

1. source code ใน โปรเจค section_2 นี้ run บน

apache airflow

2. ขอข้ามใน โจทย์ข้อที่ 2 ส่วนที่ 1 เนื่องจาก ใช้เวลาศึกษาไม่ทัน

โจทย์ข้อที่ 2 การสร้าง Pipeline, ฐานข้อมูล, และ การจัดการคุณภาพข้อมูล (คะแนนเต็ม 30 คะแนน)

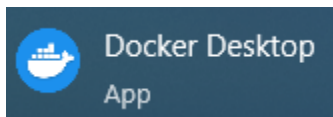
- (15 คะแนน) การจัดการคุณภาพข้อมูลด้วย OOP (Data Manipulation):
 - เขียนสคริปต์ในรูปแบบ Object-Oriented Programming (OOP) เพื่อจัดการคุณภาพข้อมูลตามมาตรฐานหลักเกณฑ์การประเมินคุณภาพข้อมูลสำหรับหน่วยงานภาครัฐ
<https://standard.dga.or.th/standard/dg-std/5785/>
 - สร้าง Class (เช่น DataProcessor) ที่มี Method สำหรับ:
 - Standardization: จัดการข้อมูลให้อยู่ในรูปแบบมาตรฐาน (เช่น รูปแบบชื่อผู้แต่ง, ชื่อวารสาร)
 - Handling Missing Values: จัดการค่าว่างในคอลัมน์ที่สำคัญ
 - Deduplication: สร้างตรรกะเพื่อระบุและรวมข้อมูลที่ซ้ำซ้อนกันจากแหล่งต่างๆ
 - เป็นต้น

3. วิธีติดตั้ง apache_airflow บน window 10 และ setup environment ที่ใช้ในโจทย์ข้อสอบ

- 3.1 ติดตั้ง environment ของ ubuntu บน window 10



- 3.2 ติดตั้ง docker desktop



- 3.3 เปิด command line ของ ubuntu ที่ได้ติดตั้งไว้



- 3.4 ติดตั้ง apache_airflow ด้วย docker บน window (ไม่มี Docker ใน ubuntu ให้ไปหาติดตั้งมาก่อน)

1. Docker-based Installation : Install Docker and Docker Compose.

```
## mkdir apache_airflow
```

```
## cd apache_airflow
```

```
## sudo apt-get update
```

```
## sudo apt-get install ca-certificates curl gnupg lsb-release
```

```
## curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o  
/usr/share/keyrings/docker-archive-keyring.gpg
```

```
## echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-  
archive-keyring.gpg] https://download.docker.com/linux/ubuntu $(lsb_release -cs)  
stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

```
## sudo apt-get update
```

```
## sudo apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin
```

Download Airflow Docker Compose file.

```
## curl -LfO 'https://airflow.apache.org/docs/apache-airflow/stable/docker-  
compose.yaml'
```

Create necessary directories and set environment variable:

```
## mkdir -p ./dags ./logs ./plugins
```

```
## echo -e "AIRFLOW_UID=$(id -u)" > .env
```

Initialize the database.

```
##docker compose up airflow-init
```

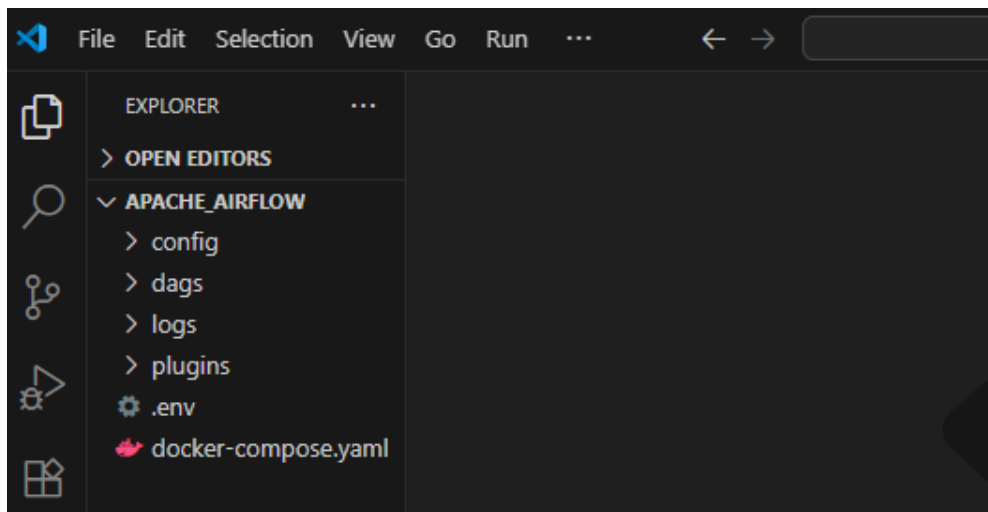
Run Airflow.

```
##docker compose up -d
```

2. Open in vs code

#code .

3. เมื่อเปิด vscode จะได้โครงสร้าง project ดังรูป



3.5 ถ้าติดตั้งถูกต้อง run บน browser ด้วย

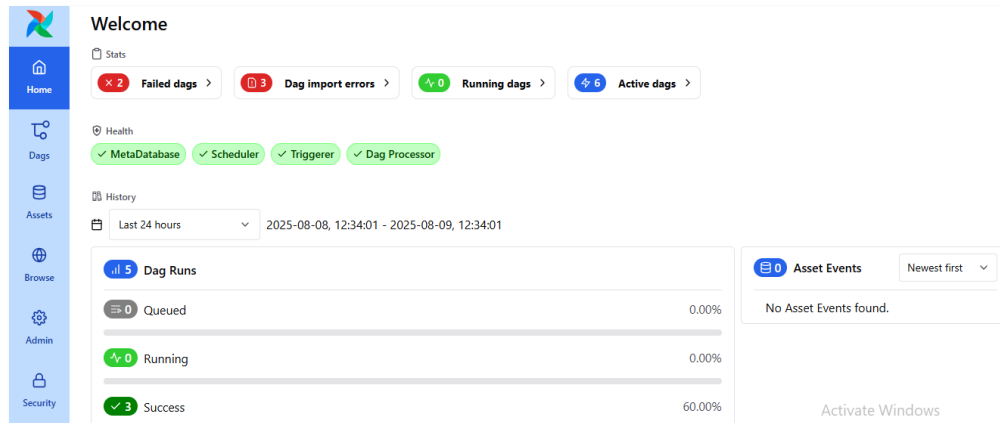
<http://localhost:8080>

Login

Username : airflow

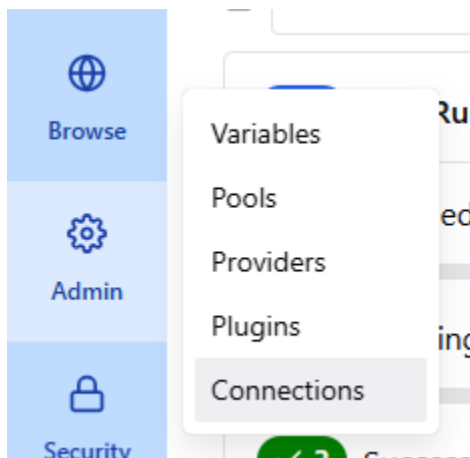
Password : airflow

จะเข้าหน้าเว็บต่อไปนี้

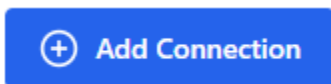


4. Set up environment ของ postgresql บน apache_airflow

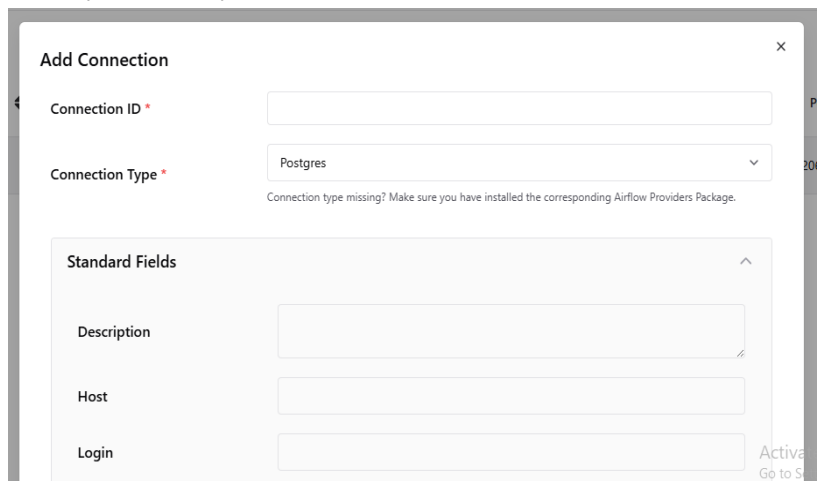
4.1 ไปที่ admin >> connection



4.2 ไปที่ add connection



4.3 ตั้งค่า postgresql connection




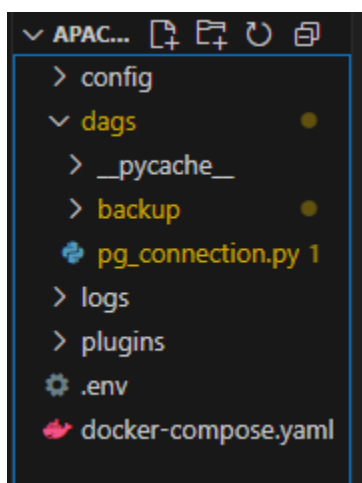
4.4 ถ้าตั้งค่าถูกต้องจะได้

Connection Id	Connection Type	Description	Host	Port	
pg-1f2e1a3e	postgres	pg-1f2e1a3e	pg-1f2e1a3e-punvarat-7a2d.b.aivencloud.com	22067	Edit Delete

5. จาก โจทย์ข้อที่ 2 ส่วนที่ 2 เพิ่มไฟล์ชื่อ pg_connection.py ใน dags

Source code อยู่ใน Folder

 section_2



Demo :

โจทย์ข้อที่ 2 ส่วนที่ 2 : https://www.youtube.com/watch?v=VQQPB-_saKM