

Ismael Moreau

TP1 du Cours Mise en production –



Amor Amami

Github : <https://github.com/ismaelMoreau/Cours-A61>

---

J'ai décidé d'utiliser un Dockerfile et de construire une image à chaque partie du TP pour faire les tests et autres, au lieu d'un environnement virtuel. J'ai trouvé ça plus propre pour avoir la version de Python avec les librairies aux versions spécifiques. Aussi, j'utilise le CLI de GitHub (<https://cli.github.com/manual/>) pour la connexion avec token entre Git (localement) et GitHub.

Voici le DOCKERFILE original :

```
# Utilise Python 3.8 comme image de base
FROM python:3.8

# Définit le répertoire de travail dans le conteneur
WORKDIR /app

# Copie le fichier requirements.txt dans le conteneur
COPY requirements.txt .

# Installe les dépendances depuis requirements.txt
RUN pip install -r requirements.txt

# Installe tox
RUN pip install tox

# Copie l'ensemble du projet dans le conteneur
COPY . .

# Commande pour exécuter l'application Python (ou dans ce cas, ouvrir un shell
bash)
CMD ["/bin/bash"]
```

\*\*\*\*il va beaucoup changer au fils des etapes\*\*\*\*

Les parties 1 à 6 ont toutes suivi la même procédure :

1. Remplacement du dossier possédant les nouvelles fonctionnalités

2. git add . , git commit -m "Partie-X-...", git push
3. docker build -t a61build . , docker run -it a61build bash
4. tox -r (resultat:

```
docs: https://docs.pytest.org/en/latest/warnings.html
===== 2 passed, 12 warnings in 0.95s =====
regression_model: OK (33.99=setup[31.62]+cmd[1.08,1.29] seconds)
congratulations :) (34.07 seconds)
)
)
```

Étapes supplémentaires des Parties suivantes

Partie 7:

1 pip install -e ./regression\_model

```
adding 'regression_model-0.1.0.dist-info/METADATA'
adding 'regression_model-0.1.0.dist-info/WHEEL'
adding 'regression_model-0.1.0.dist-info/top_level.txt'
adding 'regression_model-0.1.0.dist-info/RECORD'
removing build/bdist.linux-x86_64/wheel
install_locally: OK (14.09=setup[12.85]+cmd[0.85,0.40] seconds)
congratulations :) (14.16 seconds)
```

2 pip install -e ./regression\_model

```
Successfully installed pluggy-0.13.1 regression-model-0.1.0
```

3 python , import , version

```
root@e703964dabb3:/app# python
Python 3.8.18 (default, Jan 17 2024, 07:06:51)
[GCC 12.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import regression_model
>>> regression_model.__version__
'0.1.0'
>>>
```

Partie 8 :

J'ai ajouté un nouveau Dockerfile pour utiliser l'application Flask dans un Docker. Une seule ligne a été nécessaire de changer dans `run.py` pour permettre l'accès au Docker depuis l'hôte :

```
application.run(host='0.0.0.0', port=5000, debug=True)
```

(À réappliquer à toutes les parties suivantes.)

Partie 9 :

J'ai dû ajouter le path pour réussir à faire rouler le test de Flask dans le Docker.

```
export PYTHONPATH=/app/api:$PYTHONPATH
```

Partie 10 : J'ai combiné les Dockerfile en un seul pour le projet complet. Lors du build, Docker va créer le package et l'installer.

## Partie 11

```
root@aec8eda652a0:/app# cd ml_api/
root@aec8eda652a0:/app/ml_api# pytest tests
===== test session starts =====
platform linux -- Python 3.8.18, pytest-5.4.3, py-1.11.0, pluggy-0.13.1
rootdir: /app/ml_api
collected 3 items

tests/test_controller.py ...

===== warnings summary =====
tests/test_controller.py::test_health_endpoint_returns_200
  /usr/local/lib/python3.8/site-packages/sklearn/utils/multiclass.py:13: DeprecationWarning: Please use `spmatrix` namespace, the `scipy.sparse.base` namespace is deprecated.
    from scipy.sparse.base import spmatrix

tests/test_controller.py::test_health_endpoint_returns_200
  /usr/local/lib/python3.8/site-packages/sklearn/utils/optimize.py:18: DeprecationWarning: Please use `line_search` namespace, the `scipy.optimize.linesearch` namespace is deprecated.
    from scipy.optimize.linesearch import line_search_wolfe2, line_search_wolfe1

tests/test_controller.py::test_health_endpoint_returns_200
  /usr/local/lib/python3.8/site-packages/sklearn/utils/optimize.py:18: DeprecationWarning: Please use `line_search` namespace, the `scipy.optimize.linesearch` namespace is deprecated.
    from scipy.optimize.linesearch import line_search_wolfe2, line_search_wolfe1

-- Docs: https://docs.pytest.org/en/latest/warnings.html
===== 3 passed, 3 warnings in 0.72s =====
```

```
exit
sma@voltaire /m/s/p/C/packages (main)> sudo docker run a61build
2024-01-22 23:47:34,264 - api.app - DEBUG -create_app:18 - Application instance created
* Serving Flask app 'ml_api'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://172.17.0.2:5000
Press CTRL+C to quit
* Restarting with stat
2024-01-22 23:47:35,026 - api.app - DEBUG -create_app:18 - Application instance created
* Debugger is active!
* Debugger PIN: 458-954-981
172.17.0.1 - - [22/Jan/2024 23:47:37] "GET / HTTP/1.1" 404 -
172.17.0.1 - - [22/Jan/2024 23:47:45] "GET /version HTTP/1.1" 200 -
```

## Partie 12

```

root@7ccf4444f8e7:/app/ml_api# pytest tests
===== test session starts =====
platform linux -- Python 3.8.18, pytest-5.4.3, py-1.11.0, pluggy-0.13.1
rootdir: /app/ml_api
collected 4 items

tests/test_controller.py ...
tests/test_validation.py .

===== warnings summary =====
tests/test_controller.py::test_health_endpoint_returns_200
  /usr/local/lib/python3.8/site-packages/sklearn/utils/multiclass.py:13: DeprecationWarning: Please use `spmatrix` namespace, the `scipy.sparse.base` namespace is deprecated.
    from scipy.sparse.base import spmatrix

tests/test_controller.py::test_health_endpoint_returns_200
  /usr/local/lib/python3.8/site-packages/sklearn/utils/optimize.py:18: DeprecationWarning: Please use `line_search` namespace, the `scipy.optimize.linesearch` namespace is deprecated.
    from scipy.optimize.linesearch import line_search_wolfe2, line_search_wolfe1

tests/test_controller.py::test_health_endpoint_returns_200
  /usr/local/lib/python3.8/site-packages/sklearn/utils/optimize.py:18: DeprecationWarning: Please use `line_search` namespace, the `scipy.optimize.linesearch` namespace is deprecated.
    from scipy.optimize.linesearch import line_search_wolfe2, line_search_wolfe1

-- Docs: https://docs.pytest.org/en/latest/warnings.html
===== 4 passed, 3 warnings in 1.76s =====

```

```

=> => transferring context: 2.58MB
=> CACHED [2/8] WORKDIR /app
=> CACHED [3/8] COPY regression_model/requirements.txt /app
=> CACHED [4/8] RUN pip install -r requirements.txt
=> CACHED [5/8] RUN pip install pytest tox flask marshmallow==2.17.0
=> [6/8] COPY . /app
=> [7/8] RUN cd /app/regression_model && tox -e install_locally
=> [8/8] RUN cd /app/regression_model && pip install -e .
=> exporting to image
=> => exporting layers
=> => writing image sha256:78fcd1888be4d651d71be99b9799273d140479f0dec782b5d4f770c9ef3892e3
=> => naming to docker.io/library/a61build
sma@voltaire /m/s/p/C/packages (main)> sudo docker run a61build
* Serving Flask app 'ml_api'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://172.17.0.2:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 458-954-981
172.17.0.1 - - [23/Jan/2024 00:52:35] "GET / HTTP/1.1" 404 -
172.17.0.1 - - [23/Jan/2024 00:52:43] "GET /version HTTP/1.1" 200 -
2024-01-23 00:53:34,671 - api.controller - INFO -health:18 - health status OK
172.17.0.1 - - [23/Jan/2024 00:53:34] "GET /health HTTP/1.1" 200 -

```