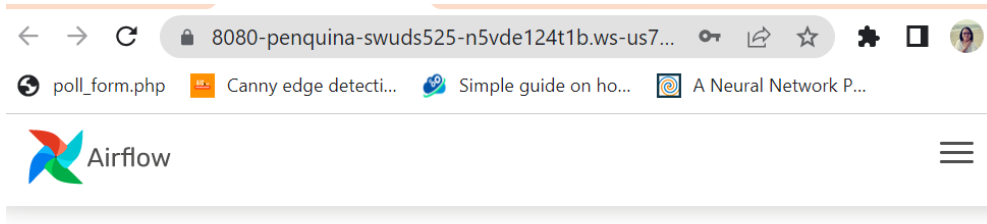


# Project: Creating and Automating a Set of Data Pipelines with Airflow

สิ่งที่คาดหวังในโปรเจกต์นี้

1. นำโค้ดจาก [Project: Building a Data Modeling with Postgres \(SQL\)](#) มาสร้าง data pipeline โดยใช้ Airflow
2. มีการเขียน documentation อธิบายสิ่งที่ตัวเองทำลงไป รวมไปถึงการออกแบบ data model

### 3. มี instruction ในการรันโค้ดของตัวเอง



Sign In

Enter your login and password below:


**Username:**

**Password:**

[Sign In](#)

The screenshot shows the Airflow DAGs page. The top navigation bar includes links for DAGs, Datasets, Security, Browse, Admin, and Docs. The current time is 10:33 UTC. The page title is DAGs. Below the title, there are filters for All (43), Active (0), and Paused (43). There is a search bar and an auto-refresh button. The main table lists DAGs with columns for DAG, Owner, Runs, Schedule, Last Run, Next Run, and Recent Tasks.

DAG	Owner	Runs	Schedule	Last Run	Next Run	Recent Tasks
<input type="checkbox"/> dataset_consumes_1 consumes dataset-scheduled	airflow	0	Dataset		0 of 1 datasets updated	
<input type="checkbox"/> dataset_consumes_1_and_2 consumes dataset-scheduled	airflow	0	Dataset		0 of 2 datasets updated	
<input type="checkbox"/> dataset_consumes_1_never_scheduled consumes dataset-scheduled	airflow	0	Dataset		0 of 2 datasets updated	
<input type="checkbox"/> dataset_consumes_unknown_never_scheduled dataset-scheduled	airflow	0	Dataset		0 of 2 datasets updated	
<input type="checkbox"/> dataset_produces_1 dataset-scheduled produces	airflow	0	@daily		2022-10-07, 00:00:00	
<input type="checkbox"/> dataset_produces_2						

 Airflow

☐

example\_branch\_dop\_operator\_v3

airflow

\*1 \* \* \* \*

i

☐

example

☐

example\_branch\_labels

airflow

@daily

i

☐

example\_branch\_operator

airflow

@daily

i


☐

example

example2

← → ↺ 8080-penquina-swuds525-n5vde124t1b.ws-us70.gitpod.io/dags/tutorial/code

poll\_form.php Canny edge detecti... Simple guide on ho... A Neural Network P... Learner Lab Image Feature Extra... Canny edge detecti...

 Airflow DAGs Datasets Security Browse Admin Docs 10:41 UTC AA

☐ DAG: tutorial A simple tutorial DAG

Schedule: 1 day, 0:00:00 Next Run: 2022-10-07, 10:40:21

☐ Grid ☒ Graph ☐ Calendar ☐ Task Duration ☐ Task Tries ☐ Landing Times ☐ Gantt ☐ Details

1 #  
2 # Licensed to the Apache Software Foundation (ASF) under one  
3 # or more contributor license agreements. See the NOTICE file

Toggle Wrap

← → ↺ penquina-swuds525-n5vde124t1b.ws-us70.gitp...

poll\_form.php Canny edge detecti... Simple guide on ho... A Neural Network P...

started

README.md U

docker-compose.yaml U

my\_dag.py U ×

...

05-creating-and-scheduling-data-pipelines > dags > my\_dag.py

```
1 from airflow import DAG
2 from airflow.utils import timezone
```



t1 >> t2

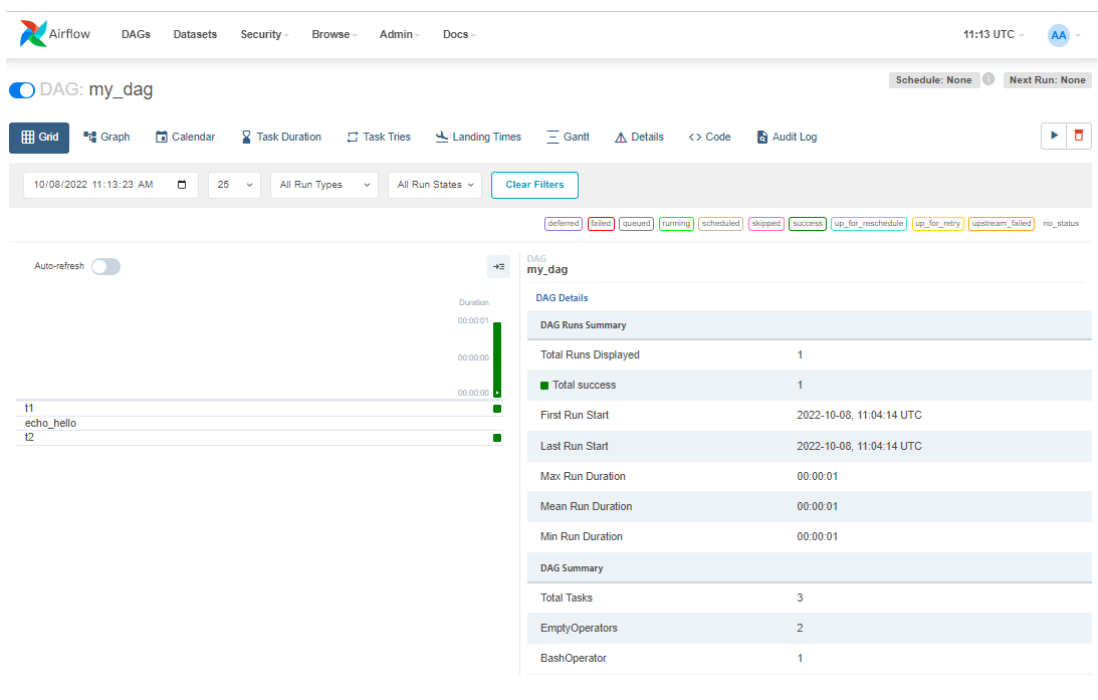
The screenshot shows the 'DAG: my\_day' interface. The top navigation bar includes a 'success' status indicator and a 'Schedule: None' label. The main toolbar contains icons for Grid, Graph, Calendar, Task Duration, Task Times, Landing Times, Gantt, Details, Code, and Audit Log. The task list displays a single task with the ID 'manual\_\_2022-10-08T11:04:13.596122+00:00' and a status of 'success'. Below the task list, a diagram shows two tasks, 't1' and 't2', connected by an arrow, indicating a dependency.

```

# Task
#1st task ยังไม่ทำอะไร
# step 1
t1 = EmptyOperator( task_id = "t1")

# 2nd task
t2 = EmptyOperator( task_id = "t2")
#step 3
t1 >> t2

```



Airflow DAGs Datasets Security - Browse - Admin - Docs - 11:19 UTC - AA -

Triggered my\_dag, it should start any moment now.

DAG: my\_dag Schedule: None Next Run: None

Grid Graph Calendar Task Duration Task Tries Landing Times Gantt Details <> Code Audit Log

10/08/2022 11:18:11 AM 25 All Run Types All Run States Clear Filters

Auto-refresh

Duration

11 echo\_hello

12

DAG my\_dag Run 2022-10-08, 11:18:13 UTC / echo\_hello

Task Instance Details Rendered Template Log XCom List instances, all runs Filter Upstream

Details Logs

(by attempts)

1

All Levels All File Sources Wrap Full Logs Download See More

```
[2022-10-08, 11:18:14 UTC] (taskinstance.py:285) WARNING - Dependency <task(_PythonDecoratedOperator): load>, transform a
[2022-10-08, 11:18:14 UTC] (taskinstance.py:285) WARNING - Dependency <task(_PythonDecoratedOperator): transform>, load a
[2022-10-08, 11:18:14 UTC] (taskinstance.py:285) WARNING - Dependency <task(_PythonDecoratedOperator): load>, transform a
[2022-10-08, 11:18:15 UTC] (taskinstance.py:184) INFO - Running <taskinstance: my_dag:echo_hello manual_2022-10-08T11
[2022-10-08, 11:18:15 UTC] (taskinstance.py:1592) INFO - Exporting the following env vars:
AIRFLOW_CTX_DAG_OWNER=***
AIRFLOW_CTX_DAG_ID=my_dag
AIRFLOW_CTX_TASK_ID=echo_hello
AIRFLOW_CTX_EXECUTION_DATE=2022-10-08T11:18:13.196026+00:00
AIRFLOW_CTX_TRY_NUMBER=1
AIRFLOW_CTX_DAG_RUN_ID=manual_2022-10-08T11:18:13.196026+00:00
[2022-10-08, 11:18:15 UTC] (subprocess.py:63) INFO - Temp dir root location:
/tmp
[2022-10-08, 11:18:15 UTC] (subprocess.py:75) INFO - Running command: ['bin/bash', '-c', 'echo 'hello''']
[2022-10-08, 11:18:15 UTC] (subprocess.py:86) INFO - Output:
[2022-10-08, 11:18:15 UTC] (subprocess.py:93) INFO - hello
[2022-10-08, 11:18:15 UTC] (subprocess.py:97) INFO - Command exited with return code 0
[2022-10-08, 11:18:15 UTC] (taskinstance.py:1486) INFO - Marking task as SUCCESS. dag_id=my_dag, task_id=echo_hello, o
[2022-10-08, 11:18:15 UTC] (local_task_job.py:164) INFO - Task exited with return code 0
[2022-10-08, 11:18:15 UTC] (local_task_job.py:275) INFO - 1 downstream tasks scheduled from follow-on schedule check
```

Airflow DAGs Datasets Security - Browse - Admin - Docs - 11:25 UTC - AA -

Triggered my\_dag, it should start any moment now.

DAG: my\_dag Schedule: None Next Run: None

Grid Graph Calendar Task Duration Task Tries Landing Times Gantt Details <> Code Audit Log

10/08/2022 11:24:48 AM 25 All Run Types All Run States Clear Filters

Auto-refresh

Duration

11 echo\_hello

12 print\_hey

12

DAG my\_dag

DAG Details

DAG Runs Summary

Total Runs Displayed	5
Total success	3
Total failed	2
First Run Start	2022-10-08, 11:15:36 UTC
Last Run Start	2022-10-08, 11:24:47 UTC
Max Run Duration	00:00:03
Mean Run Duration	00:00:02
Min Run Duration	00:00:02

DAG Summary

Total Tasks	4
EmptyOperators	2
BashOperator	1
PythonOperator	1

← → ↻ crontab.guru/#30\_8\_\*\*\_\*

poll\_form.php Canny edge detecti... Simple guide on ho... A Neural Network P... Learner Lab

**Cronitor** Cron Job Monitoring

# crontab guru

The quick and simple editor for cron schedule expressions by [Cronitor](#)

*“At 08:30.”*

[next](#) at 2022-10-09 08:30:00 [random](#)

30 8 \* \* \*

<a href="#">minute</a>	<a href="#">hour</a>	<a href="#">day</a> (month)	<a href="#">month</a>	<a href="#">day</a> (week)
*		any value		
		value list		
		separator		
		range of values		

```
with DAG(  
    "my_dag",  
    start_date = timezone.datetime(2022, 10, 8),  
    schedule = "* * * * *",  
    tags = ["workshop"],  
):
```



