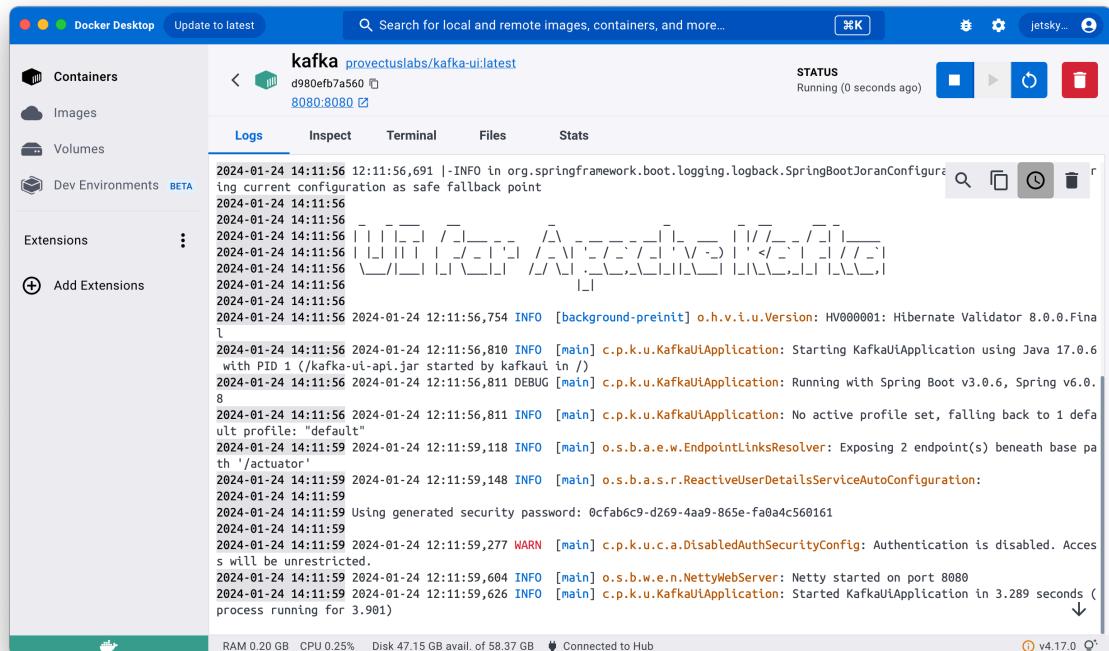
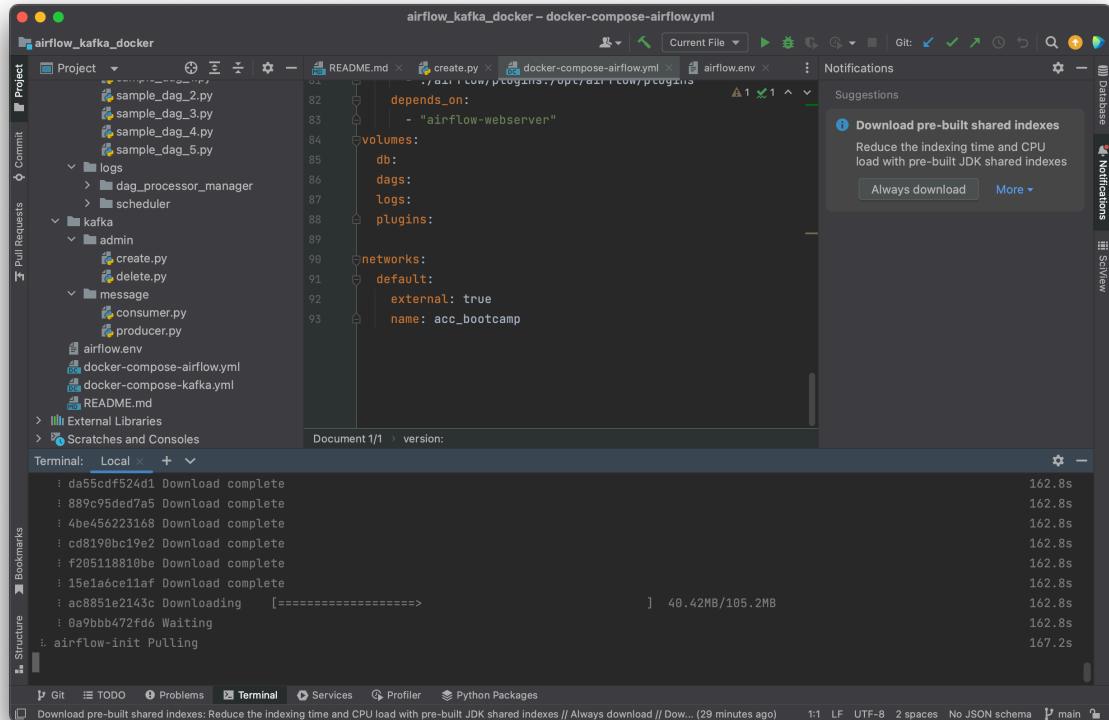


## **Day 13 – Airflow – Accenture bootcamp**

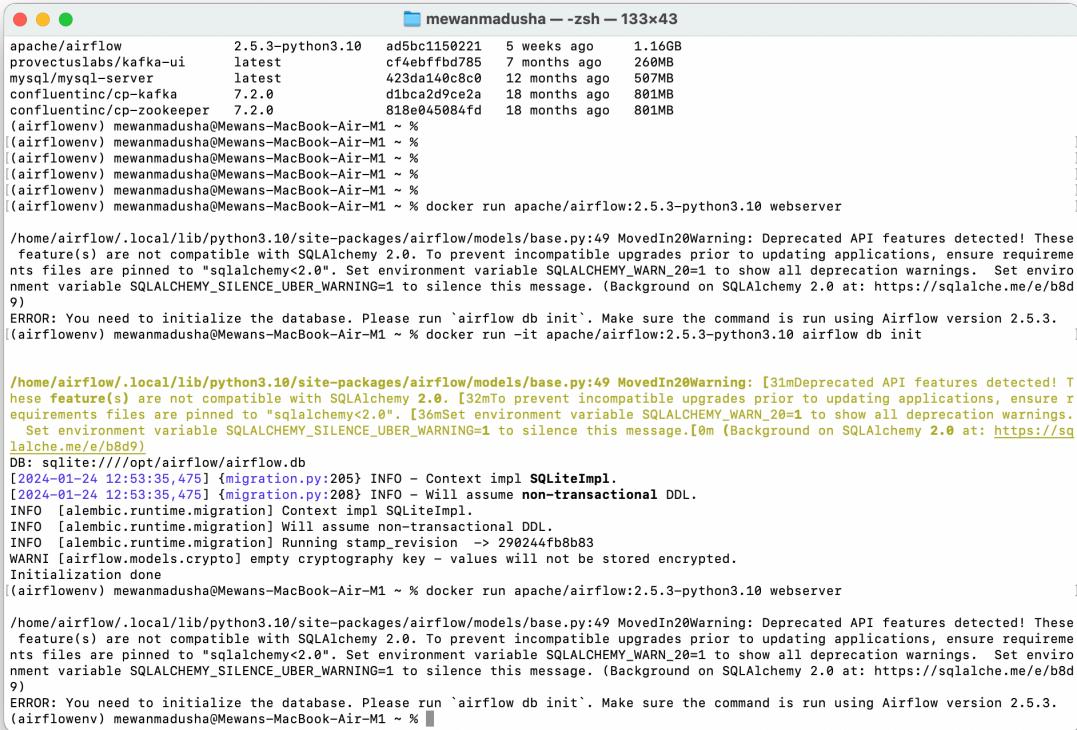
**Name: Pothumulla Kankanamge Mewan Madhusha**

## Clone and compose given docker repository

```
docker-compose -f docker-compose-airflow.yml -f docker-compose-kafka.yml up
```



```
docker run apache/airflow:2.5.3-python3.10 webserver
```



```
mewanmadusha -- zsh - 133x43
apache/airflow      2.5.3-python3.10    ad5bc1150221   5 weeks ago   1.16GB
provectuslabs/kafka-ui  latest        cf4ebffbd785   7 months ago   260MB
mysql/mysql-server  latest        423da140c8c0   12 months ago  507MB
confluentinc/cp-kafka  7.2.0       d1bca2d9ce2a   18 months ago  801MB
confluentinc/cp-zookeeper  7.2.0       818e045084fd   18 months ago  801MB
(airflowenv) mewanmadusha@Mewans-MacBook-Air-M1 ~ %
(airflowenv) mewanmadusha@Mewans-MacBook-Air-M1 ~ %
(airflowenv) mewanmadusha@Mewans-MacBook-Air-M1 ~ %
(airflowenv) mewanmadusha@Mewans-MacBook-Air-M1 ~ %
(airflowenv) mewanmadusha@Mewans-MacBook-Air-M1 ~ % docker run apache/airflow:2.5.3-python3.10 webserver
/home/airflow/.local/lib/python3.10/site-packages/airflow/models/base.py:49 MovedIn20Warning: Deprecated API features detected! These feature(s) are not compatible with SQLAlchemy 2.0. To prevent incompatible upgrades prior to updating applications, ensure requirements files are pinned to "sqlalchemy<2.0". Set environment variable SQLALCHEMY_WARN_20=1 to show all deprecation warnings. Set environment variable SQLALCHEMY_SILENCE_UBER_WARNING=1 to silence this message. (Background on SQLAlchemy 2.0 at: https://sqlalche.me/e/b8d9)
ERROR: You need to initialize the database. Please run `airflow db init`. Make sure the command is run using Airflow version 2.5.3.
(airflowenv) mewanmadusha@Mewans-MacBook-Air-M1 ~ % docker run -it apache/airflow:2.5.3-python3.10 airflow db init
[...]
/home/airflow/.local/lib/python3.10/site-packages/airflow/models/base.py:49 MovedIn20Warning: [31mDeprecated API features detected! These feature(s) are not compatible with SQLAlchemy 2.0. [32mTo prevent incompatible upgrades prior to updating applications, ensure requirements files are pinned to "sqlalchemy<2.0". [33mSet environment variable SQLALCHEMY_WARN_20=1 to show all deprecation warnings. Set environment variable SQLALCHEMY_SILENCE_UBER_WARNING=1 to silence this message.[0m (Background on SQLAlchemy 2.0 at: https://sqlalche.me/e/b8d9)
DB: sqlite:///opt/airflow/airflow.db
[2024-01-24 12:53:35,475] {migration.py:205} INFO - Context impl SQLiteImpl.
[2024-01-24 12:53:35,475] {migration.py:208} INFO - Will assume non-transactional DDL.
INFO [alembic.runtime.migration] Context impl SQLiteImpl.
INFO [alembic.runtime.migration] Will assume non-transactional DDL.
INFO [alembic.runtime.migration] Running stamp_revision -> 290244fb8b83
WARN [airflow.models.crypto] empty cryptography key - values will not be stored encrypted.
Initialization done
(airflowenv) mewanmadusha@Mewans-MacBook-Air-M1 ~ % docker run apache/airflow:2.5.3-python3.10 webserver
[...]
/home/airflow/.local/lib/python3.10/site-packages/airflow/models/base.py:49 MovedIn20Warning: Deprecated API features detected! These feature(s) are not compatible with SQLAlchemy 2.0. To prevent incompatible upgrades prior to updating applications, ensure requirements files are pinned to "sqlalchemy<2.0". Set environment variable SQLALCHEMY_WARN_20=1 to show all deprecation warnings. Set environment variable SQLALCHEMY_SILENCE_UBER_WARNING=1 to silence this message. (Background on SQLAlchemy 2.0 at: https://sqlalche.me/e/b8d9)
ERROR: You need to initialize the database. Please run `airflow db init`. Make sure the command is run using Airflow version 2.5.3.
(airflowenv) mewanmadusha@Mewans-MacBook-Air-M1 ~ %
```

Since provided docker file airflow version is not working, tried with official Apache/airflow image

<https://hub.docker.com/r/apache/airflow>

This method also didn't work.

Next install airflow standalone with pip install.

```
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}"
```

Start webserver.

Run all at once.

```
airflow standalone
```

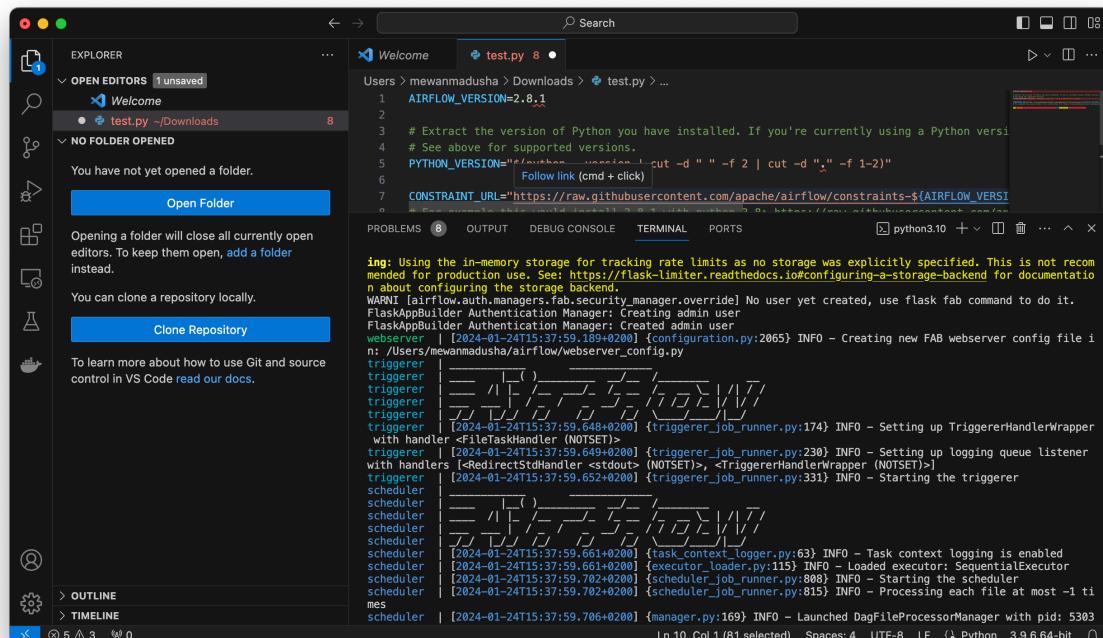
Or run each functionality separately.

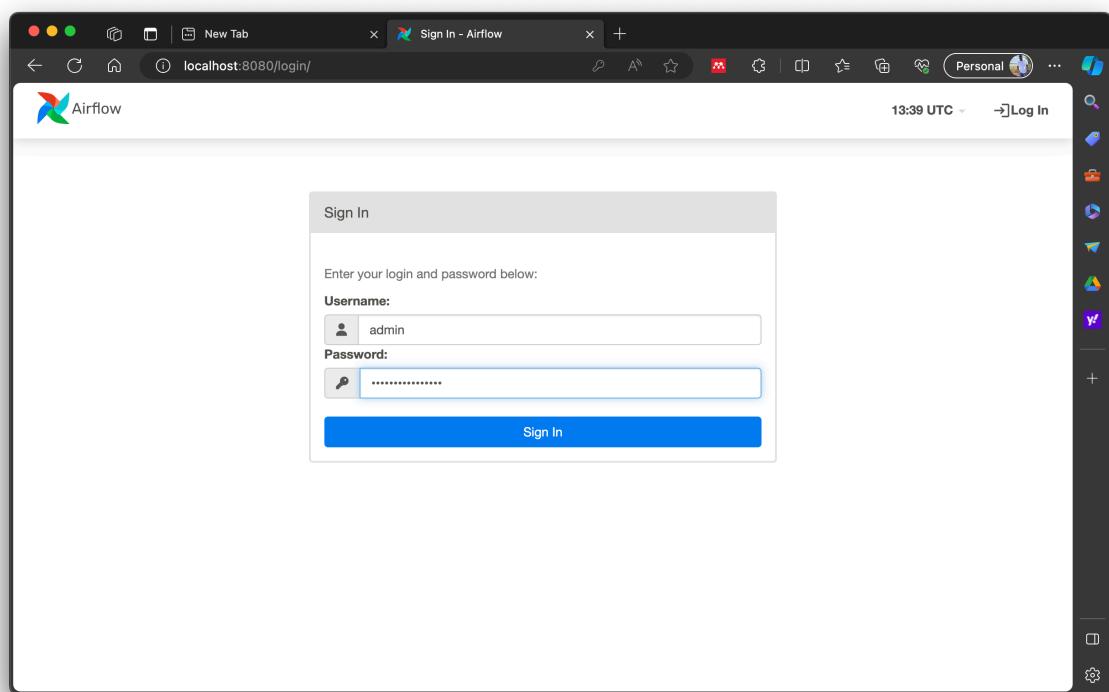
```
airflow db migrate
```

```
airflow users create \  
    --username admin \  
    --firstname Peter \  
    --lastname Parker \  
    --role Admin \  
    --email spiderman@superhero.org
```

```
airflow webserver --port 8080
```

```
airflow scheduler
```





A screenshot of a web browser showing the Airflow DAGs page. The URL is `localhost:8080/home`. The page title is "DAGs - Airflow". It displays a list of DAGs with the following details:

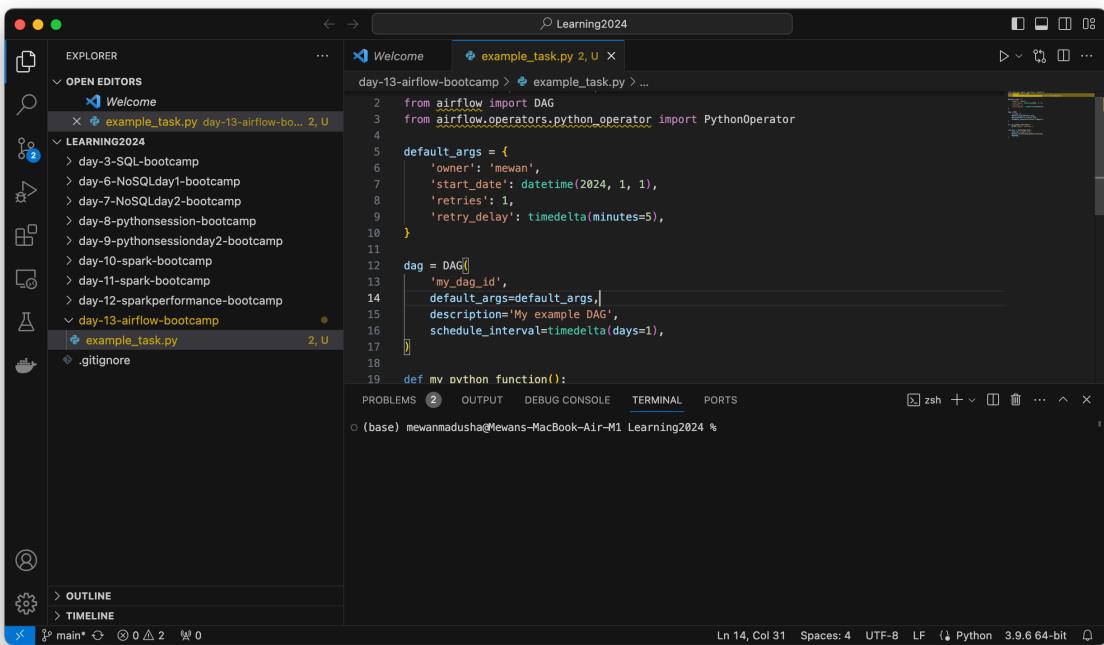
DAG	Owner	Runs	Schedule	Last Run	Next Run	Recent Ta
<code>dataset_consumes_1</code> consumes dataset-scheduled	airflow	○ ○ ○ ○	Dataset	On s3://dagf/output_1.txt		
<code>dataset_consumes_1_and_2</code> consumes dataset-scheduled	airflow	○ ○ ○ ○	Dataset	0 of 2 datasets updated		
<code>dataset_consumes_1_never_scheduled</code> consumes dataset-scheduled	airflow	○ ○ ○ ○	Dataset	0 of 2 datasets updated		
<code>dataset_consumes_unknown_never_scheduled</code> dataset-scheduled	airflow	○ ○ ○ ○	Dataset	0 of 2 datasets updated		

The page also includes navigation links for `DAGs`, `Cluster Activity`, `Datasets`, `Security`, `Browse`, `Admin`, and `Docs`, and a status bar indicating `13:40 UTC` and `AU`.

Create python file (you can do it into dag file as well but its good to process external py file with its path!)

Create dag as attached file

Create external python file.



The screenshot shows a code editor window with the title "Learning2024". The left sidebar shows a file tree with a folder named "LEARNING2024" containing several sub-folders like "day-3-SQL-bootcamp" and "day-13-airflow-bootcamp". Inside "day-13-airflow-bootcamp", there is a file named "example\_task.py" which is currently open. The code in the editor is:

```
from airflow import DAG
from airflow.operators.python_operator import PythonOperator

default_args = {
    'owner': 'mewan',
    'start_date': datetime(2024, 1, 1),
    'retries': 1,
    'retry_delay': timedelta(minutes=5),
}

dag = DAG(
    "my_dag_id",
    default_args=default_args,
    description='My example DAG',
    schedule_interval=timedelta(days=1),
)

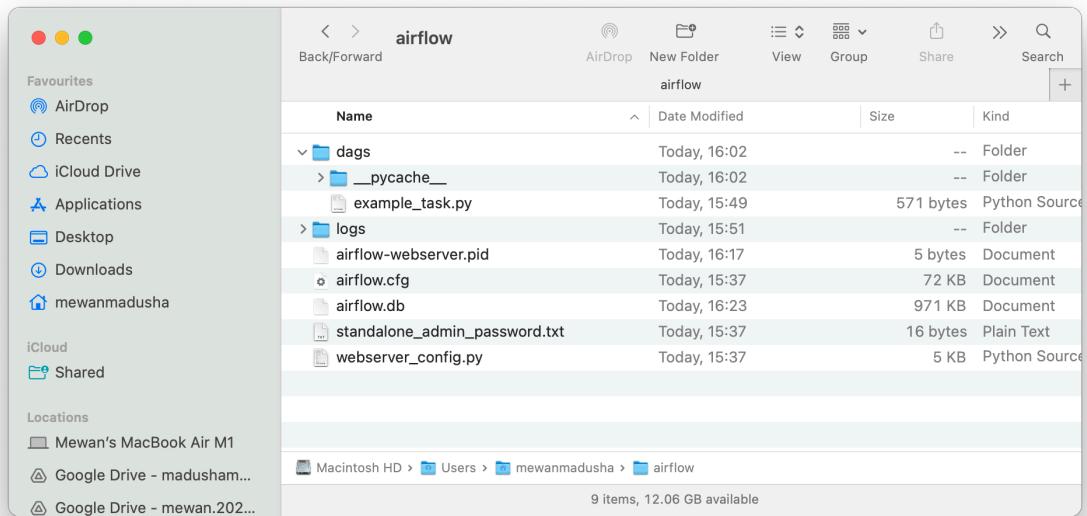
def my_python_function():
    pass
```

The code editor has tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS. The DEBUG CONSOLE tab shows the command "(base) mewanmadusha@Mewans-MacBook-Air-M1 Learning2024 %". The bottom status bar indicates "Ln 14, Col 31 Spaces: 4 UTF-8 LF Python 3.9.6 64-bit".

Put into Dag folder

Past it into the dag folder

`~/airflow/dags`



Couple of minutes later you can see on UI.

"my\_dag\_id" is external file

DAG	Owner	Runs	Last Run
tutorial_taskflow_api	airflow	0	
tutorial_objectstorage	airflow	0	
tutorial_dag	airflow	0	
tutorial	airflow	1 day, 0:00:00	2024-01-23, 00:00:00
my_dag_id	mewan	1 day, 0:00:00	2024-01-01, 00:00:00
latest_only_with_trigger	airflow	4:00:00	2024-01-24, 08:00:00
latest_only	airflow	4:00:00	2024-01-24, 08:00:00
example_xcom_args_with_operators	airflow	None	

The screenshot shows the Airflow web interface for the DAG `my_dag_id`. The top navigation bar includes links for Airflow, DAGs, Cluster Activity, Datasets, Security, Browse, Admin, and Docs. The current time is 14:26 UTC. The main title is "DAG: my\_dag\_id My example DAG". Below the title, there are tabs for Grid, Graph (which is selected), Calendar, Task Duration, Task Tries, Landing Times, Gantt, and Details. A sidebar on the right contains icons for various Airflow components like DAGs, Tasks, and Metrics.

The Graph tab displays a timeline from 2024-01-24, 14:24:38 to 2024-01-25, 00:00:00. It shows a single task named `example_task` which is a PythonOperator. The task is currently in the `running` state. There are filters at the top for All Run Types, All Run States, and a "Clear Filters" button. A "Auto-refresh" toggle is also present.

The screenshot shows the Airflow web interface for the log view of the DAG `my_dag_id`. The top navigation bar and sidebar are identical to the previous screenshot. The main title is "DAG my\_dag\_id Run 2024-01-24, 14:26:40 UTC Task example\_task". The Graph tab is selected, showing a histogram of task durations. The Log tab is selected, displaying the command-line output of the task's execution. The log output shows the task starting at 14:26:43 UTC and executing the Python code `print("Hello Airflow")`.

```
mewans-macbook-air-ml.local
*** Found local files:
[2024-01-24, 14:26:43 UTC] {taskinstance.py:1956} INFO - Dependencies all met for dep_context
[2024-01-24, 14:26:43 UTC] {taskinstance.py:1956} INFO - Dependencies all met for dep_context
[2024-01-24, 14:26:43 UTC] {taskinstance.py:2170} INFO - Starting attempt 1 of 1
[2024-01-24, 14:26:43 UTC] {taskinstance.py:2191} INFO - Executing <task[pythonOperator]>: example_task
[2024-01-24, 14:26:43 UTC] {standard_task_runner.py:87} INFO - Running on ['airflow', 'tasks']
[2024-01-24, 14:26:43 UTC] {standard_task_runner.py:88} INFO - Job 10: Subtask example_task
[2024-01-24, 14:26:43 UTC] {task_command.py:423} INFO - Running <taskinstance: my_dag_id>
[2024-01-24, 14:26:43 UTC] {taskinstance.py:2480} INFO - Exporting env vars: AIRFLOW_CTX_DAG_ID=example_task
[2024-01-24, 14:26:43 UTC] {logging_mixin.py:188} INFO - Hello, Airflow!
[2024-01-24, 14:26:43 UTC] {python.py:201} INFO - Done. Returned value was: None
[2024-01-24, 14:26:43 UTC] {taskinstance.py:1138} INFO - Marking task as SUCCESS. dag_id=my_dag_id
[2024-01-24, 14:26:43 UTC] {local_task_job_runner.py:234} INFO - Task exited with return code 0
[2024-01-24, 14:26:43 UTC] {taskinstance.py:3280} INFO - 0 downstream tasks scheduled from this task
```

Print “Hello Airflow”