

Airflow 활용 Guide

Table of Contents

1. 문서 개요
 1. 목적
 2. 범위
 3. 시스템 구성도
 4. 참고자료
2. Prerequisite
 - Python 설치
3. Airflow 설치
 - 3.2.1 가상환경에서 실행 폴더 지정해주기
 - 3.2.2 DB init
 - 3.3.3 User 생성
 - 3.3.4 WebServer 실행하기
 - 3.3.4.1 Web Server 실행
 - 3.3.4.2 Scheduler를 실행
4. 웹 크롤링을 위한 설정
 - 4.1 Config File 작성
 - 4.2 ChromeDriver 설치
 - 4.2.1 Chrome 버전 확인하기
 - 4.2.2 Google-Chrome 설치하기
 - 4.2.3 ChromeDriver 다운로드 받기
5. 데이터 파이프라인 구성
 - 5.1 작업 폴더 생성하기
 - 5.2 Dag 파일 생성하기
 - 5.3 webcraw 폴더 구성하기
 - 5.3.1 Task 설명
 - 5.3.1.1 Elements 값 추출
 - 5.3.1.2 Data 출력 경로 지정
 - 5.3.2 Task 코드 작성하기

6. Web Server 활용

7. 참고 사항

1. 문서 개요

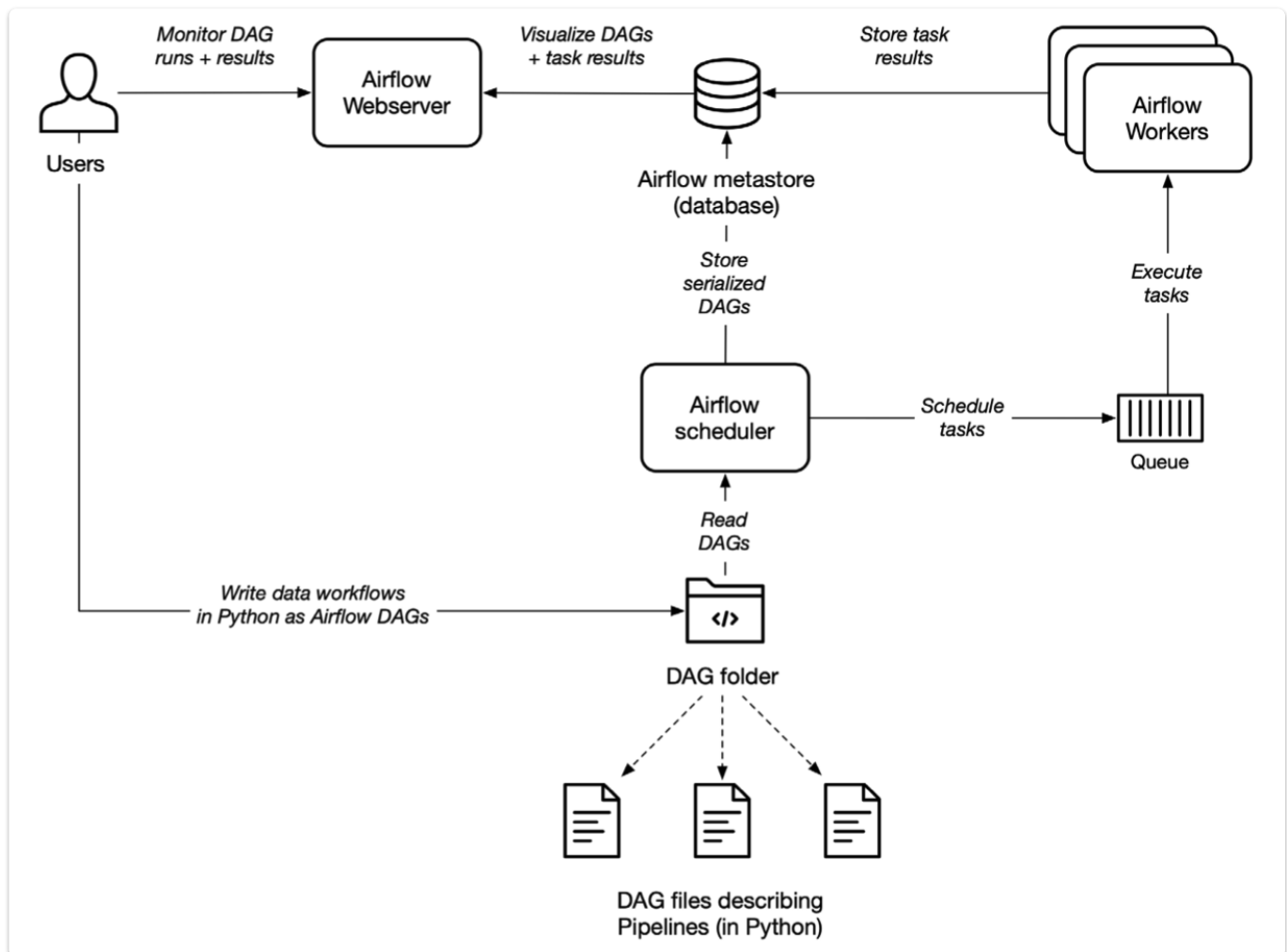
1.1 목적

본 문서는 airflow를 설치하고 데이터 파이프라인을 설정하는 방법을 기술하였다.

1.2 범위

설치 범위는 ubuntu 22.04, 가상환경에서 진행하였다.

1.3 시스템 구성도



1.4 참고자료

[Airflow Official](#)

[Airflow 튜토리얼 따라하기](#)

[ChromeDriver Official](#)

1.5 Version

- Airflow v2.8.1
- Python v3.11
- ChromDriver v121.0.6167.85

2. Prerequisite

본 설치 가이드는 Ubuntu 22.04 환경에서 설치하는 것을 기준으로 작성하였다.

2.1 Python 설치

- Python v3.11

3. Airflow 설치

3.1 Python Dependency 설치

- 작업 디렉터리 생성 및 이동한다.

```
mkdir $HOME/airflow && cd $HOME/airflow
```

- 설치를 진행할 Dependency 스크립트 작성 , 실행 한다.

```
vi install.sh
```

- install.sh

```
AIRFLOW_VERSION=2.8.1
```

```
# Extract the version of Python you have installed. If you're currently  
# using a Python version that is not supported by Airflow, you may want to  
# set this manually.
```

```
# See above for supported versions.
PYTHON_VERSION="$(python --version | cut -d " " -f 2 | cut -d "." -f 1-2)"

CONSTRAINT_URL="https://raw.githubusercontent.com/apache/airflow/constraints-${AIRFLOW_VERSION}/constraints-${PYTHON_VERSION}.txt"
# For example this would install 2.8.1 with python 3.8:
https://raw.githubusercontent.com/apache/airflow/constraints-2.8.1/constraints-3.8.txt

pip install "apache-airflow==${AIRFLOW_VERSION}" --constraint "${CONSTRAINT_URL}"
```

- 스크립트 실행

```
source install.sh
```

- 실습용 데이터 파이프라인을 위해 추가적으로 필요한 모듈을 설치한다.

```
pip install pandas
pip install requests
pip install selenium
pip install beautifulsoup4
pip install beautifulsoup4 lxml
```

3.2 Airflow 사전 설정하기

- 환경 변수를 추가한다

```
echo $PATH
export PATH=$PATH:/home/ubuntu/.local/bin
echo $PATH
```

3.2.1 가상환경에서 실행 폴더 지정해주기

- ⚠ Python을 가상환경에서 실행할 때에는 작업 디렉토리에 `virtualenv` 모듈을 활성화 해주어야 한다.
- 작업 디렉터리로 이동한다.

```
cd $HOME/airflow
```

- `virtualenv` 를 설치한다.

```
pip install virtualenv
```

- 가상 환경 폴더를 구성한다.
- ⚠️ 작업 디렉터리에서 진행되어야 한다.

```
virtualenv env
```

명령어를 실행하면 /env 폴더가 생성된다.

- `virtualenv` 를 활성화 한다.

```
source /env/bin/activate
```

i `virtualenv` 비활성화

```
source /env/bin/deactivate
```

3.2.2 DB init

- airflow의 데이터베이스를 초기화한다.

```
airflow db init
```

3.2.3 User 생성

- Webserver를 이용할 유저를 생성한다

```
airflow users create -u admin -p admin -f admin -l admin -r Admin -e  
admin@admin.com
```

3.2.4 WebServer 실행하기

- 각자 진행할 VM창을 따로 열어 진행해준다.
- server와 scheduler는 동시에 실행되고 있어야 한다.

- 작업 디렉토리에서 진행한다.

3.3.4.1 Web Server 실행

- 포트를 연결해 Web Server를 실행한다.

```
airflow webserver -p 8080
```

3.3.4.2 scheduler를 실행

- scheduler를 실행한다.

```
airflow scheduler
```



```

● ● ● ubuntu@bami: ~/airflow
activate activate.fish activate.ps1 dotenv pip pip3.10 python3 wheel wheel3
activate.csh activate.nu activate.this.py normalizer pip3 python python3.10 wheel-3.10 wheel3.10
ubuntu@bami:~/airflow$ source env/bin/activate
(env) ubuntu@bami:~/airflow$ airflow webserver -p 8080

Running the Gunicorn Server with:
Workers: 4 sync
Host: 0.0.0.0:8080
Timeout: 120
Logfiles: --
Access Logformat:

/home/ubuntu/.local/lib/python3.10/site-packages/flask_limiter/extension.py:336 UserWarning: Using the in-memory storage for tracking rate limits as no storage was explicitly specified. This is not recommended for production use. See: https://flask-limiter.readthedocs.io#configuring-a-storage-backend for documentation about configuring the storage backend.
[2024-01-29T13:22:11.113+0900] [options.py:83] WARNING - The swagger_ui directory could not be found.
Please install connexion with extra install: pip install connexion[swagger-ui]
or provide the path to your local installation by passing swagger_path=cyour path>
[2024-01-29T13:22:11.114+0900] [options.py:83] WARNING - The swagger_ui directory could not be found.
Please install connexion with extra install: pip install connexion[swagger-ui]
or provide the path to your local installation by passing swagger_path=cyour path>
[2024-01-29 13:22:11 +0900] [20419] [INFO] Starting gunicorn 21.2.0
[2024-01-29 13:22:11 +0900] [20419] [INFO] Listening at: http://0.0.0.0:8080 (20419)
[2024-01-29 13:22:11 +0900] [20419] [INFO] Using worker: sync
[2024-01-29 13:22:11 +0900] [20423] [INFO] Booting worker with pid: 20423
[2024-01-29 13:22:11 +0900] [20424] [INFO] Booting worker with pid: 20424
[2024-01-29 13:22:11 +0900] [20425] [INFO] Booting worker with pid: 20425
[2024-01-29 13:22:11 +0900] [20426] [INFO] Booting worker with pid: 20426
211.234.196.169 - - [29/Jan/2024:13:22:35 +0900] "GET /home HTTP/1.1" 200 41728 "http://180.210.81.241:8080/dags/cp-dags/grid?tab=logs&dag_run_id>manual_2024-01-29T02K3A04X3A51.74829XZB00X3A00&task_id=get_url" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36"
211.234.196.169 - - [29/Jan/2024:13:22:36 +0900] "GET /static/appbuilder/css/bootstrap-datepicker/bootstrap-datepicker3.min.css HTTP/1.1" 304 0 "http://180.210.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36"
211.234.196.169 - - [29/Jan/2024:13:22:36 +0900] "GET /static/appbuilder/css/bootstrap-datepicker.min.css HTTP/1.1" 304 0 "http://180.210.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36"

● ● ● ubuntu@bami: ~/airflow

* Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s just raised the bar for easy, resilient and secure K8s cluster deployment.

https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

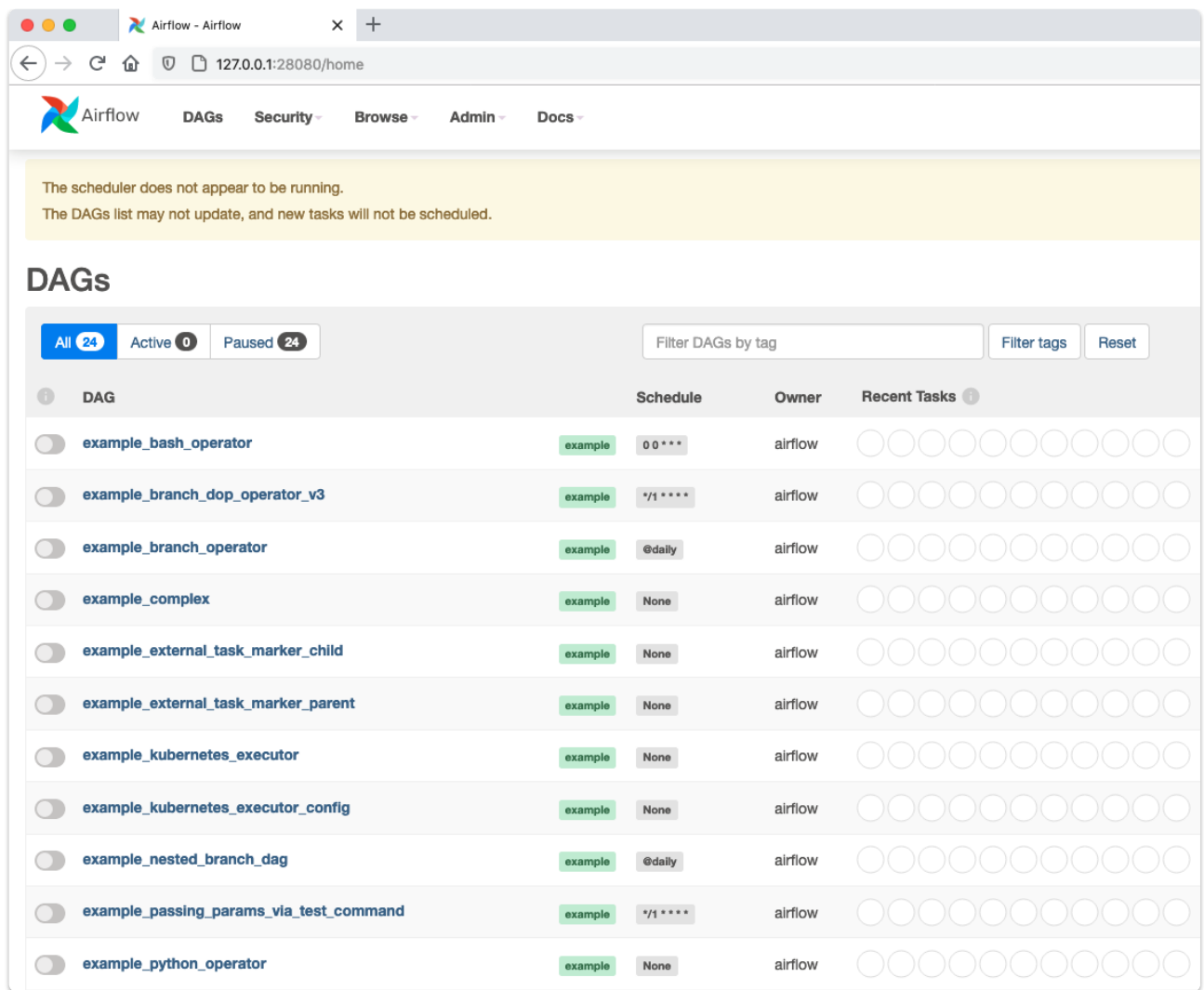
88 updates can be applied immediately.
48 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

1 additional security update can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

*** System restart required ***
Last login: Tue Jan 30 16:42:32 2024 from 211.234.192.42
ubuntu@bami:~$ cd airflow/
ubuntu@bami:~/airflow$ source env/bin/activate
(env) ubuntu@bami:~/airflow$ airflow scheduler

Running the Dag Scheduler with:
Task Context Logging Enabled
[2024-01-30T16:43:28.796+0900] [task_context_logger.py:63] INFO - Task context logging is enabled
[2024-01-30T16:43:28.796+0900] [executor_loader.py:115] INFO - Loaded executor: SequentialExecutor
[2024-01-30 16:43:28 +0900] [49503] [INFO] Starting gunicorn 21.2.0
[2024-01-30T16:43:28.829+0900] [scheduler_job_runner.py:808] INFO - Starting the scheduler
[2024-01-30 16:43:28 +0900] [49503] [INFO] Listening at: http://:::8793 (49503)
[2024-01-30T16:43:28.830+0900] [scheduler_job_runner.py:815] INFO - Processing each file at most -1 times
[2024-01-30 16:43:28 +0900] [49503] [INFO] Using worker: sync
[2024-01-30 16:43:28 +0900] [49504] [INFO] Booting worker with pid: 49504
[2024-01-30T16:43:28.837+0900] [manager.py:169] INFO - Launched DagFileProcessorManager with pid: 49505
[2024-01-30T16:43:28.840+0900] [scheduler_job_runner.py:1619] INFO - Adopting or resetting orphaned tasks for active dag runs
[2024-01-30T16:43:28.854+0900] [settings.py:60] INFO - Configured default timezone UTC
[2024-01-30T16:43:28.878+0900] [scheduler_job_runner.py:1642] INFO - Marked 1 SchedulerJob instances as failed
[2024-01-30T16:43:28.882+0900] [manager.py:392] WARNING - Because we cannot use more than 1 thread (parsing_processes = 2) when using sqlite. So we set parallelism to 1.
```


{MASTER_NODE_IP}:8080 접속



4. 웹 크롤링을 위한 설정

★ 이하 모든 과정은 `virtualenv` 가 실행중인 폴더, 즉 작업 폴더에서 진행한다

4.1 Config File 작성

```
vi airflow.cfg
```

- Dags 폴더가 작업 폴더와 일치한지 확인한다.

```
ubuntu@bami: ~/airflow
[core]
# The folder where your airflow pipelines live, most likely a
# subfolder in a code repository. This path must be absolute.
# Warning: This variable is deprecated! Use AIRFLOW__CORE__DAGS_FOLDER instead.
# Warning: This variable is deprecated! Use AIRFLOW__CORE__DAGS_FOLDER instead.
# Variable: AIRFLOW__CORE__DAGS_FOLDER
#
dags_folder = /home/ubuntu/airflow/dags
```


- Web Server에서 실습용 Dag 파일만 보일 수 있게 설정한다.

```
# get started, but you probably want to set this to False in a production
# environment
#
# Variable: AIRFLOW__CORE__LOAD_EXAMPLES
#
load_examples = False
# Path to the folder containing Airflow plugins
```

4.2 ChromeDriver 설치

4.2.1 Chrome 버전 확인하기

- Chrome에 접속하여 현재 버전을 확인한다.
- 최신 버전이 아니라면 **최신 버전으로 업데이트**를 진행한다.

Web-Chrome 접속 > 더보기 클릭 > 설정 > Chrome 정보



4.2.2 Google-Chrome 설치하기

- 최신 버전의 Chrome을 다운로드 받는다.

```
sudo wget -q -O - [https://dl-
ssl.google.com/linux/linux_signing_key.pub] (https://dl-
ssl.google.com/linux/linux_signing_key.pub) | sudo apt-key add -
```

```
sudo sh -c 'echo "deb [arch=amd64]
[http://dl.google.com/linux/chrome/deb/]
(http://dl.google.com/linux/chrome/deb/) stable main" >>
```

```
/etc/apt/[sources.list.d/google.list'](https://league-  
cat.tistory.com/sources.list.d/google.list')
```

```
sudo apt-get update  
sudo apt-get install google-chrome-stable
```

웹 버전과 같은 버전인지 확인한다.

```
(env) ubuntu@bami:~/airflow/dags$ google-chrome --version  
Google Chrome 121.0.6167.85
```

4.2.3 ChromeDriver 다운로드 받기

[ChromeDriver Download](#)

- 위의 링크를 통해 Web Chrome, Google-Chrome과 같은 버전의 Driver를 다운 받고 실행 시켜준다.

```
wget https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-  
testing/121.0.6167.85/linux64/chromedriver-linux64.zip
```

```
cd airflow/chromedriver-linux64/chromeDirver  
$ ./ChromeDirver
```

5. 데이터 파이프라인 구성

5.1 작업 폴더 생성하기

- dag: Dag 파일(`cp-dags.py`) 이 저장될 디렉토리
- webcraw: Task가 정의된 파일(`gitcraw.py`)과 `ChromeDriver` 가 위치 할 디렉토리
- data: 파이프라인 진행 후 최종 데이터가 저장될 디렉토리

```
mkdir dag  
mkdir webcraw  
mkdir data
```

- 최종 파일구조 (주요파일만 출력, 이하 env,python lib 등 생략)

```
./airflow/
├─ airflow-webserver.pid
├─ airflow.cfg
├─ airflow.db
├─ dags
│   ├── __pycache__
│   │   └─ cp-dags.cpython-310.pyc
│   └─ cp-dags.py
├─ data
│   ├── git_20240129.csv
│   └─ git_20240130.csv
├─ webcrawl
│   ├── __pycache__
│   │   └─ gitcrawl.cpython-310.pyc
│   ├── chromedriver
│   └─ gitcrawl.py
└─ webserver_config.py
```

5.2 Dag 파일 생성하기

- Dag 폴더로 이동 후 `cp-dags.py` 파일을 생성한다.
- 프로세스의 추가 설명은 주석을 참고한다.

`cp-dags.py`

```
import sys
import os
from datetime import timedelta, datetime
from airflow import DAG
from airflow.operators.bash import BashOperator
from airflow.operators.python_operator import PythonOperator
from airflow.utils.trigger_rule import TriggerRule

sys.path.append(os.path.dirname(os.path.abspath(os.path.dirname(__file__))))
from webcrawl.gitcrawl import get_folder_info

def print_result(**kwargs):
    r = kwargs["task_instance"].xcom_pull(key='result_msg')
    print("message:", r)
```

```

# Dag의 이름, 이메일 등 Dag Arguments를 정의한다.(Default)
default_args = {
    'owner': 'admin',
    'depends_on_past': False,
    'email': ['admin@admin.com'],
    'email_on_failure': False,
    'email_on_retry': False,
    'retries': 1,
    'retry_delay': timedelta(minutes=30),
    'start_date': datetime(2024, 1, 30),
}

# Dag Arguments 추가 설정
dag_args = dict(
    dag_id="cp-dags",
    default_args=default_args,
    description='tutorial DAG ml',
    schedule_interval=timedelta(minutes=50),
    tags=['cp-test-bami'],
)

# Dag 객체를 정의한다. Dag 객체 안에는 실행될 task들이 선언되어 있다.
with DAG(**dag_args) as dag:
    start = BashOperator(
        task_id='start',
        bash_command='echo "start!"',
    )

    get_folder_info_task = PythonOperator(
        task_id='get_folder_info',
        python_callable=get_folder_info,
    )

# Task의 실행 순서 그래프를 지정한다.
start >> get_folder_info_task

```

5.3 webcraw 폴더 구성하기

5.3.1 Task 설명

5.3.1.1 Elements 값 추출

- 가져올 데이터를 정하고 그에 해당하는 Element 값을 찾는다.

The screenshot shows a GitHub repository page for 'seunghyejeong/cp-cluster-install-single.md'. The repository has 893,385 commits and 847 commits. The repository contains a table with folders and files. The table has a header row and a body with 7 rows. The first row is a header row with the text 'Folders and files'. The following 6 rows are data rows with the text 'react-directory-row undefined'.

5.3.1.2 Data 출력 경로 지정

- 해당 값을 Table 형식으로 바꾼 뒤 /data 폴더에 .csv 파일로 저장한다.

5.3.2 Task 코드 작성하기

- webcraw 폴더로 이동 후 gitcraw.py 를 생성한다.
- 프로세스의 추가 설정은 주석을 참고한다.

```
import pandas as pd
import time
import datetime
import warnings
from bs4 import BeautifulSoup as bs
from selenium import webdriver
from selenium.webdriver.chrome.service import Service
from selenium.webdriver.common.by import By
from selenium.webdriver.chrome.options import Options
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
```

```
warnings.filterwarnings(action="ignore")

def get_folder_info(**kwargs):
    service = Service('/home/ubuntu/airflow/webcraw/chromedriver')
    service.start()
    chrome_options = webdriver.ChromeOptions()
    chrome_options.add_argument('--headless')
    chrome_options.add_argument('--no-sandbox')
    chrome_options.add_argument('--disable-dev-shm-usage')
    browser = webdriver.Remote(service.service_url,
options=chrome_options)
    #Chrome dirver를 통해 browser를 읽어온다.
    browser.get("https://github.com/k-paas/container-platform")
    time.sleep(5)

    # 아래의 DataFrame function을 사용시 값의 길이 제약이나 NoneType 발생을 막기 위
    해 key값을 먼저 만들고 dataForm 형식을 지정해 주었다.
    folder_ids = ['folder-row-0', 'folder-row-1', 'folder-row-2',
'folder-row-3', 'folder-row-4', 'folder-row-5', 'folder-row-6']
    data = {'Folder ID': [], 'Title': [], 'Aria Label': [], 'Data Pjax
Title': [], 'Relative Time': []}

    for folder_id in folder_ids:
        # 위에서 찾아낸 Elements에 해당하는 값을 찾는다.
        a_element = browser.find_element(By.CSS_SELECTOR, f'tr#
{folder_id} td div h3 a')
        title = a_element.get_attribute('title')
        aria_label = a_element.get_attribute('aria-label')

        data_pjax_title_element = browser.find_element(By.CSS_SELECTOR,
f'tr#{folder_id} a[data-pjax="true"]')
        data_pjax_title = data_pjax_title_element.get_attribute('title')
        relative_time = browser.find_element(By.CSS_SELECTOR, f'tr#
{folder_id} relative-time').get_attribute('title')

        data['Folder ID'].append(folder_id)
        data['Title'].append(title)
        data['Aria Label'].append(aria_label)
        data['Data Pjax Title'].append(data_pjax_title)
        data['Relative Time'].append(relative_time)

    # DataFrame을 사용해 table 형태로 정렬한다.
```

```
df = pd.DataFrame(data)
print(df)

date_today = datetime.datetime.now().strftime("%Y%m%d")
df.to_csv(f"/home/ubuntu/airflow/data/git_{date_today}.csv",
index=False)

browser.quit()
```

• DataFrame 결과물


```

0 Folder ID      Title      ... 0.0.0 Safari/537.36 Data Pjax Title      Relative Time
0 folder-row-0 architecture architecture, (Directory) Renaming K-PaaS Sep 6, 2023, 1:05 PM GMT+9 81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
0 Folder ID      Title      ... 121.0.0.0 Safari/537.36 Data Pjax Title      Relative Time
0 folder-row-0 architecture architecture, (Directory) Renaming K-PaaS Sep 6, 2023, 1:05 PM GMT+9 81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
1 folder-row-1 check-guide check-guide Replace terraform with opentofu Dec 26, 2023, 1:45 PM GMT+9 118.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
1 11.234.192.42 [30/Jun/2024:17:06:48 +0900] "POST /task_stats HTTP/1.1" 200 512 "http://180.210.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
[2 rows x 5 columns]
11.234.192.42 [30/Jun/2024:17:06:48 +0900] "POST /task_stats HTTP/1.1" 200 322 "http://180.210.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
11.234.192.42 [30/Jun/2024:17:06:48 +0900] "POST /task_stats HTTP/1.1" 200 322 "http://180.210.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
0 folder-row-0 architecture architecture, (Directory) Renaming K-PaaS Sep 6, 2023, 1:05 PM GMT+9 81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
1 folder-row-1 check-guide check-guide Replace terraform with opentofu Dec 26, 2023, 1:45 PM GMT+9 118.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
2 folder-row-2 install-guide install-guide Update cp-cluster-install-single.md Jan 8, 2024, 4:50 PM GMT+9 118.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
11.234.192.42 [30/Jun/2024:17:06:48 +0900] "POST /task_stats HTTP/1.1" 200 150 "http://180.210.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
[3 rows x 5 columns]
11.234.192.42 [30/Jun/2024:17:06:48 +0900] "POST /task_stats HTTP/1.1" 200 150 "http://180.210.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
0 folder-row-0 architecture architecture, (Directory) Renaming K-PaaS Sep 6, 2023, 1:05 PM GMT+9 81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
1 folder-row-1 check-guide check-guide Replace terraform with opentofu Dec 26, 2023, 1:45 PM GMT+9 118.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
2 folder-row-2 install-guide install-guide Update cp-cluster-install-single.md Jan 8, 2024, 4:50 PM GMT+9 118.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
3 folder-row-3 tech-edge tech-edge add to smi-remove Aug 28, 2023, 11:36 AM GMT+9 304 0 "http://180.210.81.241:8080/dags/cp-dags/grid" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36"
[4 rows x 5 columns]
0 folder-row-0 architecture architecture, (Directory) Renaming K-PaaS Sep 6, 2023, 1:05 PM GMT+9 81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
1 folder-row-1 check-guide check-guide Replace terraform with opentofu Dec 26, 2023, 1:45 PM GMT+9 118.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
2 folder-row-2 install-guide install-guide Update cp-cluster-install-single.md Jan 8, 2024, 4:50 PM GMT+9 118.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
3 folder-row-3 tech-edge tech-edge add to smi-remove Aug 28, 2023, 11:36 AM GMT+9 304 0 "http://180.210.81.241:8080/dags/cp-dags/grid" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36"
[5 rows x 5 columns]
0 folder-row-0 architecture architecture, (Directory) Renaming K-PaaS Sep 6, 2023, 1:05 PM GMT+9 81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
1 folder-row-1 check-guide check-guide Replace terraform with opentofu Dec 26, 2023, 1:45 PM GMT+9 118.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
2 folder-row-2 install-guide install-guide Update cp-cluster-install-single.md Jan 8, 2024, 4:50 PM GMT+9 118.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
3 folder-row-3 tech-edge tech-edge add to smi-remove Aug 28, 2023, 11:36 AM GMT+9 304 0 "http://180.210.81.241:8080/dags/cp-dags/grid" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36"
4 folder-row-4 use-guide use-guide Update v1.5.0 guide Dec 12, 2023, 6:20 PM GMT+9 304 0 "http://180.210.81.241:8080/dags/cp-dags/grid" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36"
[6 rows x 5 columns]
0 folder-row-0 architecture architecture, (Directory) Renaming K-PaaS Sep 6, 2023, 1:05 PM GMT+9 81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
1 folder-row-1 check-guide check-guide Replace terraform with opentofu Dec 26, 2023, 1:45 PM GMT+9 118.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
2 folder-row-2 install-guide install-guide Update cp-cluster-install-single.md Jan 8, 2024, 4:50 PM GMT+9 118.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
3 folder-row-3 tech-edge tech-edge add to smi-remove Aug 28, 2023, 11:36 AM GMT+9 304 0 "http://180.210.81.241:8080/dags/cp-dags/grid" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36"
4 folder-row-4 use-guide use-guide Update v1.5.0 guide Dec 12, 2023, 6:20 PM GMT+9 304 0 "http://180.210.81.241:8080/dags/cp-dags/grid" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36"
5 folder-row-5 LICENSE LICENSE Initial commit Sep 16, 2020, 11:17 AM GMT+9 2024-01-30 02:30:00+00:00, run_id=scheduled_2024-01-30T02:30:00+00:00, run_status=success, external_trigger=False, run_type=scheduled, data_interval_end=2024-01-30 02:30:00+00:00, run_end_date=2024-01-30 02:30:00+00:00, dag_hash=9510296d8537d8c60ad726c9212d9dda
[7 rows x 5 columns]
0 folder-row-0 architecture architecture, (Directory) Renaming K-PaaS Sep 6, 2023, 1:05 PM GMT+9 81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
1 folder-row-1 check-guide check-guide Replace terraform with opentofu Dec 26, 2023, 1:45 PM GMT+9 118.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
2 folder-row-2 install-guide install-guide Update cp-cluster-install-single.md Jan 8, 2024, 4:50 PM GMT+9 118.81.241:8080/home" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36"
3 folder-row-3 tech-edge tech-edge add to smi-remove Aug 28, 2023, 11:36 AM GMT+9 304 0 "http://180.210.81.241:8080/dags/cp-dags/grid" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36"
4 folder-row-4 use-guide use-guide Update v1.5.0 guide Dec 12, 2023, 6:20 PM GMT+9 304 0 "http://180.210.81.241:8080/dags/cp-dags/grid" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36"
5 folder-row-5 LICENSE LICENSE Initial commit Sep 16, 2020, 11:17 AM GMT+9 2024-01-30 02:30:00+00:00, run_id=scheduled_2024-01-30T02:30:00+00:00, run_status=success, external_trigger=False, run_type=scheduled, data_interval_end=2024-01-30 02:30:00+00:00, run_end_date=2024-01-30 02:30:00+00:00, dag_hash=9510296d8537d8c60ad726c9212d9dda
6 folder-row-6 README.md README.md Update service-mesh guide Dec 15, 2023, 4:41 PM GMT+9 2024-01-30 02:30:00+00:00, run_id=scheduled_2024-01-30T02:30:00+00:00, run_status=success, external_trigger=False, run_type=scheduled, data_interval_end=2024-01-30 02:30:00+00:00, run_end_date=2024-01-30 02:30:00+00:00, dag_hash=9510296d8537d8c60ad726c9212d9dda


```

6. Web Server 활용

- Web-server 접속 화면


[Airflow](#)

[DAGs](#)
[Cluster Activity](#)
[Datasets](#)
[Security](#)
[Browse](#)
[Admin](#)
[Docs](#)

08:03 UTC


Do not use **SQLite** as metadata DB in production – it should only be used for dev/testing. We recommend using Postgres or MySQL. [Click here](#) for more information.

Do not use the **SequentialExecutor** in production. [Click here](#) for more information.

DAGs

All 1

Active 1



Paused 0








































Running 8

Failed 0

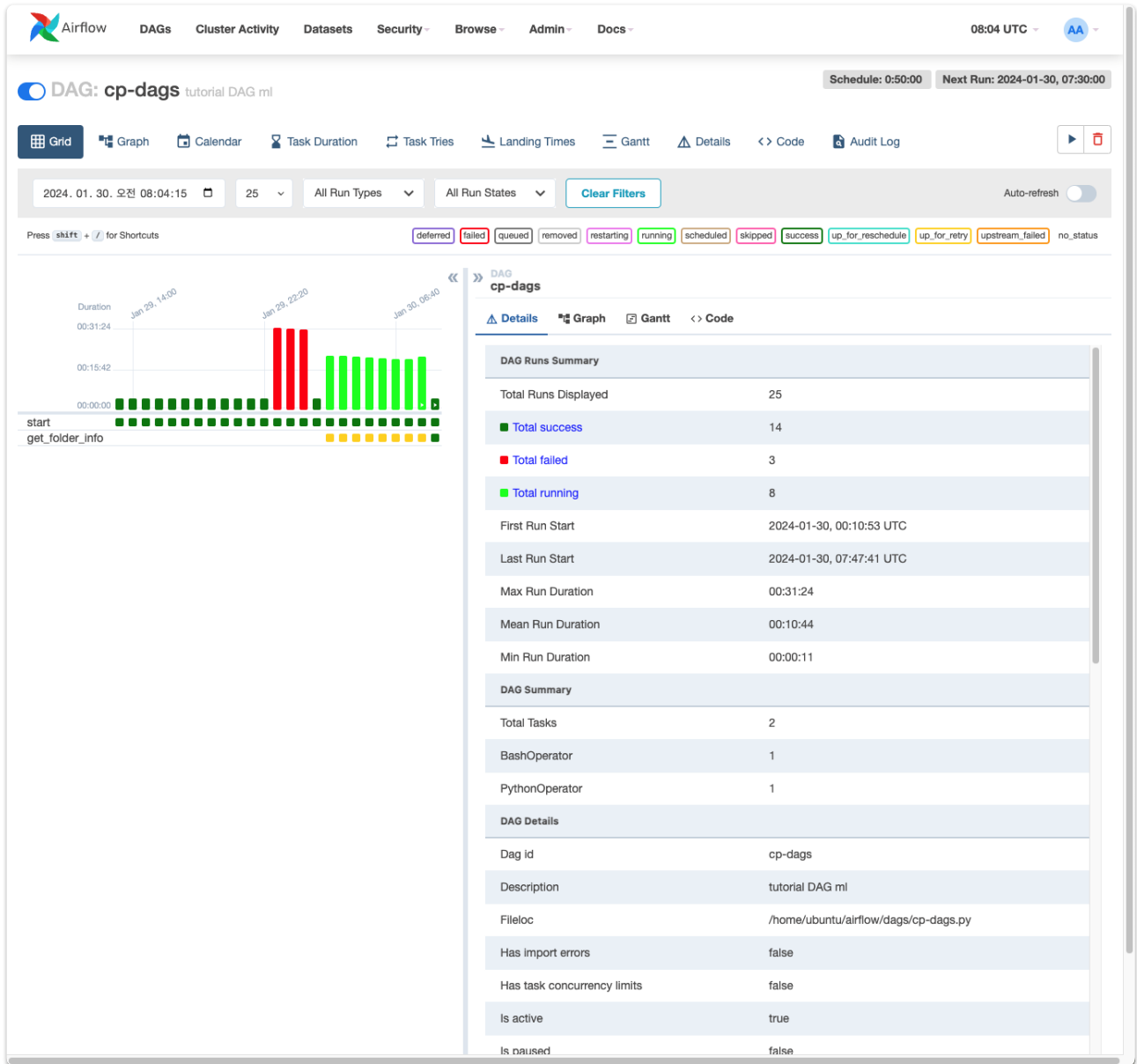
Filter DAGs by tag

Search DAGs

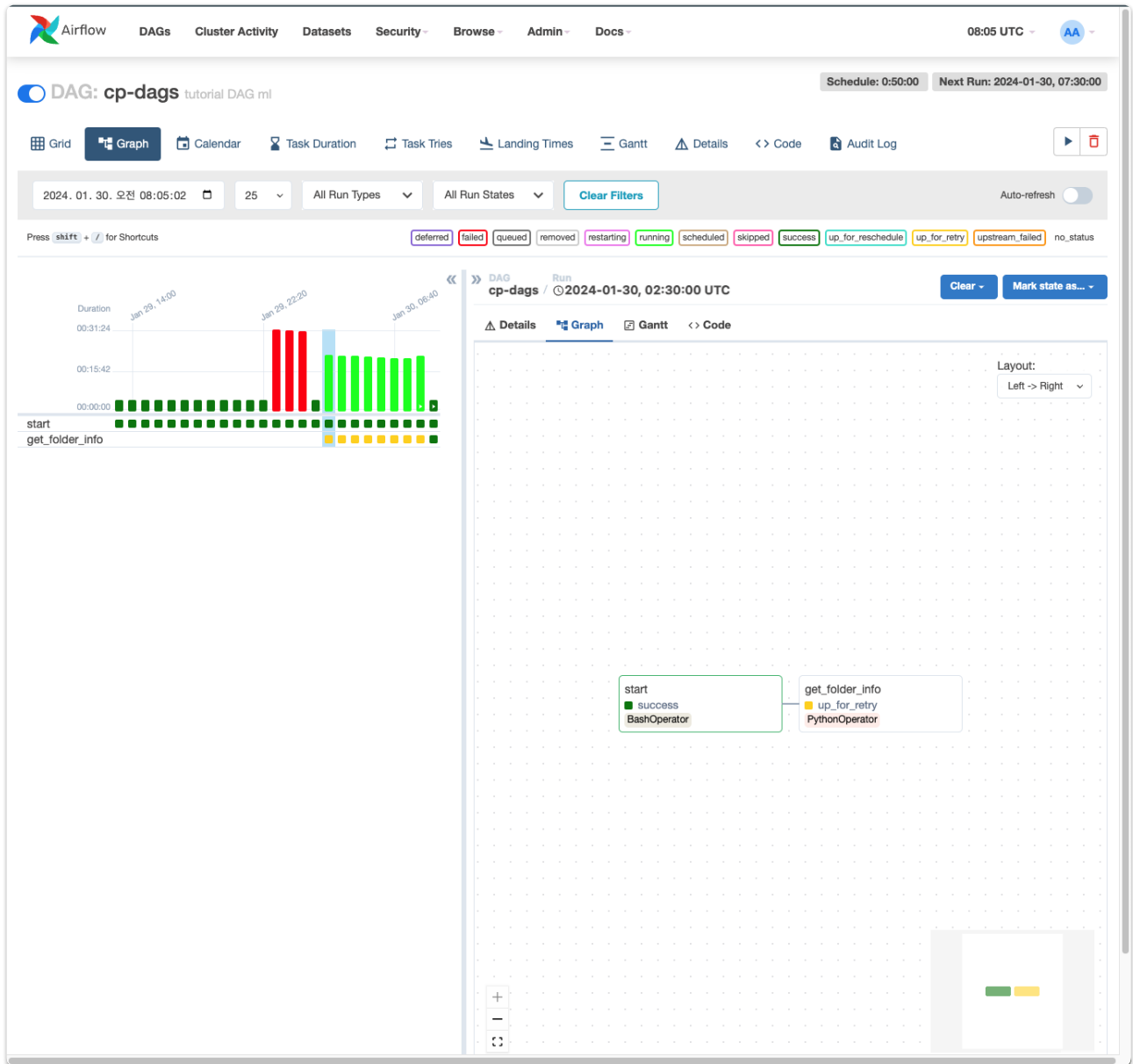
 Auto-refresh
 

DAG	Owner	Runs	Schedule	Last Run	Next Run	Recent Tasks	Actions	Links
 cp-dags cp-test-baml	admin	  	0:50:00	2024-01-30, 07:47:40	2024-01-30, 07:30:00	                                  		

- cp-dags 둘러보기



- Dags 파일에 정의된 task들의 Graph view



- Task 수행에 실패 할 경우 Log를 볼 수 있다.

Airflow DAGs Cluster Activity Datasets Security Browse Admin Docs 08:05 UTC AA

DAG: cp-dags tutorial DAG ml Schedule: 0:50:00 Next Run: 2024-01-30, 07:30:00

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

2024. 01. 30. 오전 08:05:02 25 All Run Types All Run States Clear Filters Auto-refresh

Press **shift** + **/** for Shortcuts deferred failed queued removed restarting running scheduled skipped success up_for_reschedule up_for_retry upstream_failed no_status

DAG Run **cp-dags** / 2024-01-30, 02:30:00 UTC Task **get_folder_info** Clear task Mark state as... Filter Tasks

Details Graph Gantt Code **Logs** XCom

(by attempts)

1

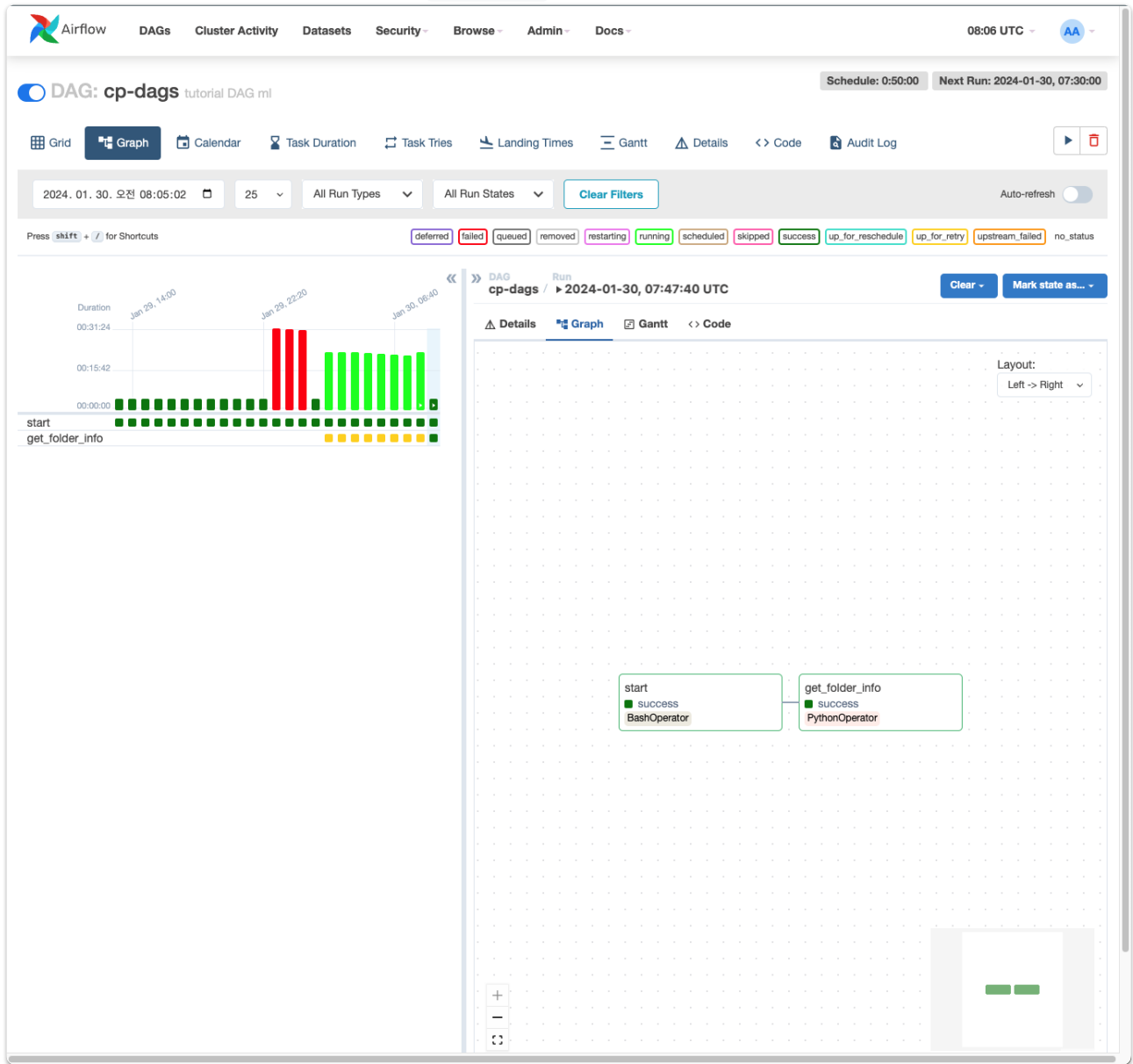
All Levels All File Sources Wrap Download See More

```

bani.novalocal
*** Found local files:
*** /home/ubuntu/airflow/logs/dag_id=cp-dags/run_id=scheduled__2024-01-30T01:40:00+00:00/task_id=get_folder_info/
[2024-01-30, 07:43:36 UTC] (taskinstance.py:1956) INFO - Dependencies all met for dep_context=non-requeueable deps ti
[2024-01-30, 07:43:36 UTC] (taskinstance.py:1956) INFO - Dependencies all met for dep_context=queueable deps ti=Ta
[2024-01-30, 07:43:36 UTC] (taskinstance.py:2170) INFO - Starting attempt 1 of 2
[2024-01-30, 07:43:36 UTC] (taskinstance.py:2191) INFO - Executing <Task(PythonOperator): get_folder_info> on 2024-01
[2024-01-30, 07:43:36 UTC] (standard_task_runner.py:60) INFO - Started process 49520 to run task
[2024-01-30, 07:43:36 UTC] (standard_task_runner.py:87) INFO - Running: ['airflow', 'tasks', 'run', 'cp-dags', 'get_f
[2024-01-30, 07:43:36 UTC] (standard_task_runner.py:88) INFO - Job 737: Subtask get_folder_info
[2024-01-30, 07:43:36 UTC] (task_command.py:423) INFO - Running <TaskInstance: cp-dags.get_folder_info scheduled__202
[2024-01-30, 07:43:36 UTC] (taskinstance.py:2480) INFO - Exporting env vars: AIRFLOW_CTX_DAG_EMAIL='admin@admin.com'
[2024-01-30, 07:43:43 UTC] (logging_mixin.py:188) INFO - Folder ID Title Aria Label
0 folder-row-0 architecture architecture, (Directory) Renaming K-PaaS Sep 6, 2023, 1:05
1 folder-row-1 check-guide check-guide, (Directory) Replace terraform with opentofu Dec 26, 2023, 1:45
2 folder-row-2 install-guide install-guide, (Directory) Update cp-cluster-install-single.md Jan 8, 2024, 4:50
3 folder-row-3 tech-edge tech-edge, (Directory) add to smi-remove Aug 28, 2023, 11:36
4 folder-row-4 use-guide use-guide, (Directory) Update vi.5.0 guide Dec 12, 2023, 6:20
5 folder-row-5 LICENSE LICENSE, (File) Initial commit Sep 16, 2020, 11:17
6 folder-row-6 README.md README.md, (File) Update service-mesh guide Dec 15, 2023, 4:41
[2024-01-30, 07:43:43 UTC] (taskinstance.py:2698) ERROR - Task failed with exception
Traceback (most recent call last):
  File "/home/ubuntu/.local/lib/python3.10/site-packages/airflow/models/taskinstance.py", line 433, in _execute_task
    result = execute_callable(context=context, **execute_callable_kwargs)
  File "/home/ubuntu/.local/lib/python3.10/site-packages/airflow/operators/python.py", line 199, in execute
    return_value = self.execute_callable()
  File "/home/ubuntu/.local/lib/python3.10/site-packages/airflow/operators/python.py", line 216, in execute_callable
    return self.python_callable(*self.op_args, **self.op_kwargs)
  File "/home/ubuntu/airflow/webcrawler/gitcrawl.py", line 47, in get_folder_info
    date_today = datetime.datetime.now().strftime("%Y%m%d")
NameError: name 'datetime' is not defined
[2024-01-30, 07:43:43 UTC] (taskinstance.py:1138) INFO - Marking task as UP_FOR_RETRY. dag_id=cp-dags, task_id=get_fo
[2024-01-30, 07:43:43 UTC] (standard_task_runner.py:107) ERROR - Failed to execute job 737 for task get_folder_info (
[2024-01-30, 07:43:43 UTC] (local_task_job_runner.py:234) INFO - Task exited with return code 1
[2024-01-30, 07:43:43 UTC] (taskinstance.py:3280) INFO - 0 downstream tasks scheduled from follow-on schedule check

```

- 정의된 Dag의 Tasks들이 성공하면 `success` 가 출력된다.



7. 참고 사항

- i** Web-Server에서 Dag를 실행해보지 않아도 airflow CLI를 통해 로그를 미리 볼 수 있다.
 - Dag에 정의된 Task 살펴보기

```
airflow tasks list {DAG_NAME} --tree
```

- Task 미리 실행해보기

```
airflow tasks test {DAG_ID} {TASK_ID} {START_DATE}
```

- i** 웹서버 강제 종료 후 Server 중지 시키기

- 이미 웹서버가 동작 중일경우 실행이 안될 수 있다.
- 아래 표시된 두 개의 이름을 가진 Pid를 종료시킨다.

```

tcp        0      0 127.0.0.1:54717      0.0.0.0:*             LISTEN      1000       70113      8711/chromedriver
tcp        0      0 0.0.0.0:8080         0.0.0.0:*             LISTEN      1000       44431      4358/gunicorn: mas
tcp        0      0 127.0.0.1:46489      0.0.0.0:*             LISTEN      1000       72477      8904/chromedriver
tcp        0      0 127.0.0.1:38330      0.0.0.0:*             LISTEN      1000       70330      8638/chromedriver

tcp6       0      0 :::1:35109           :::*                   LISTEN      1000       70918      8670/chromedriver
tcp6       0      0 :::8793              :::*                   LISTEN      1000       83304      10190/gunicorn: mas
tcp6       0      0 :::1:47639           :::*                   LISTEN      1000       74910      8998/chromedriver

```

```
kill {PID_NUM}
```

- Chrome Server 중지 시키기

```
pkill chrome
```