Deploy to Astro

It is time! Time to deploy the DAG you've built during this course on Astro! Without further ado, let's go!

Prerequisites

The DAG extract_stars.py under the folder dags should look like that:

```
from airflow import DAG
from airflow.operators.bash import BashOperator
from airflow.providers.http.operators.http import SimpleHttpOperator
from airflow.operators.python import PythonOperator
import json
from datetime import datetime
def _print_stargazers(github_stats: str, date: str):
 github_stats_json = json.loads(github_stats)
 airflow_stars = github_stats_json.get("stargazers_count")
 print(f"As of {date}, Apache Airflow has {airflow_stars} stars on Github!")
with DAG('extract_stars', schedule_interval='@daily', start_date=datetime(2022, 1, 1), cat
chup=False) as dag:
    get_date = BashOperator(
        task_id="get_date",
        bash_command="echo {{ ds.format('yyyy') }}"
    query_github_stats = SimpleHttpOperator(
        task_id="query_github_stats",
        endpoint="{{ var.value.endpoint }}",
       method="GET",
       http_conn_id="github_api",
        log_response=True
   )
    print_stargazers = PythonOperator(
        task_id="print_stars",
       python_callable=_print_stargazers,
        op_args=["{{ ti.xcom_pull(task_ids='query_github_stats') }}", "{{ ti.xcom_pull(tas
k_ids='get_date') }}"]
```

```
get_date >> query_github_stats >> print_stargazers
```

With the Connection:

• Connection Id: github_api

• Connection Type: HTTP

• Host: api.github.com

• Schema: https

And the Variable:

• Key: endpoint

• Val: repos/apache/airflow

Authenticate to Astro

Once you've tested your DAGs locally with astro dev parse

Authenticate to the Astro CLI by running astro login

```
~/astro astro login
Welcome to the Astro CLI //
To learn more about Astro, go to https://docs.astronomer.io
Logging in with saved user marc@astronomer.io
Press Enter to open the browser to log in or ^C to quit...
```

After running this command, you will be prompted to open your web browser and log in to the Cloud UI. Once you complete this login, you will be automatically authenticated to the CLI.



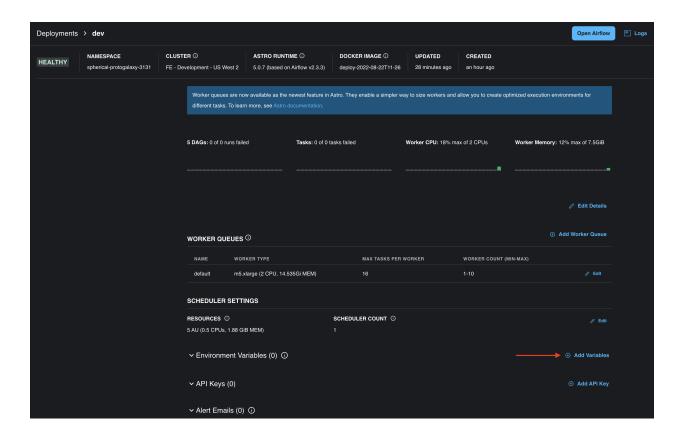


Add Variables and Connections to Astro

Before deploying your DAG, you need to recreate the connections/variables that you've created locally on Astro.

Got to your Astro deployment dev

Click Add Variables



We are going to create two environment variables. One for the connection, one for the variable.

For the connection:

KEY: AIRFLOW_CONN_GITHUB_API

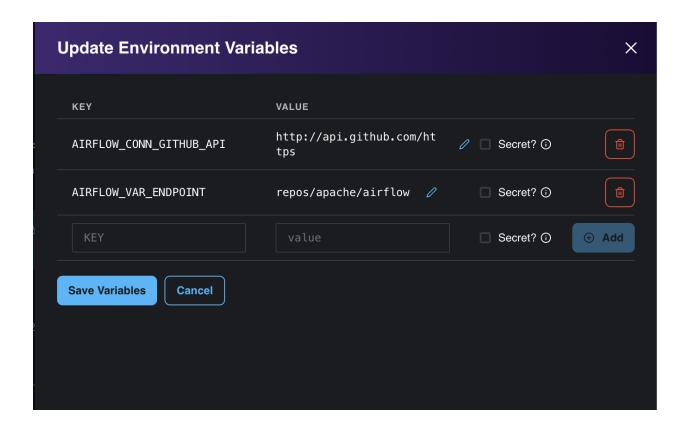
VALUE: http://api.github.com/https

Click Add

For the variable:

KEY: AIRFLOW_VAR_ENDPOINT

VALUE: repos/apache/airflow



Click Save Variables

Deploy to Astro

Back to the Astro CLI, you can list workspaces with astro workspaces list

NAME ID
Dogfood Production g4elywerg190081hzgdwc08cei
Onboarding - Customer Success 342dg4ely190081hzgdwc08cei
Education c148g4edfess234422dwc08cei
Data Engineering c174nlx32434324fyoa6vtvj86

Pick your workspace with astro workspace switch <workspace-id>

Next, run astro deploy and pick your deployment

```
Authenticated to Astro

Current Workspace: Data Engineering

Select which Deployment you want to deploy to:
# DEPLOYMENT NAME RELEASE NAME DEPLOYMENT ID
1 dev spherical-protogalaxy-3131 cl74npow0485901j0ikhpsgf4r

> 1
Deploying: spherical-protogalaxy-3131
Building image...
[+] Building 1.6s (11/11) FINISHED
```

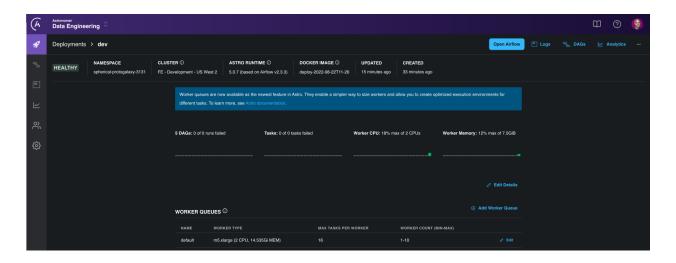
After a few minutes, you should end up with something like that:

```
Deployed Image Tag: deploy-2022-08-22T11-26
Successfully pushed Docker image to Astronomer registry. Navigate to the Astronomer UI for confirmation that your deploy was successful.

Deployment can be accessed at the following URLs:

Deployment Dashboard: cloud.astronomer.io/cknaqyipv0573levsry6cj4n0/deployments/cl74npow0485901j0ikhpsgf4r
Airflow Dashboard: astronomer.astronomer.run/dhpsgf4r?orgId=org_0FGQMRjnZdcx3EDk
```

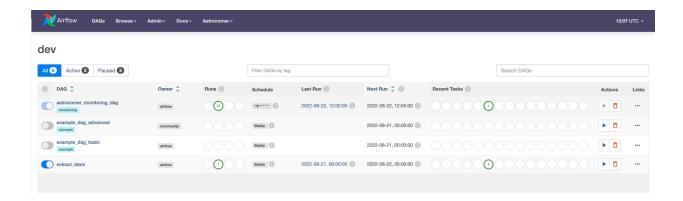
Back to your Deployment on Astro, wait until the status turned back to Healthy



Run you DAG

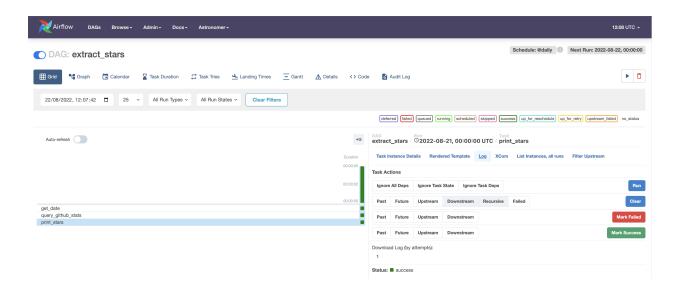
Click Open Airflow

Turn on the toggle of extract_stars DAG and wait a little bit.



Click extract_stars

Then click on the task print_stars and Log



You should get something like that



Well done! You've just successfully built and deployed your first DAG on Astro!



Additional resources

Astro Environment Variables: https://docs.astronomer.io/astro/environment-variables

Deploy on Astro: https://docs.astronomer.io/astro/deploy-code