=RIGHT(A2, 5)
3	Cherprang BNK48	
4	Elon SHIBA	
5	Great SKD69	
6	Jennie BLPNK	
7	Jisoo BLPNK	
8	Larry GOOGL	
9	Sergey GOOGL	

Result

	A	B
1	Stage Name	BAND
2	Bambam IGOT7	IGOT7
3	Cherprang BNK48	BNK48
4	Elon SHIBA	SHIBA
5	Great SKD69	SKD69
6	Jennie BLPNK	BLPNK
7	Jisoo BLPNK	BLPNK
8	Larry GOOGL	GOOGL
9	Sergey GOOGL	GOOGL

Cleaning Data with Spreadsheets

MID

ดึงส่วนหนึ่งของข้อความ (Substring)

Syntax:

`=MID(text, ตำแหน่งเริ่มต้น, จำนวน
ตัวอักษรที่ต้องการ)`

Ex. ดึงเดือนออกจากวันที่

B2	A	B
	1 Birthday	05 x Month
	2 1997-05-02	=MID(A2,6,2)
	3 1996-05-02	
	4 1971-06-28	
	5 1991-01-07	
	6 1996-01-16	
	7 1995-01-03	
	8 1973-03-26	
	9 1973-06-21	

123456789 10
| | | | | | | | | |
1997-05-02

2 ตัวอักษร

Result

	A	B
1	Birthday	Month
2	1997-05-02	05
3	1996-05-02	05
4	1971-06-28	06
5	1991-01-07	01
6	1996-01-16	01
7	1995-01-03	01
8	1973-03-26	03
9	1973-06-21	06

Cleaning Data with Spreadsheets

FIND

หาเลขตำแหน่งของสิ่งที่ต้องการค้นหาในข้อความ

Syntax:

```
=FIND("สิ่งที่ต้องการค้นหา", text,  
ตำแหน่งเริ่มต้นในการค้นหา [Optional])
```

Note: ตัวพิมพ์เล็กหรือพิมพ์ใหญ่มีผลต่อการค้นหา (Case-sensitive) เช่น ถ้าหา "a" ใน "GREAT" จะหาไม่เจอ

Ex. หาตำแหน่งของตัวอักษร "a" ในชื่อในวงการ

	A	B
1	Stage Name	
2	Bambam IGOT7	=FIND("a",A2)
3	Cherprang BNK48	
4	Elon SHIBA	
5	Great SKD69	
6	Jennie BLPNK	
7	Jisoo BLPNK	
8	Larry GOOGL	
9	Sergey GOOGL	

Result

	A	B
1	Stage Name	FIND "a"
2	Bambam IGOT7	2
3	Cherprang BNK48	7
4	Elon SHIBA	#VALUE!
5	Great SKD69	4
6	Jennie BLPNK	#VALUE!
7	Jisoo BLPNK	#VALUE!
8	Larry GOOGL	2
9	Sergey GOOGL	#VALUE!

← ทำ "a" ไม่เจอ ("a" ไม่เท่ากับ "A")



Cleaning Data with Spreadsheets

CONCATENATE

นำข้อความย่อยมาเชื่อมกันเป็นข้อความใหญ่

Syntax:

=CONCATENATE(text1, text2, ...)

Note: ฟังก์ชัน CONCATENATE เป็นการเอาข้อความย่อยมาเชื่อมกันที่เรื่อยๆ จะไม่มีช่องว่างคั่นกลางให้ ถ้าอยากให้มีต้องใส่ช่องว่างคั่นกลางเอง (อย่าลืมเครื่องหมาย quotes ครอบช่องว่างที่อยากใส่ด้วย เพราะต้องใส่ค่าในรูปแบบ string)

<https://support.google.com/docs/table/25273?hl=en>

Ex. รวมชื่อจริงกับชื่อวงเข้าด้วยกัน

	A	B	C
1	Name	Band	BambamIGOT7 ×
2	Bambam	IGOT7	=CONCATENATE(A2,B2)
3	Cherprang	BNK48	
4	Elon	SHIBA	
5	Great	SKD69	
6	Jennie	BLPNK	
7	Jisoo	BLPNK	
8	Larry	GOOGL	
9	Sergey	GOOGL	



C
Stage Name
BambamIGOT7
CherprangBNK48
ElonSHIBA
GreatSKD69
JennieBLPNK
JisooBLPNK
LarryGOOGL
SergeyGOOGL

	A	B	C
1	Name	Band	Bambam IGOT7 ×
2	Bambam	IGOT7	=CONCATENATE(A2, " ", B2)
3	Cherprang	BNK48	
4	Elon	SHIBA	
5	Great	SKD69	
6	Jennie	BLPNK	
7	Jisoo	BLPNK	
8	Larry	GOOGL	
9	Sergey	GOOGL	



C
Stage Name
Bambam IGOT7
Cherprang BNK48
Elon SHIBA
Great SKD69
Jennie BLPNK
Jisoo BLPNK
Larry GOOGL
Sergey GOOGL

Cleaning Data with Spreadsheets

VLOOKUP

หาข้อมูลในแนวดิ่ง (Vertical look-up) ถ้าเจอค่าที่ต้องการ ให้เอาค่าในคอลัมน์ทางขวามาเติมใส่ตารางอันใหม่

Syntax:

`=VLOOKUP(ค่าค้นหา, data, หมายเลขคอลัมน์ที่ต้องการเอาค่ามาใส่ตารางใหม่, FALSE)`

Notes:

- FALSE อันสุดท้ายจำเป็นต้องใส่เพื่อบอกว่า เราไม่ต้องการให้โปรแกรมค้นหาค่าใกล้เคียง แต่ต้องการให้ค้นหาค่าที่เหมือนเป๊ะตามค่าค้นหา
- ไม่สามารถเอาค่าทางซ้ายของค่าค้นหามาได้

<https://support.google.com/docs/table/25273?hl=en>

Ex. เอาปีที่ Debut ของวงมาใส่ในตารางอีกอัน

ใช้ Absolute Reference เพื่อใช้ data เดิมในการค้นหา

`=VLOOKUP(E3, A3:B6, 2, FALSE)`

	A	B	C	D	E	F	G
1							
2	Band	Debut		Name	Band	2014 × but	
3	IGOT7	2014		Bambam	IGOT7	=VLOOKUP(E3, \$A\$3:\$B\$6, 2, FALSE)	
4	BNK48	2017		Cherprang	BNK48		
5	SKD69	2016		Elon	SHIBA		
6	BLPNK	2016		Great	SKD69		
7				Jennie	BLPNK		
8	คอลัมน์ 1	คอลัมน์ 2		Jisoo	BLPNK		
9				Larry	GOOGL		
10				Sergey	GOOGL		

Result

Name	Band	Debut
Bambam	IGOT7	2014
Cherprang	BNK48	2017
Elon	SHIBA	#N/A
Great	SKD69	2016
Jennie	BLPNK	2016
Jisoo	BLPNK	2016
Larry	GOOGL	#N/A
Sergey	GOOGL	#N/A

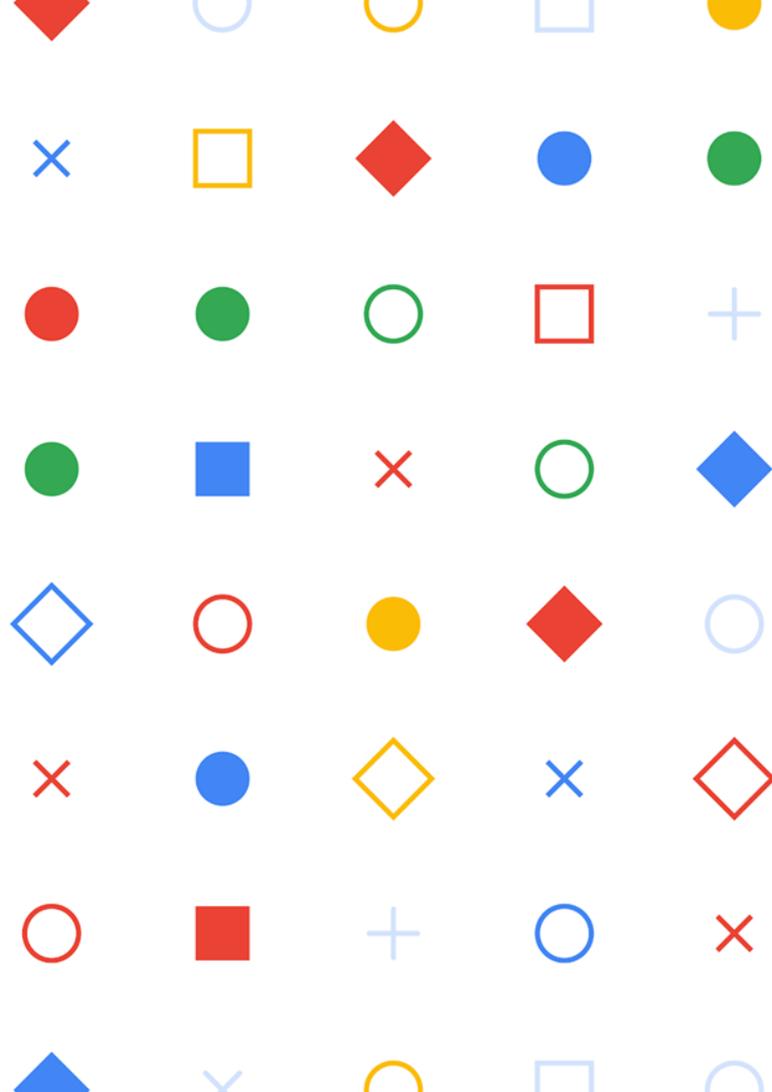
← หาค่าไม่เจอ



Cleaning Data with SQL

— *Part3:*

1. Why SQL?
2. SQL: Play with Tables
3. SQL: Text Functions
4. SQL: Advanced Data Cleaning Functions



Why SQL?

สาเหตุที่ภาษา SQL เป็นที่นิยมในการใช้ query ข้อมูลจาก Databases คือ

- ฐานข้อมูลส่วนใหญ่ไม่ว่าของบริษัทใด มักสามารถใช้ภาษา SQL แบบมาตรฐาน (Standard SQL) ได้เลย หรืออย่างมากที่สุดก็ปรับ syntax เพียงเล็กน้อย
- ทำงานกับชุดข้อมูลขนาดใหญ่มากได้ และประมวลผลได้อย่างรวดเร็ว
- มีฟังก์ชันในการทำความสะอาดข้อมูลมากมาย เพื่อเตรียมข้อมูลสำหรับนำไปวิเคราะห์ต่อในโปรแกรมอื่น เช่น Excel, Python
- มีสูตร (Formulas) และฟังก์ชัน (Functions) ในการคำนวณทางคณิตศาสตร์ (Arithmetic) รวมถึงเชื่อมข้อมูลจากหลายแหล่งเข้าด้วยกัน (Join data) เหมือนกับของ Spreadsheets

SQL: Play with Tables

ดึงข้อมูลที่มีอยู่ในฐานข้อมูลแล้วมาใช้

Syntax:

SELECT	รายชื่อคอลัมน์ที่ต้องการ
FROM	ชื่อตาราง
WHERE	เงื่อนไข (Optional)
LIMIT	จำนวนแถวที่ต้องการ (Optional)

Ex. ดึงคอลัมน์ทั้งหมดของตาราง car_info โดยเอามาแค่ 5 แถวแรกพอ

The screenshot shows a SQL query editor interface. On the left, there is a sidebar with a search bar and a list of pinned projects. The 'cars' project is expanded, showing the 'car_info' table selected. The main editor area contains the following SQL query:

```
1 SELECT *
2 FROM linejenniebot01-hfyb.cars.car_info
3 LIMIT 5
```

Below the query editor, there are buttons for 'RUN', 'SAVE', 'SHARE', and 'SCHEDULE'. The 'Query results' section is visible, showing a table with the following data:

Row	make	fuel_type	num_of_doors	body_style	driv
1	alfa-romero	gas	two	convertible	rwc
2	alfa-romero	gas	two	convertible	rwc
3	alfa-romero	gas	two	hatchback	rwc
4	audi	gas	four	sedan	fw
5	audi	gas	four	sedan	4w

SQL: Play with Tables

เลือกแถวแบบไม่ซ้ำด้วย DISTINCT

Syntax:

```
SELECT DISTINCT ชื่อคอลัมน์ที่ต้องการ  
FROM ชื่อตาราง
```

Ex. ขอดูชื่อหัวและประเภทเครื่องยนต์ที่ไม่ซ้ำกัน

```
1 SELECT make, engine_type  
2 FROM linejenniebot01-hfyb.cars.car_info
```

Query results

	JOB INFORMATION	RESULTS	JSON
Row	make	engine_type	
1	alfa-romero	dohc	
2	alfa-romero	dohc	
3	alfa-romero	ohcv	
4	audi	ohc	
5	audi	ohc	
6	audi	ohc	
7	audi	ohc	
8	audi	ohc	
9	audi	ohc	
10	audi	ohc	
11	bmw	ohc	
12	bmw	ohc	

```
1 SELECT DISTINCT make, engine_type  
2 FROM linejenniebot01-hfyb.cars.car_info
```

Query results

	JOB INFORMATION	RESULTS	JSON
Row	make	engine_type	
1	alfa-romero	dohc	
2	alfa-romero	ohcv	
3	audi	ohc	
4	bmw	ohc	
5	chevrolet	l	
6	chevrolet	ohc	
7	dodge	ohc	
8	honda	ohc	
9	isuzu	ohc	
10	jaguar	dohc	
11	jaguar	ohcv	
12	mazda	ohc	

SQL: Play with Tables

เพิ่มข้อมูลใหม่เข้าไปในตารางที่มีอยู่แล้ว

Syntax:

```
INSERT INTO ชื่อตาราง
      (List รายชื่อคอลัมน์ คั่นด้วย comma)
VALUES
      (List ค่าที่ต้องการใส่ ตรงกับชื่อคอลัมน์)
```

Note: หากใช้ BigQuery เวอร์ชัน Sandbox จะไม่สามารถเพิ่มข้อมูลใหม่ด้วย INSERT INTO ได้

Ex. เพิ่มแถวข้อมูลเข้าไปในตารางรายการสั่งซื้อสินค้า

```
▶ RUN  📄 SAVE ▾  👤 SHARE ▾  ⌚ SCHEDULE ▾  ⚙️ MORE ▾
1  INSERT INTO projectonlineshop.transactions.txn_202206
2  | (order_id, customer_id, order_date, product_id, amount)
3  VALUES
4  | ("A0005", "C1234", "2022-06-03", "P4234", 30000)
```

SQL: Play with Tables

แก้ไขข้อมูลในตาราง

Syntax:

UPDATE	ชื่อตาราง
SET	ชื่อคอลัมน์ = ค่าใหม่
WHERE	เงื่อนไขระบุแถวที่จะแก้ไขค่า

Note: หากใช้ BigQuery เวอร์ชัน Sandbox จะไม่สามารถแก้ไขข้อมูลด้วย UPDATE ได้

Ex. แก้ไขข้อมูลปริมาณการสั่งซื้อ

```
▶ RUN  📄 SAVE ▾  👤 SHARE ▾  ⌚ SCHEDULE ▾  
1 UPDATE projectonlineshop.transactions.txn_202206  
2 SET amount = 50000  
3 WHERE order_id = "A0005"
```

SQL: Play with Tables

สร้างตารางใหม่

Syntax:

```
CREATE TABLE IF NOT EXISTS ชื่อตาราง  
(List รายชื่อคอลัมน์และประเภทข้อมูล)
```

หากสนใจอ่านเพิ่มเติมได้ที่:

https://cloud.google.com/bigquery/docs/reference/standard-sql/data-definition-language#creating_a_table_only_if_the_table_doesnt_exist

Ex. สร้างตารางรายได้ไตรมาสที่สองของปี

```
▶ RUN  📄 SAVE ▾  👤 SHARE ▾  ⌚ SCHEDULE ▾  ⚙️ MORE ▾  
1 CREATE TABLE IF NOT EXISTS projectonlineshop.transactions.revenue_q2  
2 | (txn_dt DATETIME, revenue FLOAT64)
```

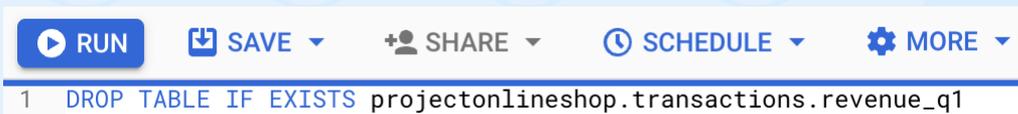
SQL: Play with Tables

ลบตารางที่ไม่ใช้

Syntax:

```
DROP TABLE IF EXISTS ชื่อตาราง
```

Ex. ลบตารางรายได้ไตรมาสแรกออกเพราะไม่ใช่แล้ว



The screenshot shows a SQL editor interface with a toolbar at the top containing buttons for RUN, SAVE, SHARE, SCHEDULE, and MORE. Below the toolbar, a single line of SQL code is entered: `1 DROP TABLE IF EXISTS projectonlineshop.transactions.revenue_q1`.

SQL: Text Functions

SQL มีฟังก์ชันสำหรับข้อมูลข้อความที่คล้ายกับ Spreadsheets แต่อาจมี syntax หรือผลลัพธ์ที่ต่างกันเล็กน้อย



Spreadsheets



SQL

หาจำนวนตัวอักษร

=LEN(text)

LENGTH(คอลัมน์)

ตัดช่องว่างส่วนเกินออก

=TRIM(text)

*ตัดช่องว่างระหว่างคำให้เหลือ 1 หน่วยให้ด้วย

TRIM(คอลัมน์)

*ตัดเฉพาะช่องว่างนำหน้าและตามหลังเท่านั้น

เลือกมาเฉพาะบางส่วนของข้อความ (Substring)

=MID(text, ตำแหน่งเริ่มต้น, จำนวนตัวอักษรที่ต้องการ)

SUBSTR(คอลัมน์, ตำแหน่งเริ่มต้น, จำนวนตัวอักษรที่ต้องการ)

นำข้อความย่อยมาเชื่อมเป็นข้อความใหญ่

=CONCATENATE(text1, " ", text2, ...)

CONCATENATE(คอลัมน์, " ", คอลัมน์,...) AS ชื่อคอลัมน์ใหม่

SQL: Advanced Data Cleaning Functions

เปลี่ยนประเภทข้อมูล

Syntax:

```
CAST(ชื่อคอลัมน์ AS ประเภทข้อมูลใหม่)
```

Note: ตัวอย่างประเภทข้อมูลที่ใช้บ่อยในเบื้องต้น

- STRING
- INT64, FLOAT64
- DATE, DATETIME, TIME, TIMESTAMP
- BOOL

ดูประเภทข้อมูลทั้งหมดได้ที่:

<https://cloud.google.com/bigquery/docs/reference/standard-sql/data-types>

Ex. แก้ไขข้อมูลวันสั่งซื้อ จากข้อความ (STRING) เป็น วัน (DATE) และ
ข้อมูลจำนวนสั่งซื้อ จากจำนวนเต็ม (INT64) เป็นจำนวนทศนิยม

```
▶ RUN  ⏴ SAVE  + SHARE  ⌚ SCHEDULE
```

```
1 SELECT
2   order_id,
3   customer_id,
4   CAST(order_date AS DATE),
5   product_id,
6   CAST(amount AS FLOAT64)
7 FROM projectonlineshop.transactions.txn_202206
```

SQL: Advanced Data Cleaning Functions

เรียงลำดับข้อมูล

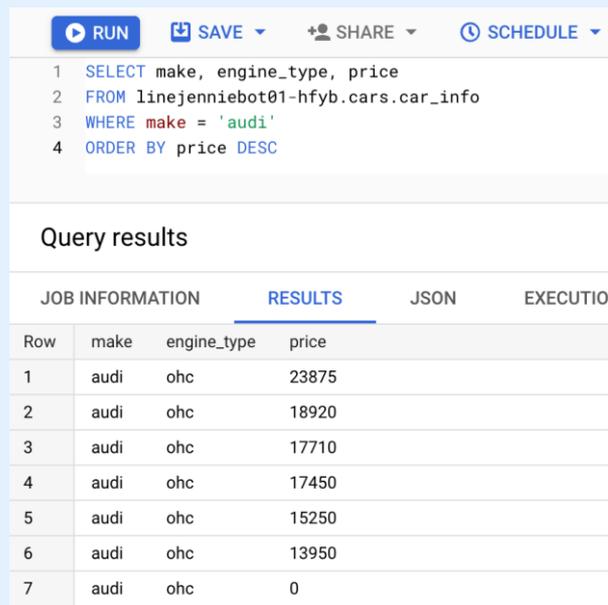
Syntax: เรียงจากน้อยไปมาก (Ascending)

SELECT รายชื่อคอลัมน์ที่ต้องการ
FROM ชื่อตาราง
ORDER BY ชื่อคอลัมน์

Syntax: เรียงจากมากไปน้อย (Descending)

SELECT รายชื่อคอลัมน์ที่ต้องการ
FROM ชื่อตาราง
ORDER BY ชื่อคอลัมน์ **DESC**

Ex. เรียงลำดับราคารถยนต์ยี่ห้อ audi จากมากไปน้อย (Descending)



The screenshot shows a SQL query editor interface with a 'RUN' button and options for 'SAVE', 'SHARE', and 'SCHEDULE'. The query is as follows:

```
1 SELECT make, engine_type, price
2 FROM linejenniebot01-hfyb.cars.car_info
3 WHERE make = 'audi'
4 ORDER BY price DESC
```

Below the query, the 'Query results' section is displayed with a table showing the results of the query. The table has columns for 'Row', 'make', 'engine_type', and 'price'. The results are sorted in descending order of price.

Row	make	engine_type	price
1	audi	ohc	23875
2	audi	ohc	18920
3	audi	ohc	17710
4	audi	ohc	17450
5	audi	ohc	15250
6	audi	ohc	13950
7	audi	ohc	0

SQL: Advanced Data Cleaning Functions

เลือกข้อมูลระหว่างสองค่า

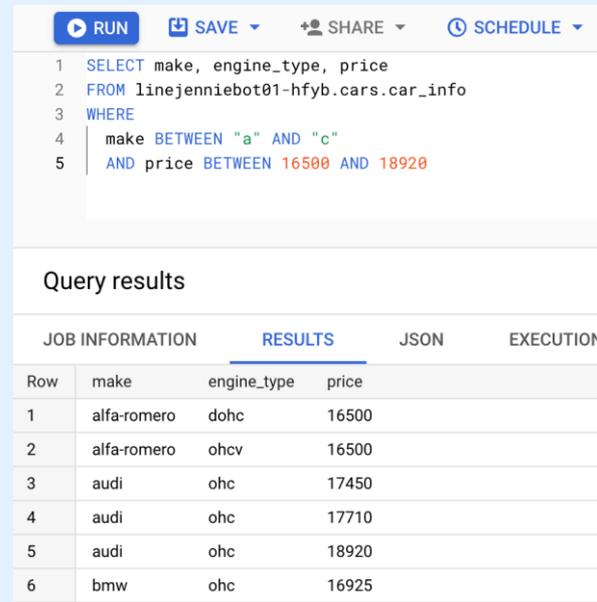
Syntax:

```
SELECT   รายชื่อคอลัมน์ที่ต้องการ
FROM     ชื่อตาราง
WHERE    ชื่อคอลัมน์ BETWEEN ขอบล่าง AND ขอบบน
```

Notes:

- เอาค่าที่เท่ากับขอบล่างและขอบบนพอดีเปะมาด้วย (Inclusive)
- ค่าที่นำมากรองไม่จำเป็นต้องเป็นตัวเลข แต่ต้องเปรียบเทียบลำดับได้ เช่น
 - วันเวลา (BETWEEN ก่อน AND หลัง)
 - ข้อความ (ตามลำดับใน "พจนานุกรม")

Ex. เลือกดูรถยนต์ที่อยู่ระหว่างตัวอักษร a กับ c และราคาระหว่าง 16,500 ถึง 18,920



The screenshot shows a SQL query interface with a 'RUN' button and options for 'SAVE', 'SHARE', and 'SCHEDULE'. The query is as follows:

```
1 SELECT make, engine_type, price
2 FROM linejenniebot01-hfyb.cars.car_info
3 WHERE
4   make BETWEEN "a" AND "c"
5   AND price BETWEEN 16500 AND 18920
```

Below the query, the 'Query results' section is visible, showing a table with columns 'JOB INFORMATION', 'RESULTS', 'JSON', and 'EXECUTION'. The 'RESULTS' column is selected, displaying the following data:

Row	make	engine_type	price
1	alfa-romero	dohc	16500
2	alfa-romero	ohcv	16500
3	audi	ohc	17450
4	audi	ohc	17710
5	audi	ohc	18920
6	bmw	ohc	16925

SQL: Advanced Data Cleaning Functions

แทนค่า NULL ด้วยค่าที่ต้องการ

Syntax:

COALESCE (คอลัมน์ตั้งต้น, ค่า/ชื่อคอลัมน์ที่เอาค่า
มาแทนกรณีคอลัมน์ตั้งต้นไม่มีข้อมูล)

Ex. แทนค่าจำนวนประตูที่ NULL ด้วย "six" (สถานการณ์สมมติ)

```
1 SELECT
2   make, num_of_doors, price,
3   COALESCE(num_of_doors, "six") AS fixed_num_of_doors
4 FROM linejenniebot01-hfyb.cars.car_info
5 WHERE make = 'dodge'
```

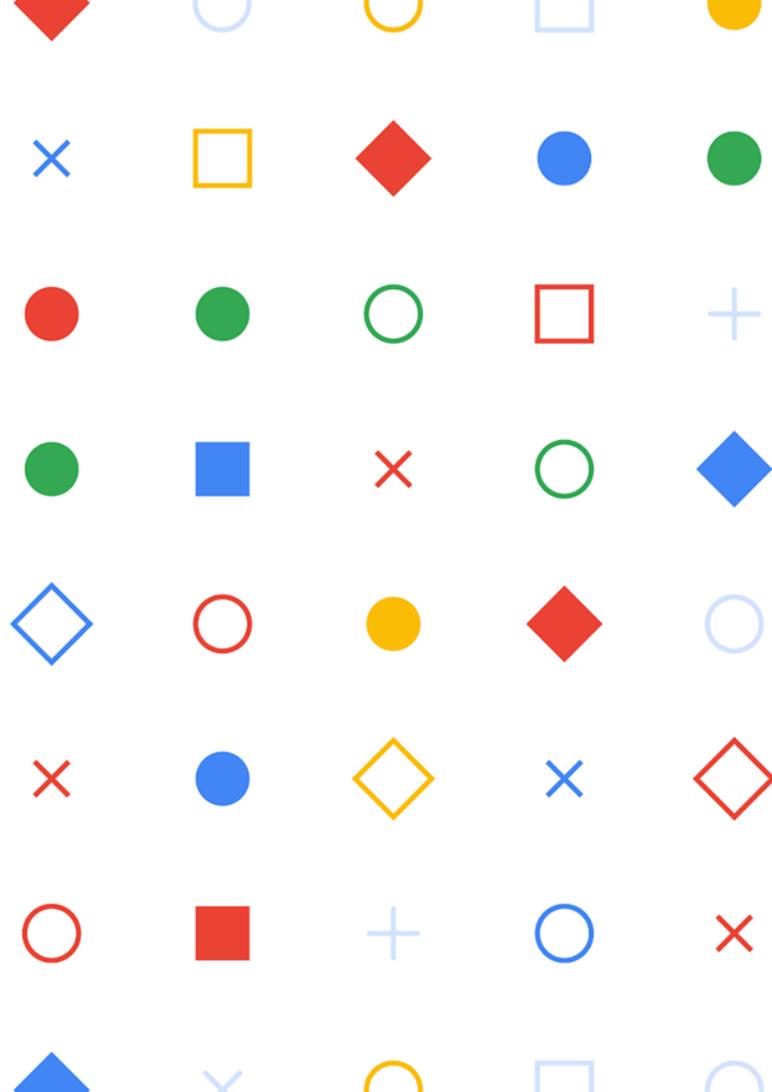
Query results

Row	make	num_of_doors	price	fixed_num_of_doors
1	dodge	two	5572	two
2	dodge	two	6377	two
3	dodge	two	7957	two
4	dodge	four	6229	four
5	dodge	four	6692	four
6	dodge	four	7609	four
7	dodge	null	8558	six
8	dodge	four	8921	four
9	dodge	two	12964	two

Verify and Report on Cleaning Results

— *Part4:*

1. Verifying Data
2. Documentation: Changelogs



Verifying Data

ก่อนจะนำข้อมูลไปใช้ต่อ เราต้อง

- มีกระบวนการในการตรวจสอบว่ากระบวนการทำความสะอาดข้อมูลเป็นไปอย่างเรียบร้อย และหากไม่เรียบร้อย ก็มีแก้ไขปัญหาให้เรียบร้อยก่อน
- ยืนยันว่าผลลัพธ์ออกมาน่าเชื่อถือและแม่นยำ
- ตัวอย่างสิ่งที่ตรวจสอบ
 - ลบแถวซ้ำด้วย Drop duplicates ใน Spreadsheets
 - นับจำนวนข้อมูลว่าเป็นไปตามที่ควรไหมด้วย COUNTA ใน Spreadsheets
 - แก้ไขข้อมูลผิดด้วย Edit → Find and Replace ใน Spreadsheets หรือคำสั่ง CASE WHEN ใน SQL
- นอกเหนือจากพิจารณาความสะอาดของข้อมูล (Data) แล้ว อย่าลืมตรวจสอบด้วยว่าเรามีข้อมูลครบ ในการตอบโจทย์ธุรกิจ (Business Problem) และบรรลุเป้าหมาย (Goals) หรือเปล่า

Verifying Data (Spreadsheets)

Ex. สรุปจำนวนครั้งที่ product_name แต่ละอันถูกสั่งซื้อ
ด้วย Pivot Table และฟังก์ชัน COUNTA

SKD Online Shop

File Edit View Insert Format Data Tools Extensions Help Last edit was second

Cells 100' 10 B I

Rows

Columns

Sheet Shift+F11

Chart

Pivot table 1

Image

Drawing

Function

Link %K

Checkbox

People chip

Comment %+Option+M

Note Shift+F2

Create pivot table

Data range: tn_2022|A1:D11

Insert to

New sheet

Existing sheet

tn_2022|B13

Cancel Create

Pivot table editor

tn_2022|A1:D11

Search

Suggested

order_id

product_name

product_price

amount

Rows Add

product_name

Order Ascending Sort by product_name

Show totals

Columns Add

Values Add

product_name

Summarize by COUNTA Show as Default

3

Note: หากใช้ COUNT เฉยๆ จะเป็นการนับข้อมูลชนิดตัวเลข แต่เนื่องจาก product_name เป็นข้อมูลชนิดข้อความ แปลว่าถ้าใช้ COUNT จะนับได้ 0 ทั้งหมด!

Result

	A	B	C	D
1	order_id	product_name	product_price	amount
2	A0001	Playstation 5	23,000	1
3	A0002	PS5 Joystick	2,000	2
4	A0003	Lactasoy 125ml	5	10,000
5	A0004	Lactasoy 125ml	5	20,000
6	A0005	Lactasoy 125ml	5	30,000
7	A0001	Playstation 5	23,000	1
8	A0006	Lactasoy 150ml	5	300
9	A0007	Lactasoy 125ml	5	40,000
10	A0008	Lactasoy 125ml		50,000
11	A0009	Lactasoy 125ml	5	100,000
12				
13	product_name	COUNTA of product_name		
14	Lactasoy 125ml	6		
15	Lactasoy 150ml	1		
16	Playstation 5	2		
17	PS5 Joystick	1		
18	Grand Total	10		

Lactasoy 5 บาทต้อง 125 มิลลิเมตรครับ
ใช้ Find and Replace แก้ได้เลย 😊

Verifying Data (SQL)

Ex. แทนที่ข้อมูลด้วยคำสั่ง CASE WHEN

Syntax:

CASE

WHEN เงื่อนไข THEN ค่าที่ต้องการ

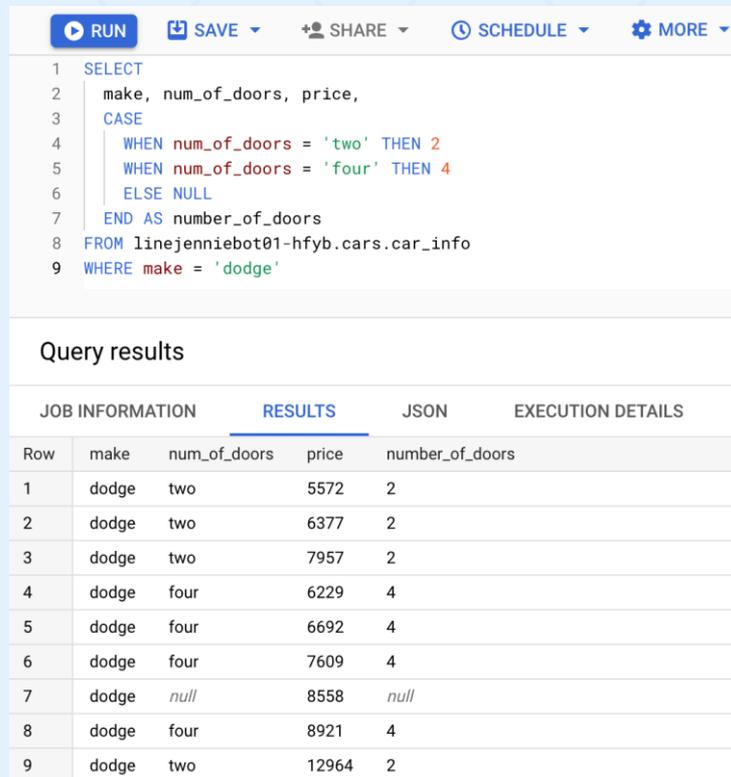
WHEN เงื่อนไข THEN ค่าที่ต้องการ

ELSE ค่าที่ต้องการกรณีไม่ตรงเงื่อนไขใด ๆ

END AS ชื่อคอลัมน์ใหม่ที่ต้องการ

Note: ในที่นี้คำสั่ง CASE อยู่ในส่วนของ SELECT คือเลือกคอลัมน์ที่สนใจมา แปลว่า เรายังไม่ได้แก้ไขข้อมูลต้นฉบับในฐานข้อมูล เพียงแค่ query มาเฉย ๆ

Ex. คอลัมน์ num_of_doors ให้แทนที่คำว่า two และ four ด้วย 2 และ 4 ไม่เช่นนั้นให้แทนด้วย NULL



```
1 SELECT
2   make, num_of_doors, price,
3   CASE
4     WHEN num_of_doors = 'two' THEN 2
5     WHEN num_of_doors = 'four' THEN 4
6     ELSE NULL
7   END AS number_of_doors
8 FROM linejenniebot01-hfyb.cars.car_info
9 WHERE make = 'dodge'
```

Query results

	JOB INFORMATION	RESULTS	JSON	EXECUTION DETAILS
Row	make	num_of_doors	price	number_of_doors
1	dodge	two	5572	2
2	dodge	two	6377	2
3	dodge	two	7957	2
4	dodge	four	6229	4
5	dodge	four	6692	4
6	dodge	four	7609	4
7	dodge	null	8558	null
8	dodge	four	8921	4
9	dodge	two	12964	2

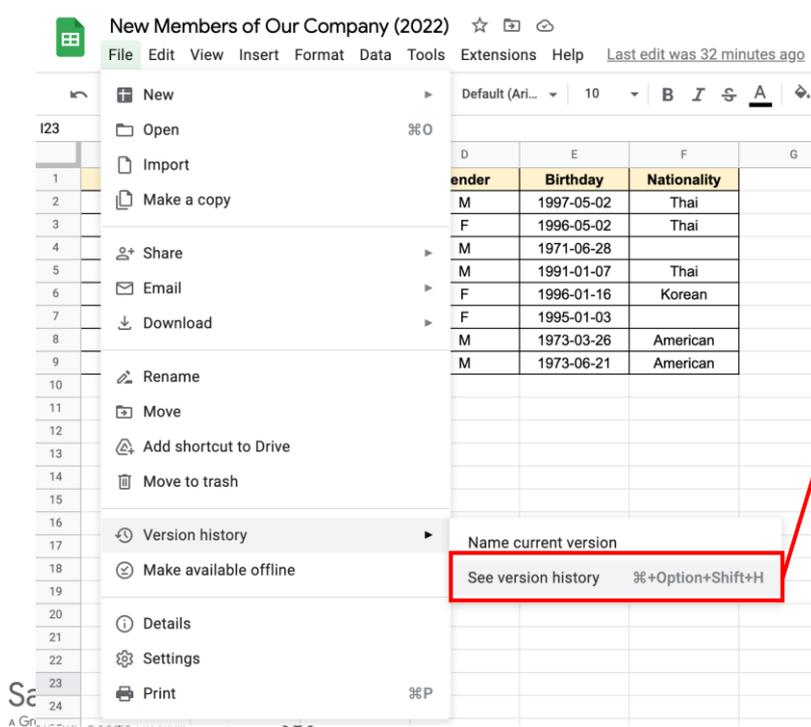
Documentation: Changelogs

ระหว่างกระบวนการทำความสะอาดข้อมูล เราควรบันทึกประวัติการเปลี่ยนแปลงของข้อมูลเอาไว้ทุกชั้นด้วย

- กระบวนการบันทึกนี้เรียกว่า **Documentation** ซึ่งเราจะบันทึก การเปลี่ยนแปลง (**Changes**) สิ่ง que เพิ่ม (**Additions**) สิ่ง que ลบ (**Deletions**) และข้อผิดพลาด (**Errors**) ระหว่างการทำความสะอาดข้อมูล
- เหตุผลในการทำ Documentation คือ
 - กู้คืน (Recover) ข้อมูลในกรณีที่เกิดข้อผิดพลาดระหว่างการทำความสะอาดข้อมูลได้
 - รายงานให้คนอื่นในทีมทราบถึงการเปลี่ยนแปลงในชั้นตอนต่าง ๆ
 - ช่วยยืนยันคุณภาพของข้อมูล que ผ่านการทำความสะอาดแล้ว
- เราเรียกไฟล์ที่บันทึกการเปลี่ยนแปลงทุกชั้นตอนตามลำดับเวลา (chronologically ordered list of modifications) ว่า **Changelog**

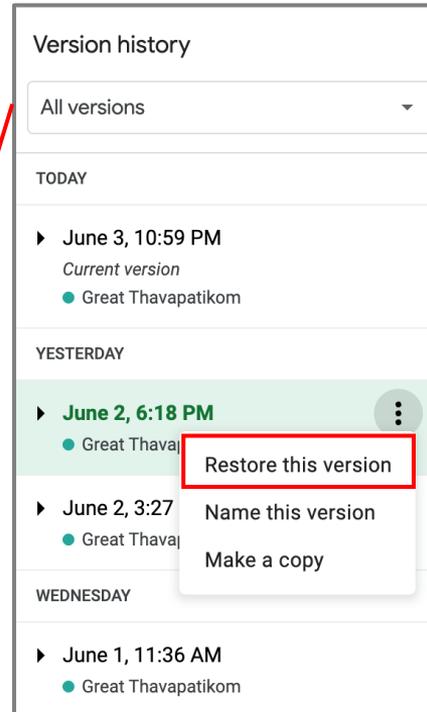
Documentation: Changelogs

ใน Spreadsheets เช่น Google Sheets เราสามารถดู Version History ได้ และย้อนกลับ (Restore) ได้ถ้าจำเป็น



The screenshot shows the Google Sheets interface for a spreadsheet titled "New Members of Our Company (2022)". The "File" menu is open, and the "Version history" option is highlighted with a red box. Below it, the "See version history" option is also highlighted with a red box. The spreadsheet data is as follows:

ender	Birthday	Nationality
M	1997-05-02	Thai
F	1996-05-02	Thai
M	1971-06-28	
M	1991-01-07	Thai
F	1996-01-16	Korean
F	1995-01-03	
M	1973-03-26	American
M	1973-06-21	American



The screenshot shows the "Version history" panel in Google Sheets. The panel displays a list of versions, with the version from June 2, 6:18 PM highlighted in green. A red box highlights the "Restore this version" option in the context menu for this version. The other options in the context menu are "Name this version" and "Make a copy".

Version history

All versions

TODAY

- ▶ June 3, 10:59 PM
Current version
● Great Thavapatikom

YESTERDAY

- ▶ **June 2, 6:18 PM**
● Great Thava
Restore this version
Name this version
Make a copy
- ▶ June 2, 3:27
● Great Thava

WEDNESDAY

- ▶ June 1, 11:36 AM
● Great Thavapatikom

Documentation: Changelogs

ใน BigQuery เราดูประวัติการเขียน query ได้จาก PROJECT HISTORY

Query results

[SAVE RESULTS](#) ▾

	JOB INFORMATION	RESULTS	JSON	EXECUTION DETAILS
Row	make	num_of_doors	price	number_of_doors
1	dodge	two	5572	2
2	dodge	two	6377	2

PERSONAL HISTORY

PROJECT HISTORY

SAVED QUERIES

 **Filter** Enter property name or value

<input type="radio"/>	Job ID	Creation time	Owner	Type	Summary
<input checked="" type="radio"/>	bqyxjob_2ba4982b_1812a59acbe	Jun 3, 2022, 11:15:29 PM	cherprang_is_very_cute	QUERY	SELECT make, num_of_doors, price, CASE WHEN num_of_doors = 'two' THEN 2 WHEN num_of_door...
<input checked="" type="radio"/>	bqyxjob_7763dc30_181286a06a3	Jun 3, 2022, 2:14:06 PM	cherprang_is_very_cute	QUERY	SELECT make, num_of_doors, price, COALESCE(num_of_doors, 'six') AS fixed_num_of_doors FROM ...
<input checked="" type="radio"/>	bqyxjob_421b1ca0_1812869d27b	Jun 3, 2022, 2:13:53 PM	cherprang_is_very_cute	QUERY	SELECT make, num_of_doors, price, COALESCE(num_of_doors, 'six') FROM linejenniebot01-hfyb.car...

