

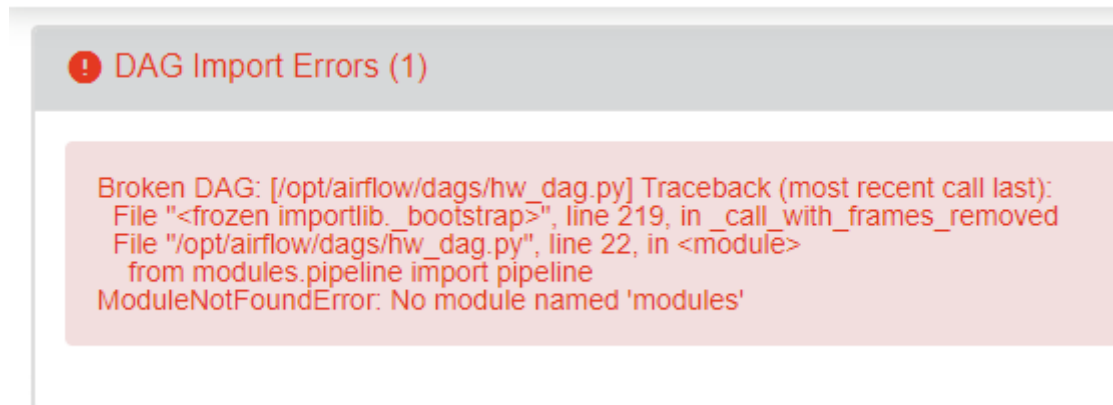


Дмитрий

02 октября, 19:33

**Вячеслав, здравствуйте!**

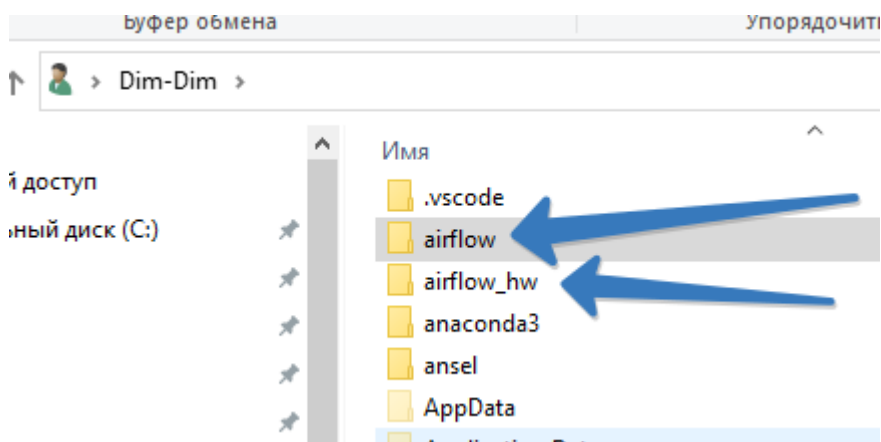
Не могу побороть ошибку:



Подробности ниже.

Когда отлаживаю в Python локальные модули, то всё работает. А вот внутри контейнера не хочет видеть файлы если я указываю путь к их расположению. Вообще никакие не видит.

Вот как разделены проект и данные Airflow:



Airflow обнаруживает dag\_hw.py запускает, но возникает та ошибка, которую указал выше - он не может подключить модули, которые находятся в папке **airflow\_hw**

Специально для проверки сделал запись в лог:

```
logging.info(f'>>>>>>: File: {sfile} {os.path.exists(sfile)}' + "\n \n")
```

Это я пытаюсь проверять наличие файла "/home/**airflow\_hw**/modules/predict.py". Но в лог всегда попадает False, то есть файл не обнаруживается.

Ладно, перенёс папку **modules** из **airflow\_hw** в **airflow** и изменил путь в начале dag\_hw.py, но ошибка осталась та же самая и всё равно файл не виден.

Вот как выглядит код логирования:

```
path = os.path.expanduser('~')
sfile = path + "/modules/predict.py"
logging.info("\n \n" + f'>>>>> PATH {path}' + "\n \n")
logging.info(f'>>>>>: File: {sfile} {os.path.exists(sfile)}' + "\n \n")
```

Всё равно не видит файл:

```
File "<frozen importlib._bootstrap_external>", line 728, in exec_module
File "<frozen importlib._bootstrap>", line 219, in _call_with_frames_removed
File "/opt/airflow/dags/hw_dag.py", line 21, in <module>
    from modules.pipeline import pipeline
ModuleNotFoundError: No module named 'modules'
[2022-10-02T15:40:06.730+0000] {processor.py:770} WARNING - No viable dags retrieved from /opt/airflow/dags/hw_dag.py
[2022-10-02T15:40:06.762+0000] {processor.py:178} INFO - Processing /opt/airflow/dags/hw_dag.py took 0.078 seconds
[2022-10-02T15:40:37.066+0000] {processor.py:156} INFO - Started process (PID=1003) to work on /opt/airflow/dags/hw_dag.py
[2022-10-02T15:40:37.073+0000] {processor.py:758} INFO - Processing file /opt/airflow/dags/hw_dag.py for tasks to queue
[2022-10-02T15:40:37.079+0000] {logging_mixin.py:117} INFO - [2022-10-02T15:40:37.078+0000] {dagbag.py:525} INFO - Filling up the DagBag from
/opt/airflow/dags/hw_dag.py
[2022-10-02T15:40:37.121+0000] {logging_mixin.py:117} INFO - [2022-10-02T15:40:37.120+0000] {hw_dag.py:11} INFO -
>>>>> PATH /home/airflow
[2022-10-02T15:40:37.122+0000] {logging_mixin.py:117} INFO - [2022-10-02T15:40:37.121+0000] {hw_dag.py:12} INFO - >>>>>: File:
/home/airflow/modules/predict.py False
[2022-10-02T15:40:37.134+0000] {logging_mixin.py:117} INFO - [2022-10-02T15:40:37.125+0000] {dagbag.py:330} ERROR - Failed to import:
/opt/airflow/dags/hw_dag.py
Traceback (most recent call last):
  File "/home/airflow/.local/lib/python3.7/site-packages/airflow/models/dagbag.py", line 326, in parse
    loader.exec_module(new_module)
  File "<frozen importlib._bootstrap_external>", line 728, in exec_module
  File "<frozen importlib._bootstrap>", line 219, in _call_with_frames_removed
  File "/opt/airflow/dags/hw_dag.py", line 21, in <module>
    from modules.pipeline import pipeline
ModuleNotFoundError: No module named 'modules'
[2022-10-02T15:40:37.137+0000] {processor.py:770} WARNING - No viable dags retrieved from /opt/airflow/dags/hw_dag.py
[2022-10-02T15:40:37.202+0000] {processor.py:178} INFO - Processing /opt/airflow/dags/hw_dag.py took 0.142 seconds
[2022-10-02T15:41:06.417+0000] {processor.py:156} INFO - Started process (PID=1022) to work on /opt/airflow/dags/hw_dag.py
[2022-10-02T15:41:06.420+0000] {processor.py:758} INFO - Processing file /opt/airflow/dags/hw_dag.py for tasks to queue
[2022-10-02T15:41:06.422+0000] {logging_mixin.py:117} INFO - [2022-10-02T15:41:06.422+0000] {dagbag.py:525} INFO - Filling up the DagBag from
/opt/airflow/dags/hw_dag.py
[2022-10-02T15:41:06.475+0000] {logging_mixin.py:117} INFO - [2022-10-02T15:41:06.475+0000] {hw_dag.py:11} INFO -
```

В ручную менял путь /home/airflow/modules/predict.py на /home/modules/predict.py всё равно не видит.

Но если перенести папку modules в папку, где находится dag\_hw.py , то ошибка исчезает.... ну в этом нет ничего странного ... но это не спортивно, так как хочется подгружать модули из другого каталога.

Но дело даже не в этом! Эта ошибка исчезает, но появляются проблемы в работе, потому что уже pipeline.py и predict.py не видят файлы train.csv и json-ы с тестами.

Скорее всего ошибка примитивная, но я не вижу где. Уже часов 5 потратил на поиск.... не нашёл.

Прошу помочь.



Дмитрий

02 октября, 22:16

Добрый вечер!

Кажется, разобрался как сделать файлы видимыми - их нужно вручную копировать, используя команды докера. Правда после перезагрузки докера эти файлы исчезли, но ладно, это уже тонкости - у меня написан батник, который копирование автоматизирует.

Обнаружилась другая проблема, оказывается, что не вызываются `pipeline()` и `predict()`

Вот эти:

```
with DAG(  
    dag_id='car_price_prediction',  
    schedule_interval="00 15 * * *",  
    default_args=args,  
) as dag:  
    pipeline = PythonOperator(  
        task_id='model_creation',  
        python_callable=pipeline,  
    )  
    predict = PythonOperator(  
        task_id='price_prediction',  
        python_callable=predict,  
    )
```

Проверил просто:

Вот эти строки не появляются в логах:

```
df = df.copy()
df.loc[:, 'short_model'] = df['model'].apply(short_model)
df.loc[:, 'age_category'] = df['year'].apply(lambda x:
return df
```

```
def pipeline() -> None:
    logging.info(f'!@! pipeline() started')
    path = "."

    logging.info(f'-----Список файлов (pipeline)
    directory = f'{path}/data/train'
    logging.info(f'----->: directory: {directory}')
    for filename in os.listdir(directory):
        logging.info(f'----->: файлы: {filename}')
    logging.info(f'-----')

    df = pd.read_csv(f'{path}/data/train/homework.csv')
    logging.info(f'----->: df = pd.read_csv: {df.shape}')
```

```
import os
import dill

def predict():
    logging.info(f'predict() started')
    # path = os.path.expanduser('~')
    path = "."
    path_models = os.path.join(path, "data", "models")
    path_test = os.path.join(path, "data", "test")
    path_preds = os.path.join(path, "data", "predictions")

    # Если несколько моделей, то будем обрабатывать каждую
    for model_filename in os.listdir(path_models):
        full_model_file_name = os.path.join(path_models, model_filename)
        model_version = model_filename.split("car")
```

Если эти строки вынести из def и поставить в самом начале файла (сразу после import ...), то записи в логах будут, но видимо, это следствие вот этих команд:

```
from modules.pipeline import pipeline
```

```
from modules.predict import predict
```

Не знаю как запустить.... подскажите, где поковырять?



Дмитрий

03 октября, 01:13

Ни как ... не хочет вызывать функции `pipeline` и `predict` и соответственно файлы не создаются.  
Что-то происходит, а что не понятно.

DAG: car\_price\_prediction

Schedule: 00 15 \*\*\*Next Run: 2022-06-10, 15:00:00

Grid

Graph

Calendar

Task Duration

Task Tries

Landing Times

Gantt

Details

<> Code

Audit Log

02.10.2022 22:09:08

25

All Run Types

All Run States

Clear Filters

deferred

failed

queued

running

scheduled

skipped

success

up\_for\_reschedule

up\_for\_retry

upstream\_failed

no\_status

Auto-refresh

→

Duration

00:03:08

00:01:34

00:00:00

Jun 15, 15:00

Jun 25, 15:00

model\_creation

price\_prediction

DAG

car\_price\_prediction

DAG Details

DAG Runs Summary

Total Runs Displayed

19

Total failed

4

Total running

15

First Run Start

2022-10-02, 22:06:02 UTC

Last Run Start

2022-10-02, 22:07:40 UTC

Max Run Duration

00:03:08

Mean Run Duration

00:02:32

Min Run Duration

00:01:13

DAG Summary

Total Tasks

2

PythonOperators

2



Вячеслав  
Куратор

03 октября, 11:05

Доброе утро, Дмитрий!

*Кажется, разобрался как сделать файлы видимыми - их нужно в ручную копировать используя команды докера. Правда после перезагрузки докера эти файлы исчезли, но ладно, это уже тонкости - у меня написан батник, который копирование автоматизирует.*

Да, всё верно. Когда скрипт предикта готов и отлажен, нам нужно положить его и сопутствующие файлы на worker, чтобы он запускался, а файлы считывались. В инструкции не указано, но всё это нужно сложить ещё и на scheduler, иначе веб-интерфейс будет показывать ошибку при импорте дага. На всякий случай – копировать лучше директорию целиком. Скажем, как-то так:

```
docker cp ~/airflow_hw <worker id>:/home/airflow/airflow_hw
```

¶ По дагу:

```
path = os.path.expanduser('~/.airflow_hw')
```

Вот эту строку менять не требовалось, чтобы в переменную среды PROJECT\_PATH положился корректный путь к файлам, которые мы скопировали в контейнер.

¶ pipeline

Соответственно, в пайлайне оставляем код:

```
path = os.environ.get('PROJECT_PATH', '.')
```

Иными словами, в качестве initial-пути будет браться либо путь, лежащий в переменной среды PROJECT\_PATH (при запуске в airflow, т.е. когда даг записал туда путь), либо, если переменной PROJECT\_PATH нет – будет браться текущая директория для локального запуска, т.е. ".".

¶ predict

1. Аналогично и здесь – верните строку (можно поставить её сразу после импортов в глобальной области видимости):

```
path = os.environ.get('PROJECT_PATH', '.')
```

2. Функцию predict можно декомпозировать на две составляющие: получение последней модели из папки с моделями, получение предсказаний. Их уже можно объединить в главной функции predict. Это сделает код чуть более читабельным.

Если будут ошибки, которые не получится поправить – присылайте логи, будем посмотреть.

Успехов!

Всего доброго,  
Вячеслав

Работа отправлена на доработку



Дмитрий

03 октября, 12:05

Доброе утро!

Вячеслав, Вы пишете:

"// По дагу:

[path = os.path.expanduser\('~/.airflow\\_hw'\)](#)

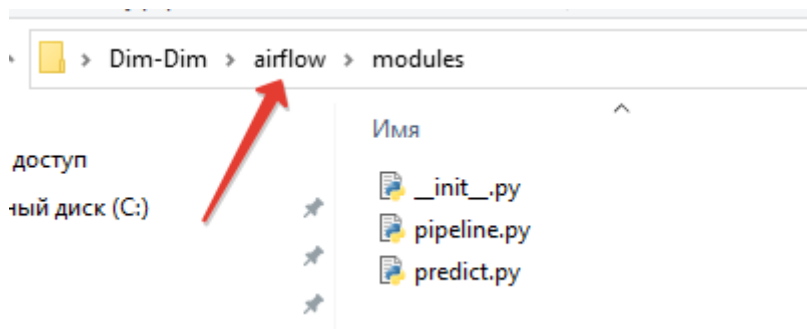
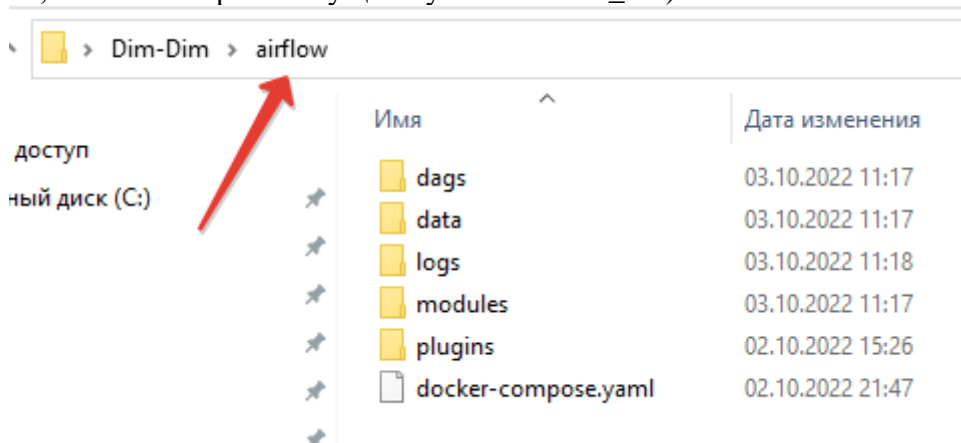
Вот эту строку менять не требовалось, чтобы в переменную среды PROJECT\_PATH положился корректный путь к файлам, которые мы скопировали в контейнер."

Так я уже пробовал.

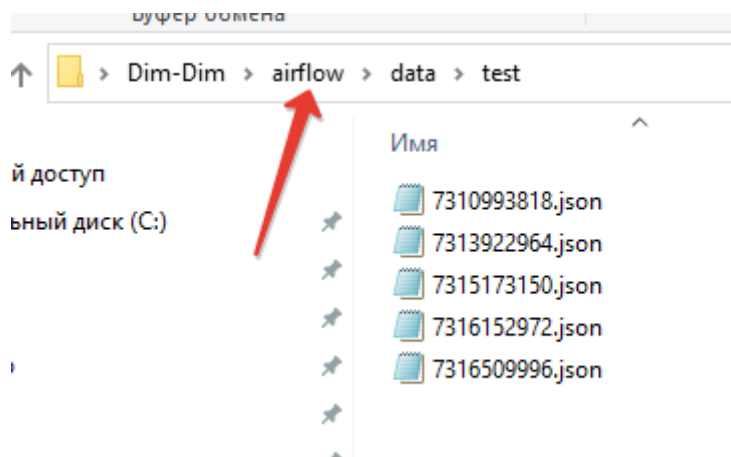
Мой контейнер не только внешние папки не видит, он и свои собственные не видит!

Только что всё удалил и установил заново, вот смотрите:

1) Вот как выглядят локальные папки перед загрузкой и созданием контейнеров (здесь я не показываю то, что папки проекта существуют в **airflow\_hw**):







2) Запускаю поочередно:

**docker-compose up airflow-init**

**docker-compose up**

Всё, после этого у меня в Докере установлено всё что нужно, так?

3) А теперь захожу в терминал одного из контейнеров:


**docker exec -it airflow-airflow-scheduler-1 bash**

**docker exec -it airflow-airflow-worker-1 bash**

И у них внутри нет таких папок как:

- modules

- data

 airflow@8af0f9fbaec8: /opt/airflow

```
C:\Users\Dim-Dim\airflow>docker exec -it airflow-airflow-scheduler-1 bash
airflow@8af0f9fbaec8:/opt/airflow$ ls
airflow.cfg dags logs plugins webserver_config.py
airflow@8af0f9fbaec8:/opt/airflow$
```

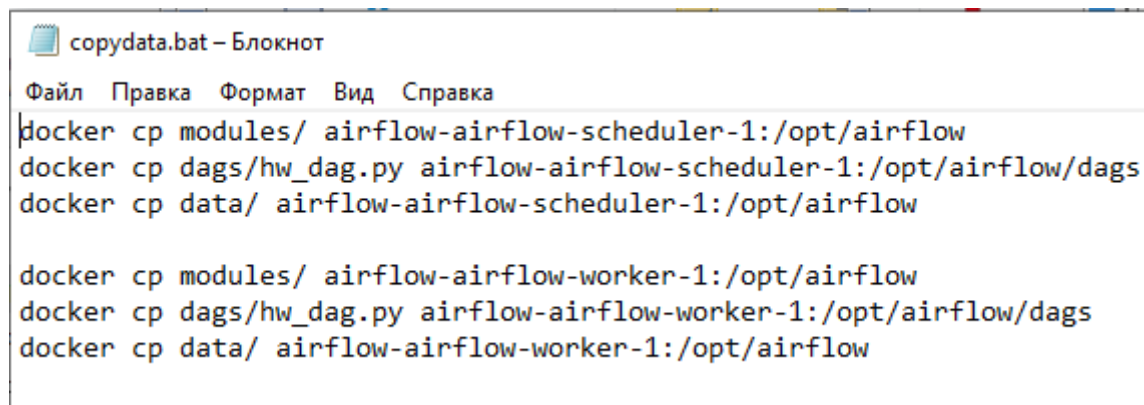
Поэтому модули и не подключались.

Поэтому и файлы с данными были не видны.

Тогда с какой стати должен быть виден внешний каталог `~/airflow_hw` если внутренние не видны?

**Проблема с доступом к данным – это первая проблема.** Её я победил тем, что вручную создал папки в контейнерах и скопировал туда код и данные:

```
dags
data
modules
.gitignore
copydata.bat
```



```
copydata.bat - Блокнот
Файл  Правка  Формат  Вид  Справка
docker cp modules/ airflow-airflow-scheduler-1:/opt/airflow
docker cp dags/hw_dag.py airflow-airflow-scheduler-1:/opt/airflow/dags
docker cp data/ airflow-airflow-scheduler-1:/opt/airflow

docker cp modules/ airflow-airflow-worker-1:/opt/airflow
docker cp dags/hw_dag.py airflow-airflow-worker-1:/opt/airflow/dags
docker cp data/ airflow-airflow-worker-1:/opt/airflow
```

**Вторая проблема заключается том, что DAG регистрируется, а связанные с ним задачи не запускаются.** И я не смог это победить.

Был или нет запуск задачи я контролировал через логи - в начале прописывал некоторое сообщение для информирования того запуск был. По какой-то причине запускает только та часть кода, которая прописана в "глобальной" области модуля, а функция, прописанная в модуле, не запускается.



Вячеслав  
Куратор

03 октября, 12:51

Дмитрий,

*2) Запускаю поочередно:*

*`docker-compose up airflow-init`*

*`docker-compose up`*

*Всё, после этого у меня в Докере установлено всё что нужно, так?*

Инициализация нужна только при первом запуске. Повторно можно запускать так:

```
docker-compose up
```

После первого запуска достаточно положить в контейнеры вашу папку `airflow` и поставить нужные для работы пайплайна/предикта пакеты (пример для `worker`):

1. Узнаём id контейнера с воркером:

```
docker ps | grep worker
```

2. Копируем исполняемый код и данные на воркер:

```
docker cp ~/airflow <worker_id>:/home/airflow/airflow_hw
```

При условии, что "Dim-Dim" – это имя вашего пользователя (т.е. `C:/Users/Dim-Dim` или просто `~`). После команды `docker cp` сначала указываем, что копируем, а затем то, куда копируем. Путь, куда копируем, трогать не нужно.

3. Заходим на воркер и ставим нужные для работы пайплайна пакеты:

```
docker exec -it <worker_id> bash  
pip install scikit-learn
```

По аналогии делаем с `scheduler`-ом.

P.S. Перед копированием скриптов на `worker` и `scheduler`, очистите, пожалуйста, папку с моделями (`data/models`).

*3) А теперь захожу в терминал одного из контейнеров:*

*`docker exec -it airflow-airflow-scheduler-1 bash`*

*`docker exec -it airflow-airflow-worker-1 bash`*

*И у них внутри нет таких папок как:*

*- `modules`*

*- `data`*

```
airflow@8af0f9fbaec8: /opt/airflow
```

```
C:\Users\Dim-Dim\airflow>docker exec -it airflow-airflow-scheduler-1 bash
airflow@8af0f9fbaec8:/opt/airflow$ ls
airflow.cfg  dags  logs  plugins  webserver_config.py
airflow@8af0f9fbaec8:/opt/airflow$
```

*Поэтому модули и не подключались.*

*Поэтому и файлы с данными были не видны.*

*Тогда с какой стати должен быть виден внешний каталог ~/airflow\_hw если внутренние не видны?*

Смотреть нужно в другом месте. Заходим в контейнер от root-пользователя:

```
docker exec -it -u root <id контейнера> bash
```

И следуем туда, куда копировали директорию:

```
cd /home/airflow/airflow hw
```

Для надёжности можно выдать права папкам – иногда с правами бывают беды при копировании с Windows:

```
chmod -R 777 data dags modules
```

Поправьте, пожалуйста, а затем, если вторая проблема не уйдёт, будем разбираться с ней.

Всего доброго,  
Вячеслав

Работа отправлена на доработку



Вячеслав  
Куратор

03 октября, 13:10

P.S. Решил перечитать, вдруг что-то упустил:

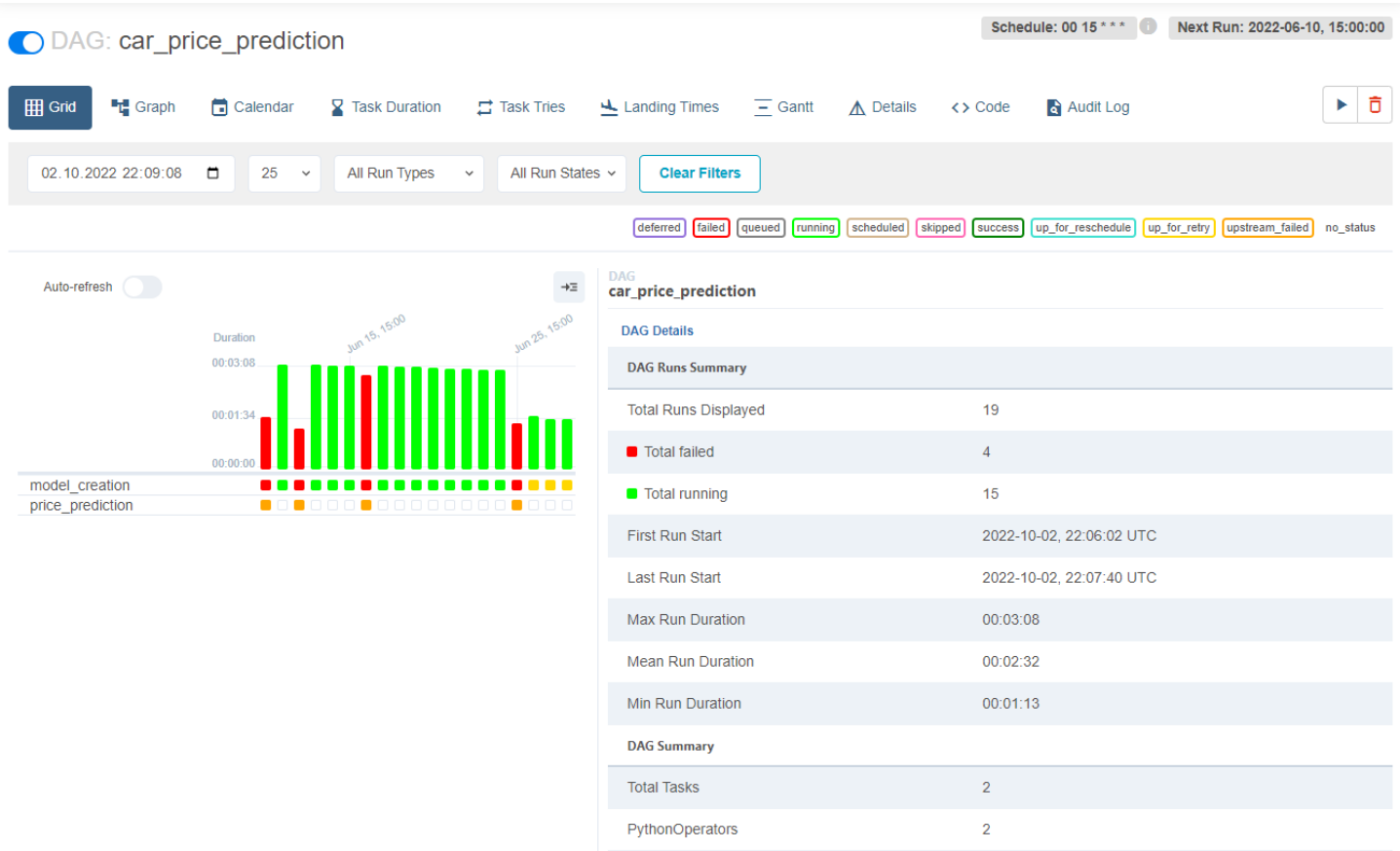
*Тогда с какой стати должен быть виден внешний каталог ~/airflow\_hw если внутренние не видны?*

~/airflow\_hw внутри контейнера – это, как я чуть ранее написал, папка по пути /home/airflow/airflow\_hw. /home/airflow – это наш пользователь внутри контейнера, от которого запускаются даги и стартуют таски, поэтому файлы чуть удобнее сложить там.

P.P.S.

*Ни как ... не хочет вызывать функции pipeline и predict и соответственно файлы не создаются.*

*Что-то происходит, а что не понятно.*



Если бы функции просто не запускались, то и ошибок бы не было – таски бы показывали заветный success. Но, поскольку имеет место быть ошибка (статус failed), то дело в чём-то другом.

Если так и не будет получаться – присылайте архив с проектом целиком: и папку с домашкой (airflow\_hw) и папку с докером (airflow-docker). Если декомпозиции по папкам нет, то лучше её сделать, дабы не загрязнять рабочее пространство.

\* Иными словами, в airflow-docker:

dags, logs, plugins, docker-compse.yaml

(отсюда поднимаем контейнеры)

\* В airflow\_hw:

dags, data, modules



Дмитрий

04 октября, 01:13

**Вячеслав, здравствуйте!**

Структуру папок создал как Вы написали.

Вы предлагаете использовать вот такую команду: `path = os.path.expanduser('~/.airflow_hw')`

Как я понял, Вы имеете в виду, что это присвоение укажет на то место где находится папка `modules`, при условии, что


1. `modules` вложена в `airflow_hw`
2. `airflow_hw` вложена в папку `airflow`

Вот смотрите, всё так и есть - вложение такое как надо и файлы там есть:

```
Командная строка - docker exec -it -u root airflow-airflow-scheduler-1 bash

C:\Users\Dim-Dim>docker exec -it -u root airflow-airflow-scheduler-1 bash
root@ff67d6af3b00:/opt/airflow# ls
airflow.cfg airflow_hw dags data logs modules plugins webserver_config.py
root@ff67d6af3b00:/opt/airflow# cd airflow_hw
root@ff67d6af3b00:/opt/airflow/airflow_hw# ls
dags data modules
root@ff67d6af3b00:/opt/airflow/airflow_hw# cd modules
root@ff67d6af3b00:/opt/airflow/airflow_hw/modules# ls
__init__.py __pycache__ pipeline.py predict.py
root@ff67d6af3b00:/opt/airflow/airflow_hw/modules#
```

Но ошибка осталась:

 DAG Import Errors (1)

Broken DAG: [/opt/airflow/dags/hw\_dag.py] Traceback (most recent call last):  
File "<frozen importlib.\_bootstrap>", line 219, in \_call\_with\_frames\_removed  
File "/opt/airflow/dags/hw\_dag.py", line 21, in <module>  
from modules.pipeline import pipeline  
ModuleNotFoundError: No module named 'modules'

Более того, я поставил в самое начало файла dag.py запись в логи проверки значений и наличие некоторого файла в папке (обведены голубыми линиями):

```

import datetime as dt
import logging
import os
import sys

from airflow.models import DAG
from airflow.operators.python import PythonOperator

# -----
logging.info(f'-----')
stext = '~'
logging.info(f'{stext}: {os.path.expanduser(stext)}')

stext = '~/airflow_hw/modules/pipeline.py'
logging.info(f'{stext}: {os.path.expanduser(stext)}')

stext = os.path.expanduser(stext)
logging.info(f'{stext}: {os.path.exists(stext)}')
logging.info(f'-----')
# -----

```

И вот что попало в логи:

```

[2022-10-03T13:04:00.892+0000] {logging_mixin.py:117} INFO - [2022-10-03T13:04:00.891+0000] {hw_dag.py:10} INFO -
-----
[2022-10-03T13:04:00.892+0000] {logging_mixin.py:117} INFO - [2022-10-03T13:04:00.892+0000] {hw_dag.py:12} INFO - ~: /home/airflow
[2022-10-03T13:04:00.893+0000] {logging_mixin.py:117} INFO - [2022-10-03T13:04:00.893+0000] {hw_dag.py:15} INFO -
~/airflow_hw/modules/pipeline.py: /home/airflow/airflow_hw/modules/pipeline.py
[2022-10-03T13:04:00.895+0000] {logging_mixin.py:117} INFO - [2022-10-03T13:04:00.894+0000] {hw_dag.py:18} INFO -
/home/airflow/airflow_hw/modules/pipeline.py: False
[2022-10-03T13:04:00.896+0000] {logging_mixin.py:117} INFO - [2022-10-03T13:04:00.896+0000] {hw_dag.py:19} INFO -
-----

```

Не видит программа этих путей, а соответственно не подключает модули.

А теперь поменяем на "точку":


```

# -----
logging.info(f'-----')
stext = '.'
logging.info(f'{stext}: {os.path.expanduser(stext)}')

stext = './airflow_hw/modules/pipeline.py'
logging.info(f'{stext}: {os.path.expanduser(stext)}')

stext = os.path.expanduser(stext)
logging.info(f'{stext}: {os.path.exists(stext)}')
logging.info(f'-----')
# -----

```



```

[2022-10-03T13:16:34.317+0000] {logging_mixin.py:117} INFO - [2022-10-03T13:16:34.316+0000] {hw_dag.py:12} INFO - .: .
[2022-10-03T13:16:34.317+0000] {logging_mixin.py:117} INFO - [2022-10-03T13:16:34.317+0000] {hw_dag.py:15} INFO -
./airflow_hw/modules/pipeline.py: ./airflow_hw/modules/pipeline.py
[2022-10-03T13:16:34.318+0000] {logging_mixin.py:117} INFO - [2022-10-03T13:16:34.318+0000] {hw_dag.py:18} INFO -
./airflow_hw/modules/pipeline.py: True
[2022-10-03T13:16:34.319+0000] {logging_mixin.py:117} INFO - [2022-10-03T13:16:34.319+0000] {hw_dag.py:19} INFO -
-----

```

И ошибка сразу исчезла:



# DAGs

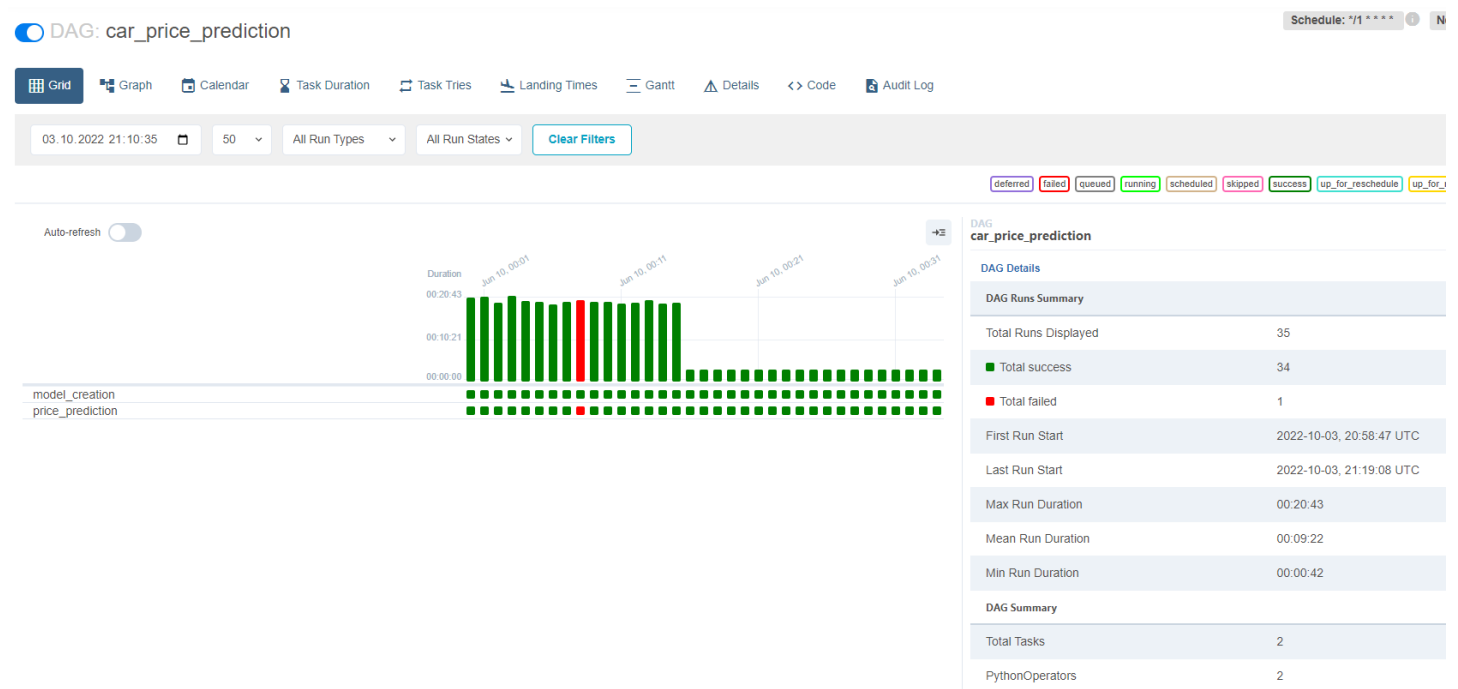
All 1 Active 0 Paused 1

*i* DAG

car\_price\_prediction

**Вот почему была ".", а не "~".**

В общем после ряда экспериментов о том какие файлы в какие контейнеры копировать, у меня получился вот что:



В логах посмотрел какие ошибки и понял, что я ничего не понимаю.... ошибка возникает на импортах библиотек. Вот, например, ошибка 9-го запуска (красный столбик на картинке выше):

```
[2022-10-03, 21:16:59 UTC] {taskinstance.py:1363} INFO - Starting attempt 1 of 2
[2022-10-03, 21:16:59 UTC] {taskinstance.py:1364} INFO -
-----
[2022-10-03, 21:16:59 UTC] {taskinstance.py:1383} INFO - Executing <Task(PythonOperator): price_prediction> on 202
[2022-10-03, 21:16:59 UTC] {standard_task_runner.py:54} INFO - Started process 1385 to run task
[2022-10-03, 21:16:59 UTC] {standard_task_runner.py:82} INFO - Running: ['***', 'tasks', 'run', 'car_price_predict
[2022-10-03, 21:16:59 UTC] {standard_task_runner.py:83} INFO - Job 21: Subtask price_prediction
[2022-10-03, 21:16:59 UTC] {dagbag.py:525} INFO - Filling up the DagBag from /opt/***/dags/hw_dag.py
[2022-10-03, 21:17:29 UTC] {timeout.py:68} ERROR - Process timed out, PID: 1385
[2022-10-03, 21:17:30 UTC] {dagbag.py:330} ERROR - Failed to import: /opt/***/dags/hw_dag.py
Traceback (most recent call last):
  File "/home/airflow/.local/lib/python3.7/site-packages/airflow/models/dagbag.py", line 326, in parse
    loader.exec_module(new_module)
  File "<frozen importlib._bootstrap_external>", line 728, in exec_module
  File "<frozen importlib._bootstrap>", line 219, in _call_with_frames_removed
  File "/opt/airflow/dags/hw_dag.py", line 16, in <module>
    from modules.pipeline import pipeline_func
  File "./airflow_hw/modules/pipeline.py", line 9, in <module>
    from sklearn.ensemble import RandomForestClassifier
  File "/home/airflow/.local/lib/python3.7/site-packages/sklearn/ensemble/__init__.py", line 16, in <module>
    from ._gb import GradientBoostingClassifier
```

Вот результат:

```
C:\WINDOWS\system32\cmd.exe - docker exec -it -u root airflow-airflow-worker-1 bash
Microsoft Windows [Version 10.0.19044.2075]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\Dim-Dim\airflow>docker exec -it -u root airflow-airflow-worker-1 bash
root@1aa2dd071cfd:/opt/airflow# cd airflow_hw
root@1aa2dd071cfd:/opt/airflow/airflow_hw# ls
data modules
root@1aa2dd071cfd:/opt/airflow/airflow_hw# cd data
root@1aa2dd071cfd:/opt/airflow/airflow_hw/data# ls
models predictions test train
root@1aa2dd071cfd:/opt/airflow/airflow_hw/data# ls models/*.
ls: cannot access 'models/*.': No such file or directory
root@1aa2dd071cfd:/opt/airflow/airflow_hw/data# ls /models/*.
ls: cannot access '/models/*.': No such file or directory
root@1aa2dd071cfd:/opt/airflow/airflow_hw/data# cd models
root@1aa2dd071cfd:/opt/airflow/airflow_hw/data/models# ls
cars_pipe_9f0e883aeb.pkl
root@1aa2dd071cfd:/opt/airflow/airflow_hw/data/models# cd..
bash: cd..: command not found
root@1aa2dd071cfd:/opt/airflow/airflow_hw/data/models# cd ..
root@1aa2dd071cfd:/opt/airflow/airflow_hw/data# ls
models predictions test train
root@1aa2dd071cfd:/opt/airflow/airflow_hw/data# cd predictions
root@1aa2dd071cfd:/opt/airflow/airflow_hw/data/predictions# ls
preds_9f0e883aeb.csv
root@1aa2dd071cfd:/opt/airflow/airflow_hw/data/predictions# cat preds_9f0e883aeb.csv
ModelName,DataFile,Prediction
cars_pipe_9f0e883aeb,7316509996.json,high
cars_pipe_9f0e883aeb,7316152972.json,medium
cars_pipe_9f0e883aeb,7310993818.json,low
cars_pipe_9f0e883aeb,7313922964.json,high
cars_pipe_9f0e883aeb,7315173150.json,low
root@1aa2dd071cfd:/opt/airflow/airflow_hw/data/predictions#
```

Репозиторий на GitHub:

{ skipped }

Сделал установку достаточно простой и не привязанной к домашнему каталогу пользователя: копируем репозиторий (папку airflow\_hw) в любое место и запускаем по очереди три "батника":

- InstallAndStart.bat (Создаёт и запускает контейнеры)
- PiPInstall.bat (Запускать только по завершении работы первого)
- CopyData.bat (Запускать только по завершении работы первого)

"Убил" на эту задачу в течение двух дней часов 18-20....

**Вячеслав, прошу посмотреть мою работу на предмет зачёта.**



Вячеслав  
Куратор

04 октября, 10:27

Доброе утро, Дмитрий! 🙌

Как я понял, Вы имеете в виду, что это присвоение укажет на то место, где находится папка modules, при условии, что


1. modules вложена в airflow\_hw
2. airflow\_hw вложена в папку airflow

Вот смотрите, всё так и есть - вложение такое как надо и файлы там есть:

```
Командная строка - docker exec -it -u root airflow-airflow-scheduler-1 bash

C:\Users\Dim-Dim>docker exec -it -u root airflow-airflow-scheduler-1 bash
root@ff67d6af3b00:/opt/airflow# ls
airflow.cfg airflow_hw dags data logs modules plugins webserver_config.py
root@ff67d6af3b00:/opt/airflow# cd airflow_hw
root@ff67d6af3b00:/opt/airflow/airflow_hw# ls
dags data modules
root@ff67d6af3b00:/opt/airflow/airflow_hw# cd modules
root@ff67d6af3b00:/opt/airflow/airflow_hw/modules# ls
__init__.py __pycache__ pipeline.py predict.py
root@ff67d6af3b00:/opt/airflow/airflow_hw/modules#
```

Но ошибка осталась:

 DAG Import Errors (1)

Broken DAG: [/opt/airflow/dags/hw\_dag.py] Traceback (most recent call last):  
File "<frozen importlib.\_bootstrap>", line 219, in \_call\_with\_frames\_removed  
File "/opt/airflow/dags/hw\_dag.py", line 21, in <module>  
 from modules.pipeline import pipeline  
ModuleNotFoundError: No module named 'modules'

Повторюсь, вы не совсем там просматриваете. Нам нужна директория не /opt/airflow, а /home/airflow. Я предлагал посмотреть так:

```
cd /home/airflow/airflow_hw
ls
```

При условии, что при копировании директории вы указали корректный путь "до" (/home/airflow/airflow\_hw):

```
docker cp ~/airflow <worker_id>:/home/airflow/airflow_hw
```

А теперь поменяем на "точку":

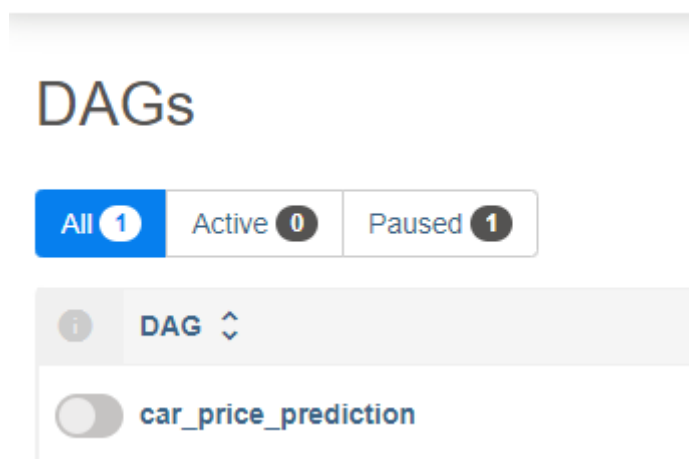
```
# -----
logging.info(f'-----')
stext = '.'
logging.info(f'{stext}: {os.path.expanduser(stext)}')

stext = './airflow_hw/modules/pipeline.py'
logging.info(f'{stext}: {os.path.expanduser(stext)}')

stext = os.path.expanduser(stext)
logging.info(f'{stext}: {os.path.exists(stext)}')
logging.info(f'-----')
# -----
```

```
[2022-10-03T13:16:34.317+0000] {logging_mixin.py:117} INFO - [2022-10-03T13:16:34.316+0000] {hw_dag.py:12} INFO - .: .
[2022-10-03T13:16:34.317+0000] {logging_mixin.py:117} INFO - [2022-10-03T13:16:34.317+0000] {hw_dag.py:15} INFO -
./airflow_hw/modules/pipeline.py: ./airflow_hw/modules/pipeline.py
[2022-10-03T13:16:34.318+0000] {logging_mixin.py:117} INFO - [2022-10-03T13:16:34.318+0000] {hw_dag.py:18} INFO -
./airflow_hw/modules/pipeline.py: True И файл сразу нашёлся!
[2022-10-03T13:16:34.319+0000] {logging_mixin.py:117} INFO - [2022-10-03T13:16:34.319+0000] {hw_dag.py:19} INFO -
-----
[2022-10-03T13:16:34.319+0000] {logging_mixin.py:117} INFO - [2022-10-03T13:16:34.319+0000] {hw_dag.py:19} INFO -
```

И ошибка сразу исчезла:



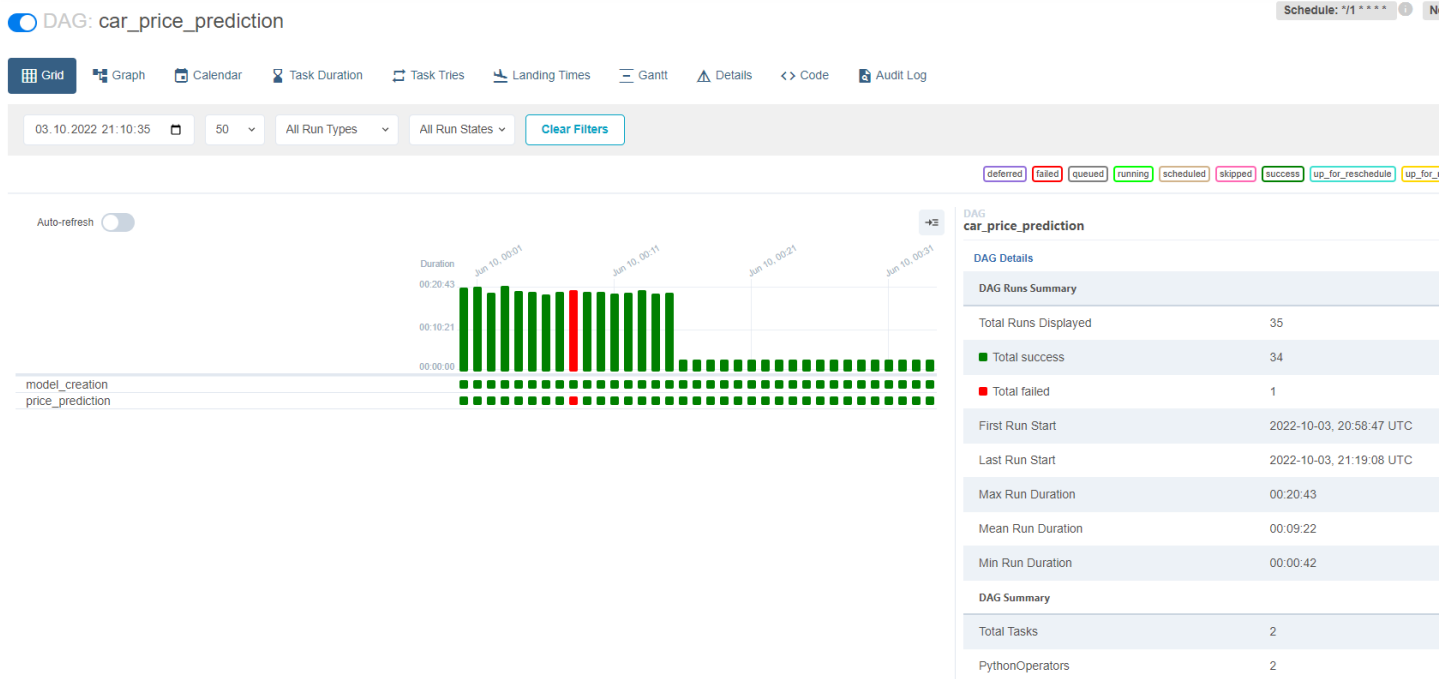
**Вот почему была ".", а не "~".**

Логично, что "." срабатывает, а "~" нет, т.к. модули вы сложили в /opt/airflow, а не в /home/airflow. По умолчанию просматривается именно opt, поэтому так и получается.

В общем-то, здесь можно разными путями идти, но складывать наши сабмодули и данные в /opt будет не совсем корректно – эта директория служит для других целей, в неё обычно кладутся всякие проприетарные пакеты.

---

В общем после ряда экспериментов о том какие файлы в какие контейнеры копировать, у меня получился вот что:



В логах посмотрел какие ошибки и понял, что я ничего не понимаю.... ошибка возникает на импортах библиотек. Вот, например, ошибка 9-го запуска (красный столбик на картинке выше):

```
[2022-10-03, 21:16:59 UTC] {taskinstance.py:1363} INFO - Starting attempt 1 of 2
[2022-10-03, 21:16:59 UTC] {taskinstance.py:1364} INFO -
-----
[2022-10-03, 21:16:59 UTC] {taskinstance.py:1383} INFO - Executing <Task(PythonOperator): price_prediction> on 202
[2022-10-03, 21:16:59 UTC] {standard_task_runner.py:54} INFO - Started process 1385 to run task
[2022-10-03, 21:16:59 UTC] {standard_task_runner.py:82} INFO - Running: ['***', 'tasks', 'run', 'car_price_predict
[2022-10-03, 21:16:59 UTC] {standard_task_runner.py:83} INFO - Job 21: Subtask price_prediction
[2022-10-03, 21:16:59 UTC] {dagbag.py:525} INFO - Filling up the DagBag from /opt/***/dags/hw_dag.py
[2022-10-03, 21:17:29 UTC] {timeout.py:68} ERROR - Process timed out, PID: 1385
[2022-10-03, 21:17:30 UTC] {dagbag.py:330} ERROR - Failed to import: /opt/***/dags/hw_dag.py
Traceback (most recent call last):
  File "/home/airflow/.local/lib/python3.7/site-packages/airflow/models/dagbag.py", line 326, in parse
    loader.exec_module(new_module)
  File "<frozen importlib._bootstrap_external>", line 728, in exec_module
  File "<frozen importlib._bootstrap>", line 219, in _call_with_frames_removed
  File "/opt/airflow/dags/hw_dag.py", line 16, in <module>
    from modules.pipeline import pipeline_func
  File "./airflow_hw/modules/pipeline.py", line 9, in <module>
    from sklearn.ensemble import RandomForestClassifier
  File "/home/airflow/.local/lib/python3.7/site-packages/sklearn/ensemble/__init__.py", line 16, in <module>
    from ._gb import GradientBoostingClassifier
```


Отлично! Ошибка не критичная – просто задача устала ждать своей очереди на выполнение и решила выкинуть failed.

Спасибо за работу, принимается! 🙏

P.S. Как замену встроенному модулю logging, от себя ещё советую попробовать библиотеку loguru для виртуозного логирования. Почитать/посмотреть можно здесь:

- <https://youtu.be/3ndEeGDVqD4>

- [Delgan/loguru: Python logging made \(stupidly\) simple \(github.com\)](https://github.com/Delgan/loguru)

P.P.S. На этом модуле мы с вами прощаемся (но, возможно, когда-нибудь ещё встретимся!). Спасибо за неизменное стремление к знаниям. Желаю не останавливаться в обучении и приобретать все больше навыков, используя их на практике! 

Всего доброго,  
Вячеслав

Работа принята

Можете переходить к следующему модулю