



Road to Analytics Engineer EP 1

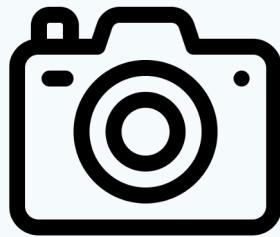
INTRO TO DATA LAKE & GIT

คอร์สเนี่ยเราจะเรียนอะไรบ้าง & เช็ตอัพ AWS Data Lake + Git

กติกา: จดได้ แคปได้ คุยเล่นได้ ถามได้



เนื้อหาที่จดไว้ไม่ลืม
แน่นอน



เนื้อหารองใหม่สำคัญ
แคปจ่อได้



ช่องแชทใน Live ไว้คุยเล่นได้
แต่ถ้าคำถามลึก ๆ แนะนำ
Sli.do



มาคุยเพิ่มเติมในกรุ๊ป
Facebook ต่อ กัน เลอะ
(คำถามนอกเหนือคอร์ส
จะตอบเท่าที่ทำได้)

ระหว่างพิงค์มีคำถาม พิมพ์ได้ที่นี่เลย



ยิ่งถาม ยิ่งได้เรียนรู้

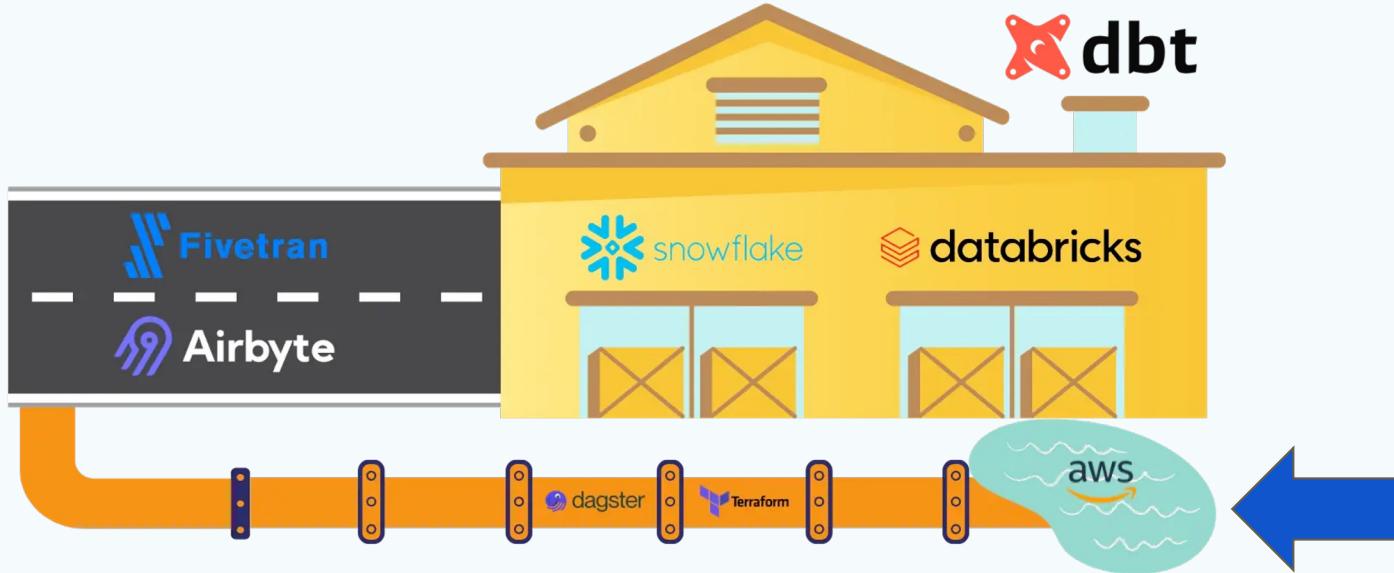
sli.do

เข้าไปที่เว็บ sli.do พิมพ์ชื่อห้อง
#r2ae-class1

(สำหรับคนที่ไม่ต้องการออกสื่อ เว็บนี้ถ้าแบบ Anonymous ได้เลย ไม่ต้องพิมพ์ชื่อ)



ເຮົາຍຸຕອງໃໝນຂອງ R2AE



คลาสนี้ เราจะคุยกันเรื่อง...

- คอร์ส R2AE นี้ เราจะเรียนอะไรกันบ้าง
- Modern Data Stack คืออะไร
- Analytics Engineer คือใคร ต่างกับ Data Engineer ยังไง
- Intro to Data Lake
 - Intro to AWS S3
 - Demo: สมัคร AWS & ลองใช้ S3
- Intro to Git
 - Demo: สมัคร & ลองใช้ GitHub
- Live Q&A คุยกันชิลๆ

ระยะเวลา: 2-3 ชั่วโมง



แนะนำตัวกันเล็กน้อย



แอดเพิร์ธ

*Founder at DataTH
Analytics Engineer at Canva*



DATA TH



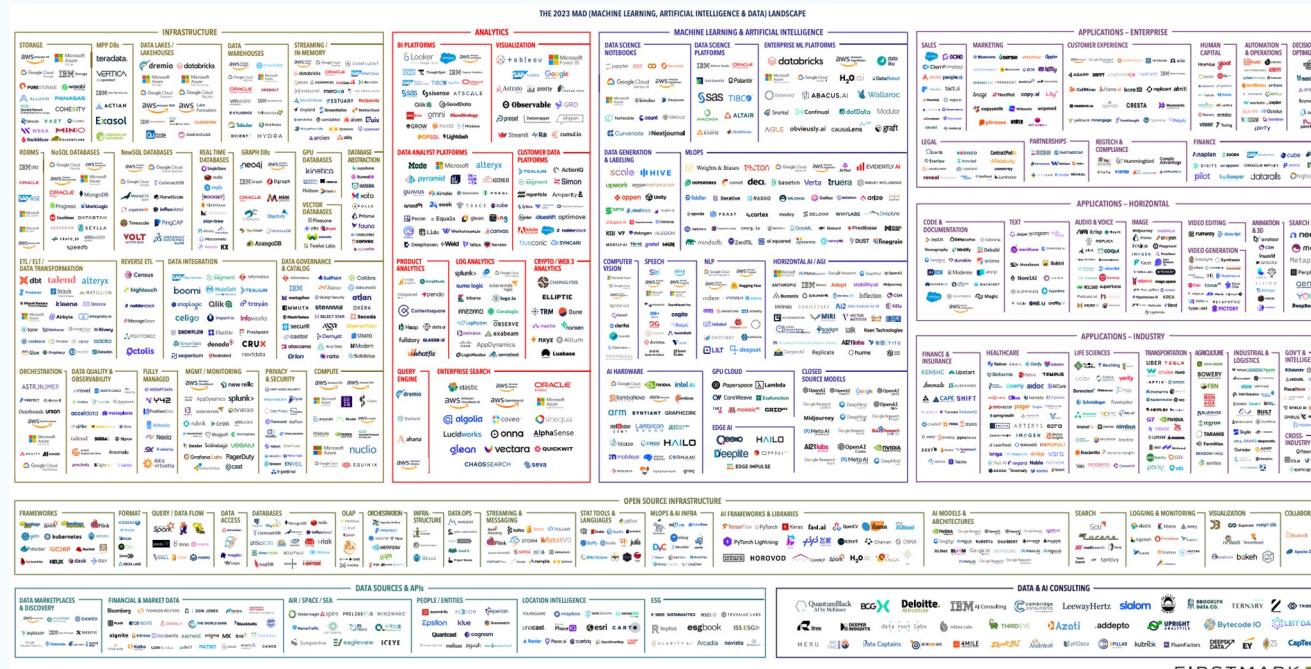
เราจะเรียนอะไรกันบ้าง

ในคอร์ส Road to Analytics Engineer



ทำไมถึงเกิดคอร์สนี้ขึ้นมา

เครื่องมือ Data ในปัจจุบัน มีเยอะมากกกก



The 2023 MAD (Machine Learning, Artificial Intelligence & Data) Landscape

<https://mattturck.com/mad2023/>

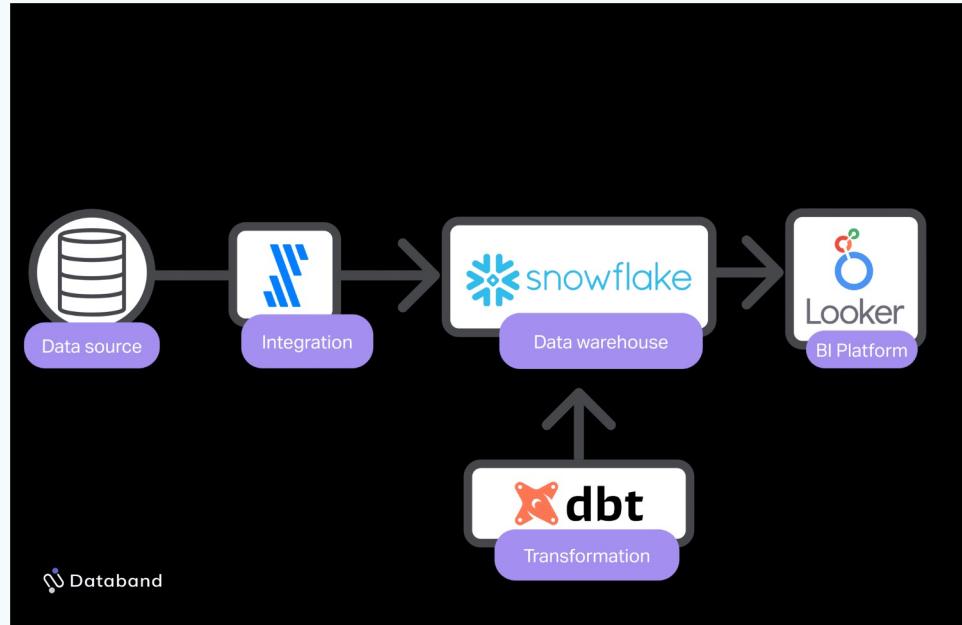
ทำไมถึงเกิดคอร์สบีบี้ขึ้นมา (2)

สร้างระบบ Data แต่ละครั้ง เมื่อการมุ่งเข้มเล็ก ๆ ในมหาสมุทร
(บางทีเลือกผิด ชีวิตเปลี่ยนอีก)



ทำไมถึงเกิดคอร์สนี้ขึ้นมา (3)

แต่งาน Data ที่ดี ไม่ได้ต้องใช้ Tool ทุกตัว



Is The Modern Data Stack Leaving You Behind?

<https://databand.ai/blog/modern-data-stack-data-engineering/>

ทำไมถึงเกิดคอร์สนี้ขึ้นมา (4)

คอร์สนี้จึงอยากมาเล่าถึง Tool Data ที่

คุณภาพดีมากกก

ใช้อย่างแพร่หลายในระดับโลก

เพื่อให้เรียนแล้วไม่ต้องไปงมเข้มกันเอง

= ประหยัดเวลาชีวิต + อัปความรู้ Data ให้เรารู้กว้าง รู้ยะ



ทำไมต้องเรียน R2AE ตอนนี้



เครื่องมือ Data ในปัจจุบันมีเยอะ
เหมือนกับเข้มในมหาสมุทร
คอร์สนี้จะพาไปรู้จักเครื่องมือ Data
ที่ได้รับความนิยมในบริษัททุกขนาด
ให้เราหยิบใช้ได้เลย

รู้จัก 7+ เครื่องมือ Data ชื่อดัง
ต่อยอดไปใช้ได้ทั่วโลก
ไม่ว่าคุณจะสนใจทำงานต่างประเทศ
หรืออยากอัปสกิลให้รองรับอนาคตงาน
สาย Data ในไทย คอร์สนี้ตอบโจทย์

พูดคุย แลกเปลี่ยนความคิดกับ
คนสอน และ TA ได้เต็มที่
เราเรียนกันแบบสบาย ๆ
สามารถได้ทั้งตอนสอนสด
และทางกรุ๊ป Facebook ลับ

ในคอร์ส R2AE เราจะเรียนอะไรกัน

EP 1 – EP 9
คลาสละ 2+ ชั่วโมง



สรุปสิ่งที่เรียน + Modern Data Concepts + Prompt Engineering

วันที่ - เวลาเรียน

เรียนสดออนไลน์ ทุกวันพุธและเสาร์ เวลา 19:00 - 21:00 น.



✓ มี Recording ดูย้อนหลัง ดูซ้ำได้ทุกคลาส 12 เดือน

ทางเข้ากรุ๊ป Facebook ลับ: เวพะผู้เรียน R2AE



facebook.com/groups/r2ae2023/



กี๊มผู้สอน



Perth Woratana

ผู้ดูแลคอร์ส Road to Analytics
Engineer

Analytics Engineer at Canva



Shane Nuttatum

Data Engineer at Chalhoub
Group



Meow Pattranit

Data Engineer at The Data
Foundry



Fai Narisara

Data Engineer Intern at T2P &
Rentspree



DATA TH

เรียนจบ สอสอบผ่าน รับ Certificate



ทำข้อสอบหลังเรียนจบ
ตามเกี่ยวกับทุกเครื่องมือ
ที่เราเรียนกัน

รับไปเลย Certificate



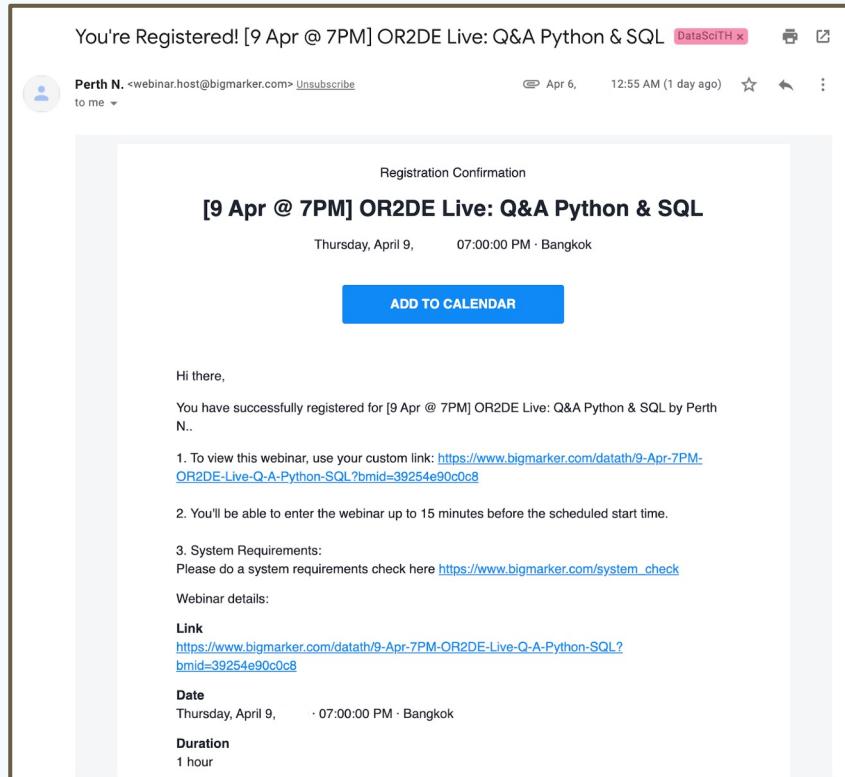
วิธีการเข้าเรียน

1) แบบเรียนสด จะได้รับลิงค์เข้าร่วมไลฟ์ทางอีเมล จากระบบ BigMarker ก่อนวันไลฟ์

ภายในไลฟ์ คุณจะ

- ได้พูดคุยกับผู้เรียนท่านอื่นระหว่างเรียน และเปลี่ยนความคิดเห็นสนุกสนานกัน (ผ่านแชท)
- เราจะมี TA คอยช่วยตอบคำถามต่าง ๆ ช่วงสอน (ผ่านแชท)
- ตอนท้ายทุกไลฟ์จะมีช่วง Live Q&A ที่คุณจะสามารถสอบถาม พูดคุยกับผู้สอนได้อย่างเต็มที่

(หากไม่เจอลิงค์ไลฟ์ สามารถทัก Inbox เพจ DataTH มาได้เลย)



วิธีการเข้าเรียน (2)

2) แบบย้อนหลัง วีดีโອะจะอยู่ใน school.datath.com
จะอัพโหลดวีดีโอย้อนหลัง หลังจากไฟล์ 0 - 2 วัน
และจะมีการแจ้งทางอีเมล + ทางกรุ๊ป Facebook

Road to Analytics Engineer 2023

ROAD TO ANALYTICS ENGINEER

0% Complete

Search for a lesson

อัญใจระหว่างจัดเรียนช่องมือ

เล่าเรียนที่ท่าน

ทางเข้ากรุ๊ป Facebook

EP 1: Intro to R2AE + S3 + Git

วิธีการสมัคร AWS

Extra Video: วิธีสมัคร G ใช้งาน Git

EP 1: Intro to R2AE + S3 + Git / Extra Video: วิธีสมัคร G ใช้งาน Git

Live: Git Essentials

มาทำความรู้จัก Git และวิธีใช้งานเข้าใจง่าย ๆ ได้ลองทำกันจริง

แอดเพิร์ธ

ฟัน Fony

git

GitHub

1:59:22

Slide 3

EXIT

R2DE vs R2AE



คอร์ส Road to Data Engineer

- เหมาะกับทุกคน ที่อยากเข้ามาทำสายงาน Data Engineer
- ไม่ต้องมีพื้นฐานมาก่อน
- สอนตั้งแต่พื้นฐาน Python, SQL ไปจนถึงคอนเซปต์ที่ใช้ในงานจริง Cloud Computing, Data Pipeline, Data Warehouse ฯลฯ
- ช่วยรีวิว Resume, สอนทำ LinkedIn, สอนการเตรียมสัมภาษณ์งาน



คอร์ส Road to Analytics Engineer

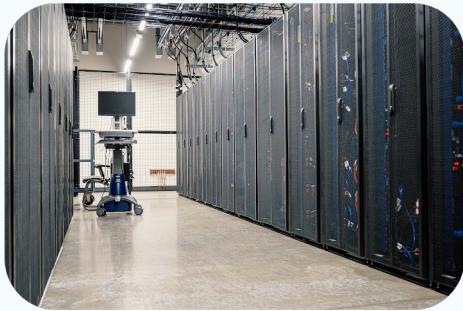
- เหมาะกับทุกคน ที่่อยากอัปสกิลในด้าน Data ให้เท่าทันเทรนด์โลก
- หากเคยทำงานสาย data มาจะเรียนสนุก แต่ถ้าไม่มีก็เรียนได้ เราเล่าพื้นฐานให้ครบ หรือมาถามเพิ่มเติมได้
- เน้นเครื่องมือใหม่ ๆ สาย Modern Data Stack ที่ดังในระดับโลก เรียนแล้วเลือกเครื่องมือได้ดี ใช้ได้แบบมองเห็นภาพกว้าง
- เล่าคอนเซปต์ Data ใหม่ ๆ ที่ใช้ในสาย Data ซึ่งในไทยไม่ค่อยมีคนรู้จัก



Modern Data Stack คืออะไร

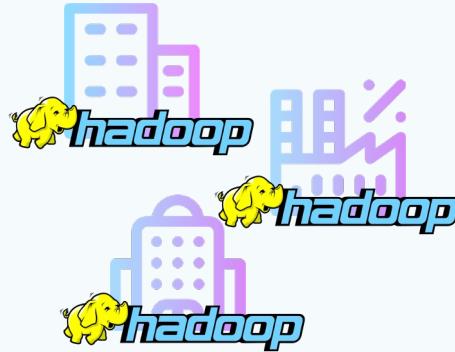


ยุคก่อน Traditional Data Stack เป็นอย่างไร



แต่ละบริษัท ตั้งเครื่อง Server เอง (On-Premise)

การจะทำงานด้าน
เทคโนโลยี ต้องลงทุนซื้อ
Server และจ้างคนมาดูแล



ทุกบริษัทต้องซื้อ Server เอง
ทำเครื่องมือ Data เพิ่มมาใช้เอง

หลายองค์กรใช้งาน Data เมื่อกัน
เจอปัญหาเมื่อกัน แก้โดยการ
สร้างเครื่องมือของตัวเอง



Modern Data Stack เกิดขึ้นมาได้ยังไง

1.

Cloud Computing
เป็นที่นิยมมากขึ้น

2.

องค์กรมี Use Case และ^ๆ
เจอปัญหาคล้าย ๆ กัน



1. Cloud Computing เป็นที่นิยมมากขึ้น



On-Premise (On-Prem)

วางเครื่องเซิร์ฟเวอร์ (Server) เอง

- - ถ้าอยากรายหุ้นระบบ ต้องซื้อเครื่องใหม่
- - ใช้เงินลงทุนก้อนแรกเยอะ
- + ค่าบริการรายเดือนไม่สูง



Google Cloud



Cloud Computing

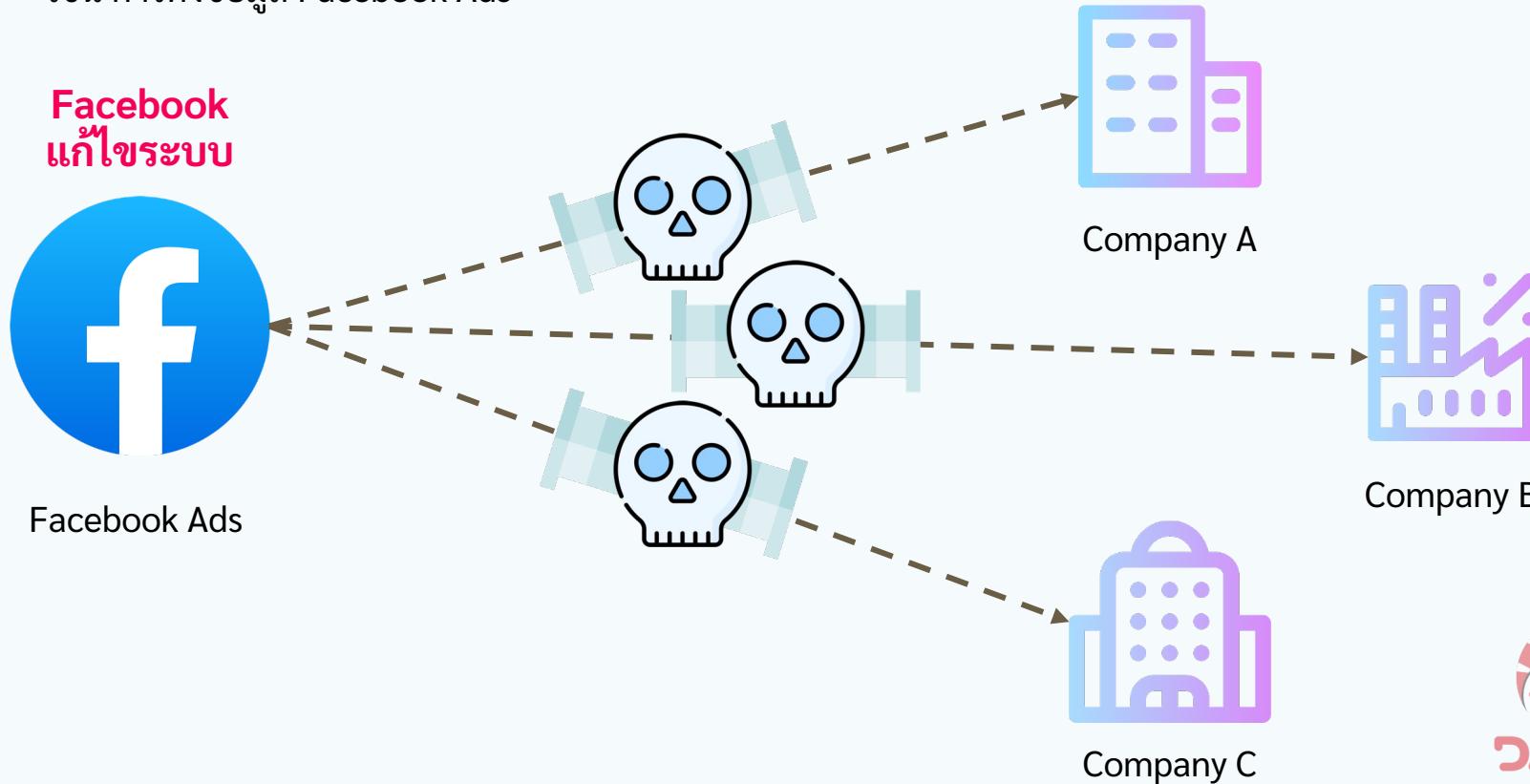
เช่าเซิร์ฟเวอร์จากบริษัทอื่น เช่น Amazon, Google, Microsoft

- + ขยายได้ไม่จำกัด
- + ไม่ต้องลงทุนก้อนแรก
- + ความปลอดภัยของ Data Center ได้มาตรฐาน
- - ค่าบริการรายเดือนอาจสูงกว่า



2. องค์กรมี Use Case และเจอปัญหาคล้าย ๆ กัน

เช่น การดึงข้อมูล Facebook Ads



Modern Data Stack เกิดขึ้นมาได้ยังไง

1.

Cloud Computing
เป็นที่นิยมมากขึ้น

2.

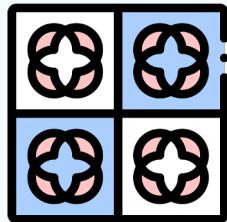
องค์กรมี Use Case และ^ๆ
เจอปัญหาคล้าย ๆ กัน

จึงเกิดมาเป็น Modern Data Stack



Modern Data Stack คืออะไร

เครื่องมือสาย Data สมัยใหม่ ที่ออกแบบมาแก้ปัญหา Data แบบใช้งานง่าย ดูแลง่าย แก้ปัญหา Data ได้ฉับไว



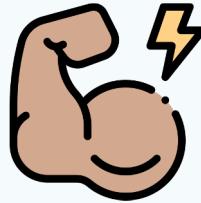
Common Patterns

ไม่ต้องทำซ้ำ
เสรีจเร็ว เสร็จไว



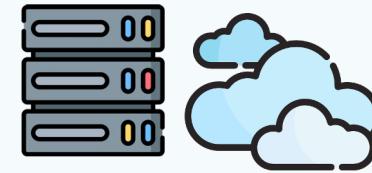
Scalability

บริษัทเล็กใช้ได้
บริษัทใหญ่ใช้ได้



Efficiency

ทีมเล็ก แต่ได้งานดี
เหมือนทีมใหญ่

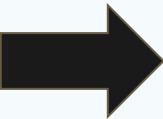


Support on-prem &
Cloud-based version

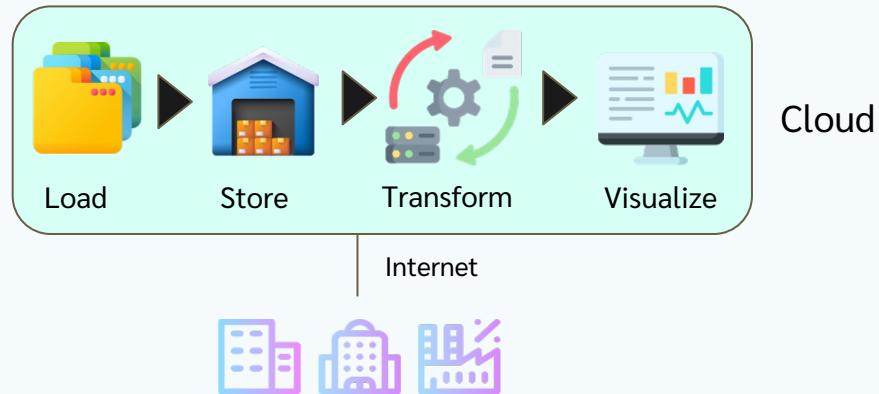
(ส่วนใหญ่) ติดตั้งบน
Server ตัวเอง ดูแลเอง
ได้ แต่บริษัทมักชอบ
แบบ Cloud มากกว่า ไม่
ต้องดูแลเอง

Traditional Data Stack -> Modern Data Stack

Traditional Data Stack



Modern Data Stack



ทุกบริษัทต้องแล Server เอง
+ ทำเครื่องมือ Data เพิ่มมาใช้เอง

ใช้เครื่องมือ Modern Data Stack
สำหรับ Use Case พื้นฐาน

- + Server บน Cloud ไม่ต้องดูแลเอง
- + ถ้าเจอ Use Case ใหม่ที่ไม่มีเครื่องมือ
ค่อยพัฒนาเพิ่ม

ตัวอย่างเครื่องมือ Modern Data Stack (บางส่วน)

Modern Data Stack ครอบคลุมทุกขั้นตอนของการทำงานกับ Data

Extract & Load
ดึงข้อมูลเข้ามา



Data Platform
ที่เก็บ & ประมวลผลข้อมูล



BigQuery



snowflake



databricks



Azure
Synapse
Analytics

Transform
แปลงข้อมูล



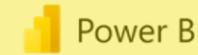
Data Analysis & Visualization
วิเคราะห์ข้อมูล, ทำ Report /
Dashboard



data
iku



Amazon
SageMaker

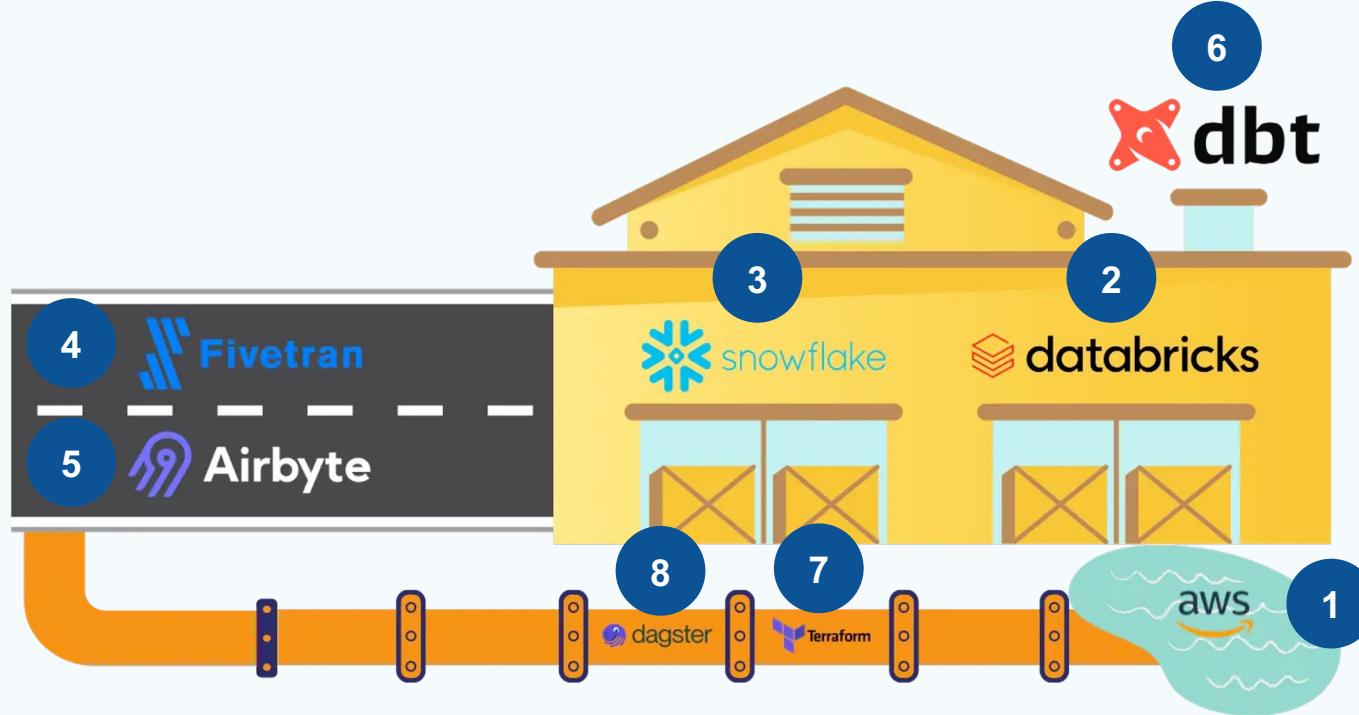


Data Orchestration
ควบคุมการไหลของข้อมูล



DATA TH

Modern Data Stack ในโครงสร้าง





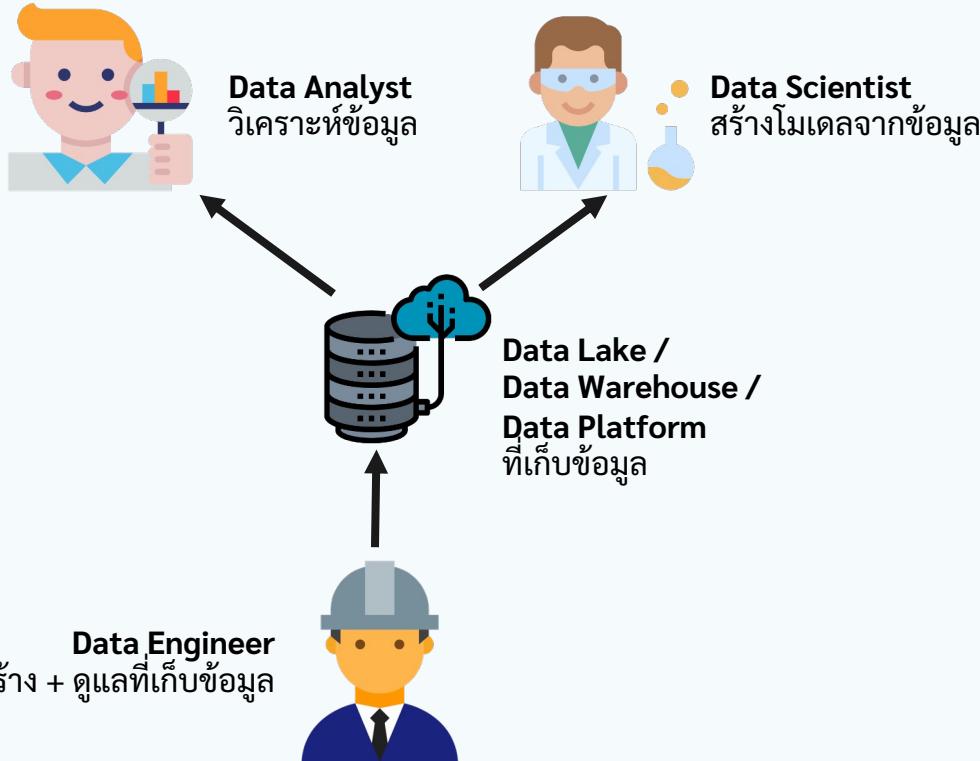
Analytics Engineer คือใคร ต่างกับ Data Engineer ยังไงบ้าง



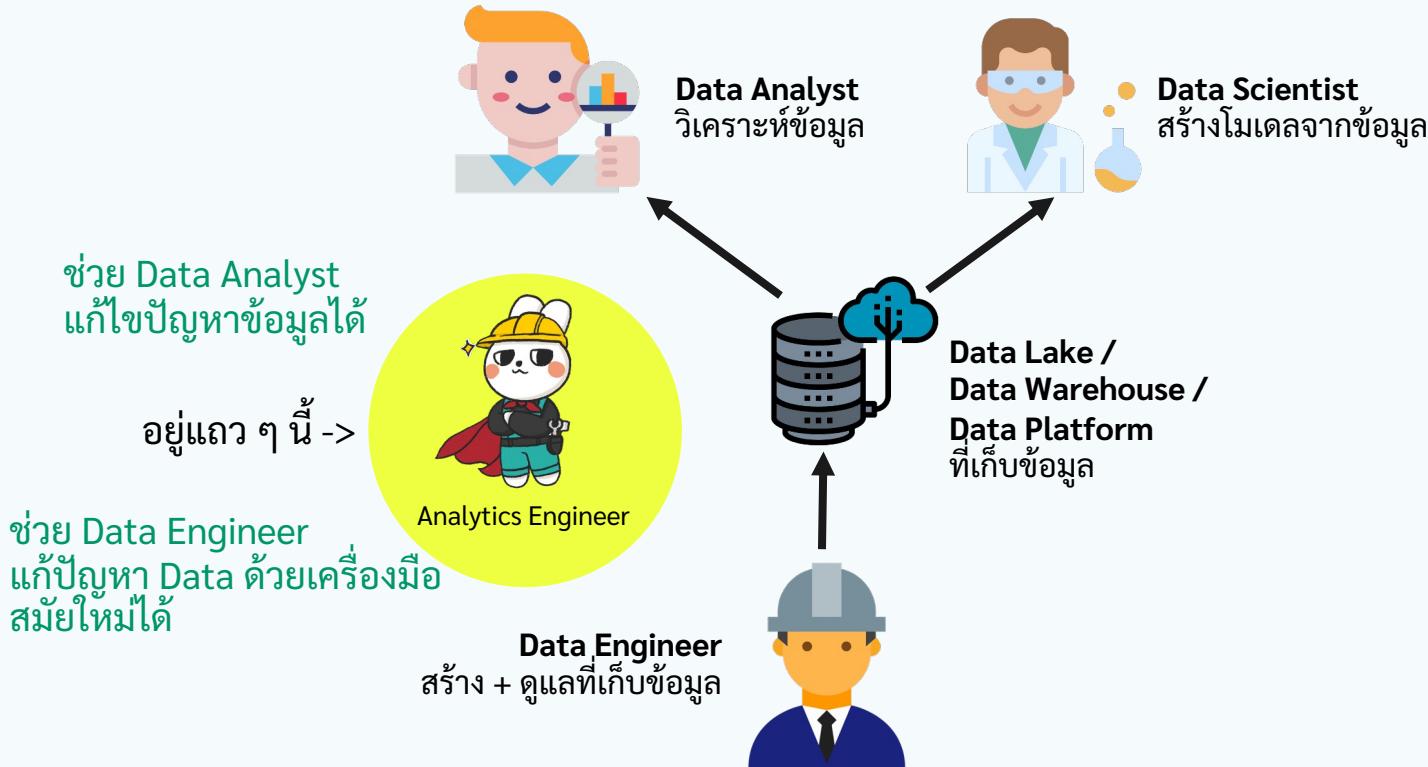
DATA TH

Analytics Engineer คือใคร

🤔 “เจอปัญหาการดึงข้อมูล,
ข้อมูลไม่มีคุณภาพ ฯลฯ
Data Engineer ก็ไม่ว่างช่วยดู”



Analytics Engineer คือใคร (2)



บริษัทแรกที่ใช้คำว่า Analytics Engineer



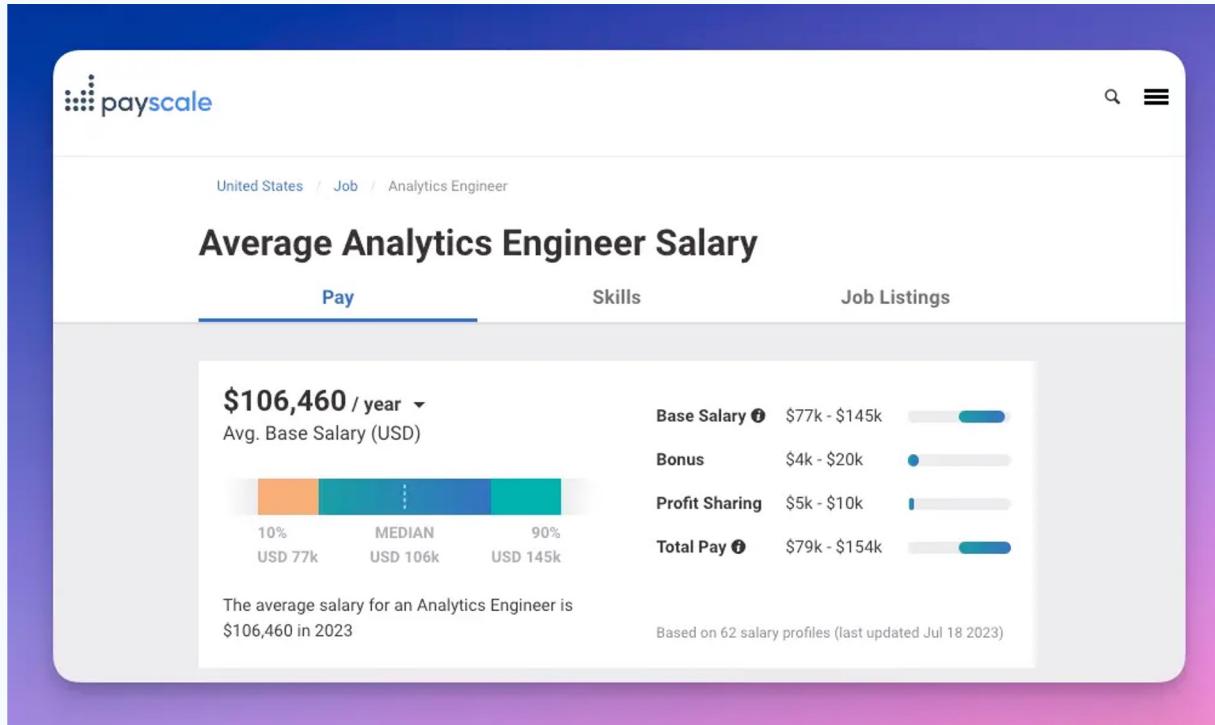
เริ่มเรียกชื่อตำแหน่งที่ไม่ใช่ Data Engineer และไม่ใช่ Data Analyst
ว่า “Analytics Engineer” ในปี 2018

บริษัทต้องมีทั้ง Data Engineer และ Analytics Engineer เหรอ?



Scope งาน DE กับ AE ค่อนข้างซ้อนทับกันมาก
จนหลายครั้งเป็นคนเดียวกันได้ แล้วแต่ความต้องการองค์กร

ตลาดงานสาย Analytics Engineer เริ่มมีให้เห็น



ANALYTIC ENGINEER
Data Engineer Team

Responsibilities

- Collect business needs in collaboration with Data Analyst team and your business user.
- Collaborate with Data Analyst team to optimize data model and dashboard on BI tools.
- Collaborate with data user to build data model from business needs.

Qualifications

- programming skills (SQL, Python, Scala)
- At least 2 years of experience in data analytics
- Strong logical thinking and problem-solving skills.
- Experience with data visualization and dashboard creation with Tableau (or other tools).
- Experience in Spreadsheet for data preparations.
- Having a growth mindset, and willing to learn new things or share knowledge with others

contact: burasakorn.sab@cjexpress.co.th



DATA TH

อย่างไรก็ตาม ชื่อตำแหน่งสาย Data เมื่อนั่นกัน ไม่ได้ลักษณะงานเหมือนกันเสมอไป อยู่ที่ความต้องการขององค์กรด้วย

blognone^{jobs} JOBS COMPANIES FOR EMPLOYERS Job Title, S

Home > Jobs > ARINCARE > Data Analyst

Data Analyst

- Entry-Level, Full time - พระโขนง กรุงเทพมหานคร

ทีม เปิดรับสมัครเพื่อนร่วมทีมตำแหน่ง Data Analyst มาร่วมทีม เพื่อพัฒนาบริการใหม่ที่จะเป็นแพลตฟอร์มช่วยยกระดับสาธารณสุขและการเข้าถึงบริการด้านสุขภาพสำหรับคนไทยทั่วประเทศ

Job Description:

- Conduct analysis and research to provide actionable insights that will be used to design marketing & sales strategies
- Identify, analyze, and devise plan to spot trends and patterns in complex datasets
- Deliver custom reports for different teams within Arincare
- Deliver custom reports for key partners and customers
- Track and analyze key industry trends and movements, then report to team leads and management
- Follow process to develop and suggest improvement on data analytics and data integrity

blognone^{jobs} JOBS COMPANIES FOR EMPLOYERS Job Title, S

Home > Jobs > AltoTech Global Co. Ltd. > Data Analyst - Energy Mgmt and Sustainability

Data Analyst

- Middle-Level, Full time - ปทุมธานี กรุงเทพมหานคร

Job Description

We are seeking a talented and experienced Data Analyst to join our team, focusing on building energy management and sustainability. In this role, you will be responsible for collecting, processing, and analyzing large amounts of data to help our clients (buildings, hotels, factories, and retail) meet its Net-Zero goals.

Responsibilities

- Collect and process raw data from various sources, including IoT devices, energy meters, and other relevant sources.
- Analyze energy consumption patterns, identify inefficiencies, and suggest ways to optimize energy use.
- Collaborate with AI engineers to develop predictive models for system maintenance, energy consumption, and other related areas.
- Develop and maintain dashboards and reports to provide actionable insights on energy consumption, system efficiency, and progress towards sustainability goals.
- Work closely with different teams to implement data-driven solutions for energy management.
- Keep abreast of latest developments in data analysis methodologies and tools in the area of energy management and sustainability.
- Ensure data privacy and security in all data-related activities.

ขอบคุณข้อมูลงานจาก Blognone jobs



DATA TH



สรุป

อย่ายึดติดกับชื่อตำแหน่งในสาย Data
ตำแหน่งเดียวกัน คนละองค์กรกัน => งานก็จะไม่เหมือนกัน





ເບີກ 3 ນາທີ



ໜັງເບຣກ

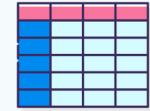
- Intro to Data Lake
- Intro to S3
- Demo: S3



Intro to Data Lake & Intro to AWS S3

Data ແປ່ງອອກເປັນ 3 ແບບໃຫຍ່ ຣ

1. Structured Data
2. Semi-Structured Data
3. Unstructured Data



1. Structured Data คืออะไร

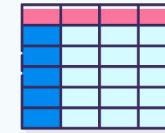
ข้อมูลแบบที่อยู่ในรูปแบบตาราง
แบ่งเป็น列 x คอลัมน์

เช่น Excel, CSV, TSV

เพิ่มเติม:

- CSV ย่อมาจาก Comma-Separated Values คือ Text file ที่ไว้สำหรับเก็บข้อมูลรูปแบบของตาราง นิยมคั่นด้วย Comma (,) หรือ Pipe (|)
- TSV ย่อมาจาก Tab-Separated Values

A	B	C	D	E	F	G	H	I
term	site	Popularity	Competition	Competition_paid	Traffic_score	Paid_impression_score	Share_of_voice	Percent_of_traffic
woocommerce ???	designil.com	27	40	0	67		15.6731876	8.17067682
adobe xd	designil.com	59	66	9	61		0.03968074	2.73210345
bootstrap 4	designil.com	60	32	0	61		0.03243026	2.52159489
???woocommerce	designil.com	28	23	1	60		3.40286743	2.33023584
font thai	designil.com	23	47	0	60		9.81897263	2.09183006
lorem ipsum	designil.com	68	63	0	60		0.012835	1.99960353
font ???	designil.com	22	47	0	58		8.73875751	1.60369875
infographic	designil.com	55	34	6	58		0.07675551	1.35195664
bounce rate ???	designil.com	9	45	0	57		45.14197988	1.32611949
google font thai	designil.com	20	24	0	57		5.43834797	1.25140507
thai font	designil.com	25	57	0	57		4.65907223	1.13151291
woocommerce	designil.com	62	72	28	56		0.02017068	0.94929497
???????	designil.com	22	22	5	56		4.33455945	0.91912392
bootstrap	designil.com	68	17	0	55		0.00602208	0.84748578
getty image ????	designil.com	6	11	0	55		26.7051048	0.77074522
jquery	designil.com	62	52	0	55		0.01405276	0.75528394
responsive web design	designil.com	42	64	22	54		0.13143857	0.70343553
lorem ipsum thai	designil.com	14	44	0	54		8.1528485	0.68265439
landing page	designil.com	54	63	42	54		0.02484704	0.62524488
thai font for mac	designil.com	11	51	0	54		22.87633838	0.61262878
????????????????? html	designil.com	6	20	0	54		31.97742702	0.60416615
sukhumvit font	designil.com	12	47	0	53		9.87686444	0.59991033
infographic ???	designil.com	10	43	0	53		13.679723	0.58999238
???????? wordpress	designil.com	14	46	5	53		10.23742658	0.57643415
font thai download free	designil.com	9	51	0	53		27.21822853	0.5711661
responsive	designil.com	45	68	1	53		0.11200085	0.56716538
wireframe	designil.com	50	59	14	53		0.07768544	0.55139784
sukhumvit set font	designil.com	14	55	0	53		5.43973756	0.55101134
page speed	designil.com	54	34	1	53		0.01378355	0.54127025
???????? ???	designil.com	12	47	0	53		10.61349979	0.53552906
???????????????????	designil.com	44	40	0	53		7.77054372	0.50000000



2. Semi-Structured Data คืออะไร

ข้อมูลแบบที่มีโครงสร้างข้อมูลที่มีความยืดหยุ่น สามารถขยายโครงสร้างข้อมูลได้ในอนาคต

หมายความว่า: ข้อมูลที่สามารถปรับเปลี่ยนโครงสร้างได้ตลอดเวลา

Semi-structured data มีหลายประเภท
เช่น Key-value, Document, JSON, XML, Graph



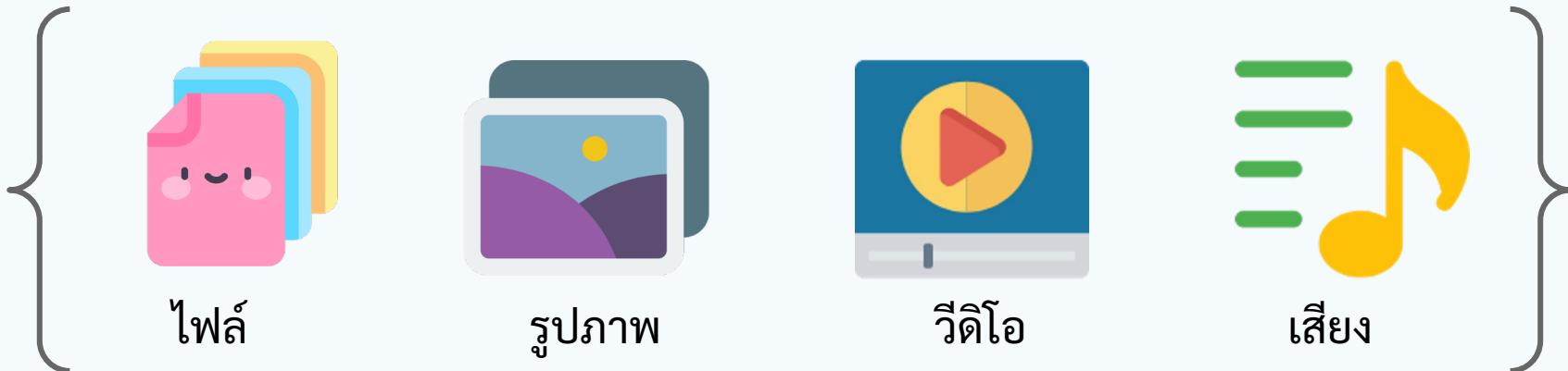
ตัวอย่างข้อมูลแบบ JSON
(JavaScript Object Notation)

```
{
  "users": [
    {
      "userId": 1,
      "firstName": "Adam",
      "lastName": "Lee",
      "emailAddress": "adam@gmail.com"
    },
    {
      "userId": 2,
      "firstName": "John",
      "lastName": "Doe",
      "phoneNumber": "0406817201",
      "emailAddress": "john.doe@gmail.com"
    }
  ]
}
```

3. Unstructured Data คืออะไร

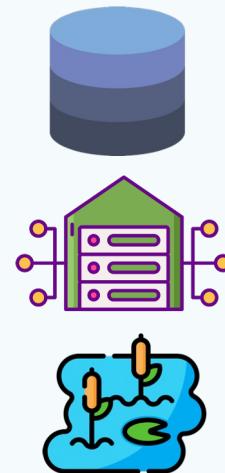
ข้อมูลที่ไม่มีโครงสร้าง

เช่น



ประเภทของ Data Storage (ที่เก็บข้อมูล) มีแบบไหนบ้าง

1. Database
2. Data Warehouse
3. Data Lake



1. Database (DB)

ใช้ในการเก็บ Data ที่มีโครงสร้าง (Structured & Semi-Structured) อาจไม่รองรับข้อมูลขนาดใหญ่ เหมาะกับการเก็บ Data ที่ต้องการ เขียนและอ่าน อย่างรวดเร็ว

มักใช้ในงาน Software Development

เช่น Website, Application มือถือ ฯลฯ (เช่น อ�토โนม่า, เว็บ & แอพ Social Media)

Database แบ่งออกเป็น 2 ประเภทใหญ่ ๆ

1.1) SQL Database

สำหรับเก็บข้อมูลแบบตาราง (Structured Data)



1.2) NoSQL Database

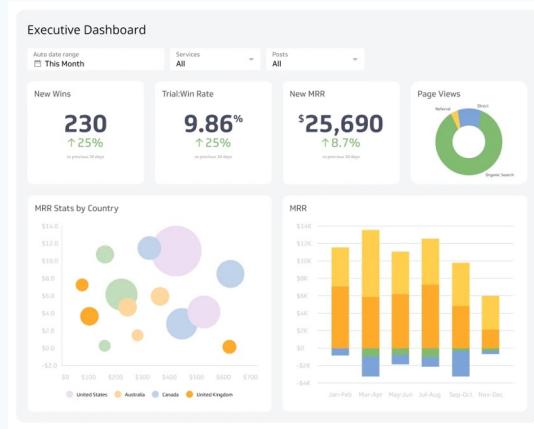
สำหรับเก็บข้อมูลที่โครงสร้างมีความยืดหยุ่น (Semi-Structured Data)



DATA TH



2. Data Warehouse (DW)



ตัวอย่าง Dashboard จาก [Klipfolio.com](https://www.klipfolio.com)

ใช้เก็บ Data แบบตาราง (Structured Data) รองรับข้อมูลขนาดใหญ่
เหมาะสมกับข้อมูลที่ต้องการ อ่าน อย่างรวดเร็ว

มักใช้ในงาน Data เช่น

- ✓ การทำ Dashboard / Report หรือ การทำข้อมูลเพื่อประกอบการตัดสินใจ (บางบริษัทเรียกว่าการทำ Business Intelligence)
- ✓ การวิเคราะห์ข้อมูล Data Analytics หรือการนำข้อมูลไปเพื่อไปสร้างโมเดล โดย Data Analyst และ Data Scientist



3. Data Lake (DL)



Data Lake เป็นที่ เก็บข้อมูลได้ทุกแบบ ขอแค่เป็นไฟล์

รองรับข้อมูลขนาดใหญ่

เหมาะสมกับการ เขียนและอ่าน แบบไม่เน้นความเร็ว

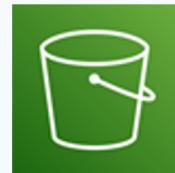
ใช้กับทั้งในงาน Software Development และ Data เช่น แอปมือถือใช้ Data Lake เก็บรูปบัตรประชาชนลูกค้า

อีกชื่อของ Data Lake
คือ **object storage**
หรือ **blob storage**
 เพราะ เก็บข้อมูลเป็นไฟล์

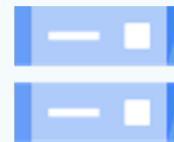
ตัวอย่าง
Data Lake



Apache Hadoop HDFS
Hadoop Distributed File
System



Amazon S3



Google Cloud
Storage



Azure Blob
Storage



สรุป ประเภทของ Data Storage (กี๊เก็บข้อมูล)

	Database 	Data Warehouse 	Data Lake 
ประเภทข้อมูล	Structured & Semi-Structured	Structured	Structured, Semi-Structured, Unstructured
ขนาดข้อมูล	⚠️ ได้ทุกขนาด แต่อาจเก็บข้อมูลได้ใหญ่ไม่เท่าตัวอื่น (DB บางตัวรองรับข้อมูลขนาดใหญ่)	ได้ทุกขนาด	ได้ทุกขนาด
เหมาะสมกับ	เขียน & อ่าน	อ่านอย่างเดียว	เขียน & อ่าน
งานที่มักจะใช้	Software	Data	Software + Data



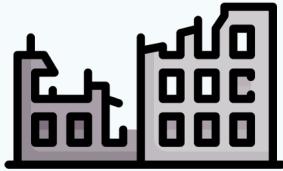


เก็บข้อมูล On-Premise VS Data Lake uu Cloud



DATA TH

ปัญหาของการเก็บข้อมูล ใน Server ของตัวเอง (On-Premise)



ข้อมูลสูญหายได้

ไม่ทันต่อภัยธรรมชาติ เช่น
ไฟไหม้ น้ำท่วม ตึกระเบิด

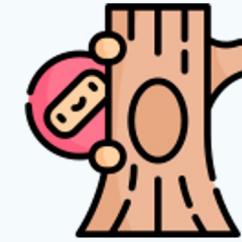
= ข้อมูลหาย



พื้นที่จำกัด ขยายไม่ได้

พอเรามีข้อมูลเยอะมากขึ้นเรื่อยๆ ๆ
Server ก็จะพื้นที่เต็ม

= ต้องสั่งซื้อ Server ใหม่มาเพิ่ม



ค่าใช้จ่ายที่ซ่อนอยู่

เช่น ค่าจ้างคนเพื่อมาดูแล Server,
ค่าจ้างคนมาติดตั้ง, ค่าเสียโอกาส
จากการรอนุ่มติ Server ใหม่

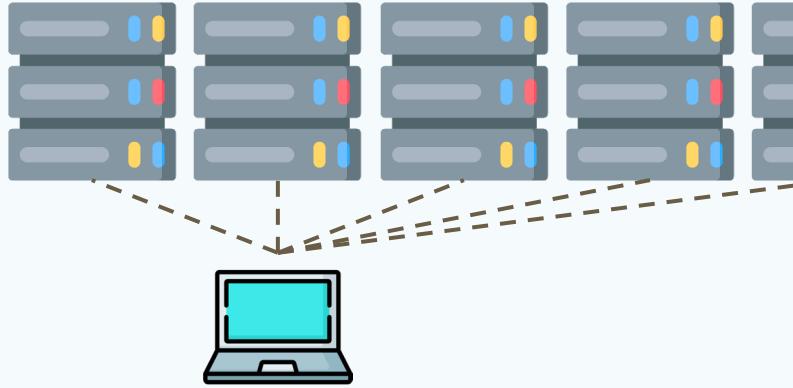


ข้อดีของการย้ายมา Data Lake UU Cloud



~~ข้อมูลสูญหายได้~~
ข้อมูลมีหลาย Backup ไม่หายง่าย ๆ

เกิดภัยธรรมชาติไม่เป็นไร เพราะมี
การเก็บข้อมูลทั้งชุดไว้หลายที่



~~พื้นที่จำกัด ขยายไม่ได้~~
พื้นที่ไม่จำกัด ขยายได้อิสระ

เซิร์ฟเวอร์ไม่มีวันเต็ม
(แต่เงินในกระเป๋าอาจจะหมดก่อน)



ประโยชน์ ๑ ของ Data Lake



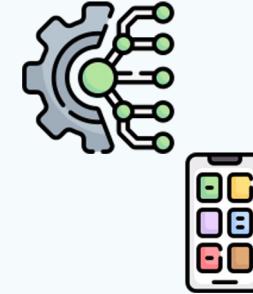
ไม่ต้องลงทุนเยอะ

ไม่ต้องค่าจ้างคนเพื่อมาดูแล Server, ไม่ต้องค่าจ้างคนมาติดตั้ง, และไม่ต้องรองอนุมัติ Server



ความปลอดภัยสูง

กำหนดสิทธิ์การเข้าถึงไฟล์ได้ละเอียด, มีการเข้ารหัสไฟล์ (Encryption) อัตโนมัติ



ใช้งานได้หลากหลาย
โดยที่ค่าเก็บไม่แพง

ใช้งาน Data (เก็บข้อมูล,
โมเดล Machine Learning)
หรืองาน Software ก็ได้
เก็บข้อมูลรูปแบบไหนก็ได้
ไม่แพงเท่าระบบอื่น

Data Lake มีตัวไหนบ้าง (On-Prem & Cloud)

Cloud Computing ซึ่งดังมีบริการ
Object Storage ทุกค่าย



S3



Google Cloud

Azure Blob Storage, Google Cloud
Azure Data Lake Storage (ใช้ใน R2DE)
Storage



Object Storage แบบ Open-source
ติดตั้งบน On-premise ได้

MINIO

ceph



DATA TH



Intro to **AWS S3**



DATA TH

AWS ในประเทศไทย

ประกาศแผนงานที่จะเปิดตัว AWS Bangkok Region

by Chatcharoen Chivakanit | on 18 OCT 2022 | in [Announcements](#), [Government](#), [News](#), [Regions](#) | [Permalink](#) |  Share

วันนี้ Amazon Web Services (AWS) ประกาศแผนงานที่จะเปิดตัว **AWS Asia Pacific (Bangkok) Region** ในประเทศไทย ซึ่งจะนำเทคโนโลยีคลาวด์ล่าสุด – ตั้งแต่ compute, storage ไปจนถึง artificial intelligence (AI) และ machine learning (ML) เพื่อให้ใกล้กับลูกค้าในประเทศไทยมากที่สุด



Amazon Web Services (AWS) เป็นบริการ Public Cloud Computing ที่ใหญ่ที่สุดในโลก เปิดให้บริการอยู่ทั่วโลก มีแผนจะมาเปิดในประเทศไทยแล้ว

หลังจากที่ AWS มาเปิด น่าจะได้เห็นงานสาย Cloud / Data เพิ่มขึ้น และบริการต่าง ๆ ที่เราร่วมกันในคอร์สนี้จะผูกกับ Cloud ทำให้น่าจะได้รับความนิยมมากขึ้น

เกล็ดเสริม:

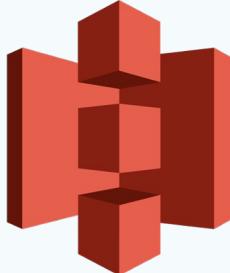
ปัจจุบัน บริการ AWS ที่ใกล้ไทยที่สุด คือ สิงคโปร์ ซึ่งความเร็วอินเตอร์เน็ตไทย -> สิงคโปร์ค่อนข้างดี ทำให้บริษัทไทยหลายที่ใช้ AWS กันสักพักแล้ว

เกล็ดเสริมกว่า: ได้ข่าวร้อนมาว่า บางบริการ กำลังเริ่มเทพในไทยแล้ว!



DATA TH

AWS S3 (Simple Storage Service)



<- โลโก้เก่า

โลโก้ใหม่ ->



บริการ Data Lake บนระบบ Cloud ของ AWS ที่เกิดมาตั้งแต่ปี 2006 และเป็นหนึ่งในบริการที่ได้รับความนิยมสูงสุดในหมู่ Data Lake

จุดเด่นของ S3:

- ใช้งานได้ง่าย เชื่อมต่อกับบริการทั้งใน AWS + นอก AWS
- โอกาสไฟล์หายน้อยมาก (Durability = 99.999999999% (11 nines))
 - เก็บไฟล์ไว้อย่างน้อย 3 ที่ ห่างกันอย่างน้อย 1 ไมล์ (3+ Availability Zones)
 - โอกาสไฟล์หาย = 1 ใน 10 ล้าน ใน 10,000 ปี
- ค่าเก็บไฟล์ไม่แพง (\$0.02 = 80 สตางค์ / GB)
- ความปลอดภัยสูง ตั้งค่าได้ละเอียด

ค่อนเชปต์ S3 แสนเรียบง่าย: แบ่งเป็น Bucket (ถังเก็บไฟล์) และ Object (ไฟล์)



Bucket

Bucket ใน S3 โดยทั่วไปห้ามชื่อซ้ำกัน
จะเก็บกี่ไฟล์ ใช้พื้นที่เท่าไหร่ก็ได้



Objects

ไฟล์ที่เก็บใน Bucket
แต่ละไฟล์จะมีชื่อของตัวเอง เรียกว่า Object Key

ความปลอดภัย ของ S3

เราสามารถตั้งกฎการเข้าถึงไฟล์ ได้ทั้งระดับ Bucket และระดับ Object

เช่น

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Principal": { กฎสำหรับ แก่กลุ่มจอห์น  
                "AWS": "arn:aws:iam::111122223333:role/JohnDoe"  
            },  
            "Effect": "Allow",  
            "Action": [ "s3:GetObject", "s3:GetObjectVersion" ],  
            "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*", กฎสำหรับ Bucket ชื่อ DOC-  
            "Condition": {  
                "StringEquals": {  
                    "s3:ExistingObjectTag/environment": "production" } }  
        } ]  
    }  
}
```

ให้ดูได้เฉพาะ Object ที่ติดแท็ก production

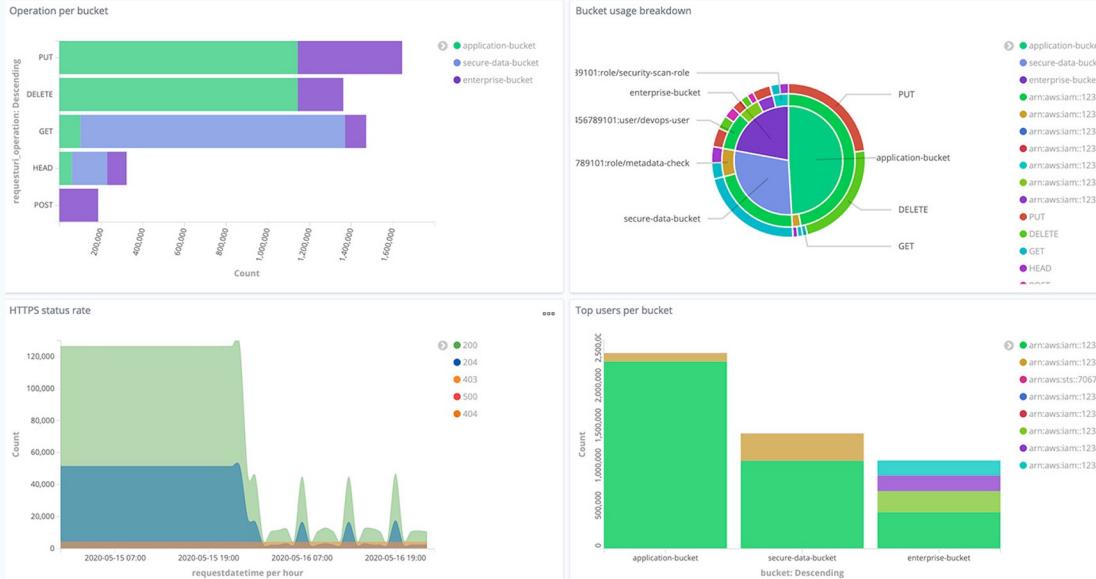
สำหรับท่านที่สนใจเจาะลึก: ข้อมูลเกี่ยวกับ IAM Policy, Bucket Policy, ACLs

<https://aws.amazon.com/blogs/security/iam-policies-and-bucket-policies-and-acls-oh-my-controlling-access-to-s3-resources/>



S3 เก็บข้อมูลการเข้าถึงไฟล์กั้งหมด

เก็บข้อมูลได้หมดว่า ใคร ทำอะไร กับไฟล์ไหน



ตัวอย่างการวิเคราะห์ S3 Log ด้วยบริการ Amazon OpenSearch

วิธีเช็คอัพ:

<https://aws.amazon.com/blogs/big-data/analyzing-amazon-s3-server-access-logs-using-amazon-opensearch-service/>

สำหรับท่านที่สนใจเจาะลึก: การทำ Monitoring บน S3 แบบละเอียด
<https://docs.aws.amazon.com/AmazonS3/latest/userguide/monitoring-overview.html>

บริการ S3 มีหลายระดับ (Tier) เราเลือกได้ในแต่ละไฟล์

สำหรับคนไม่อยากเลือก -> S3 Intelligent-Tiering - AWS จะเข็คและอัพเดทโน้มติว่าไฟล์ไหนใช้งานบ่อย-ไม่บ่อย (คิดเงิน)

S3 Standard

S3 Standard -
IA (Infrequent
Access)

S3 OneZone -
IA

S3 แบบทั่วไป
เหมือนกับไฟล์ทั่วไป
ที่เราต้องใช้บ่อย

S3 สำหรับไฟล์ที่ใช้งานไม่บ่อย

S3 Glacier
Instant
Retrieval

S3 Glacier
Flexible
Retrieval

S3 Glacier
Deep
Archive

S3 สำหรับไฟล์ที่ใช้งานไม่บ่อยมาก ๆ
เหมือนกับไว้ทำ Backup

สำหรับห่านที่สนใจเจาะลึก: ข้อแตกต่างของแต่ละ Tier ของ S3 แบบละเอียด
<https://aws.amazon.com/s3/storage-classes/>



DATA TH

Bonus: Availability 99%, 99.5%, 99.9%, 99.99% ต่างกันยังไง

คุณค่ามีส่วนตัวกับการจ่ายเงินเพิ่ม เพื่อ Availability ที่เพิ่มขึ้นน้อยกว่า 1% ?

	การล่มต่อเดือน	การล่มต่อปี
Availability 99%	7.31 ชั่วโมง	3.65 วัน
Availability 99.5%	3.65 ชั่วโมง	1.83 วัน
Availability 99.9%	42.83 นาที	8.77 ชั่วโมง
Availability 99.99%	4.38 นาที	52.6 นาที

ข้อมูลจาก https://en.wikipedia.org/wiki/High_availability

ขึ้นอยู่กับความสำคัญของระบบ ว่าเรายอมรับการล่มได้แค่ไหน

ค่าใช้จ่ายของ S3

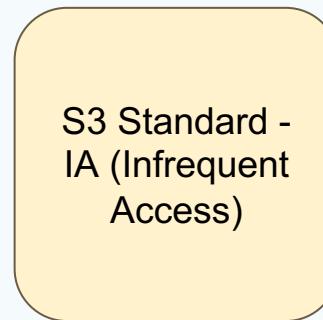
S3 ประกอบไปด้วยค่าใช้จ่าย 3 อย่าง

ค่าเก็บไฟล์ (storage) + ค่าเปิดไฟล์ (retrieval) + ค่าอ่านไฟล์ (data transfer)

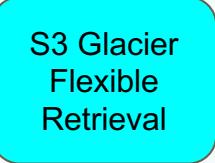
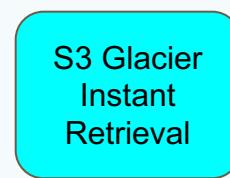
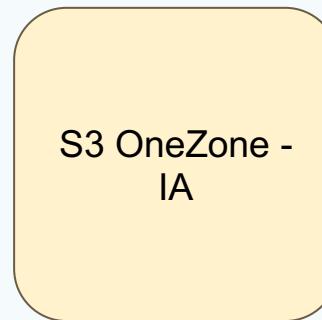
ราคาถูก-แพง ตาม Tier ของ S3 ที่ใช้



S3 แบบทั่วไป
ค่าเก็บไฟล์แพงสุด
ไม่มีค่าเปิดข้อมูล



S3 สำหรับไฟล์ที่ใช้งานไม่บ่อย
ค่าเก็บถูกกว่า S3 Standard
แต่มีค่าเปิดข้อมูล

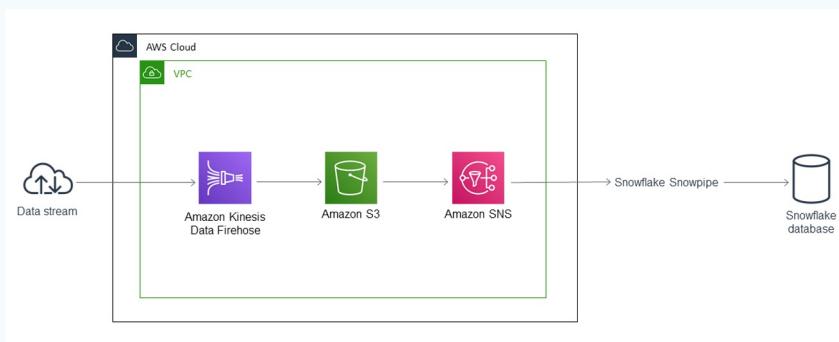


S3 สำหรับไฟล์ที่ใช้งานไม่บ่อยมาก ๆ
ค่าเก็บถูกกว่าแบบ IA อีก
แต่มีค่าเปิดข้อมูลแพง
และอาจใช้เวลาเปิดนาน

สำหรับท่านที่สนใจเจาะลึก: การคิดค่าใช้จ่ายของ S3 แบบละเอียด
<https://aws.amazon.com/s3/pricing/>

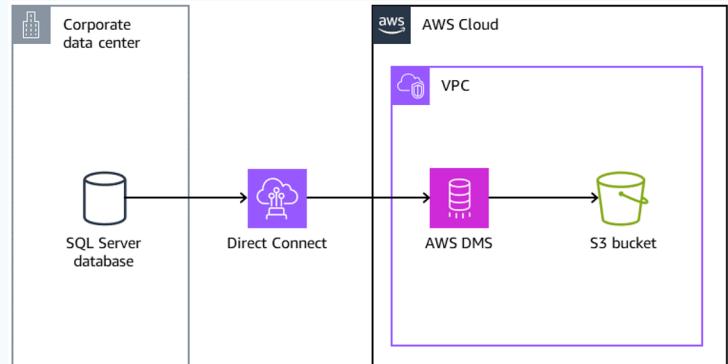
S3 เป็นส่วนหนึ่งของงาน Data บน AWS แบบทั้งหมด

การเก็บไฟล์บน S3 นำไปใช้ประโยชน์ได้เยอะ



ตัวอย่าง - การส่งข้อมูลแบบ Streaming เข้า Snowflake (เราจะเรียน Snowflake กันในคลาส 3)
ใช้ S3 เก็บไฟล์ ก่อนนำไปประมวลผล

อ่านเพิ่มเติม https://docs.aws.amazon.com/prescriptive-guidance/latest/patterns/automate-data-stream-ingestion-into-a-snowflake-database-by-using-snowflake-snowpipe-amazon-s3-amazon-sns-and-amazon-kinesis-data-firehose.html?did=pq_card&trk=pq_card



ตัวอย่าง - การสร้าง Backup จากข้อมูลใน Microsoft SQL Server
ใช้ S3 เก็บไฟล์ เพื่อใช้เป็น Backup

อ่านเพิ่มเติม <https://docs.aws.amazon.com/prescriptive-guidance/latest/patterns/export-a-microsoft-sql-server-database-to-amazon-s3-by-using-aws-dms.html>

Demo: มาลองใช้ AWS S3 กัน

ใน Demo นี้เราจะทำอะไรบ้าง

- ก่อนเริ่ม: สมัครสมาชิก AWS (ดูวิธีในวิดีโอ go.datath.com/aws-free-tier)
- Part 1: สร้าง S3 Bucket & อัปโหลดไฟล์
- Part 2: อัปโหลดไฟล์ & อ่านไฟล์จาก S3 Bucket บนเว็บไซต์



ก่อนเริ่ม: การสมัครสมาชิก AWS



go.datath.com/aws-free-tier

Part 1.1: สร้าง S3 Bucket (1)

1. เสิร์ชหา s3

The screenshot shows the AWS search interface. A red box highlights the search bar at the top containing the query 's3'. Below the search bar, the results are displayed under the heading 'Search results for 's3'' with the sub-instruction 'Try searching with longer queries for more relevant results'. The results are categorized into 'Services (7)', 'Features (22)', 'Resources (New)', 'Blogs (1,275)', 'Documentation (23,107)', 'Knowledge Articles (20)', and 'Tutorials (13)'. A specific result for 'S3' is highlighted with a blue box, showing its description as 'Scalable Storage in the Cloud' and listing 'Top features' such as Buckets, Access points, Storage Lens dashboards, and Batch Operations.

2. กด Create bucket

The screenshot shows the 'Create a bucket' wizard. It contains a large text area explaining that every object in S3 is stored in a bucket and that users need to create a bucket where objects will be stored. At the bottom is a prominent orange 'Create bucket' button.

สำหรับ User ใหม่

The screenshot shows the 'Amazon S3' service page. On the left, there's a sidebar with links for Buckets, Access Points, Object Lambda Access Points, Multi-Region Access Points, and Batch Operations. The main content area displays an 'Account snapshot' and a list of 'Buckets (7)'. A red box highlights the 'Create bucket' button at the bottom right of the bucket list.

สำหรับ User เก่า



Part 1.2: สร้าง S3 Bucket (2)

3. กรอกข้อมูลเพื่อสร้าง Bucket และกด Create Bucket

Create bucket Info

Buckets are containers for data stored in S3. [Learn more](#)

General configuration

Bucket name Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)

AWS Region

Copy settings from existing bucket - optional
Only the bucket settings in the following configuration are copied.
[Choose bucket](#)

[Cancel](#) [Create bucket](#)

ข้อมูลสำคัญ: Bucket name ชื่อ S3 Bucket จะต้องห้ามซ้ำกับ S3 Bucket อื่นทั่วโลก

ชื่อ S3 Bucket ห้ามมีซ่องว่าง ต้องคั่นคำด้วย - หรือ . เท่านั้น



Part 2.1: อัพโหลดไฟล์ขึ้น S3 Bucket บนเว็บไซต์

- เข้าไปที่เมนู S3 > Buckets คลิกที่ชื่อ Bucket ที่เราเพิ่งสร้าง

Buckets (8) [Info](#)
Buckets are containers for data stored in S3. [Learn more](#)

Name	AWS Region	Access
[REDACTED]	[REDACTED]	[REDACTED]
perth-test-bucket-ja	Asia Pacific (Singapore) ap-southeast-1	Bucket and objects not public

- กดปุ่ม Upload

Objects (0)
Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permission.

[C](#) [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create folder](#) [Upload](#)

Part 2.2: อัปโหลดไฟล์ขึ้น S3 Bucket บนเว็บไซต์

3. กด Add Files หรือ Add Folder และเลือก ไฟล์ที่แนบเครื่อง ที่ต้องการอัปโหลด และกด Upload

Files and folders (0)

All files and folders in this table will be uploaded.

Find by name

Name	Folder	Type	Size
No files or folders			

You have not chosen any files or folders to upload.

Destination

Destination
s3://perth-test-bucket-ja

▶ Destination details
Bucket settings that impact new objects stored in the specified destination.

▶ Permissions
Grant public access and access to other AWS accounts.

▶ Properties
Specify storage class, encryption settings, tags, and more.

Cancel **Upload**

เกร็ดเสริม:

กดเมนู Properties เพื่อเลือก S3 Tier สำหรับไฟล์ที่จะอัปโหลดได้เลย

Properties
Specify storage class, encryption settings, tags, and more.

Storage class
Amazon S3 offers a range of storage classes designed for different use cases. [Learn more](#) or see [Amazon S3 pricing](#)

Storage class	Designed for	Availability Zones	Min storage duration
<input checked="" type="radio"/> Standard	Frequently accessed data (more than once a month) with milliseconds access	≥ 3	-
<input type="radio"/> Intelligent-Tiering	Data with changing or unknown access patterns	≥ 3	-



Part 2.3: อ่านไฟล์จาก S3 Bucket บนเว็บไซต์

4. ในหน้าหลักของ Bucket คลิกที่ชื่อไฟล์

The screenshot shows the AWS S3 console interface. At the top, there are tabs for Objects, Properties, Permissions, Metrics, Management, and Access Points. The Objects tab is selected. Below the tabs, the heading 'Objects (1)' is displayed. A sub-instruction says: 'Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)'.

Below the heading, there is a toolbar with buttons for Copy S3 URI, Copy URL, Download, Open, Delete, Actions, Create folder, and Upload. There is also a search bar labeled 'Find objects by prefix' and a pagination indicator showing page 1 of 1.

The main list displays one object:

Name	Type	Last modified	Size	Storage class
DataTH R2AE Live Analytics Engineer & Modern Data Stack (3).png	png	August 19, 2023, 11:55:35 (UTC+10:00)	638.8 KB	Standard

The file name 'DataTH R2AE Live Analytics Engineer & Modern Data Stack (3).png' is highlighted with a red box.

5. คลิกปุ่ม Download เพื่อเชฟไฟล์จาก S3 Bucket

The screenshot shows the AWS S3 object preview page for the file 'DataTH R2AE Live Analytics Engineer & Modern Data Stack (3).png'. The top navigation bar shows the path: Amazon S3 > Buckets > perth-test-bucket-ja > DataTH R2AE Live Analytics Engineer & Modern Data Stack (3).png.

The main content area shows the file name 'DataTH R2AE Live Analytics Engineer & Modern Data Stack (3).png' with a 'Info' link. To the right of the file name is a toolbar with buttons for Copy S3 URI, Download (which is highlighted with a red box), Open, and Object actions.

เกร็ดเสริม: เราสามารถเขียนโค้ด เพื่ออัปโหลด & อ่านไฟล์จาก S3 Bucket ได้ด้วย

วิธีอัปโหลด: <https://boto3.amazonaws.com/v1/documentation/api/latest/guide/s3-uploading-files.html>

วิธีอ่านไฟล์: <https://boto3.amazonaws.com/v1/documentation/api/latest/guide/s3-example-download-file.html>

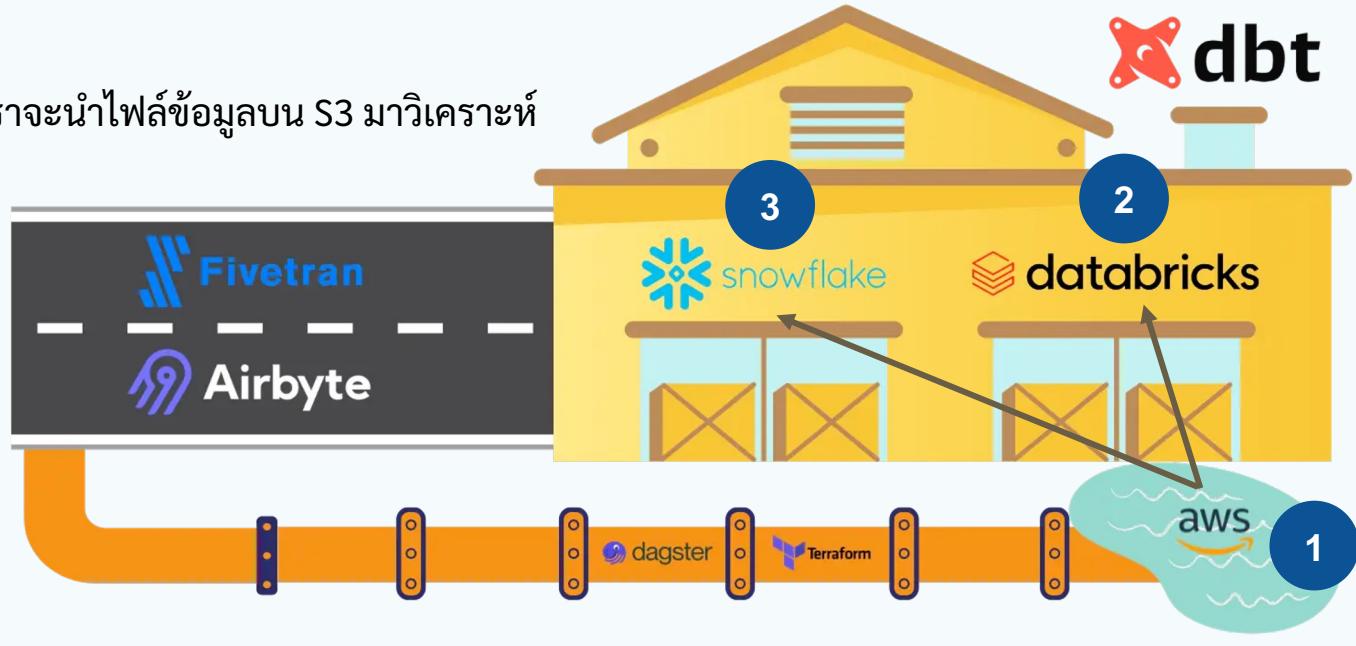


DATA TH

การใช้ S3 ในงานจริง

โปรเจค Web, App, Data ใช้ S3 ได้หมด นิยมนำมาเก็บข้อมูล เพื่อนำไปใช้งานต่อ หรือเพื่อใช้ Backup

ในคอร์สนี้ เราจะนำไฟล์ข้อมูลบน S3 มาวิเคราะห์





เบรก 3 นาที

หลังเบรก

- Intro to Git
- Demo: GitHub

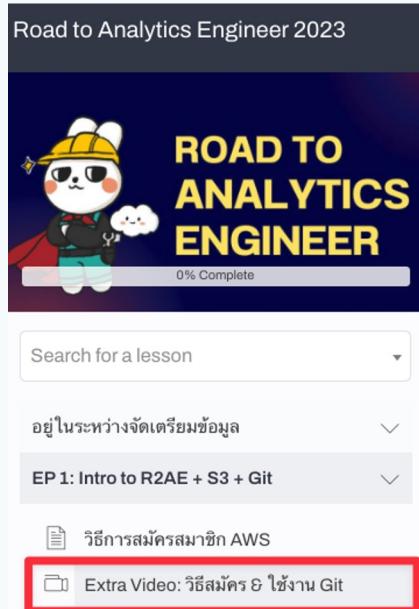


Intro to Git

Version Control System & GitHub

วันนี้เราจะแนะนำพื้นฐานสำคัญของ Git
แนะนำให้ชม วิดีโอ Special Live: Git ในระบบเรียน
school.datath.com เพื่อให้รู้จัก Git แบบเต็ม ๆ

Road to Analytics Engineer 2023



ROAD TO ANALYTICS ENGINEER

0% Complete

Search for a lesson

อยู่ในระหว่างจัดเรียนข้อมูล

EP 1: Intro to R2AE + S3 + Git

วิธีการสมัครสมาชิก AWS

Extra Video: วิธีสมัคร & ใช้งาน Git

Live: Git Essentials

มาทำความรู้จัก Git และวิธีใช้งานเข้าใจง่าย ๆ ได้ลองทำกันจริง



Aod Pich
Senior Data Engineer
at Servian

Fony Fon
Senior Machine Learning
Engineer at Sertis

git

GitHub

DataTH.com

ปัญหาของการทำงานกับโค้ด

ทุกอาชีพที่ต้องทำงานกับโค้ด (Software, Data etc.) จะเจอปัญหาเดียวกัน คือ



FinalOfFinalJingJing.zip



ไม่มีการเก็บประวัติการแก้ไข

แก้ไขที่ ก็เชฟทับไฟล์ล่าสุดที่
ไม่มีวิธีดูว่าไฟล์ไหน
เปลี่ยนยังไงบ้าง

ทำงานหลายคนยก

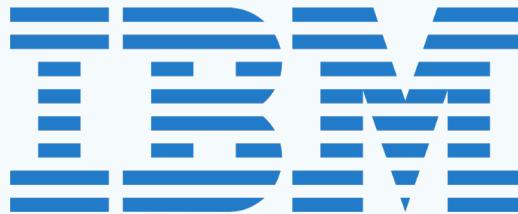
ต้องดาวน์โหลดโค้ดที่อัพเดท
ล่าสุดให้คนอื่นในทีม
ตลอดเวลา

คอมพัง = โค้ดหาย

ถ้าเก็บโค้ดไว้ในคอมเรา ถ้าคอม
พังโค้ดก็หาย
(ส่วนถ้าใครจะใช้ Dropbox
ย้อนไปข้อแรก)



จึงมีการคิดค้นระบบ Version Control System (VCS)



Linus Torvalds
ผู้พัฒนา Linux

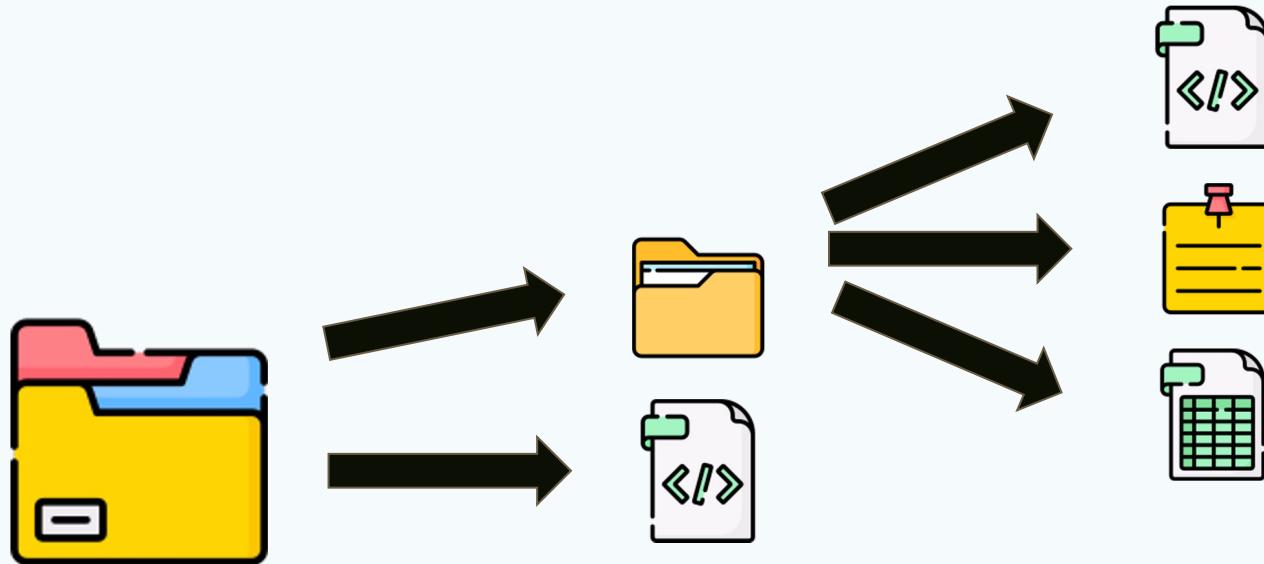


เริ่มคิดค้น VCS ตั้งแต่ 1962

พัฒนา และเปิดตัว Git ในปี 2005



Git = ที่เก็บโค้ด คล้ายการเก็บในคอมเรา



Repository (= โฟลเดอร์ใหญ่)
สามารถสร้างกี Repository ก็ได้

Folders & Files
สร้าง Folder และ File ด้านใน
Repository ได้อิสระ

ปัญหาที่ Git ช่วยแก้ 1: เก็บประวัติ & Backup ไฟล์

The image shows two screenshots from GitHub. The left screenshot displays a list of commits for August 16, 2023, and August 15, 2023. The right screenshot shows a detailed code diff between two versions of a file, highlighting changes made to the 'runner.Start()' function.

Commits on Aug 16, 2023

- Update BUSL LICENSE to include licensed product and version. (#33697) Verified 0706867
- Merge pull request #33678 from hashicorp/update-linux-package-license ... Verified dc25562
- Introduce 'run' keyword for referencing outputs from earlier run bloc... Verified 9742f22

Commits on Aug 15, 2023

- Add support for sensitive values in test assertions Verified 6a04e98
- simplify test_context evaluation (#33677) Verified 836c64f
- Update error message from an Terraform Test condition Verified cb42237

Showing 22 changed files with 888 additions and 167 deletions.

Line	Line	Change
23	24	"github.com/hashicorp/terraform/internal/plans"
251	252	defer stop()
252	253	defer cancel()
253	254	
254	-	runner.Start(variables)
255	+	runner.Start()
256	256	}
257	257	// Wait for the operation to complete, or for an interrupt to occur.
258	258	@@ -299,8 +300,10 @@ func (c *TestCommand) Run(rawArgs []string) int {
299	300	return 0

จากรูป: โครงสร้างไฟล์ในบรรทัดไหน ดูได้หมด (จาก [GitHub: Terraform](#))

Git เก็บ “ประวัติการแก้ไข” ให้กับ “ทุกไฟล์”

- ไม่ว่าเราจะอยากรู้ประวัติไฟล์เลย ๆ หรือลูกค้าอยากย้อนไปใช้โค้ดเมื่อ 3 วันก่อน ก็จัดให้ได้
- โค้ดไม่มีวันหาย เพราะทุกการแก้ไขจะเก็บอยู่บน Git เขียนโค้ดผิดก็ย้อนกลับได้ตลอด



ปัญหาที่ Git ช่วยแก้ 2: การทำงานเป็นทีม



ทุกคนมี ชุดโค้ด ที่มา
จากแหล่งเดียวกัน

ทุกคนทำงานบนคอมพิวเตอร์
ตัวเองได้ แบ่งแยกโค้ดของแต่
ละคนออกจากกัน
(ระบบ Branch)

```
README.txt
```

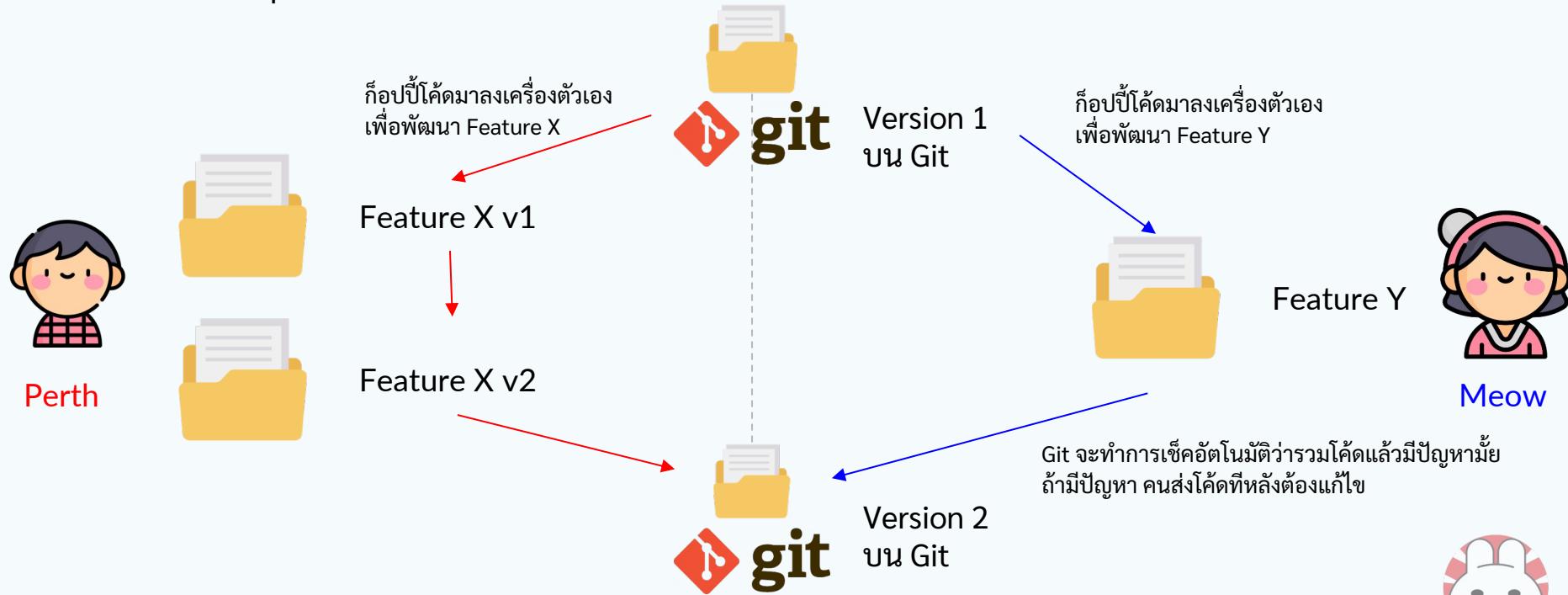
```
1 <<<<< patch-2
2 Paradigm Shift by HTML5 UP 12345
3 =====
4 Paradigm Shift by HTML5 UP
5 >>>>> master
6 html5up.net | @ajlkn
```

ทำงานเป็นทีมได้ง่าย
ไม่เซฟโค้ดทับกัน

หากมีหลายคนแก้ไขไฟล์เดียวกัน
Git จะป้องกันไม่ให้โค้ดทับกัน
(เรียกว่า การแก้ไข Merge Conflict)

ສະuu Branch ใน Git ทำงานอย่างไร

1 Branch = 1 ชุดโค้ด



เกร็ดเสริม: 1 คน มีกี่ Branch ก็ได้นะ (สำหรับคนชอบ Multi-tasking)

Git Server เป็นตัวกลางในการรับ-ส่ง-เก็บข้อมูล ด้วยระบบ Git

มีหลายระบบให้เลือกใช้



Bitbucket



AWS CodeCommit



Google Cloud
Source Repositories



GitHub



GitLab



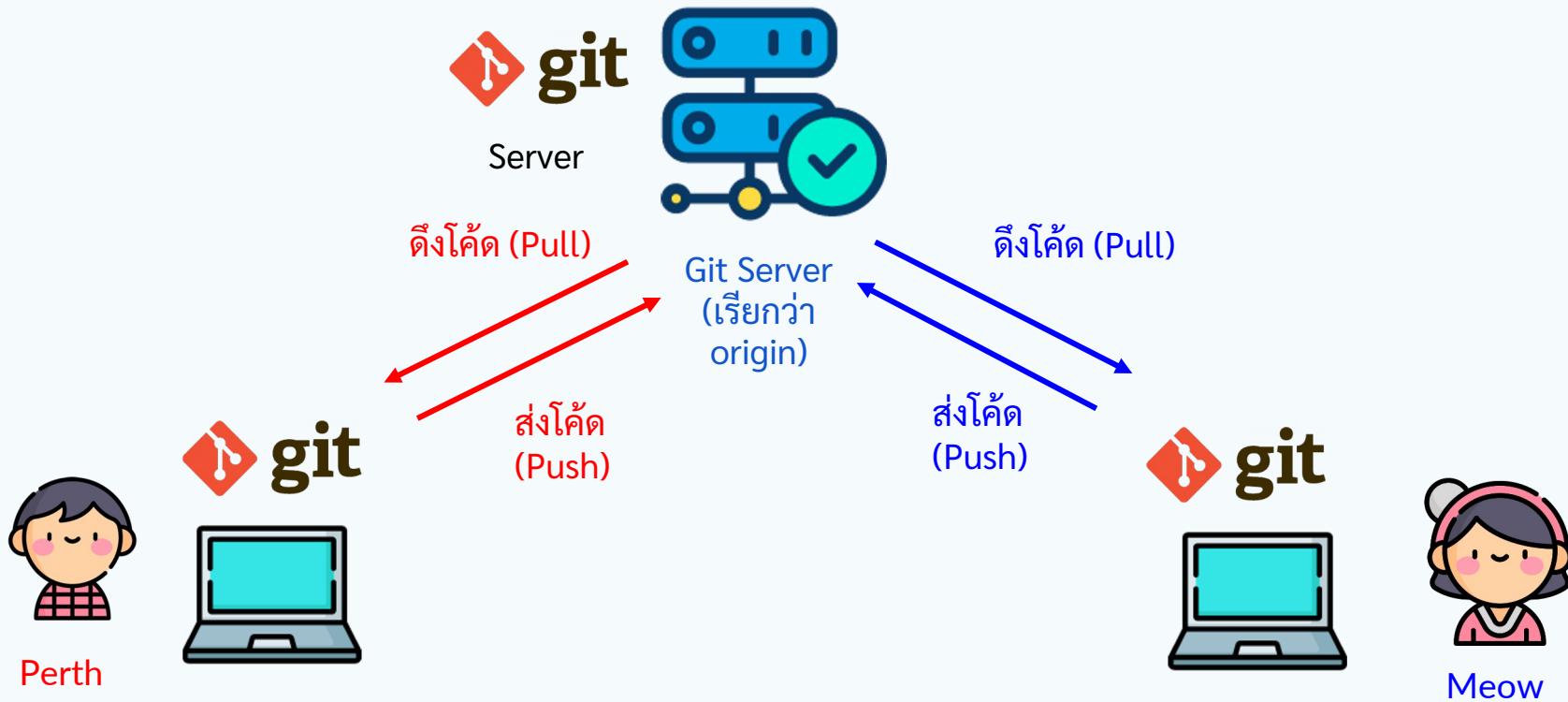
Gitea



Azure Repos



การใช้ Git ระหว่าง Computer ของคุณ <-> Git Server



Github คืออะไร?



GitHub

ผู้ให้บริการระบบ Git Server ออนไลน์ ที่ได้รับความนิยมมากที่สุดแห่งหนึ่ง เริ่มให้บริการตั้งแต่ปี 2008

ปัจจุบันมีผู้ใช้กว่า 40 ล้านคนทั่วโลก และมี Repository กว่า 190 ล้าน Repository

ค่าใช้จ่าย

- ใช้งานได้ฟรี สำหรับฟีเจอร์พื้นฐาน สร้าง Repository ได้ไม่จำกัด
- ชำระเงิน เพื่อฟีเจอร์ที่เยอะขึ้น / ดีขึ้นได้



เกร็ดเสริม: GitHub ถูกซื้อด้วย Microsoft ในปี 2018



Repository (Repo) 2 ประเกกบูน GitHub



Public Repository

Repository ที่ทุกคนสามารถเข้ามาดูโค้ดได้
หมายความว่า กับโค้ดบน Github ที่

- ใช้สมัครงาน เช่น โปรเจค Data
- อยากแชร์ให้คนทั่วไป (Open Sourced Projects)



Private Repository

Repository ที่เห็นเฉพาะเราและคนในทีม
เท่านั้น

หมายความว่า กับโค้ดของโปรเจคส่วนตัว หรือโปรเจค
ของบริษัท

โปรแกรมทำงานกับ Git

โปรแกรมเขียนโค้ดที่แนะนำในทำงานกับ Git

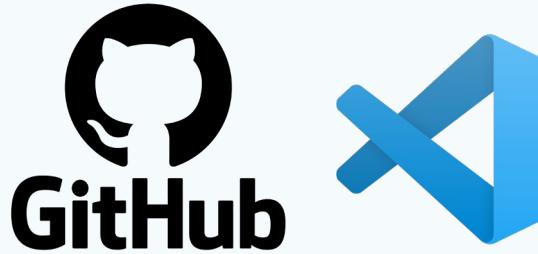


Visual Studio Code

โปรแกรมเขียนโค้ด รองรับ
Windows, Mac, Linux ดาวน์โหลด
ฟรี จาก Microsoft

<https://code.visualstudio.com/>

สำคัญ: สำหรับคนที่ติดตั้ง Visual Studio Code บน Windows
จะต้องติดตั้ง Git เพิ่ม ดาวน์โหลดที่ <https://git-scm.com/>



Github.dev, vscode.dev

หน้าตาเหมือน Visual Studio Code แต่
เป็นเวอร์ชันออนไลน์ (ฟีเจอร์น้อยกว่าnid
หน่อย) ใช้งานได้ฟรี

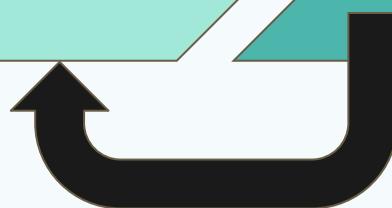
ขั้นตอนการทำงานกับ Git

การทำงานกับ Git แบ่งเป็น 3 ขั้นตอนหลัก ๆ

ขั้นตอนที่ 1:
สร้าง Repo ใหม่
หรือดึง Repo บน Git Server มาลง
เครื่องเรา

ขั้นตอนที่ 2:
แก้ไขไฟล์ในเครื่อง
เรา จนกว่าจะพอใจ

ขั้นตอนที่ 3:
ส่งโค้ดที่แก้ไขแล้ว
ขึ้น Git Server



ทำวนไป จนกว่างานจะเสร็จ



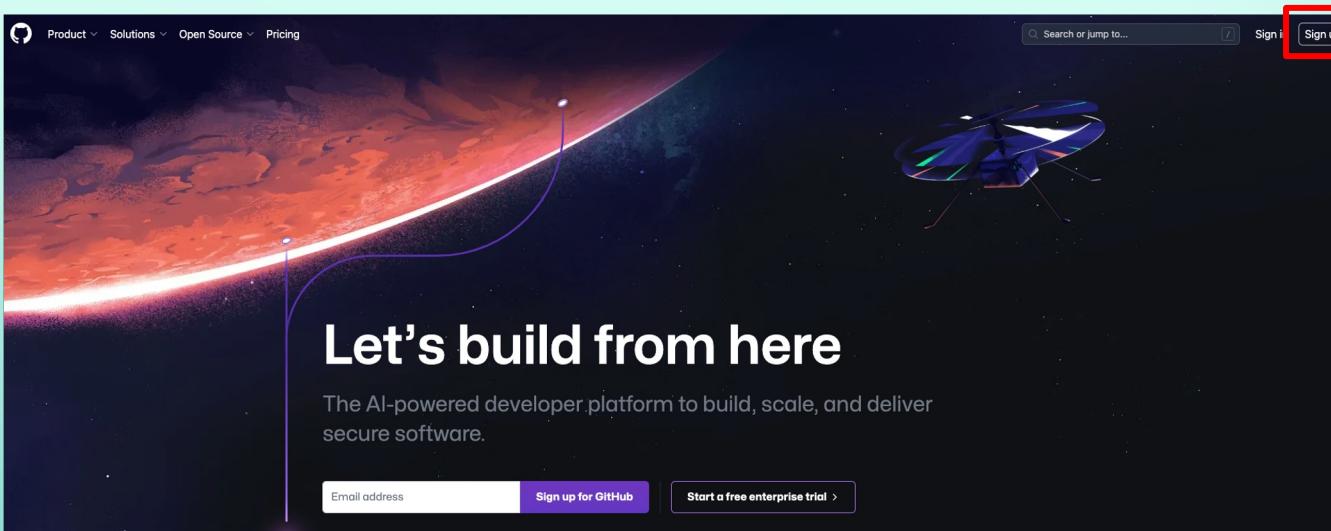
Demo: มาลองใช้ Git บน GitHub กัน

ใน Demo นี้เราจะทำอะไรบ้าง

- ก่อนเริ่ม: สมัครสมาชิก GitHub.com (ฟรี)
- Part 1: สร้าง Repo ของตัวเอง & เปิดใช้ใน github.dev
- Part 2: ก็อปปี้ Repo ชาวบ้าน & เปิดใช้ใน github.dev



ก่อนเริ่ม: สมัครสมาชิก บนเว็บไซต์ Github.com



ขั้นตอนในการสมัคร

1. กรอกฟอร์มสมัครสมาชิก
2. กรอกโค้ดยืนยันตัวตน ซึ่งจะส่งไปที่อีเมลของคุณ



DATA TH

สไลด์ต่อไป จะเป็นขั้นตอน 1

การทำงานกับ Git แบ่งเป็น 3 ขั้นตอนหลัก ๆ

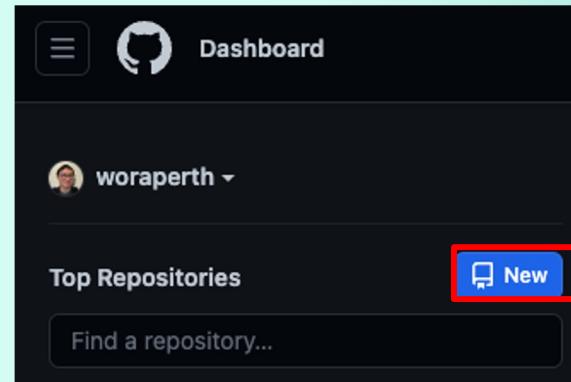
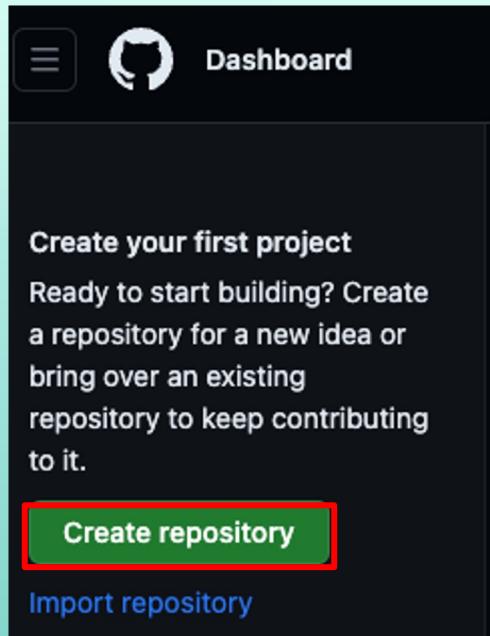
ขั้นตอนที่ 1:
สร้าง Repo ใหม่
หรือดึง Repo บน Git Server มาลง
เครื่องเรา

ขั้นตอนที่ 2:
แก้ไขไฟล์ในเครื่อง
เรา จนกว่าจะพอใจ

ขั้นตอนที่ 3:
ส่งโค้ดที่แก้ไขแล้ว
ขึ้น Git Server



Part 1.1: สร้าง Repo ใหม่ & เปิดอ่าน หลังจากสมัครสมาชิกแล้ว เข้า Github.com



เกร็ดเสริม: ถ้าเป็นการทำงานในบริษัท ปกติจะมีคนสร้าง Repo อยู่แล้ว เราไปดึงมาใช้ได้เลย (ข้ามไป Part 1.3 เลย)

Part 1.2: สร้าง Repository

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Required fields are marked with an asterisk (*).

Owner * Repository name *

 perhtester / test-repo
test-repo is available.

Great repository names are short and memorable. Need inspiration? How about [fuzzy-octo-parakeet](#)?

Description (optional)
This is my first repo

 Public
Anyone on the internet can see this repository. You choose who can commit.
  Private
You choose who can see and commit to this repository.

Initialize this repository with:
 Add a README file
This is where you can write a long description for your project. [Learn more about READMEs](#).

Add .gitignore
.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files](#).

Choose a license
License: None

A license tells others what they can and can't do with your code. [Learn more about licenses](#).

This will set  main as the default branch. Change the default name in your [settings](#).

ⓘ You are creating a public repository in your personal account.

Create repository

Repository name: ชื่อ Repo

Description: คำอธิบายว่า Repo นี้เกี่ยวกับอะไร

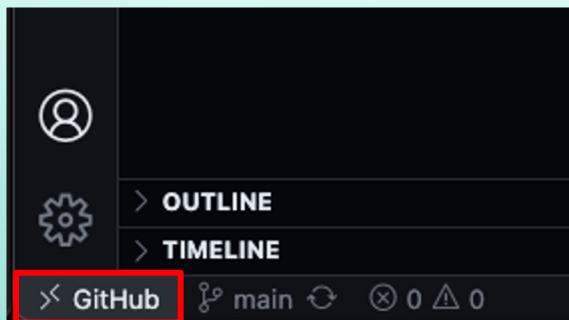
- เลือก Public Repo หรือ Private Repo ก็ได้

- ติํก Add a README file เพื่อให้สร้าง Repo ขึ้นมาพร้อมไฟล์ชื่อ README.md



Part 1.3: เปิดโค้ดใน Github.dev

1. เข้า GitHub.dev
2. คลิกคำว่า GitHub ที่มุ่งซ้ายล่าง



3. เลือกเมนู Open Remote Repository > Repository from GitHub > เลือก Repository ที่เราต้องการ



สไลด์ต่อไป จะเป็นขั้นตอน 2-3

การทำงานกับ Git แบ่งเป็น 3 ขั้นตอนหลัก ๆ

ขั้นตอนที่ 1:
สร้าง Repo ใหม่
หรือดึง Repo บน Git Server มาลง
เครื่องเรา

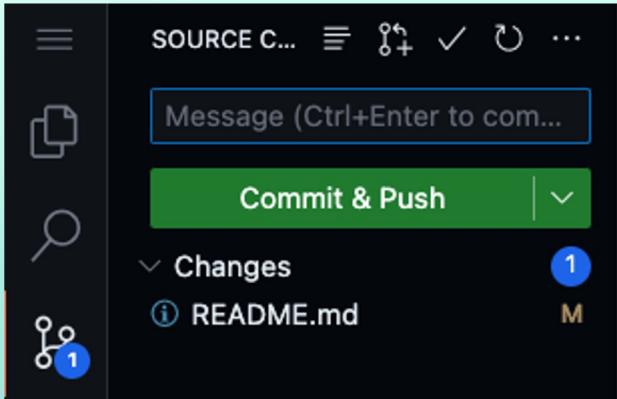
ขั้นตอนที่ 2:
แก้ไขไฟล์ในเครื่อง
เรา จนกว่าจะ
พอใจ

ขั้นตอนที่ 3:
ส่งโค๊ดที่แก้ไขแล้ว
ขึ้น Git Server



Part 1.4: การแก้ไขไฟล์ > ส่งขึ้น GitHub

1. เปิดไฟล์ที่ต้องการ ขึ้นมาแก้ไข แล้วเซฟ
2. เปิดแท็บ Source Control จะเห็นว่าเราแก้ไขไฟล์ไหนไปบ้าง



3. เอาเม้าส์ไปซื้อไฟล์ แล้วกด + เพื่อเป็นการบอก Git ว่าเราจะส่งอัพเดทของไฟล์นี้ไป (เรียกว่าขั้นตอน git add)



4. พิมพ์ข้อความในช่อง Message เพื่อบอกว่าเราทำอะไรไป แล้วคลิกปุ่มเขียว Commit & Push (เรียกว่าขั้นตอน git commit และ git push)

เกร็ดเสริม:

ในการทำงานในบริษัท เรามักจะแก้ไขไฟล์โดยสร้าง Branch ขึ้นมาก่อน เพื่อให้แต่ละคนทำงานแยกกันได้โดยไม่ตีกัน

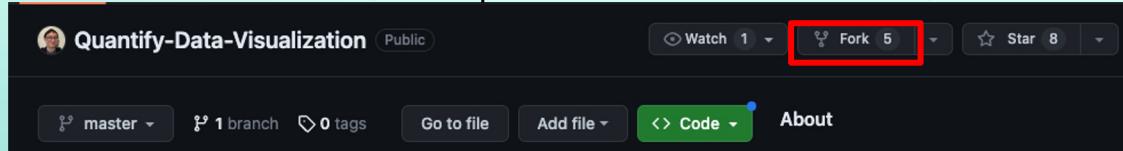
วิธีสร้าง Branch: <https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/proposing-changes-to-your-work-with-pull-requests/creating-and-deleting-branches-within-your-repository>



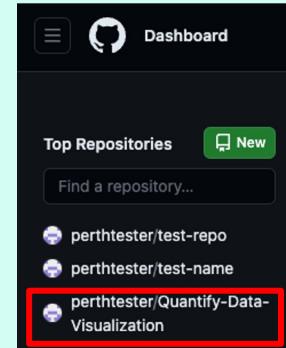
Part 2: ก็อปปี Repo ชาวบ้าน & เปิดบน GitHub.dev

ในคอร์สนี้ อาจจะมีการก็อปปี Repository ของผู้สอน บน GitHub เพื่อนำมาแก้ไขโค้ดเป็นของเรา

1. เข้าลิ้งค์ของ Repo คนอื่น หรือเข้า github.com/ชื่อคนอื่น ไปเลือก Repo ที่ต้องการ ก็ได้
 - a. เช่น github.com/woraperth
2. ในหน้าหลักของ Repo กดปุ่ม Fork (Fork = การก็อปปี Repo มาใส่ใน Account ของเรา)



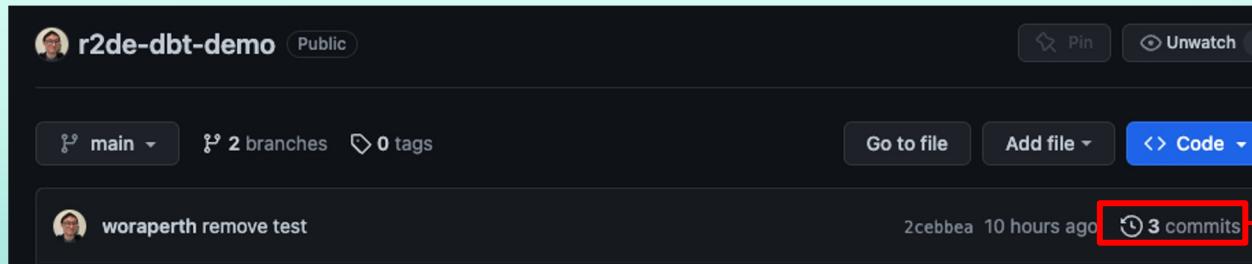
3. จะมีหน้าเว็บให้ยืนยันรายละเอียดขึ้นมา กด Create Fork
4. จะเป็นการก็อปปี Repo ที่ต้องการ มาใส่ใน GitHub account ของเรา (เช็คได้ที่ GitHub.com ดังรูป ->)
5. เปิดดูโค้ดบน GitHub.dev ด้วยวิธีเดียวกับ Part 1.3



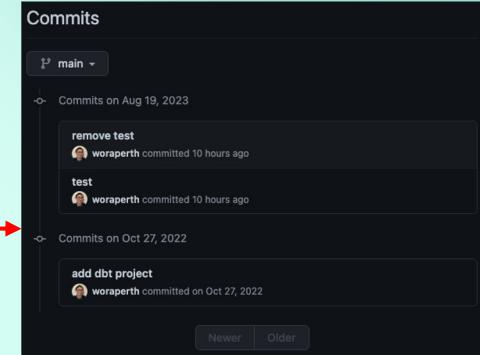
Bonus: การดูข้อมูลประวัติโค้ด บน GitHub

วิธีดูประวัติโค้ดทั้งโปรเจค

ในหน้าหลักของ Repo คลิกที่ Commit > คลิกเข้าไปในแต่ละ Commit เพื่อดูบรรทัดโค้ดที่เปลี่ยนได้



The screenshot shows the GitHub repository page for 'r2de-dbt-demo'. At the bottom of the commit list, there is a timestamp '2cebbea 10 hours ago' followed by a link '3 commits'. This link is highlighted with a red box and an arrow pointing to the detailed commit history shown in the adjacent window.

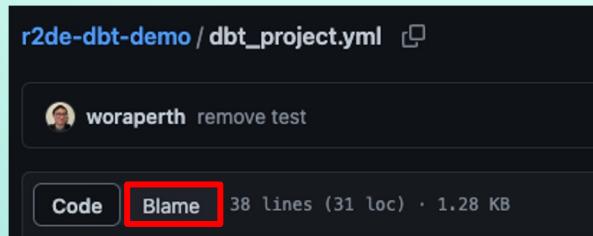


The screenshot shows the detailed commit history for the 'main' branch. It lists three commits:

- remove test (committed 10 hours ago)
- test (committed 10 hours ago)
- add dbt project (committed on Oct 27, 2022)

วิธีดูประวัติโค้ดเฉพาะไฟล์ที่ต้องการ (มีประโยชน์มาก เวลาอยากรู้ว่าไฟล์นี้ใครแก้ตรงไหนล่าสุด)

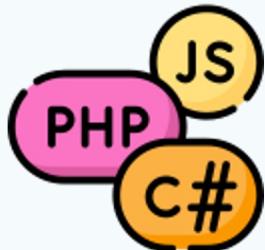
กดชื่อไฟล์ใน Repo > กด Blame



The screenshot shows the GitHub blame view for the file 'dbt_project.yml'. The 'Blame' tab is selected, highlighted with a red box. The view displays the blame information for the file, showing contributions from 'woraperth'.



Use Cases: Git ในโลกการทำงานจริง



ใช้ได้กับทุกโปรเจคที่ต้องเขียนโค้ด

Git เป็นที่นิยมทั้งในงาน Software & Data
ใช้กับภาษาโปรแกรมมิ่งอะไรก็ได้



ใช้คนเดียวก็ได้ ใช้เป็นทีมก็ยิ่งดี

Git ช่วยการในการทำให้โปรเจคโค้ดเป็นระเบียบ
และเก็บ History ทุกขั้นตอนโดยอัตโนมัติ



นอกจาก GitHub มีตัวอื่นมั้ย (Cloud & On-prem version)

Git Server ที่อยู่บน Cloud



Bitbucket



GitLab



AWS CodeCommit



Google Cloud
Source Repositories



Azure Repos

Git Server ที่ติดตั้ง On-prem ได้



GitLab

Self-Managed (ฟรี - เสียเงิน)



Gitea



Bitbucket

Data Center



DATA TH

สรุป คีย์สำคัญของ AWS S3 & Git



Data Lake & AWS S3

- Data Lake คือ บริการเก็บไฟล์ นำไปใช้ประโยชน์ได้หลากหลาย ทั้งงาน Software & Data
 - ไฟล์ใน Data Lake จะไม่หายง่าย ๆ ความปลอดภัยสูง
- AWS S3 เป็น บริการ Data Lake ที่ได้รับความนิยมมาก ของ Cloud เจ้าใหญ่ที่สุด
 - ใช้งานง่าย ใช้ผ่านเว็บ หรือผ่านโค้ดก็ได้



Git & GitHub

- Git คือระบบ Version Control
 - เก็บประวัติโค้ดไว้หมด
 - หมายความว่าการทำงานเป็นทีม
- GitHub เป็นบริการ Git Server ออนไลน์ที่ได้รับความนิยมมาก
 - โค้ดทุกอย่างเก็บบนระบบออนไลน์ คอมพ์ ไฟฟ้า แม้ น้ำท่วม โค้ดไม่หาย

ช่วง Live Q&A

ยิ่งถาม ยิ่งได้เรียนรู้

sli.**do**

เข้าไปที่เว็บ sli.do พิมพ์ชื่อห้อง
#r2ae-class1

(สำหรับคนที่ไม่ต้องการออกสื่อ เว็บนี้ถูกออกแบบ
Anonymous ได้เลย ไม่ต้องพิมพ์ชื่อ)



DATA TH