

# **ЛР 1. Airflow + docker** compose

# ▼ Задача

Развернуть Apache Airflow с помощью docker-compose

### ▼ Описание

Для непосредственно деплоя можно использовать композ файл с этой страницы: <a href="https://airflow.apache.org/docs/apache-airflow/stable/howto/docker-compose/index.html">https://airflow.apache.org/docs/apache-airflow/stable/howto/docker-compose/index.html</a>
Подготовка

docker-compose.yml:

- вырезать сервисы redis, airflow-worker, airflow-triggerer, airflow-cli, flower - главное, чтоб были webserver, scheduler, init и БД к ним (postgres)
- поменять дефолтный образ на apache/airflow:2.7.1 (заменить image ЛИБО прописать переменную AIRFLOW\_IMAGE\_NAME)
- поменять значение переменной AIRFLOW\_core\_executor с

  CeleryExecutor на LocalExecutor т.к. мы отключаем сервис airflowworker
- убрать из блока <a href="mailtow-common-depends-on">&airflow-common-depends-on</a> две строчки, касающийся redis - т.к. мы его отключаем, нам не нужна такая зависимость контейнеров друг от друга
- <u>при желании</u> можно заполнить значения переменных \_\_airflow\_www\_user\_username и \_\_airflow\_www\_user\_password, тогда для подключения через браузер будут использоваться эти креды (вместо дефолтных airflow:airflow)

Если все ок, то после выполнения docker-compose up -d и docker ps должен показаться список из двух контейнеров airflow и одного postgres. Если у контейнеров состояние (health: starting), то необходимо дождаться (healthy)

```
S docken-compose yes

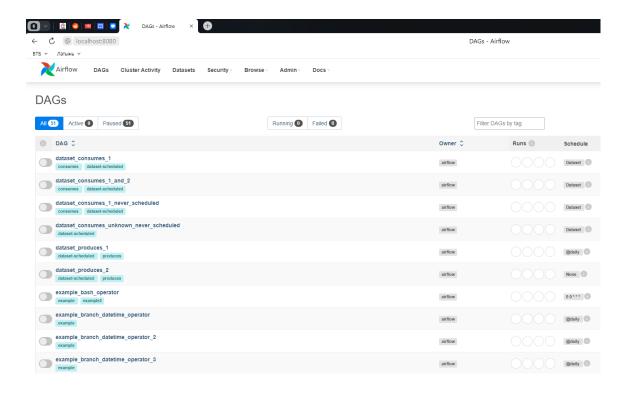
$ docken-compose up 93

$ docken-compose up 94

$ docken
```

Попасть на Airflow через браузер можно по адресу <a href="http://localhost:8080/">http://localhost:8080/</a>
. Там будет множество примеров DAG-ов

ЛР 1. Airflow + docker compose 2



# ▼ Вопросы

- 1. Для чего нужен docker-compose? Своими словами, без определений из интернета?
- 2. Как в docker-compose сделать ограничения для контейнера по ресурсам (CPU, RAM)?

## ▼ Отчетность

Ссылка на репозиторий, где содержатся

- docker-compose.yml
- ответы на вопросы