

Deploy Code For Production

Capstone Step 9

Approach

Created Docker Container Packaging with Scripts to Deploy:

1. Azure Data Lake Gen2
2. Azure MySQL Database
3. Azure Databricks
4. Azure Data Factory

Decisions

First Attempted Pipeline Orchestration via Airflow:

Connectors from Airflow to ADF were riddled with bugs

Then Attempted with ADF, using Batch jobs for logic handling:

Realized spinning up pools/jobs/tasks was overkill complexity for ETL

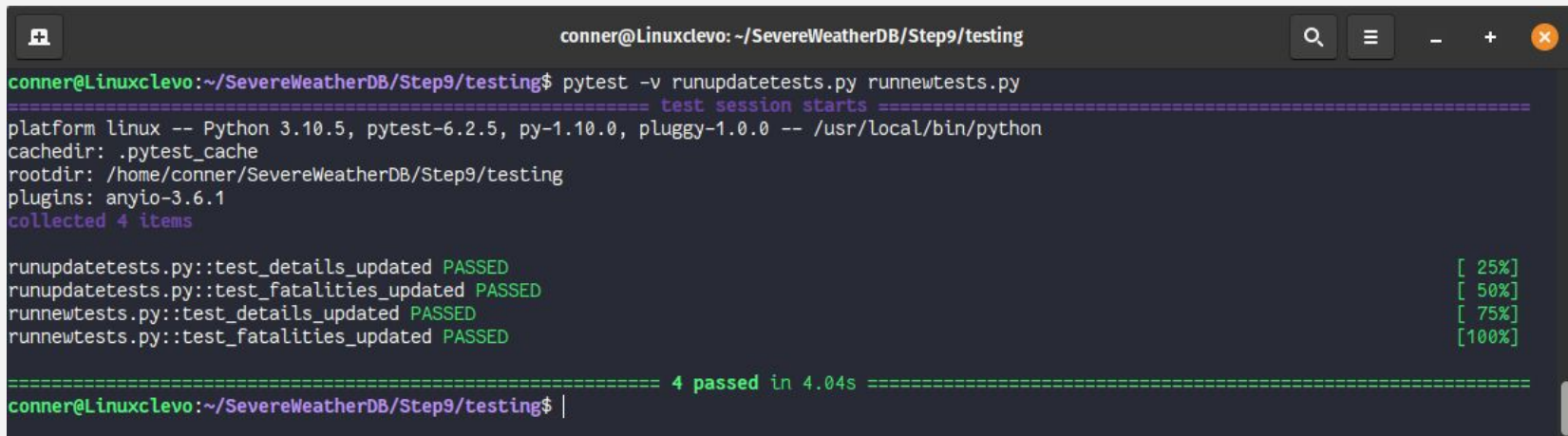
Finally settled on calling Python script via Databricks runtime in ADF:

Possibly over resourced, but future-proofs against future uncertainties

Verification

I had already vetted the basic ingestion in Step 8 tests

This final deployment added tests for all edge cases and iterative updates

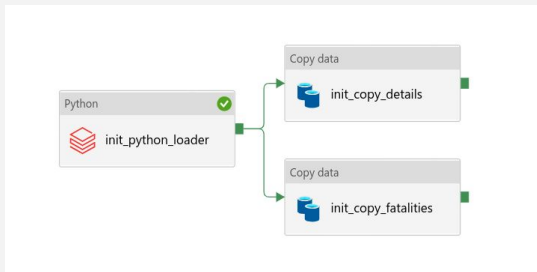


```
conner@Linuxclevo: ~/SevereWeatherDB/Step9/testing
conner@Linuxclevo:~/SevereWeatherDB/Step9/testing$ pytest -v runupdatetests.py runnewtests.py
===== test session starts =====
platform linux -- Python 3.10.5, pytest-6.2.5, py-1.10.0, pluggy-1.0.0 -- /usr/local/bin/python
cachedir: .pytest_cache
rootdir: /home/conner/SevereWeatherDB/Step9/testing
plugins: anyio-3.6.1
collected 4 items

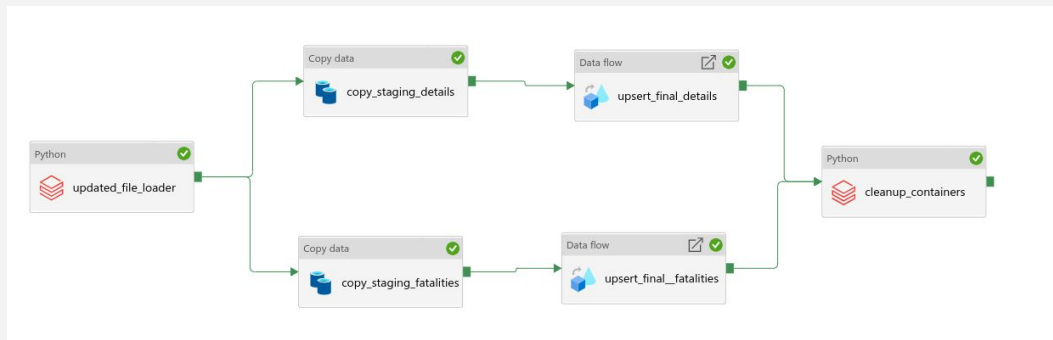
runupdatetests.py::test_details_updated PASSED [ 25%]
runupdatetests.py::test_fatalities_updated PASSED [ 50%]
runnewtests.py::test_details_updated PASSED [ 75%]
runnewtests.py::test_fatalities_updated PASSED [100%]

===== 4 passed in 4.04s =====
conner@Linuxclevo:~/SevereWeatherDB/Step9/testing$ |
```

Initial



Update



New

