

การตั้งค่า API Service สำหรับดึงข้อมูลจาก Google Sheet ด้วย Python



Mr. Rukpong [Follow](#)

Mar 22 · 4 min read

ถ้าพูดถึงการเก็บข้อมูลแล้ว นอกจาก Database ที่เก็บในเครื่อง Local หรือ Server แล้ว เราสามารถดึงข้อมูลจาก Google Sheet ได้ด้วย

ข้อมูลใน Google Sheet ที่เราใช้การในปัจจุบัน เพื่อเก็บข้อมูล online ในรูปแบบ excel ซึ่งเราค้นเคยและใช้เป็นประจำอยู่แล้ว

ตัวอย่างข้อมูลการเก็บรายชื่อหนังสือ

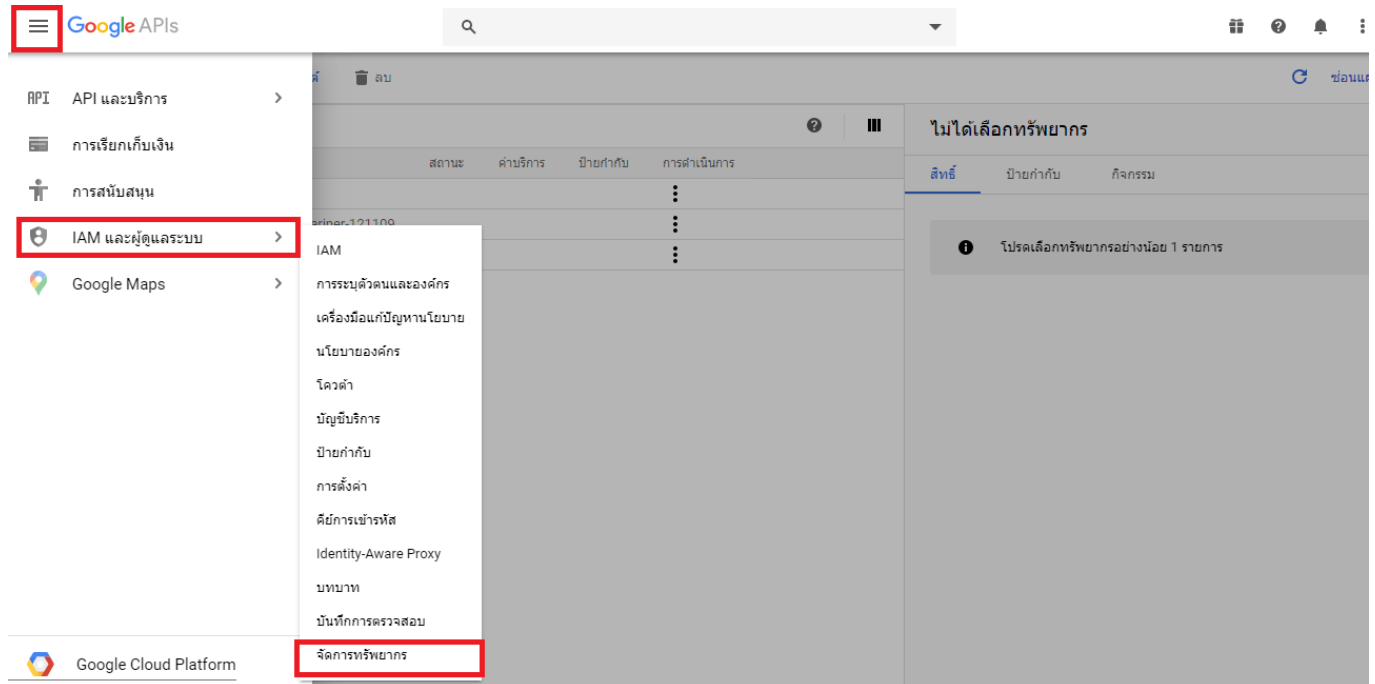
fx							
	A	B	C	D	E	F	G
1	Id	รายชื่อหนังสือ	จำนวนหน้า	ประเภท	สำนักพิมพ์	ผู้แปล	ราคา
2	001	Biomolecule World	1239	วิทยาศาสตร์	Ano	เกสร	1250
3	002	การใช้งาน Python เบื้องต้น	253	คอมพิวเตอร์	สามารถ		259
4	003	สามก๊ก	2557	วรรณกรรม	สุวรรณ		2137
5	004	ขั้นตอนสร้างรายได้ จากการถ่ายรูป	390	คอมพิวเตอร์	กฤษณ์		990
6	005	Crezy Animation	954	คอมพิวเตอร์	Philip	จิน	1500
7	006	Expert SQL Database	1211	คอมพิวเตอร์	John-D	มรกต	1330
8							
9							

ที่นี่เราจะมาดึงข้อมูลเหล่านี้กัน เพื่อนำไปแสดงผล, กรอก หรือค้นหา ได้สะดวก และเข้าถึงข้อมูลได้ โดยไม่ต้องนำข้อมูลจาก Google Sheet ไปเก็บใน Database ที่ไหนเลย

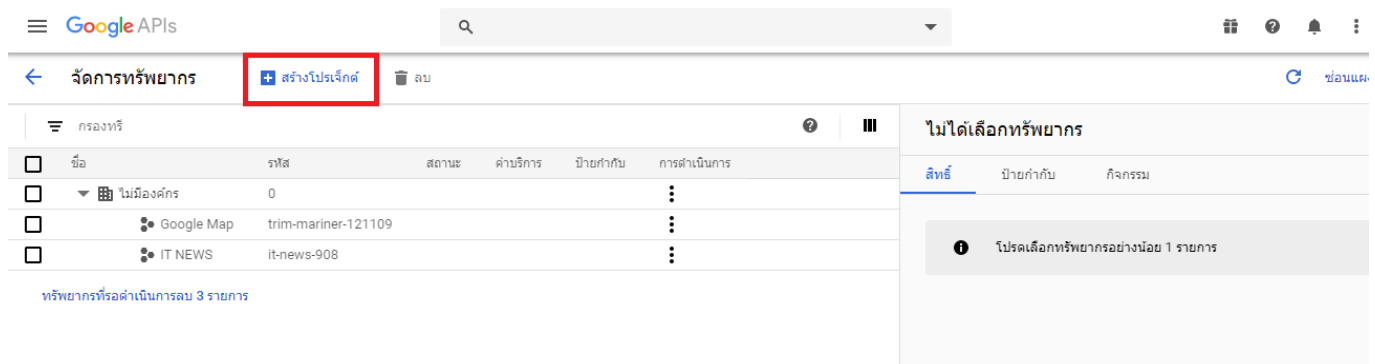
โดยการเปิดช่องทางการเข้าถึง Google Sheet และกำหนดสิทธิ์ให้ Application (Python) ที่เราต้องการไปเข้าถึงข้อมูลนั้นๆ ผ่าน Google Sheet API for Python โดยมีวิธีการดังนี้

1. เริ่มสร้าง Project เพื่อเตรียมช่องทางการเข้าถึงข้อมูลใน Google Sheet ตามลึ้งนี้เลย
ครบ <https://console.developers.google.com/> และ เลือกที่ปุ่ม Hamburger -> ไปที่ IAM

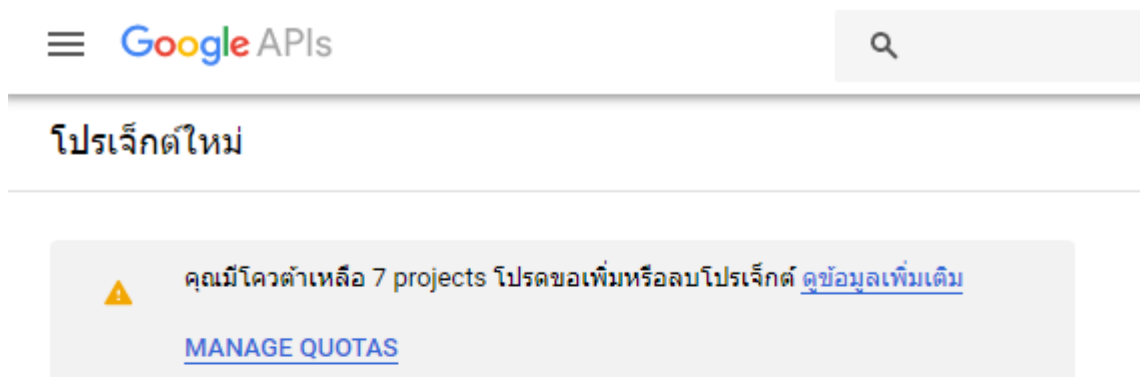
และผู้ดูแลระบบ -> เลือกจัดการทรัพยากร



2.หน้าจัดการทรัพยากร จะแสดงรายการ Project ทั้งหมดที่เราเคยสร้างไว้ เราสามารถเลือก Project เดิม หรือสร้าง Project ใหม่ก็ได้ครับ ในที่นี้ ผมขอสร้าง Project ใหม่ครับ



ตั้งชื่อให้ Project และกดสร้าง



ชื่อโปรเจกต์ *

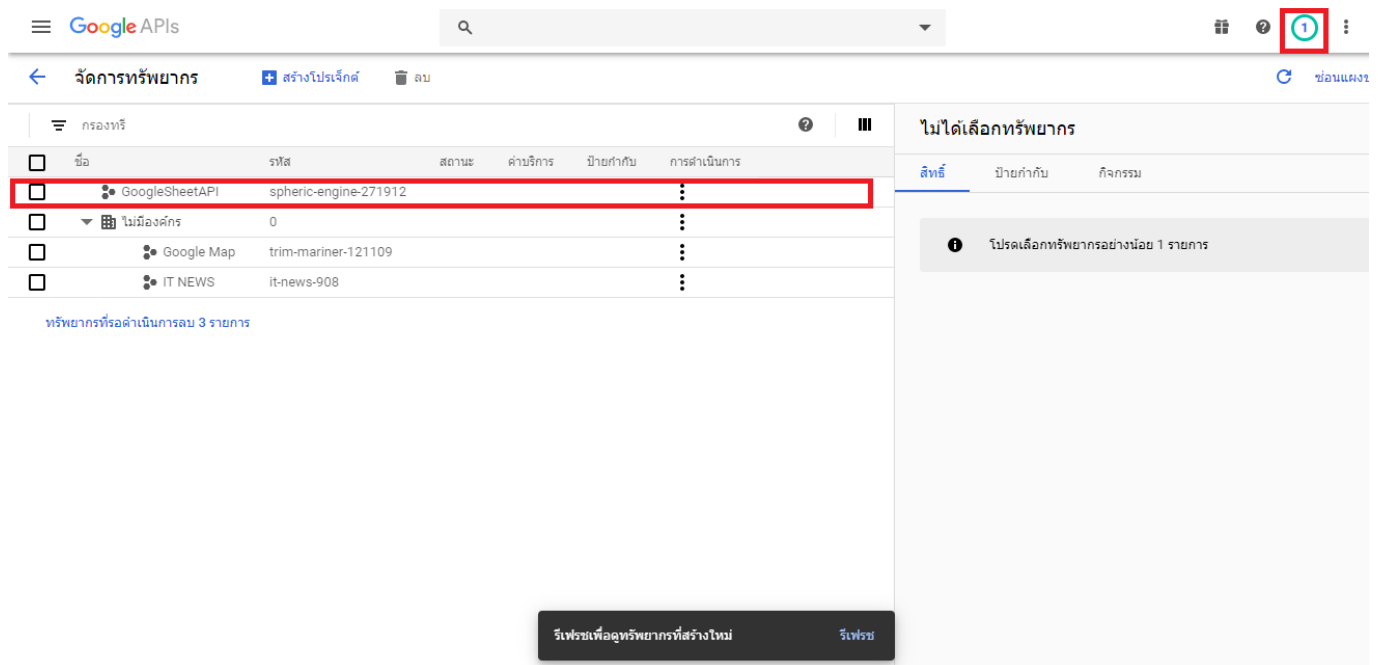
รหัสโปรเจกต์: spheric-engine-271912 จะเปลี่ยนแปลงไม่ได้ในภายหลัง แก้ไข

ตำแหน่งที่ตั้ง * [เรียนรู้](#)

องค์กรหรือโพลเดอรระดับบนสุด

[สร้าง](#) [ยกเลิก](#)

หลังจากสร้างเสร็จสมบูรณ์จะกลับมาที่หน้า จัดการทรัพยากร และจะมีรายการ Project ใหม่ที่เราสร้างขึ้นมา พร้อมกับรายการแจ้งเตือนที่มุมบนขวาด้วย



Google APIs

จัดการทรัพยากร [สร้างโปรเจกต์](#) [ลบ](#)

ชื่อ	รหัส	สถานะ	ค่าบริการ	ป้ายกำกับ	การดำเนินการ
<input type="checkbox"/> GoogleSheetAPI	spheric-engine-271912				
<input type="checkbox"/> ไม่มีองค์กร	0				
<input type="checkbox"/> Google Map	trim-mariner-121109				
<input type="checkbox"/> IT NEWS	it-news-908				

ทรัพยากรที่รอดำเนินการลง 3 รายการ

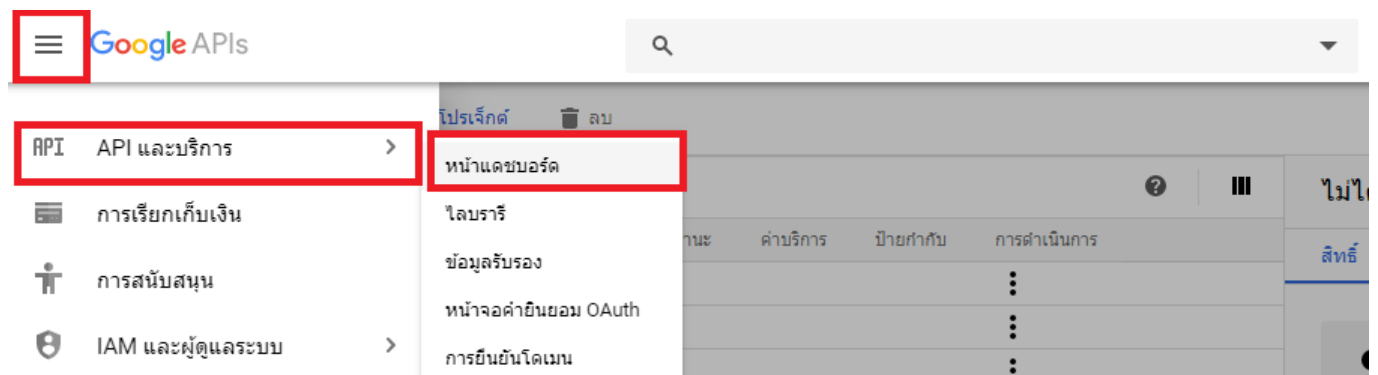
ไม่ได้เลือกทรัพยากร

สิทธิ์ ป้ายกำกับ กิจกรรม

โปรดเลือกทรัพยากรอย่างน้อย 1 รายการ

รีเฟรชเพื่อดูทรัพยากรที่สร้างใหม่ รีเฟรช

3. กดที่ปุ่ม Hamburger อีกครั้ง แล้วเลือกไปที่ API และบริการ -> เลือกหน้าแดชบอร์ด



Google APIs

API และบริการ >

- การเรียกเก็บเงิน
- การสนับสนุน
- IAM และผู้ดูแลระบบ >

โปรเจกต์ [ลบ](#)

- หน้าแดชบอร์ด
- ไลบรารี
- ข้อมูลรับรอง
- หน้าจอดำเนินการ OAuth
- การยืนยันตัวตน



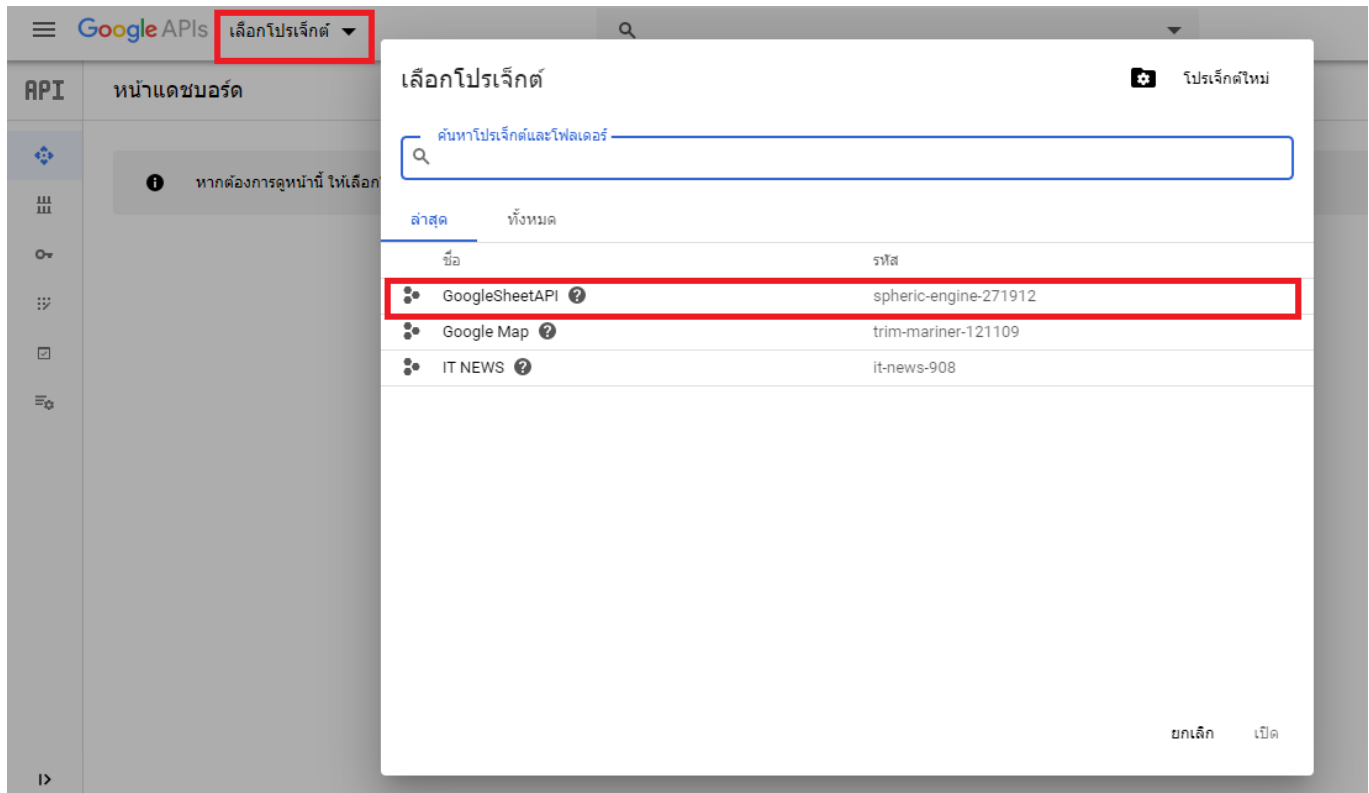
Google Maps



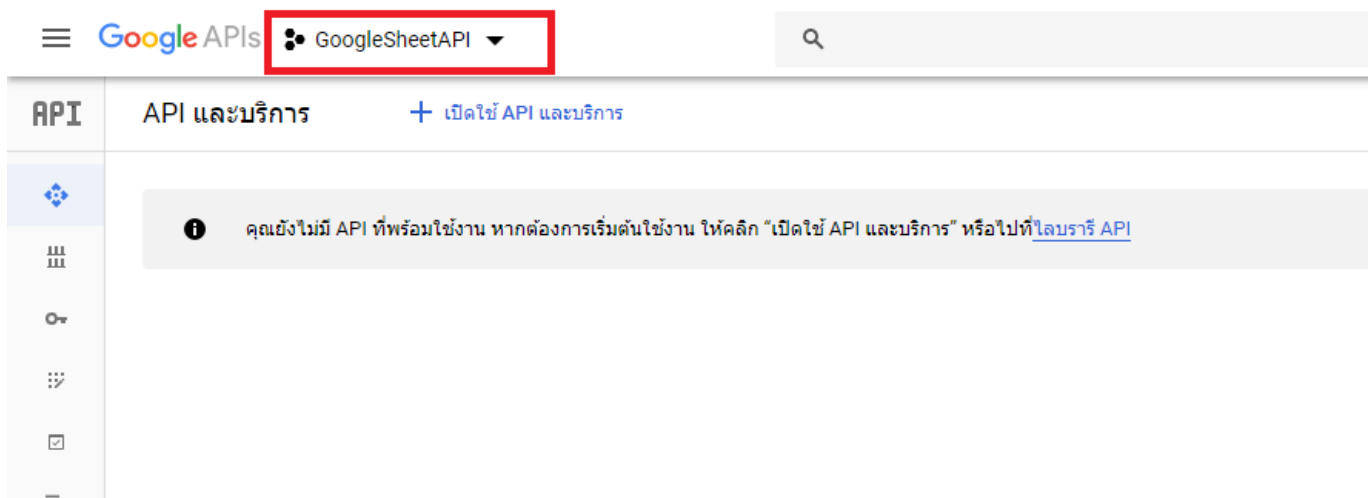
ข้อตกลงการใช้งานหน้าเว็บ



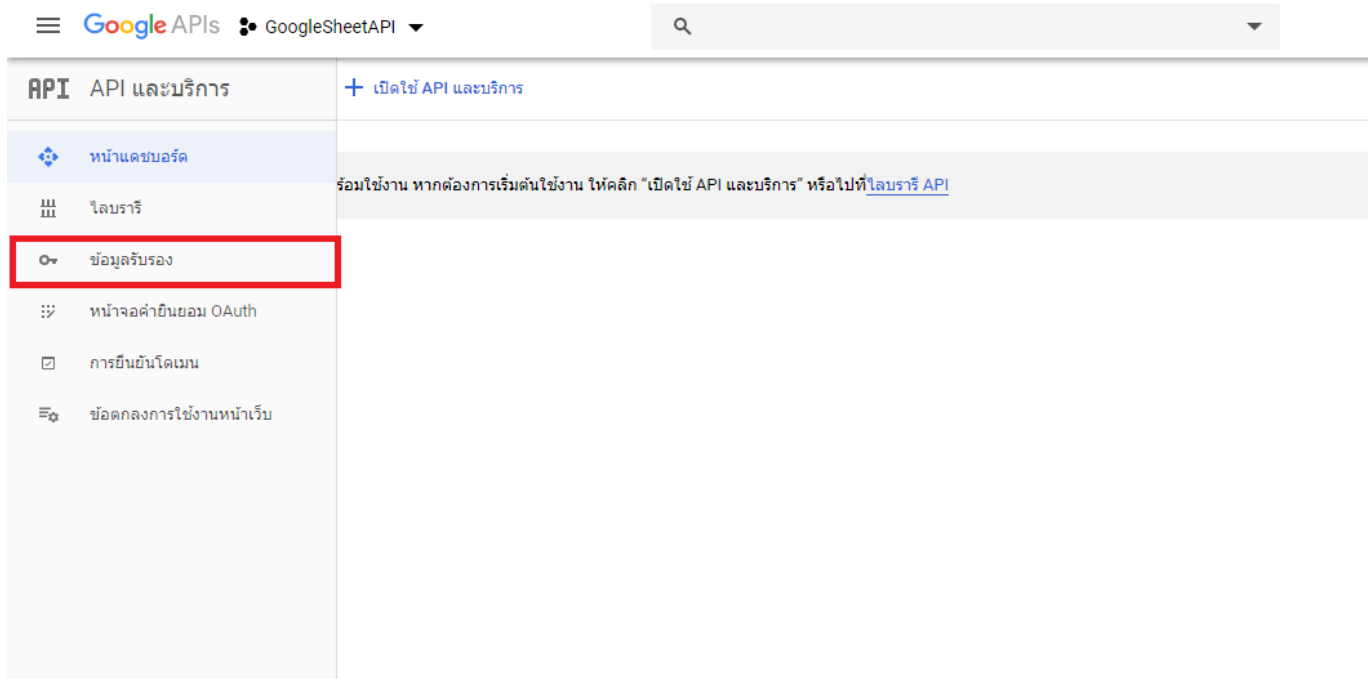
เมื่อเราไปหน้าแดชบอร์ดแล้ว จะเห็นหน้าเปล่าๆ ขึ้นมา ให้กดปุ่มเลือก Project และเลือก Project ที่เราสร้างมา



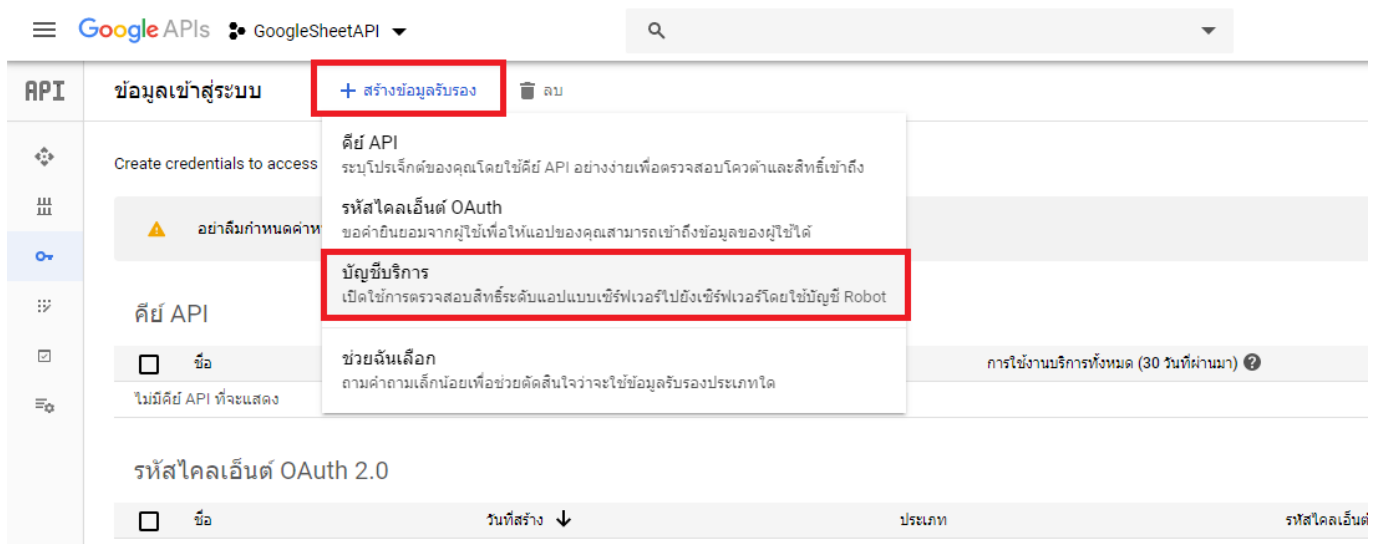
ชื่อ Project จะแสดงแทนที่ปุ่ม เลือก Project



เลือกข้อมูลรับรองที่เมนูด้านข้าง



จะเข้าไปหน้า ข้อมูลเข้าสู่ระบบ (เราจะสามารถสร้างช่องทางการเข้าถึงข้อมูลได้โดย Key API, Client OAuth และ Account Service) ในกรณีนี้ผมจะสร้างข้อมูลรับรองเป็นบัญชีบริการครับ (Credentials -> Account Service)



โมดูลโคลเลชัน OAuth2 ที่จะแสดง

บัญชีบริการ

☐ อีเมล ชื่อ ↑ การใช้งานบริการทั้งหมด (30 วันที่ผ่านมา) ?

ไม่มีบัญชีบริการที่จะแสดง

ตั้งชื่อบัญชีบริการ จะเห็นว่า รหัสบัญชีบริการจะเปลี่ยนไปตามชื่อบัญชีบริการที่ตั้ง เสร็จแล้วกดปุ่ม สร้าง

Google APIs GoogleSheetAPI

สร้างบัญชีบริการ

1 รายละเอียดบัญชีบริการ — 2 มอบสิทธิ์เข้าถึงโปรเจกต์ให้กับบัญชีบริการนี้ (ไม่บังคับ) — 3 มอบสิทธิ์เข้าถึงบัญชีบริการนี้ให้กับผู้ใช้ (ไม่บังคับ)

รายละเอียดบัญชีบริการ

ชื่อบัญชีบริการ
GoogleSheetAPI

ชื่อที่แสดงสำหรับบัญชีบริการนี้

รหัสบัญชีบริการ
googlesheetapi @spheric-engine-271912.iam.gserviceaccount.com X C

คำอธิบายบัญชีบริการ
อธิบายหน้าที่ของบัญชีบริการนี้

สร้าง ยกเลิก

ขั้นตอนต่อไป จะตั้งสิทธิ์สำหรับช่องทางการเข้าถึงข้อมูลที่เรากำลังสร้างนี้ ไม่จำเป็นต้องเลือกอะไรครับ (แต่ถ้าต้องการแก้ไข Google Sheet ได้ด้วย ให้เลือก โปรเจกต์ (Project) -> ผู้แก้ไข(Editor))

Google APIs GoogleSheetAPI

สร้างบัญชีบริการ

✓ รายละเอียดบัญชีบริการ — 2 มอบสิทธิ์เข้าถึงโปรเจกต์ให้กับบัญชีบริการนี้ (ไม่บังคับ) — 3 มอบสิทธิ์เข้าถึงบัญชีบริการนี้ให้กับผู้ใช้ (ไม่บังคับ)

สิทธิ์สำหรับบัญชีบริการ (ไม่บังคับ)

มอบสิทธิ์การเข้าถึง GoogleSheetAPI ให้กับบัญชีบริการนี้เพื่อให้ดำเนินการที่เฉพาะเจาะจงกับทรัพยากรต่างๆ ในโปรเจกต์ได้ [ดูข้อมูลเพิ่มเติม](#)

เลือกบทบาท

Type to filter

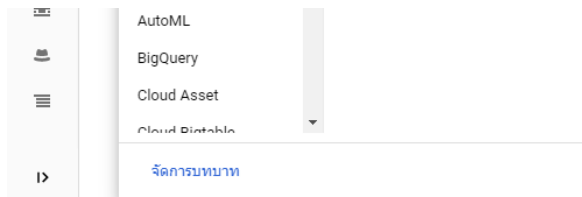
โปรเจกต์

เจ้าของ

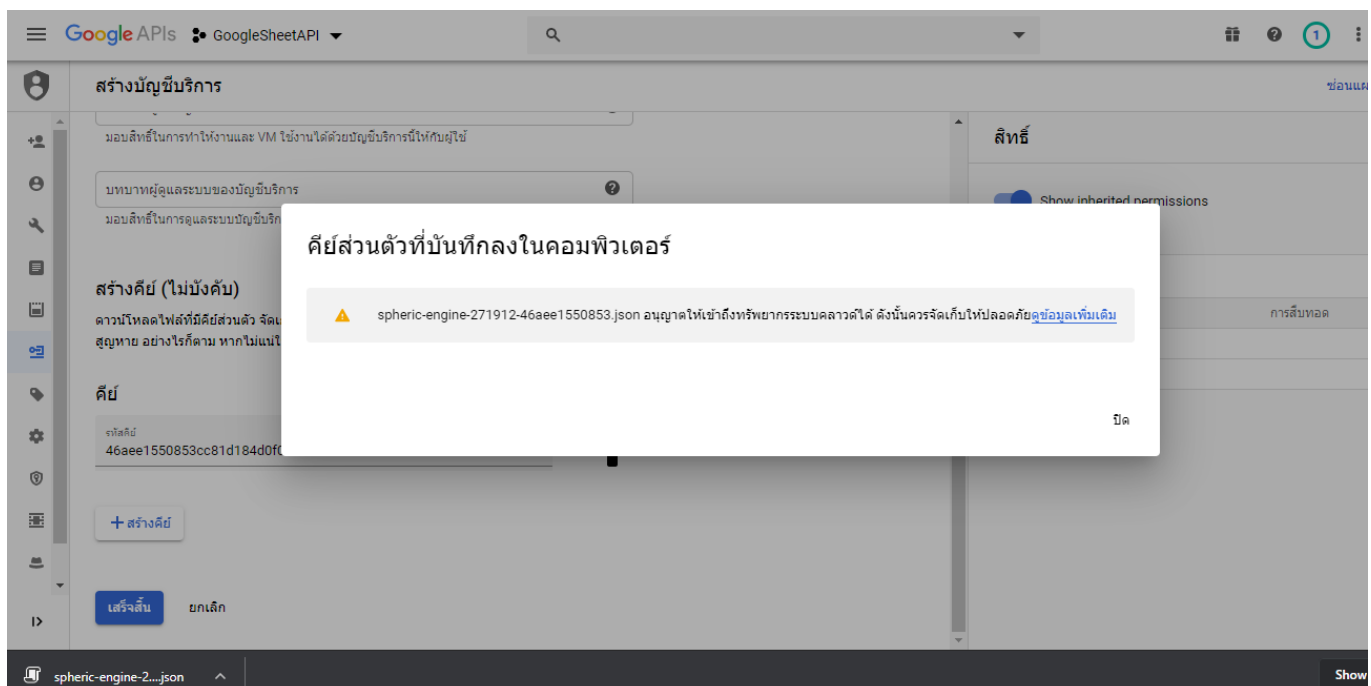
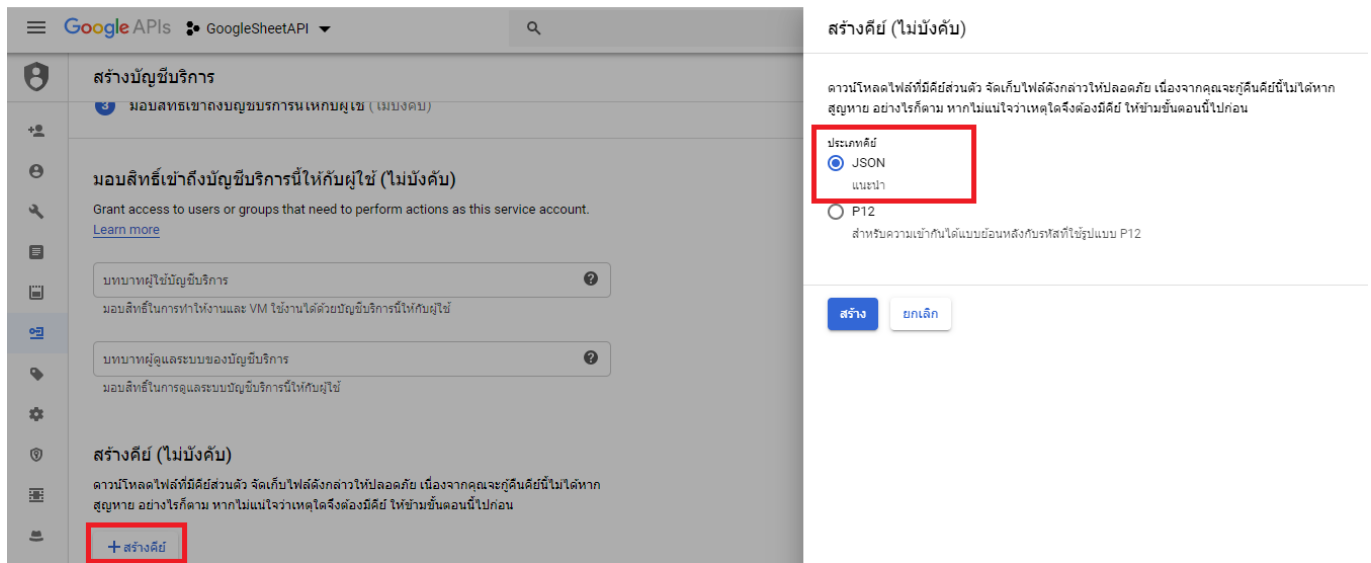
เบร้าวเซอร์

ผู้แก้ไข

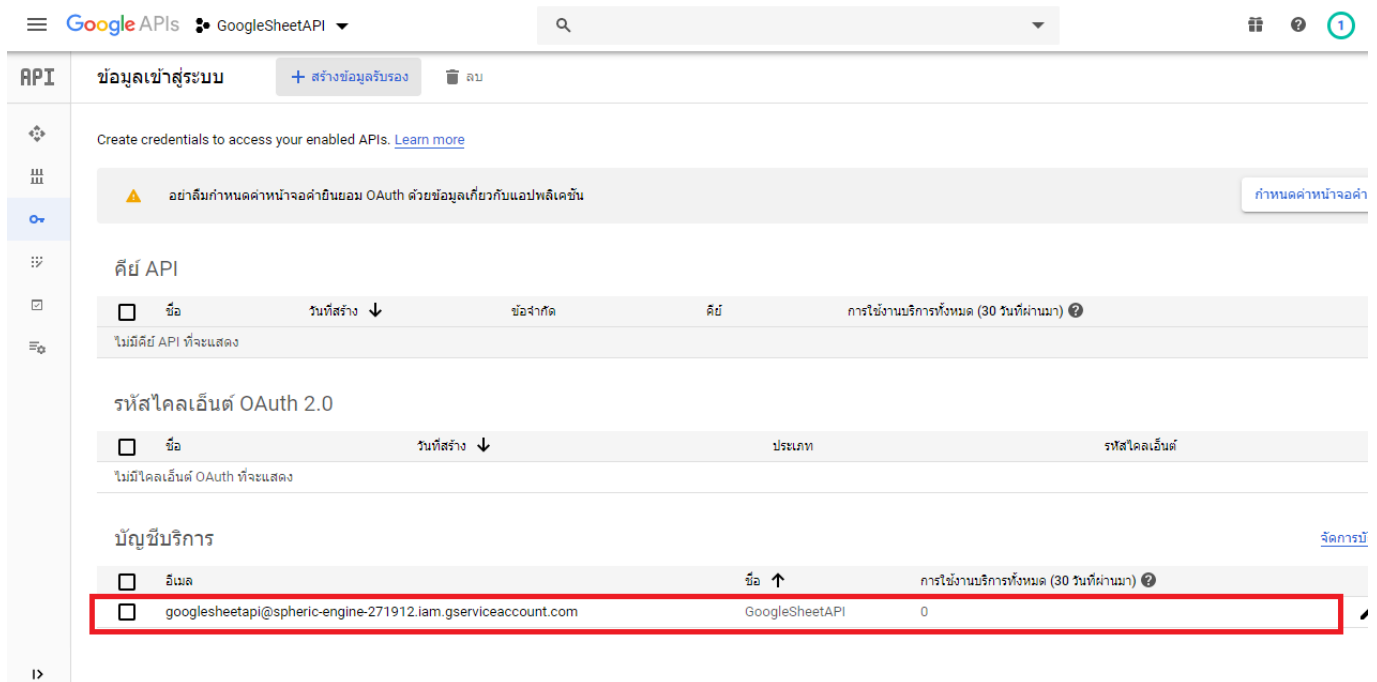
ผู้ดู



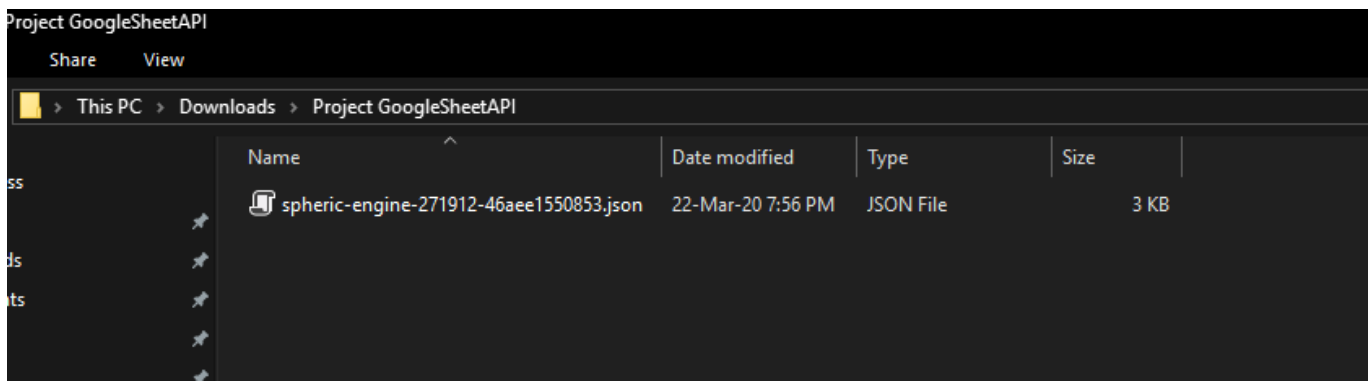
ส่วนสุดท้ายจะเป็นการมอบสิทธิ์การเข้าถึงให้กับผู้ใช้ ให้กดปุ่ม สร้างคีย์ และเลือก JSON เมื่อกดสร้างระบบจะ Download ข้อมูล Credentials สำหรับสิทธิ์การเข้าถึงข้อมูลออกมา



เมื่อกดปุ่มเสร็จสิ้น จะกลับมาที่หน้าหลัก ข้อมูลเข้าสู่ระบบ จะเห็นรายการ บัญชีบริการที่เราสร้างไว้ แสดงขึ้นมามีภาพ



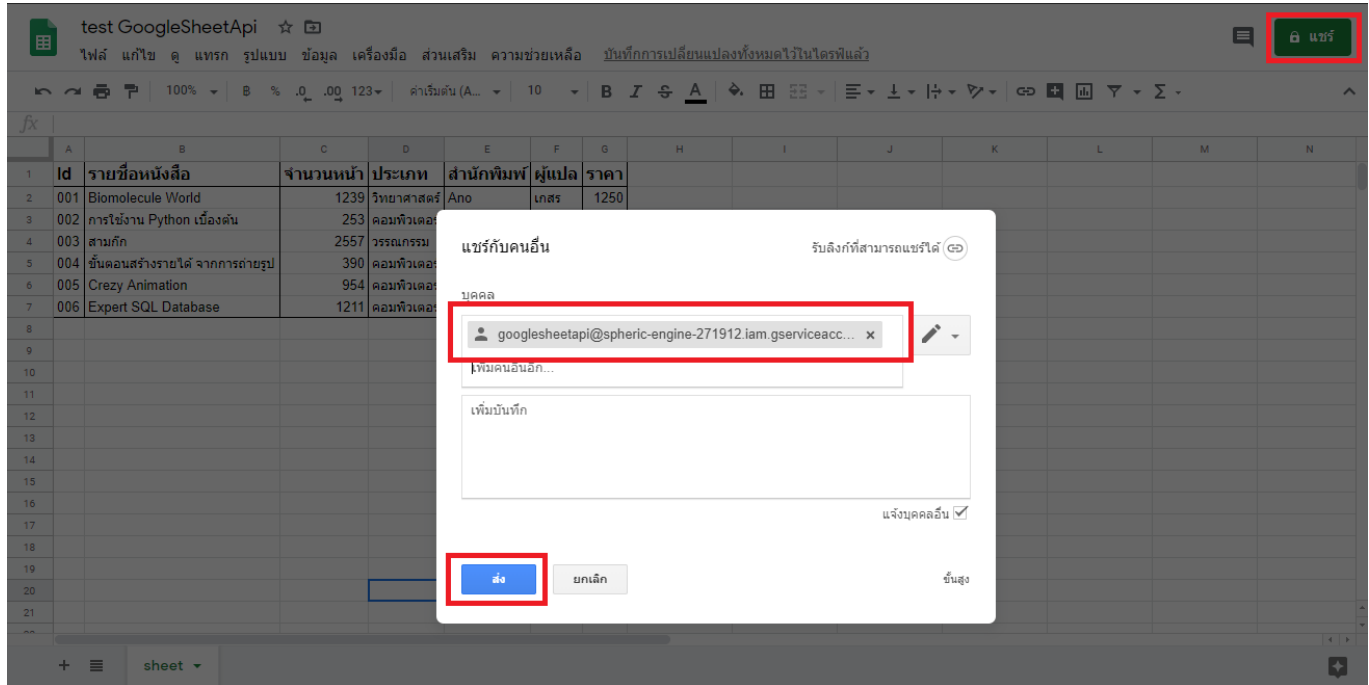
4. แก้ไขชื่อไฟล์ JSON ที่โหลดออกมาเป็นชื่อ credentials.json



เมื่อเปิดไฟล์ออกมามาดูข้อมูล Credentials ที่เรากำลังใช้ไว้ให้ Copy ข้อมูลส่วน client_email ไว้ เมื่อนำไป set ให้เข้าถึง Google Sheet ที่เราต้องการ



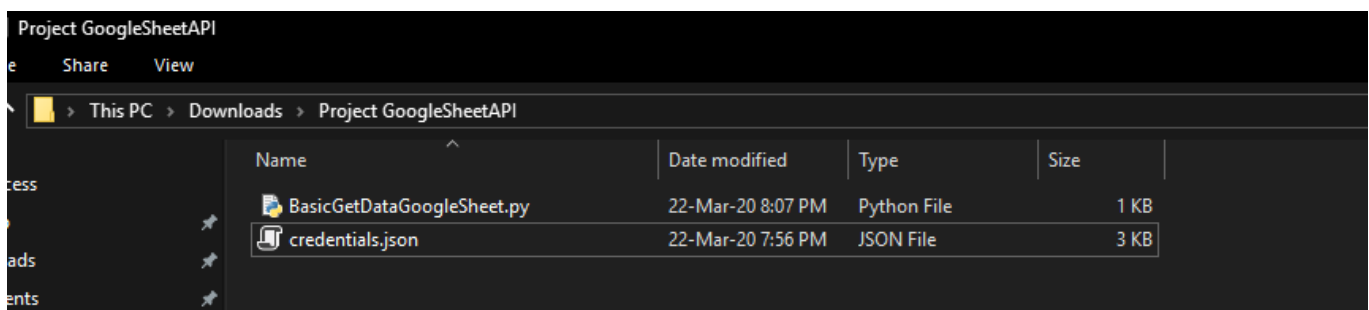
ทีนี้เราเข้า Google Sheet ตัวที่เราต้องการ set ให้เข้าถึงข้อมูลได้ โดยการกดปุ่ม แชร์ และใส่ข้อมูล email เป็น ข้อมูล client_email ส่วนที่เรา Copy มาและกดส่ง เพื่อ set Permission ให้ Application เราเข้าถึง Google Sheet นี้ได้



5.ขั้นตอนต่อไป เราจะมาเขียน Python ให้ดึงข้อมูลออกมาจาก Google Sheet กันครับ ก่อนอื่นต้อง Install Library ที่จำเป็นกันก่อนครับ (oauth2client, gspread)

```
pip install oauth2client gspread
```

ต่อไปเริ่มเขียน Code กันเลยครับ โดย Save file Python ที่เราเขียนที่เดียวกับ file credentials.json ที่เราสร้างไว้ครับ



```

from oauth2client.service_account import ServiceAccountCredentials
import gspread
import pprint

scope = ['https://www.googleapis.com/auth/spreadsheets']
credentials =
ServiceAccountCredentials.from_json_keyfile_name('credentials.json',
scope)
client = gspread.authorize(credentials)

sheet = client.open_by_url("<Link Url>")
worksheet = sheet.get_worksheet(0)

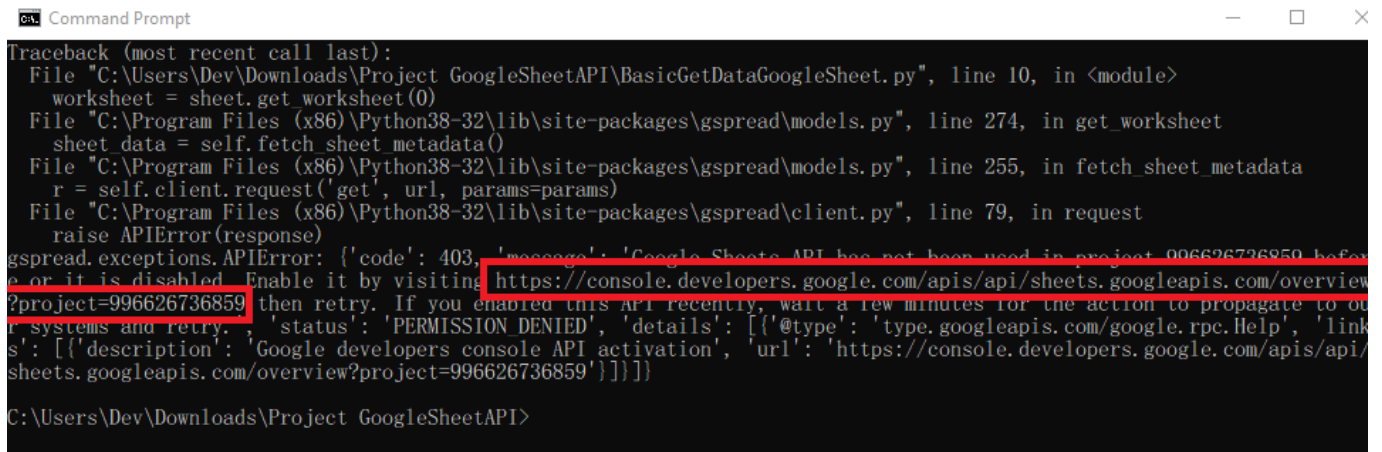
results = worksheet.get_all_records()

pp = pprint.PrettyPrinter()
pp.pprint(results)

```

- ประกาศ scope ตามลิงก์นี้ <https://www.googleapis.com/auth/spreadsheets> เพื่อ authenticate ไปยัง spreadsheets
- ประกาศ credentials ไปยังไฟล์ credentials.json ของเรา (ซึ่งถ้าเรา save file ไว้ที่ folder เดียวกัน การสามารถเรียกไฟล์ได้โดยการเขียนชื่อไฟล์ได้เลย)
- <Link Url> ให้นำ Link GoogleSheet ที่เราให้สิทธิ์การเข้าถึงเอาไว้มาใส่

ถ้าขึ้นผลลัพธ์แบบนี้



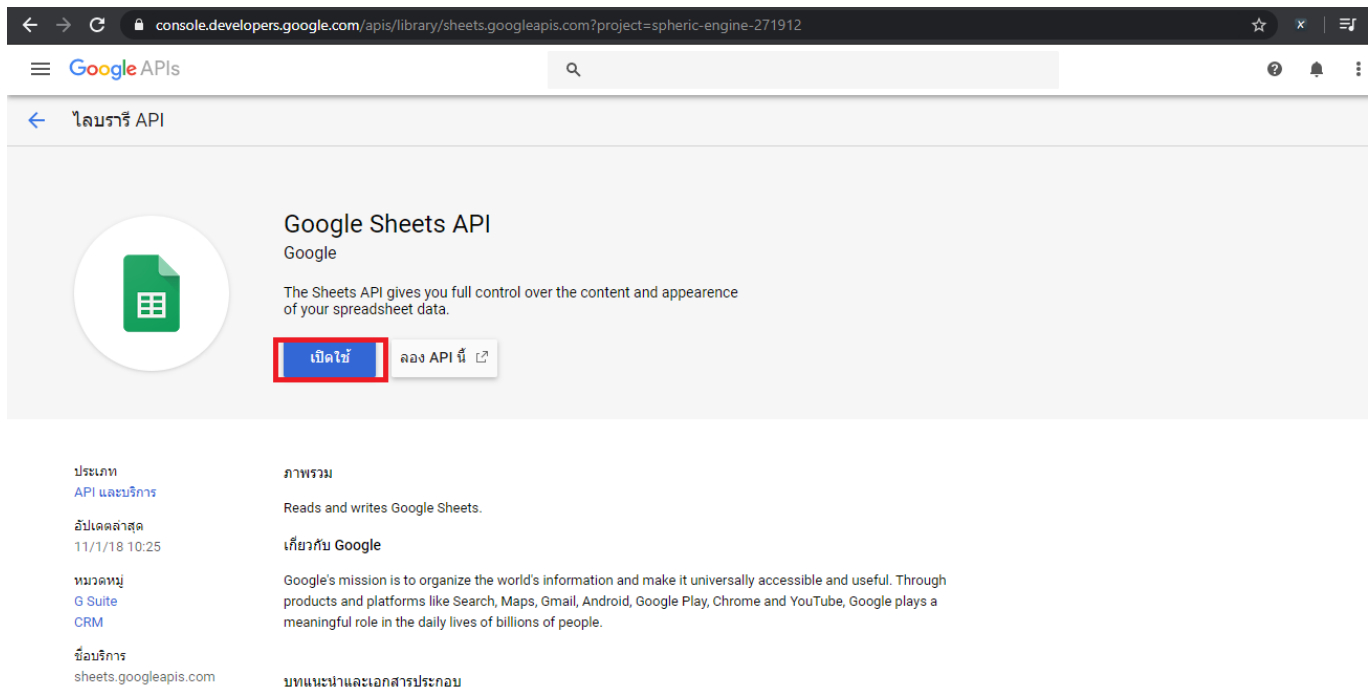
```

C:\Users\Dev\Downloads\Project GoogleSheetAPI>
Traceback (most recent call last):
  File "C:\Users\Dev\Downloads\Project GoogleSheetAPI\BasicGetDataGoogleSheet.py", line 10, in <module>
    worksheet = sheet.get_worksheet(0)
  File "C:\Program Files (x86)\Python38-32\lib\site-packages\gspread\models.py", line 274, in get_worksheet
    sheet_data = self.fetch_sheet_metadata()
  File "C:\Program Files (x86)\Python38-32\lib\site-packages\gspread\models.py", line 255, in fetch_sheet_metadata
    r = self.client.request('get', url, params=params)
  File "C:\Program Files (x86)\Python38-32\lib\site-packages\gspread\client.py", line 79, in request
    raise APIError(response)
gspread.exceptions.APIError: {'code': 403, 'message': 'Google Sheets API has not been used in project 996626736859 before or it is disabled. Enable it by visiting https://console.developers.google.com/apis/api/sheets.googleapis.com/overview?project=996626736859 then retry. If you enabled this API recently, wait a few minutes for the action to propagate to our systems and retry.', 'status': 'PERMISSION_DENIED', 'details': [{'@type': 'type.googleapis.com/google.rpc.Help', 'links': [{'description': 'Google developers console API activation', 'url': 'https://console.developers.google.com/apis/api/sheets.googleapis.com/overview?project=996626736859'}]}]}

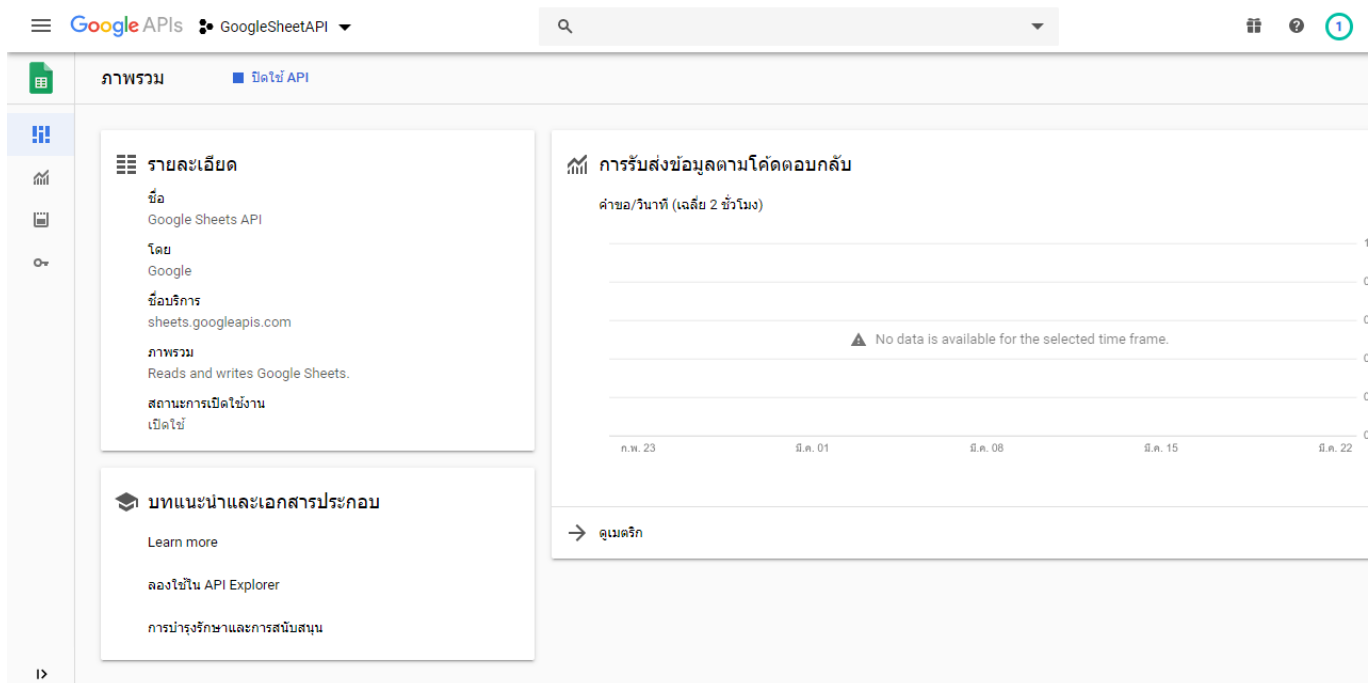
C:\Users\Dev\Downloads\Project GoogleSheetAPI>

```


PERMISSION DENIED เราจะต้องเข้าไปยืนยันการเข้าถึง Application อีกครั้งผ่านลิงก์ ขึ้นตาม Result



เมื่อเข้าไปตามลิงก์ จะเป็นหน้าขออนุญาต API Service ของเรา ให้ใช้งาน Google Sheets API
ให้กดปุ่ม เปิดใช้ จะแสดงหน้า Dashboard แสดงรายละเอียดการเรียกข้อมูลทั้งหมด



ทีนี้เรากลับเข้าไปรัน Code Python ของเรากันอีกครั้งกัน จะได้ผลลัพธ์ดังนี้ครับ

 Select Command Prompt

```
{
  'จำนวนหน้า': 1239,
  'ประเภท': 'วิทยาศาสตร์',
  'ผู้แปล': 'เกสร',
  'ราคา': 1250,
  'รายชื่อนหนังสือ': 'Biomolecule World',
  'สำนักพิมพ์': 'Ano'},
{'Id': 2,
  'จำนวนหน้า': 253,
  'ประเภท': 'คอมพิวเตอร์',
  'ผู้แปล': '',
  'ราคา': 259,
  'รายชื่อนหนังสือ': 'การใช้งาน Python เบื้องต้น',
  'สำนักพิมพ์': 'สารคดี'},
{'Id': 3,
  'จำนวนหน้า': 2557,
  'ประเภท': 'วรรณกรรม',
  'ผู้แปล': '',
  'ราคา': 2137,
  'รายชื่อนหนังสือ': 'สามก๊ก',
  'สำนักพิมพ์': 'สวรินทร์'},
{'Id': 4,
  'จำนวนหน้า': 390,
  'ประเภท': 'คอมพิวเตอร์',
  'ผู้แปล': '',
  'ราคา': 990,
  'รายชื่อนหนังสือ': 'ขั้นตอนสร้างรายได้จากการขายรูป',
  'สำนักพิมพ์': 'กฤษณ'},
{'Id': 5,
  'จำนวนหน้า': 954,
  'ประเภท': 'คอมพิวเตอร์',
  'ผู้แปล': 'อัน',
  'ราคา': 1500,
  'รายชื่อนหนังสือ': 'Crezy Animation',
  'สำนักพิมพ์': 'Philip'},
{'Id': 6,
  'จำนวนหน้า': 1211,
  'ประเภท': 'คอมพิวเตอร์',
  'ผู้แปล': 'มรกต',
  'ราคา': 1330,
  'รายชื่อนหนังสือ': 'Expert SQL Database',
  'สำนักพิมพ์': 'John-D']}]
```

C:\Users\Dev\Downloads\Project GoogleSheetAPI>_

ข้อมูลทั้งหมดใน GoogleSheet จะสามารถเข้าถึงได้แล้วครับ โดยการแสดงออกมาในรูปแบบ Dictionary

เมื่อเราเข้าถึงข้อมูลใน Google Sheet ผ่านทาง Python ได้แล้ว ทีนี้เราก็สามารถนำไปต่อยอดได้ โดยการนำ Dictionary ไปแสดงใน Application ที่เราสร้างขึ้นมาได้ แสดงผล, กรองข้อมูล หรือค้นหาได้สะดวกขึ้น ดังที่กล่าวไว้ข้างต้นได้แล้วครับ